

COMPUTER SOFTWARE APPLICATION

TRADE PRACTICAL
NSQF LEVEL - 4

VOLUME - 1

HANDBOOK FOR CRAFTS INSTRUCTOR
TRAINING SCHEME



Directorate General of Training

**DIRECTORATE GENERAL OF TRAINING
MINISTRY OF SKILL DEVELOPMENT & ENTREPRENEURSHIP
GOVERNMENT OF INDIA**



**NATIONAL INSTRUCTIONAL
MEDIA INSTITUTE, CHENNAI**

Post Box No. 3142, CTI Campus, Guindy, Chennai - 600 032

Published by



National Instructional Media Institute

Post.Box.No. 3142,

Guindy, Chennai - 600032

Email : chennai-nimi@nic.in

Website: www.nimi.gov.in

All Rights Reserved

First Edition, 2024

Rs. 570/-

Printed in India at

National Instructional Media Institute

Post. Box. No. 3142,

Guindy,

Chennai - 600032

Copyright©2024 NIMI

Disclaimer

The information contained herein has been obtained from sources reliable to Directorate General of Training, New Delhi. NIMI disclaims all warranties to the accuracy, completeness or adequacy of such information. NIMI shall have no liability for errors, omissions, or inadequacies in the information contained herein, or for interpretations thereof. Every effort has been made to trace the owners of the copyright material included in the book. The publishers would be greatfull for any omissions brought to their notice for acknowledgements in future editions of the book. No entity in NIMI shall be responsible for any loss whatsoever, sustained by any person who relies on this material. The material in this publication is copyrighted. No parts of this publication may be reproduced, stored or distributed in any form or by any means either on paper or electronic media, unless authorized by NIMI.

A Comprehensive Training Program
under Crafts Instructor Training Scheme (CITS)
for Instructors

**HANDBOOK ON
TECHNICAL INSTRUCTOR TRAINING
MODULES**

© NIMI
NOT TO BE REPUBLISHED

अतुल कुमार तिवारी, I.A.S.
सचिव
ATUL KUMAR TIWARI, I.A.S.
Secretary



भारत सरकार
कौशल विकास एवं उद्यमिता मंत्रालय
GOVERNMENT OF INDIA
MINISTRY OF SKILL DEVELOPMENT
AND ENTREPRENEURSHIP



Foreword

In today's rapidly evolving world, the role of skilled craftsmen and women is more crucial than ever. The Craft Instructor Training Scheme (CITS) stands at the forefront of this transformation, shaping the educators who will train the next generation of artisans and technicians. This book aims to provide an in-depth understanding of the subject, exploring its significance, methodologies, and impact on vocational training.

The Craft Instructor Training Scheme was established with the objective of enhancing the quality of instruction in industrial training institutes and other vocational training institutions. By equipping instructors with advanced skills and knowledge, the scheme ensures that they are well-prepared to impart high-quality training to their students. This, in turn, contributes to the creation of a highly skilled workforce capable of meeting the demands of modern industry.

The initial chapters provide the importance of specialized instructor training. Following this, detailed chapters delve into the curriculum covering advanced techniques, safety protocols, and instructional strategies. Each section is designed to offer both theoretical insights and practical applications, ensuring a well-rounded understanding of the subject.

The book offers recommendations for overcoming obstacles and enhancing the effectiveness of the program, with the ultimate goal of producing highly skilled instructors capable of shaping the future workforce.

This book is intended for a diverse audience, including current and aspiring instructors, vocational training administrators, policymakers, and industry stakeholders. It serves as a valuable resource for understanding the intricacies of the subject and its pivotal role in vocational education.

I extend my heartfelt gratitude to all contributors who have shared their experiences and expertise, enriching this book with their valuable insights. Special thanks to the contribution of the development team, reviewers and NIMI that have supported this endeavor, providing essential data and resources.

It is my sincere hope that this book will inspire and guide readers in their efforts to enhance vocational training, ultimately contributing to the development of a skilled and competent workforce.

ATUL KUMAR TIWARI, I.A.S.
Secretary, MSDE

त्रिशलजीत सेठी
महानिदेशक

Trishaljit Sethi, IPoS
Director General



भारत सरकार
कौशल विकास एवं उद्यमशीलता मंत्रालय
प्रशिक्षण महानिदेशालय
GOVERNMENT OF INDIA
MINISTRY OF SKILL DEVELOPMENT &
ENTREPRENEURSHIP
DIRECTORATE GENERAL OF TRAINING

FOREWORD

The Craftsmen Training Scheme (CTS) implemented by the Directorate General of Training (DGT) provides skill training to the youth and ensures a steady flow of skilled manpower for the industry. It aims to raise quantitatively and qualitatively the industrial production by systematic training, and to reduce unemployment among the youth by providing them with employable skills.

The Craft Instructor Training Scheme (CITS) is an indispensable part of the Craftsmen Training Scheme (CTS). It offers comprehensive training both in 'skills' and in 'training methodology' to the instructor trainees to make them conversant with techniques of transferring hands-on skills.

I congratulate NIMI for taking the initiative of preparation of the course content for CITS. This will help institutionalize the mechanism for imparting training to the trainers all across the ecosystem. I also extend my gratitude to the Instructors and Officials of National Skill Training Institutes (NSTIs) and the DGT for their invaluable contribution in preparation of the CITS course content.

As we navigate the complexities of a rapidly changing world and the technological disruptions, the significance of CTS and CITS has increased manifold. It not only empowers individuals with practical skills but also lays the foundation for a prosperous future. I am confident that this book will serve as a guiding light to all instructor trainees for skill development and nation-building.


(Trishaljit Sethi)

PREFACE

The Craft Instructor Training Scheme is an indispensable module of the Craftsmen Training Scheme, which has been an integral part of the Indian skill development industry since its inception. This program aims to equip instructors with the necessary skills and teaching methodology to effectively transfer hands-on skills to trainees and promote a holistic learning experience. The first Craft Instructor Training Institute was established in 1948, followed by six more institutes across India in 1960. Today, these institutes, including the National Skill Training Institute (formerly Central Training Institute for Instructors), offer the CITS course, which is mandated by the Directorate General of Training (DGT).

The Craft Instructor training program is designed to develop skilled manpower for industries. The course aims to offer instructors an opportunity to improve their instructional skills, engage learners effectively, offer impactful mentoring, and make efficient use of resources, leading to a more skilled workforce in various industries. The program emphasizes collaborative and innovative approaches to teaching, resulting in high-quality course delivery. Overall, the Craft Instructor Training Scheme is a pivotal program that helps instructors grow in their careers and make a significant contribution to society. This program is essential for developing skilled manpower and promoting a robust learning environment that benefits both trainees and instructors alike.

ACKNOWLEDGEMENT

National Instructional Media Institute (NIMI) sincerely acknowledges with thanks for the co-operation and contribution extended by the following experts to bring out this Instructional material **(Trade Practical)** for **CITS Computer Software Application (Volume - I of II)** **(NSQF Level - 4)** under the **IT & ITES** Sector for **Instructors**.

MEDIA DEVELOPMENT COMMITTEE MEMBERS

Smt. P.A. Sulabha	-	Training Officer, NSTI (W), Trivandrum, Kerala.
Shri. M.K. Sunil	-	Training Officer, NSTI, Chennai.

COORDINATORS

Shri. G.C. Ramamurthy	-	Joint Director, CD - Section, DGT.
Shri. T.V. Rajasekar	-	Joint Director, NIMI, Chennai.
Shri. Shiv Kumar	-	Training Officer, CD - Section, DGT.

NIMI records its appreciation of the Data Entry, CAD, DTP Operators for their excellent and devoted services in the process of development of this Instructional Material.

NIMI also acknowledges with thanks, the invaluable efforts rendered by all other staff who have contributed for the development of this Instructional Material.

NIMI is grateful to all others who have directly or indirectly helped in developing this IMP.

ABOUT THE TEXT BOOK

The Vocational Instructor Training Program is a comprehensive initiative designed to equip aspiring students with the necessary skills and knowledge to effectively teach in vocational education settings. This program encompasses a range of pedagogical strategies, instructional techniques, and subject-specific content tailored to the diverse vocational fields. Participants engage in coursework that covers curriculum development, assessment methods, classroom management, and the integration of industry-relevant technologies. Practical experience and hands-on training are emphasized, allowing participants to apply theoretical concepts in real-world teaching environments. Through collaborative learning experiences and mentorship opportunities, aspiring vocational instructors develop the confidence and competence to facilitate engaging and impactful learning experiences for their students. This training program aims to cultivate a new generation of educators who are not only proficient in their respective vocational fields but also adept at fostering the success and employability of their students in today's competitive workforce.

This text book covers communication, self-management, information and communication technology, entrepreneurial and green skills. It has been developed as per the learning outcome-based curriculum.

**G C Rama Murthy,
Joint Director,
Curriculum Development, DGT,
MSDE, New Delhi.**

CONTENT

Ex. No.	Table of Contents	Page No.
	Module 1: Network Architecture	
1	Straight cabling and cross cabling	1
2	Switch Configuration	3
3	LAN and WAN setup	6
4	Setting TCP/IP	10
5	Network Monitoring and Control	12
6	Wireless Networking Design	15
7	Implementing Voice over IP	19
8	Configuring DHCP, IPV4/IPV6	21
9	Configuring Network Security for preventing Attacks	30
10	Setting password policy	32
11	Sniffing on Switched Networks	33
12	IP Address Spoofing	35
13	DNS Spoofing	36
14	Password Cracking: Dictionary vs Brute- Force vs Hybrid methods	39
15	Handling Denial-of-Service (DoS)	41
16	Using Tools like John the Ripper, Cain & Abeletc	42
17	Configuring Firewalls	46
	Module 2: Database Concepts	
18	Installation Steps of MySQL	49
19	Troubleshooting Basic Installation Issues	58
20	Creating and Using a Database in MySQL	59
21	Designing of Tables	64
22	Applying Data Integrity Rules	69
23	Using the DDL, DCL, DML statements	72
24	Enforcing Constraints, Primary Key and Foreign Key	77
25	Adding indices to tables	79
26	Simple Select Queries	82
27	Insert, Update and Delete Queries	84
28	Using the Number, Date and Character Functions	89
29	Joins, Group by, Having, Sub query	93
30	Indexing and Optimizing query	101
31	Creating & Using Stored procedure	103
32	Creating and executing MySQL table level Triggers	105

Ex. No.	Table of Contents	Page No.
33	Creating Cursors in MYSQL	107
34	Using Cursors in MySQL	110
35	Implementing MySQL Security	113
36	Simple application on database using SP, Triggers, Cursors & Indexing	115
	Module 3: Introduction to Java Script	
37	Using the JAVA Script Syntax	118
38	Using Variables, Operators and Writing Expressions	120
39	Programming with Control Flow statements	130
40	Creating and using Objects in JavaScript	141
41	Creating and using Functions	146
42	Using Java Script with Forms	151
43	Creating Cookies with JavaScript	155
44	Creating CSS	159
45	Error Handling in JavaScript	169
46	Implementing an AJAX application	173
	Module 4: PHP (Hyper text pre processor)	
47	Demonstrate on- Paginators, popovers, progress, spinner	175
48	Demonstrate on Table, toasts, tooltips	189
49	Demonstrate on Bootstrap Styling essentials like Breakpoints for components, layouts and grid systems. Practice on typography, floats, flex, alignment, borders, position of elements, shadow and visibility	198
50	Perform Installation of Apache Web Server Practice simple PHP programs. Practicing on programming to test events	213
51	Demonstrate on if statement Using the else clause with if statement, switch statement Using the? operator, while statement, do while statement, for statement, Breaking out of loops, Nesting loops	226
52	Demonstrate on Function and returning value from function, user defined functions, dynamic functions, variable scope, accessing variable with the global statement, Function calls with the static statement, setting default values for arguments, Passing arguments to a function by value, Passing arguments to a function by reference, Testing for function existence	235
53	Demonstrate the Writing to the browser, Getting input from forms, Output buffering, Session handling, Regular expression, Common math, Random numbers, File upload, File download, Environment variables	244
54	Demonstrate on E-mail in PHP, anatomy of a cookie, setting a cookie with PHP, deleting a cookie, creating session cookie, working with the query string, creating query string, starting a session, Working with session, variables, Destroying session, passing session IDs, Encoding and decoding session variables	255

Ex. No.	Table of Contents	Page No.
55	Demonstrate Creating and deleting a file, Reading and writing text files, working with directories in PHP, checking for existence of file, determining file size, opening a file for writing, reading, or appending, Writing Data to the file Reading characters	269
56	Work With Forms - Super global variables the server array A script to acquire user input, importing user input Accessing user input, Combine HTML and PHP code., using hidden fields	283
57	Demonstrate redirecting the user, File upload and scripts.	291
58	Demonstrate Regular Expressions - The basic regular expressions, PCRE, Matching patterns, Finding matches, Replace patterns, Modifiers, Breakup Strings.	294
59	Demonstrate working with Classes And Objects - Creating an object, Object properties, Object methods, Object constructors and destructors.	311
60	Perform class constants, Class inheritance Abstract classes and methods, Object serialization, checking for class and method, existence, Exceptions, Iterators	315
61	Creating and using Functions	329
62	Using Java Script with Forms	343
	Module 5: Advanced Excel	
63	Protect sheet using password	345
64	Use flash fill techniques	353
65	Perform Goal Seek, Solver & Scenarios on Data	358
66	Use different types of cell references	380
67	Use R1C1 notation	388
68	Use array formula	397
69	Audit excel formula	405
70	Create and modify simple macros	414
71	Perform form controls and create simple data entry form with macros	425
72	Look up data by using functions	433
73	Use advanced date functions	438
74	Demonstrate advanced charts	444
75	Demonstrate pivot tables	455
76	Demonstrate a power query, power query function. Invoking the power query function and combining queries. Organize the workbook queries	473
77	Demonstrate Power BI for simple data visualizations	499

◆ Module 1 : Network Architecture ◆

EXERCISE 1 : Straight cabling and cross cabling

Objectives

At the end of this exercise you shall be able to

- crimp a straight through ethernet cable
- crimp a crossover through ethernet cable.

Requirements

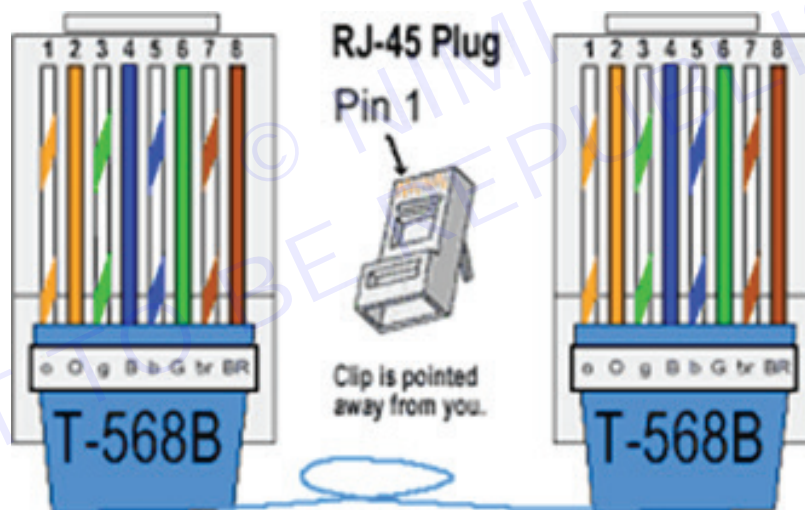
Tools/Materials

- Ethernet cable
- RJ 45 connector
- Crimping tool
- Wire stripper/cutter
- LAN Tester
- PCs/Laptops

Procedure

TASK 1: Crimping a straight through ethernet cable

Straight through ethernet cables are the standard cable used for almost all purposes, and are often called "patch cables". It is highly recommend you duplicate the color order as shown on the below.

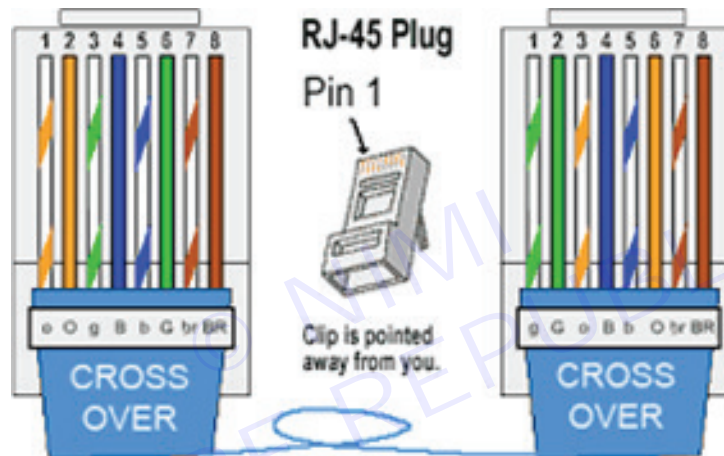


- 1 Take the Cat6 Cable of 1 meter.
- 2 Strip 1 to 2 inches (2.5 to 5.1 cm) of the outer skin at the end of the cable wire by making a shallow cut in the skin with a crimping Tool.
- 3 Unwind and pair the similar colours.
- 4 Fold each pair of wires backwards to expose the core of the cable.
- 5 Straighten the twisted wires.
- 6 Arrange the untwisted wires in a row, placing them into the position, running from right to left, in which they will go into the RJ-45 connector:
 - a Orange with a white stripe
 - b Orange
 - c Green with a white stripe
 - d Blue

- e Blue with a white stripe
 - f Green
 - g Brown with a white stripe
 - h Brown
- 7 Trim the untwisted wires to a suitable length by holding the RJ-45 connector next to the wires.
 - 8 Insert the wires into the RJ-45 connector, making sure that they stay aligned and each color goes into its appropriate channel.
 - 9 Use the crimping tool to crimp the RJ-45 connector to the cable by pressing the jacket and cable into the connector so that the wedge at the bottom of the connector is pressed into the jacket.
 - 10 Follow the instructions above to crimp an RJ-45 connector to the opposite end of the cable.
 - 11 Use a cable tester to assure that your cable is working properly when both ends are crimped.

TASK 2: Crimping a Crossover Through Ethernet Cable

Crossover cables - The purpose of a Crossover Ethernet cable is to directly connect one computer to another computer (or device) without going through a router, switch or hub.



- 1 Crimp one end following the steps in Task 1
- 2 For the next end Follow the similar steps for preparation and then
- 3 Arrange the untwisted wires in a row, placing them into the position, running from right to left, in which they will go into the RJ-45 connector:
 - a Green with a white stripe
 - b Green
 - c Orange with a white strip
 - d Blue
 - e Blue with a white strip
 - f Orange
 - g Brown with a white strip
 - h Brown
- 4 Trim the untwisted wires to a suitable length by holding the RJ-45 connector next to the wires.
- 5 Insert the wires into the RJ-45 connector, making sure that they stay aligned and each color goes into its appropriate channel.
- 6 Use the crimping tool to crimp the RJ-45 connector to the cable by pressing the jacket and cable into the connector so that the wedge at the bottom of the connector is pressed into the jacket.
- 7 Use a cable tester to assure that your cable is working properly when both ends are crimped.

EXERCISE 2 : Switch Configuration

Objectives

At the end of this exercise you shall be able to

- use cisco packet tracee application
- configure a switch by using cisco packet tracer.

Requirements

Tools/Materials

- PC/Laptop
- Cisco pocket tracee software

Procedure

TASK 1: Configure the Switch

Step 1: Open the packet tracer desktop and take a switch (PT-Switch) from the devices.

Step 2: Configure the Host name of the swith0.

- Click on switch0 and go to Command Line Interface.
- Then change the hostname to “sh”

Command:

```
switch>
switch>en
switch#conf t
switch(config)#hostname sh
sh(config)#exit
```

Step 3: Set a message of the day (MOTD) banner for the users.

Command:

```
sh(config)#banner motd $
```

- Then, enter MOTD and end it with '\$' to exit.

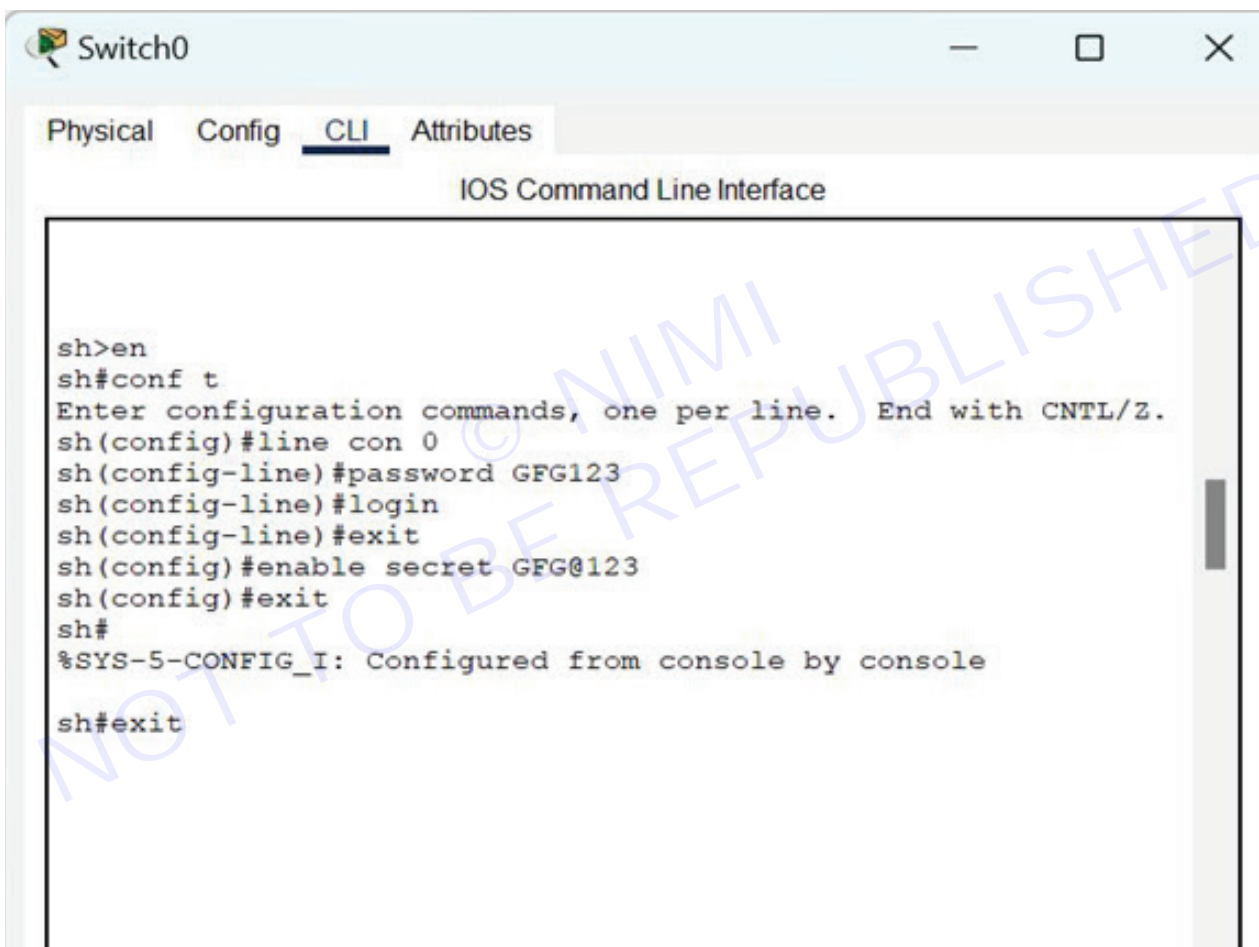


```
Switch0
Physical Config CLI Attributes
IOS Command Line Interface
sh#en
sh#conf t
Enter configuration commands, one per line. End with CNTL/Z.
sh(config)#banner motd?
motd
sh(config)#banner motd $
Enter TEXT message. End with the character '$'.
!!! only Authorized users are allowed!!!
```

Step 4: Set up line control password and enable secret password.

To configure the Line Control password and Enable secret follow the below commands:

```
sh#conf t
sh(config)#
sh(config)#line con 0
sh(config-line)#password GFG123
sh(config-line)#login
sh(config-line)#exit
sh(config)#enable secret GFG@123
sh(config)#exit
```

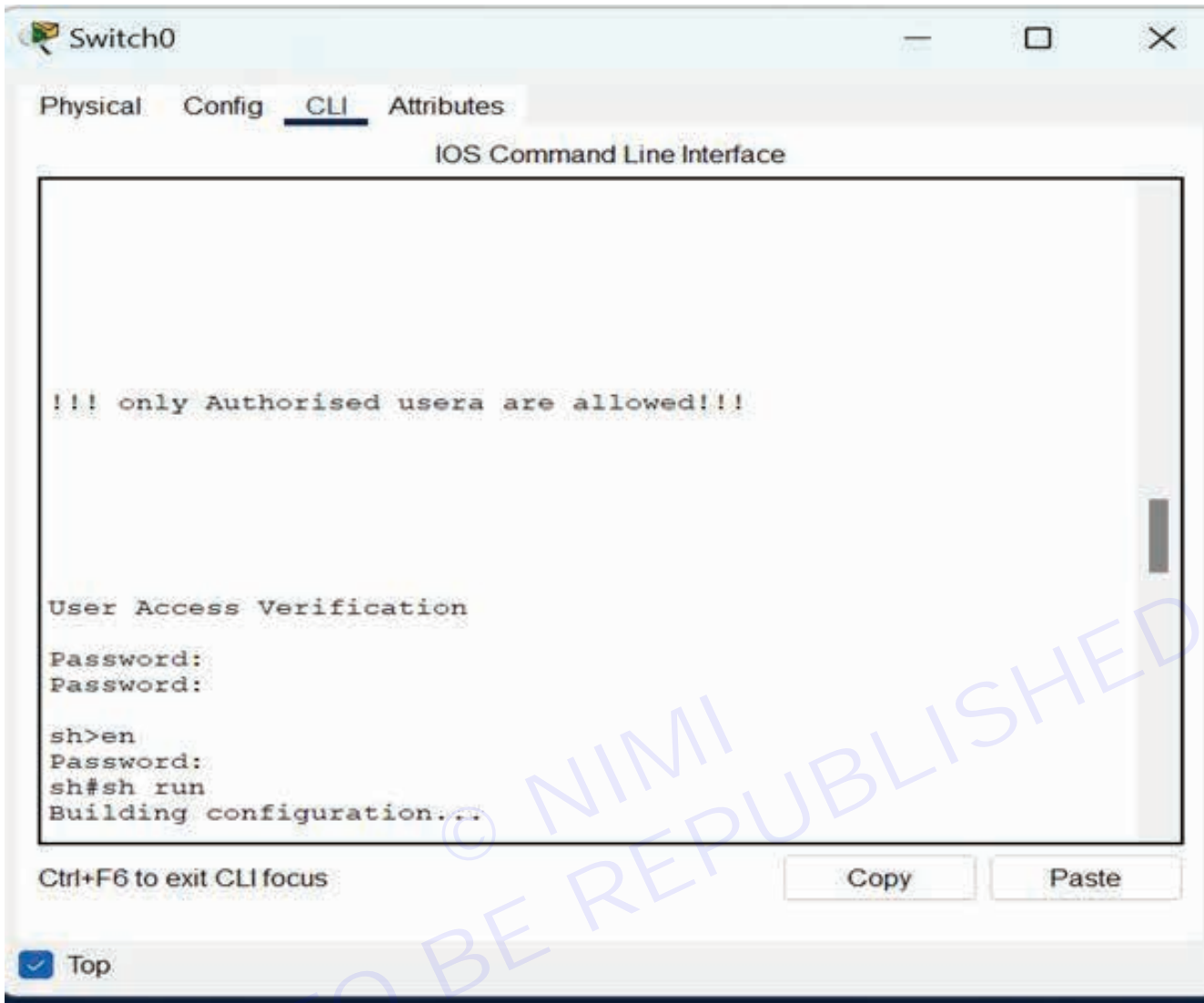


The screenshot shows a terminal window titled "Switch0" with tabs for "Physical", "Config", "CLI", and "Attributes". The "CLI" tab is active, displaying the "IOS Command Line Interface". The terminal output shows the following commands and responses:

```
sh>en
sh#conf t
Enter configuration commands, one per line. End with CNTL/Z.
sh(config)#line con 0
sh(config-line)#password GFG123
sh(config-line)#login
sh(config-line)#exit
sh(config)#enable secret GFG@123
sh(config)#exit
sh#
%SYS-5-CONFIG_I: Configured from console by console
sh#exit
```

Step 5: Verify the password

- When you try to log in first, it will ask for the line control password.
- Then, to configure the terminal it will ask to enable a secret password.



The screenshot shows a window titled "Switch0" with tabs for "Physical", "Config", "CLI", and "Attributes". The "CLI" tab is active, displaying the "IOS Command Line Interface". The interface shows a warning message: "!!! only Authorised users are allowed!!!". Below this, it prompts for "User Access Verification" with two "Password:" fields. The user has entered "sh" and "en", and the system has responded with "sh#sh run" and "Building configuration...". At the bottom of the CLI window, there is a "Ctrl+F6 to exit CLI focus" message and "Copy" and "Paste" buttons. A "Top" button is also visible at the bottom left of the window.

```
Switch0
Physical Config CLI Attributes
IOS Command Line Interface

!!! only Authorised users are allowed!!!

User Access Verification
Password:
Password:

sh>en
Password:
sh#sh run
Building configuration...
```

To save the configuration use the below command:

Command:

```
sh#copy run startup-config
```

EXERCISE 3 : LAN and WAN setup

Objectives

At the end of this exercise you shall be able to

- create a LAN and check connectivity by between two computer
- share a file, folder and a printer over a network
- configure a router and set up a WAN.

Requirements

Tools/Materials

- 2 or more PC/Laptop connected to a network
- Printer
- Router

Procedure

Connecting two computers using LAN

TASK 1: Creating a LAN

- 1 Determine whether or not your computers have Ethernet ports.
- 2 Check to see if you have a crossover Ethernet cable.
- 3 Plug one end of the Ethernet cable in to one computer.
- 4 Plug the other end of the Ethernet cable in to the other computer.
- 5 Finish.

TASK 2: Checking the IP address of a given computer

- 1 Open the command prompt. press windows + R and type cmd into the field. Press Enter to open the command prompt.
- 2 Type IP config and press Enter. This will display a list of your network connection information. All of the network connections on your computer will be displayed.
- 3 Find your IP address your active connection may be labelled wireless network connection, Ethernet adapter, or Local Area Connection. It may also be labelled by the manufacturer of your network adapter. Find your active connection and look for ipv4 address.
 - The IP address is four sets of digits, with up to three digits per set. For example, it might look like 192.168.1.
- 4 Finish.

TASK 3: Check connectivity between two computers

- 1 Open the command prompt. Press windows + R and type cmd into the field. Press Enter to open the command prompt.
- 2 Type the word "ping" Followed by a space and then your IP address at the DOS prompt (e.g.ping 111.22.33.4). Press the "Enter" key once.
- 3 View the result of the ping. If a result of "request timed out" appears, then there is a problem with either the computer or the network. A successful ping will result in a response of "Reply from" followed by the IP address.
- 4 Repeat the above steps in the other computer
- 5 Finish.

TASK 4: Sharing a File or Folder over a Network

- 1 Switch on the system.
- 2 Open control panel and click network and internet. Click network and sharing Centre.
 - a Click change advanced sharing settings to the left of your network name.
 - b Find the folder you wish to share. Right click on it.
- 3 Select the “share with” option. This will open the sharing submenu. You can choose to share it with everyone in your Home group or select specific people to share it with.
 - a When choosing a Home group option, you can allow other Home group members to both read and write to the folder, or limit them to just read from it.
- 4 Click the specific people option to select which user you want to share with. This will open a new window with a list of all the users that currently have access to the folder. You can add users to this list and give them specific permissions for the folder.
 - a To share the folder with everyone, click the dropdown menu at the top and select everyone. Click the add button.
 - b To share with specific users, click the dropdown menu and select them or type in the name and click add.
- 5 Set permissions for users on the list. Find a user on the list that you want to change the permissions for. Look in the permissions level column, and click the arrow next to the existing permission. Select the new one from the list.
 - a Read – user can see, copy, and open files from the folder, but cannot change files or add new ones.
 - b Read/write – besides read abilities, users can change files and add new files to the shared folder. Files can be deleted by users with read/write permissions.
 - c Remove- removes permission for this user, and removes them from the list.
- 6 Click the share button. Your permission settings will be saved, and the folder will be available on the network for all allowed users.
- 7 Finish.

TASK 5: Opening a shared file or folder over a network

- 1 Start the computer.
- 2 Make sure you are on the correct network. In order to open a shared folder from a different computer, you must be using the same internet network as the computer which is sharing the file.
- 3 Open start click the windows logo in the bottom-left corner of the screen. The start menu will pop up.
- 4 Open settings click the settings gear in the lower-left corner of the start menu. This will open the settings window.
- 5 Click network and internet. This is in the middle of the settings window.
- 6 Click the network and sharing center. It is a link toward the bottom of the page.
 - a You may need to scroll down to see this link. If you still cannot find the link after scrolling down, click the status tab in the upper left corner of the window and look again.
- 7 Click change advanced sharing settings. This is in the upper-left side of the window. A new window will open.
- 8 Turn on network discovery and file sharing. Check both the turn on network discovery box and the turn on file and printer sharing box.
- 9 Click save changes. It is at the bottom of the window. Doing so saves your settings.
- 10 Open file explorer click the file explorer app icon at the bottom of the screen, or right click the start icon and then click file explorer in the resulting pop-up menu.
 - a You can also press Win+E to open the file explorer.

- 11 Click network. It's is near the bottom of the sidebar that is on the left of the file explorer window.
 - a You may have to scroll down in the file explorer's left-hand sidebar to see this option.
- 12 Select a computer. Double-click the name of the computer from which the folder you want to open is being shared.
- 13 Select a folder. Double-click the folder you want to open is being shared.
- 14 Enter a username and password if prompted. This will usually be the username and password used to log into the computer which is sharing the folder. Doing so correctly will cause the folder to open.
 - a If a folder isn't protected, double clicking it will open it immediately.
- 15 Finish.

TASK 6: Sharing a Printer over a Network

- 1 Switch on the system.
- 2 Click on the Windows start button and navigate to setting, control panel, printers. Right click on the printer to be shared.
- 3 Select "change sharing options" if network and print sharing has not already been enabled. Follow the prompts to allow sharing.
- 4 Check the button next to "share this printer". Enter a share name for the printer. This is the name that other users on the network will see when searching for printers.
 - a Limit the name to 8 letters with no characters or spaces.
- 5 Select "additional drivers" if there are other computers on the network with older Windows operating systems. Follow the promotes to install drivers for these computers.
 - a This will save time since the drivers won't have to be downloaded and installed separately on the other computers.
- 6 Finish.

TASK 7: Operating a Shared File or Folder over a Network.

- 1 Start the computer.
- 2 Follow the same steps to get to the printers setting on the other computer.
- 3 Right click" add printer" and select "network printer". Allow Windows to search the network for printers.
 - a If the printer isn't found automatically, select "the printer I want is not listed. Select "browse for printers" and find the computer that attached to the USB printer. Click the plus sign to expand it, and then select the printer.
- 4 Finish

TASK 8: Sharing a Printer over a Network

- 1 Turn on your printer into an electrical outlet if necessary, then press the Printers "Power" button to turn on the printer if it isn't already on.
- 2 Connect your printer to your network. The process for this varies from printer to printer. Most modern printers can connect your network via WI-FI. Some printers may be able to connect via Ethernet cable, though this requires that they be close enough to the router for Ethernet to be viable.
 - a If your printer is WI-FI capable, you can generally connect it to the network using the built-in menu display. Refer to your printer's documentation or look up the model online for exact instructions.
 - b Make sure your Wi-Fi printer is close enough to the router to get a solid signal.
- 3 Connect to the printer (Windows). Now that the printer is your network, you can use the Windows "Add a Printer" wizard to automatically install the necessary software on your on your Computer to use it. If you are using OS X, Skip down to the next step.[1]



- a) Click the start menu and select Control Panel. Windows 8 users can press Win and type “Control panel”.
 - b) Select “Devices and Printers” or “View Devices and Printers”.
 - c) Click Add a printer at the top of the window.
 - d) Select “Add network, Bluetooth printer”.
 - e) Select your network printer from the list and click Next.
 - f) Install the necessary drivers if prompted. Windows should be able to find and install the correct drivers for most printers.
- 4 Printer to the network printer. Once you’ve added to the printer to the operating system, you can print it just as you would a printer connected directly to your computer. Simply select the printer from the “Print” window of any program.
- a) Make sure that the printer is turned and that you are connected to the same network. That will save time since the drivers won’t have to be downloaded and installed separately on the other computers.
- 5 Finish.

TASK 9: Setup a WAN

1 Select WAN Connection Type

Choose the appropriate WAN connection type based on your needs.

2 Sign Up with an ISP

Select an ISP and sign up for a WAN connection plan.

3 Install Modem or Router

Install a modem or router provided by your ISP or purchase one separately.

Connect the cables:

- 1 Once you’ve acquired a wireless router, you’ll need to connect it to your existing Internet modem.
- 2 Connect an Ethernet cable from your modem to the wireless router (there is usually a short Ethernet cable included with your wireless router for this purpose).
- 3 Plug in the power cable for the wireless router.
- 4 Wait at least 30 to 60 seconds, and make sure the lights on your router are working correctly.

Configure your router:

- 1 Using your web browser, enter the router’s default IP address into the address bar and then press Enter. Your router’s instructions should include this information, but some of the most common addresses include 192.168.0.1, 192.168.1.1, and 192.168.2.1.
- 2 The router’s sign-in page will appear. Again, the exact sign-in details should be included with your router’s instructions, but most routers use a standard user name and password combination, such as admin and password.
- 3 Your router’s settings page will appear. Locate and select the Network Name setting, then enter a unique network name.
- 4 Locate and select the Network Password setting, and choose an Encryption option. There are several types of encryption you can use, but we recommend WPA2, which is generally considered to be the most secure.
- 5 Enter your desired password. Make sure to use a strong password to help ensure no one else can access your network.
- 6 Locate and select the Save button to save your settings.

EXERCISE 4 : Setting TCP/IP

Objectives

At the end of this exercise you shall be able to

- assign IP address to a PC
- setup TCP/IP connection

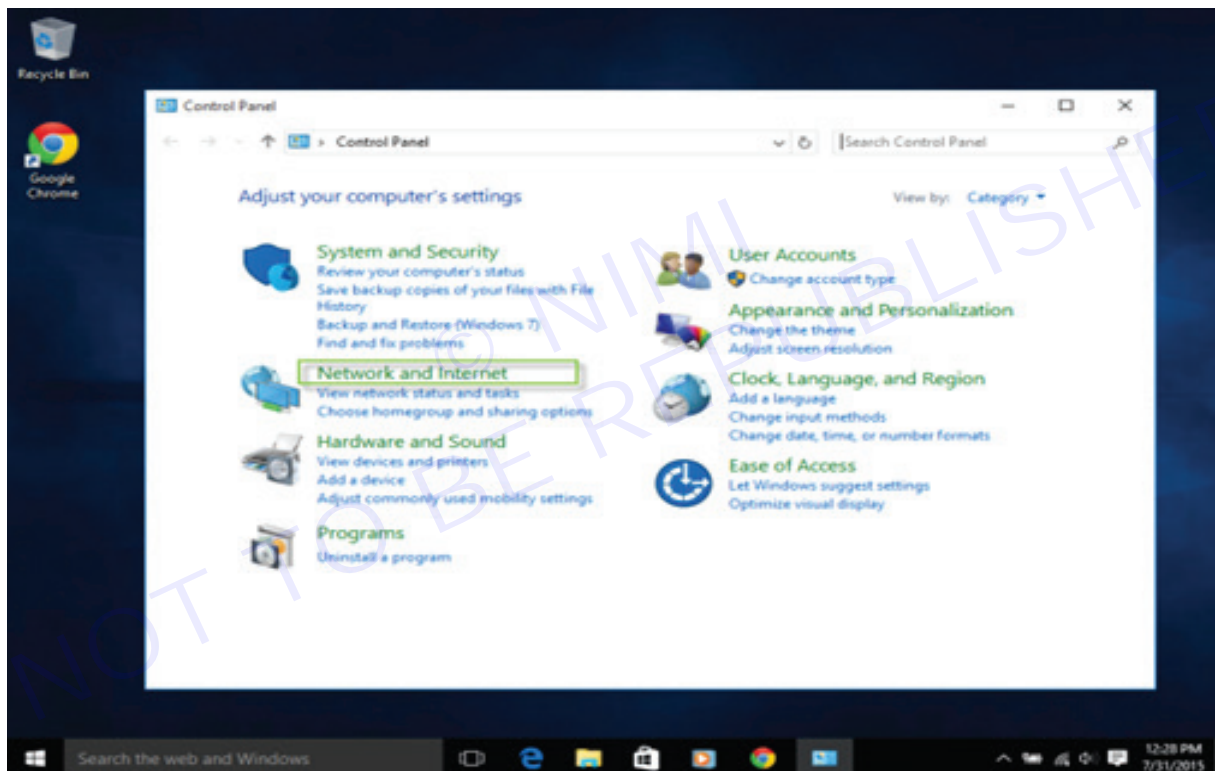
Requirements

Tools/Materials

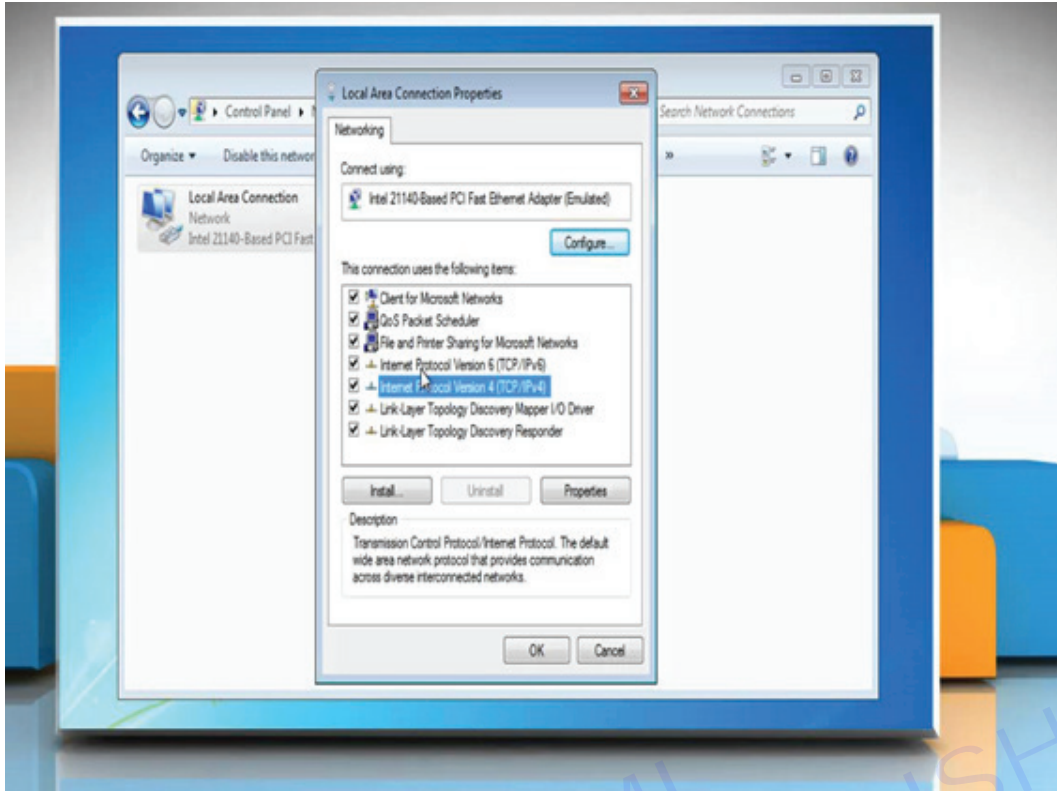
- Windows PC/Laptop connected to a network

Procedure

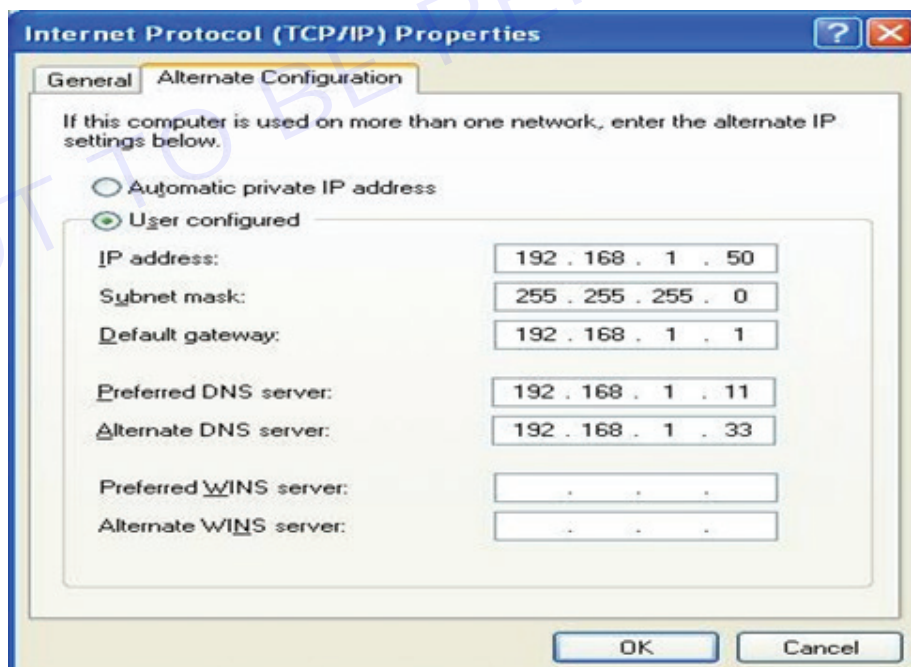
- 1 Select Start, then select Settings > Network & Internet .



- 2 Do one of the following:
 - Choose the network you want to change the settings for, then select Properties.
 - For an Ethernet network, select Ethernet, then select the Ethernet network you're connected to.
- 3 Under IP assignment, select Edit.
- 4 Under Edit IP settings, select Manual.
- 5 To specify IPv4 settings manually
 - i Under Edit IP settings, choose Manual, then turn on IPv4.
 - ii To specify an IP address, in the IP address, Subnet prefix length, and Gateway boxes, type the IP address settings.
 - iii To specify a DNS server address, in the Preferred DNS and Alternate DNS boxes, type the addresses of the primary and secondary DNS servers.



- When you select Automatic (DHCP), the IP address settings and DNS server address setting are set automatically by your router or other access point (recommended).
- When you select Manual, you can manually set your IP address settings and DNS server address.



- 6 When you're done, select OK/Save.

EXERCISE 5 : Network Monitoring and Control

Objectives

At the end of this exercise you shall be able to

- install SNMP agent
- configure the SNMP service
- configure RMON.

Requirements

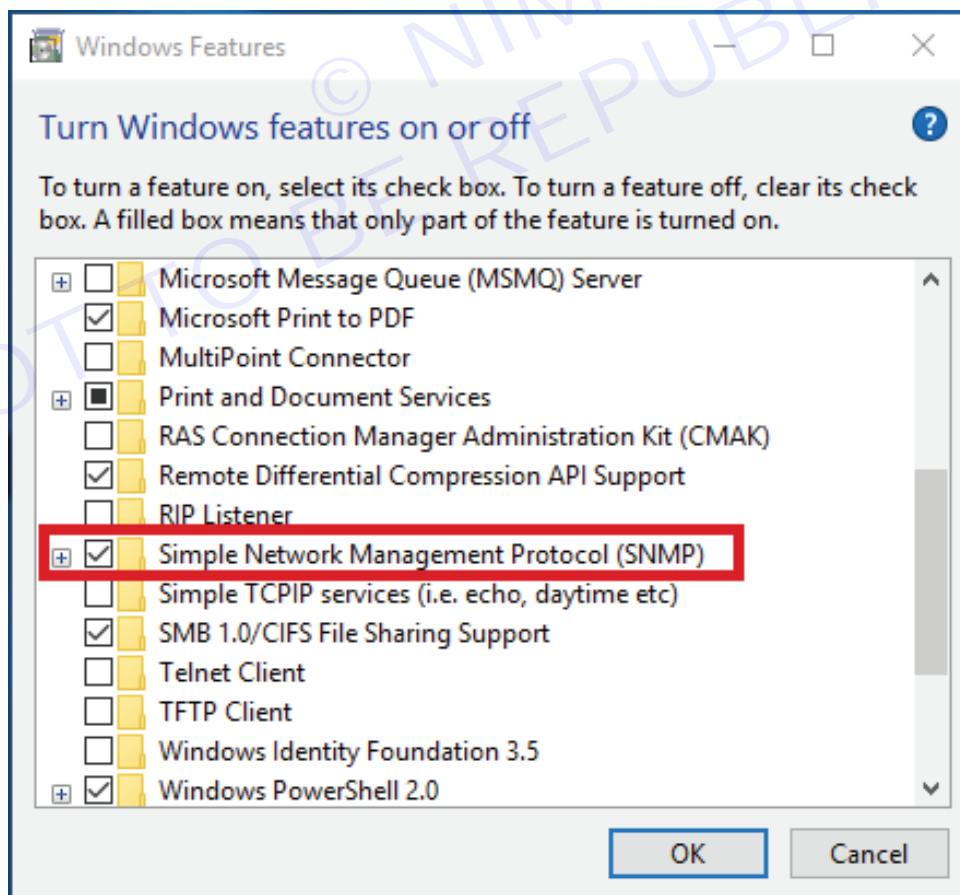
Tools/Materials

- Windows server
- PC/Laptop
- Network connection

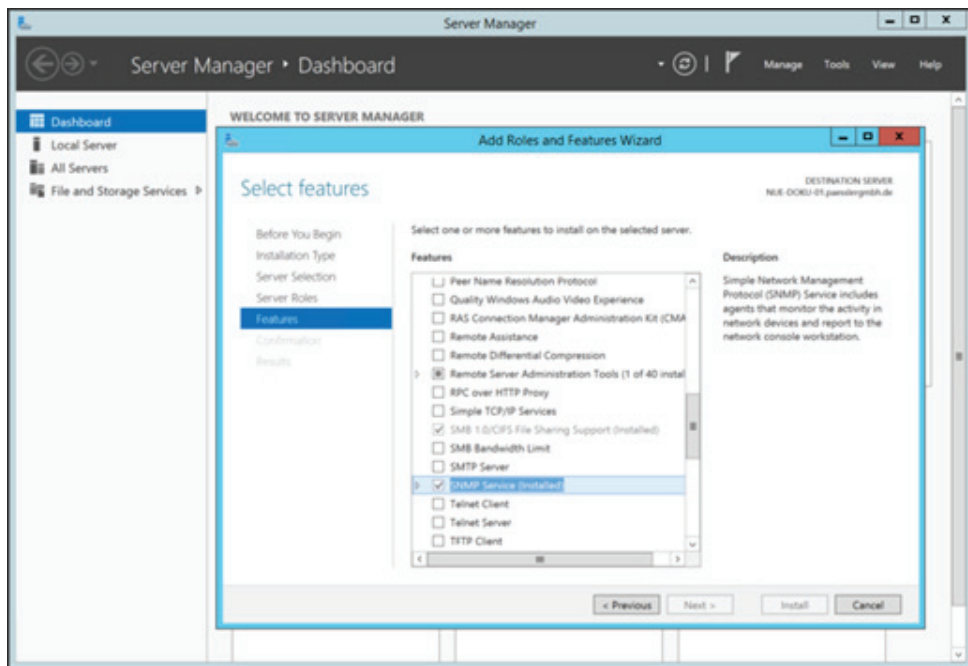
Procedure

TASK 1: Install SNMP Agent and configure the community string

- 1 Open the Control Panel on your Windows machine.
- 2 Open the Programs and Features.
- 3 Select Turn Windows features on or off.
- 4 On Windows workstations (Windows 10) select Simple Network Management Protocol (SNMP) and install it.



- 5 On Windows Server 2016 and above you'll have to click Next in the Add Roles and Features Wizard until you reach the Features sections where you can install the SNMP Service.

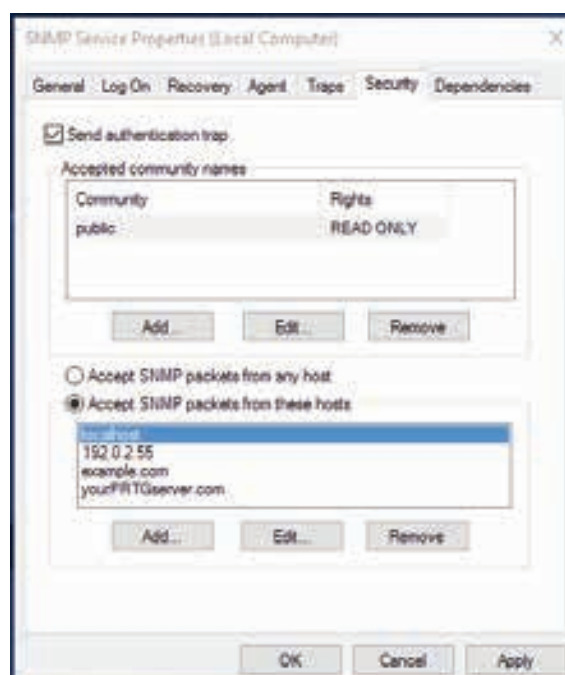


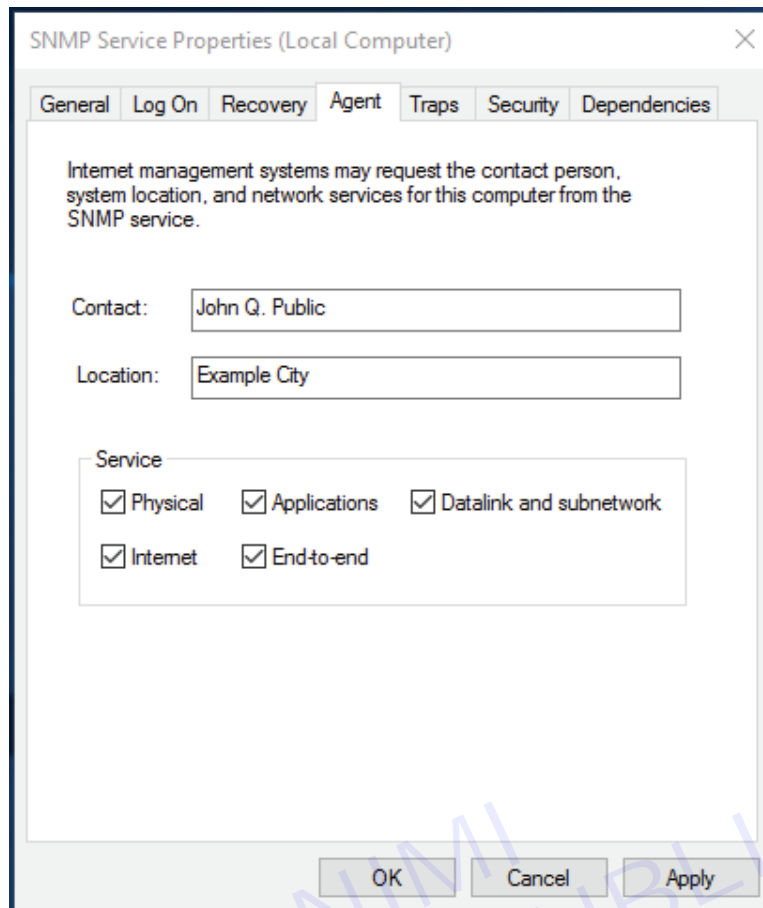
6 Install SNMP Service

Note : The system will automatically install SNMP Tools required by the service.

TASK 2: configure the SNMP service

- 1 Run services.msc as administrator.
- 2 Navigate to the Properties of the SNMP service.
- 3 Select Automatic as the Startup type. The service always runs, even after turning your computer off and on again.
- 4 For monitoring purposes, select all of the services on the Agent tab to have all SNMP properties available.
- 5 Click on the Security tab and adjust the security parameters like the community string and the IP/host filter list to your security compliance. For example, add the community string public with READ ONLY rights and accept SNMP packets from at least the address of your monitoring server.





TASK 3: Enable RMON on devices

- 1 Access the device's configuration interface.
- 2 Enable RMON functionality on interfaces or VLANs that need to be monitored.
- 3 Specify RMON groups and parameters based on monitoring requirements.

Configure RMON Probes (For RMON2):

- 1 Install RMON2 probe software on devices capable of running probe applications.
- 2 Configure probes to monitor specific network segments or traffic types.

Define RMON Groups and Alarms:

- 1 Determine which RMON groups (Hosts, Matrix, Filters, etc.) are needed for monitoring.
- 2 Configure RMON groups to collect and analyze relevant statistics (e.g., packet counts, errors, protocol distribution).
- 3 Set up alarms and thresholds for triggering notifications or actions based on predefined criteria.

Monitor and Analyze Data:

- 1 Use SNMP management tools or dedicated RMON management software to access and analyze collected data.
- 2 Monitor network performance, identify trends, and troubleshoot issues using RMON statistics and reports.

Fine-Tune Configuration:

- 1 Periodically review RMON configurations and adjust parameters as needed based on network changes or performance requirements.
- 2 Optimize RMON filters and alarms to focus on critical network metrics and reduce unnecessary overhead.

EXERCISE 6 : Wireless Networking Design

Objectives

At the end of this exercise you shall be able to

- design a wireless network

Requirements

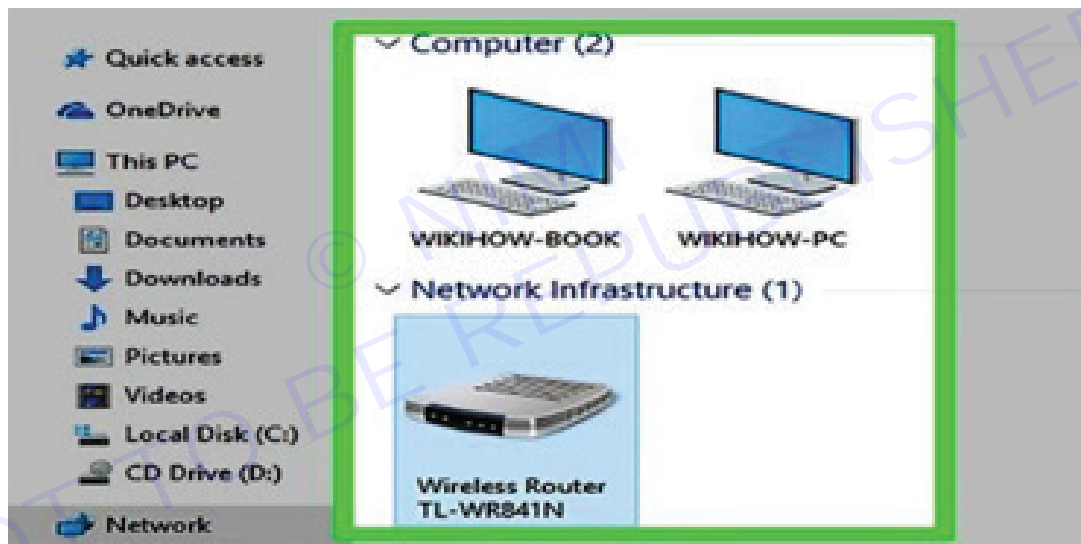
Tools/Materials

- Windows PC/Laptop
- Wireless router

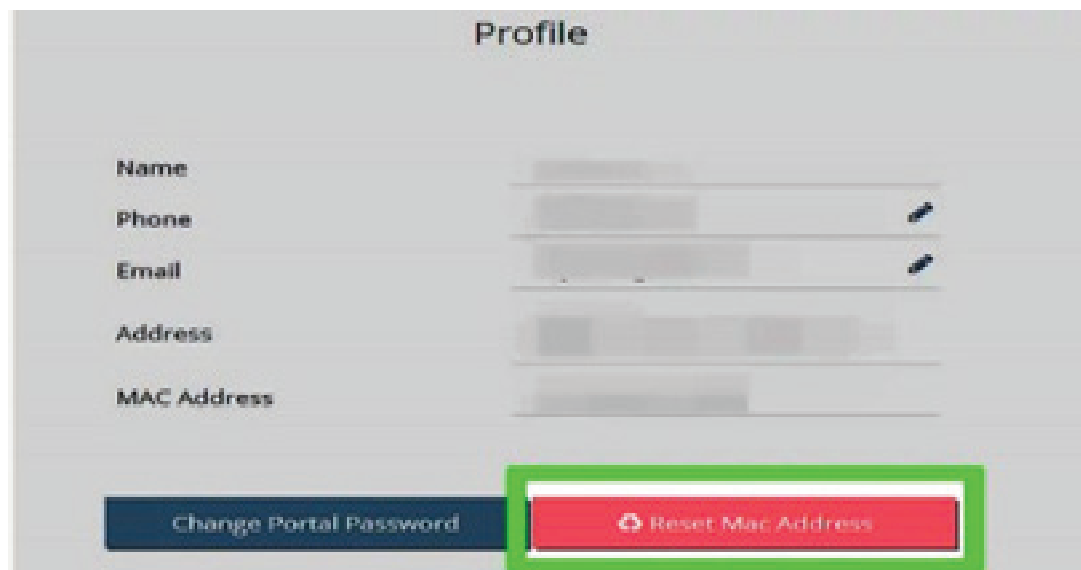
Procedure

TASK 1: Wireless network design

- 1 When you power on the router, it will only generate its wi-fi network, and the device will be connected to the router's wi-fi connection, not the internet. To connect the router to the internet need a MAC address to the internet service provider's website.



- 2 The MAC address will display already the old one need to Reset the MAC address



3 Enter the administrator name and password and click on login

A login form with a green border. It contains two input fields: the first is labeled 'administrator' and has a person icon; the second is a password field with a key icon and a series of dots. Below the fields is a green 'Login' button.

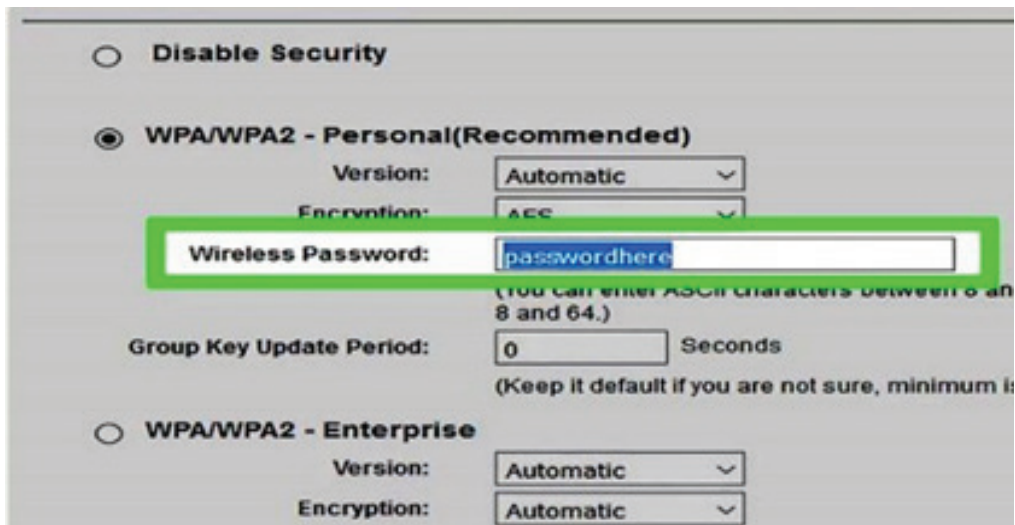
4 Select the wireless network name and select the enable the wireless router radio and SSID broadcast

Wireless Network Name: (also called the
 Region:
 Warning: Ensure you select a correct country to conform local law. Incorrect settings may cause interference.
 Mode:
 Channel Width:
 Channel:
 Enable Wireless Router Radio
 Enable SSID Broadcast
 Enable WDS Bridging

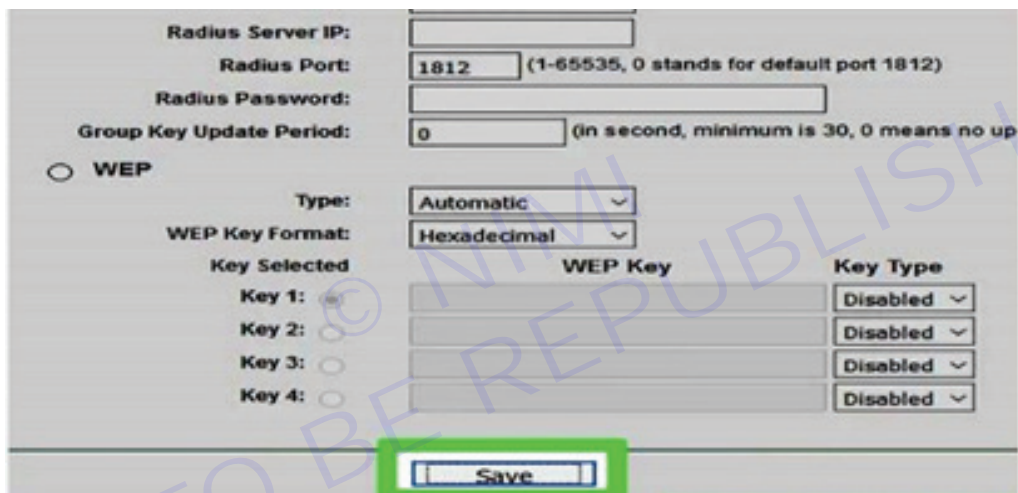
5 Select the WPA/WPA2 and fill all the fields.

Wireless Security
 Disable Security
 WPA/WPA2 - Personal(Recommended)
 Version:
 Encryption:
 Wireless Password:
 (You can enter ASCII characters between 8 and 63 or Hexadecimal characters between 8 and 64.)
 Group Key Update Period: Seconds
 (Keep it default if you are not sure, minimum is 30, 0 means no update)
 WPA/WPA2 - Enterprise
 Version:
 Encryption:
 Radius Server IP:
 Radius Port: (1-65535, 0 stands for default port 1812)

6 Enter the wireless password and click on the save button



Disable Security
 WPA/WPA2 - Personal(Recommended)
 Version:
 Encryption:
Wireless Password:
(You can enter ASCII characters between 8 and 64.)
 Group Key Update Period: Seconds
(Keep it default if you are not sure, minimum is 30)
 WPA/WPA2 - Enterprise
 Version:
 Encryption:



Radius Server IP:
 Radius Port: (1-65535, 0 stands for default port 1812)
 Radius Password:
 Group Key Update Period: (in second, minimum is 30, 0 means no update)
 WEP
 Type:
 WEP Key Format:

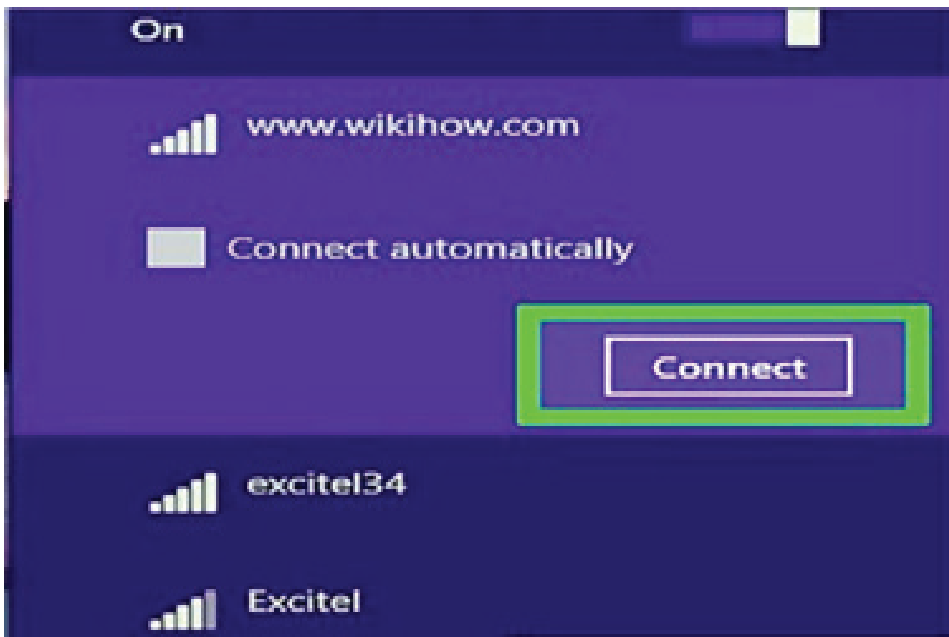
Key Selected	WEP Key	Key Type
Key 1: <input type="radio"/>	<input type="text"/>	<input type="text" value="Disabled"/>
Key 2: <input type="radio"/>	<input type="text"/>	<input type="text" value="Disabled"/>
Key 3: <input type="radio"/>	<input type="text"/>	<input type="text" value="Disabled"/>
Key 4: <input type="radio"/>	<input type="text"/>	<input type="text" value="Disabled"/>

7 Gave old administrator name and password and a new username and password then click on save.



name and password can contain between 1 - 15 characters and may not include special characters
 Old User Name:
 Old Password:
 New User Name:
 New Password:
 Confirm New Password:

8 Open the WIFI portal see the name is displaying then click on connect.



9 Enter the password or security key which was given before and click on next button.



10 The WIFI is connected finally it is displaying in WIFI portal.

EXERCISE 7 : Implementing Voiceover IP

Objectives

At the end of this exercise you shall be able to

- configure VPN
- download and install skype.

Requirements

Tools/Materials

- Windows enabled PC/Laptop
- Internet connectivity with VPN
- Skype software setup

Procedure

TASK 1: Planning and preparation

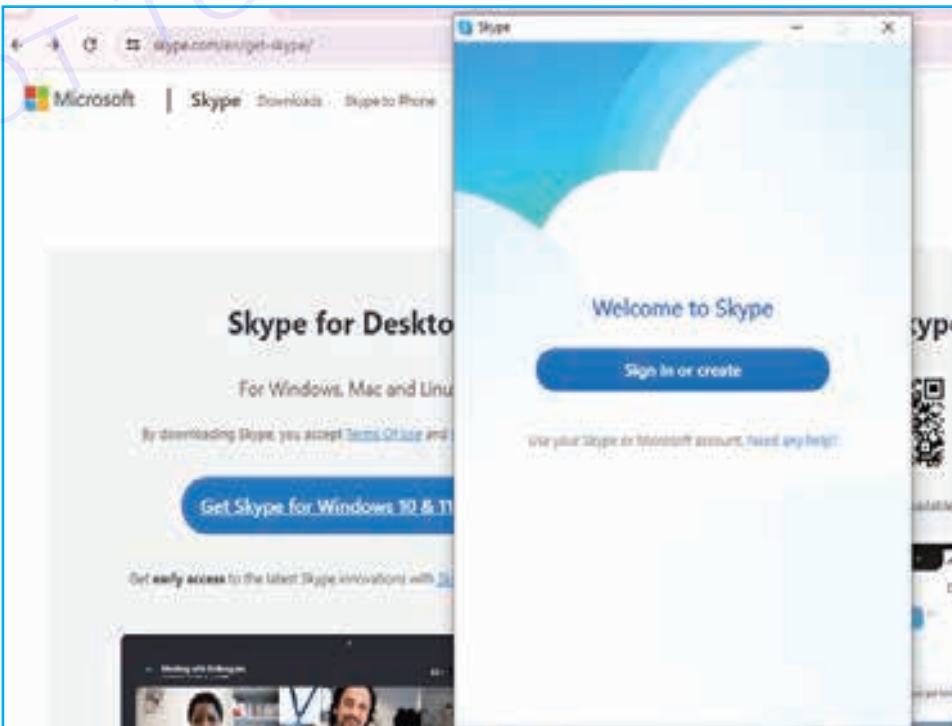
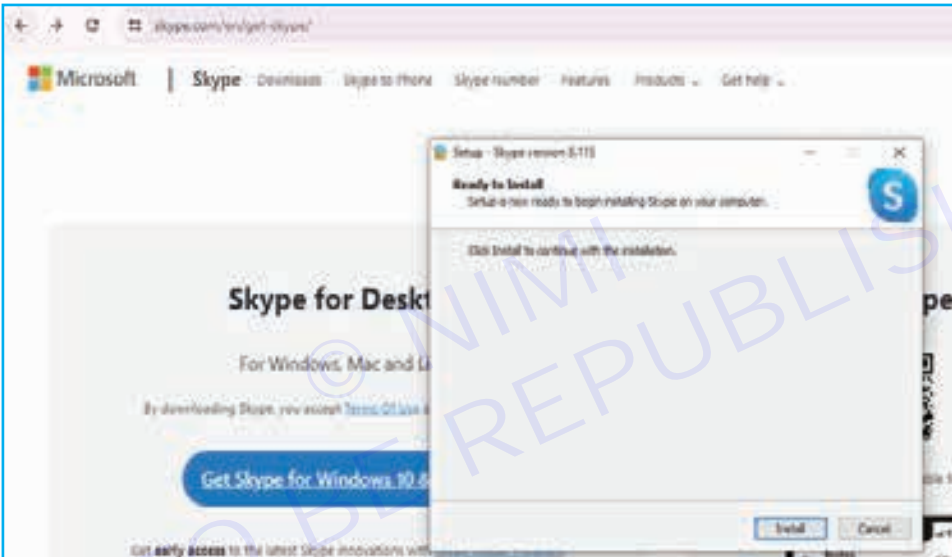
- 1 Assess your needs:
- 2 Determine how many users will need VoIP, what features you require (e.g., call forwarding, voicemail), and your budget.
- 3 Test your internet connection:
- 4 Ensure your upload and download speeds are sufficient for VoIP calls (generally, at least 3 Mbps for both). You can use online speed tests to check.
- 5 Choose a VoIP provider:
- 6 Select a provider that meets your needs and budget, offering reliable service and features you need.
- 7 Decide on hardware:
- 8 Consider whether you'll use physical VoIP phones, softphones (software apps on computers), or a combination.

TASK 2: Installation

- 1 Follow your provider's instructions:
- 2 Each provider will have specific setup steps, so refer to their documentation or website for detailed guidance.
- 3 Configure your network:
- 4 Depending on your setup, you may need to adjust firewall settings, enable Quality of Service (QoS) for VoIP traffic prioritization, or configure a Virtual Private Network (VPN) for added security.
- 5 Set up your devices:
- 6 Connect your VoIP phones or install the softphone app on your devices. Configure settings like your phone number, voicemail, and call forwarding.

TASK 3: Download and Install Skype

- 1 Download skype link - ms-windows-store://pdp/?productid=9WZDNCRFJ364&cid=scom-web-store
- 2 Click on Download Button for download Skype in windows.
- 3 Click on Install button for installation process.
- 4 After installation you have to create an account using Sign in button.



EXERCISE 8 : Configuring DHCP, IPV4/IPV6

Objectives

At the end of this exercise you shall be able to

- install DHCP server
- configure DHCP server and creating scope
- provide IP address range along with subnet.

Requirements

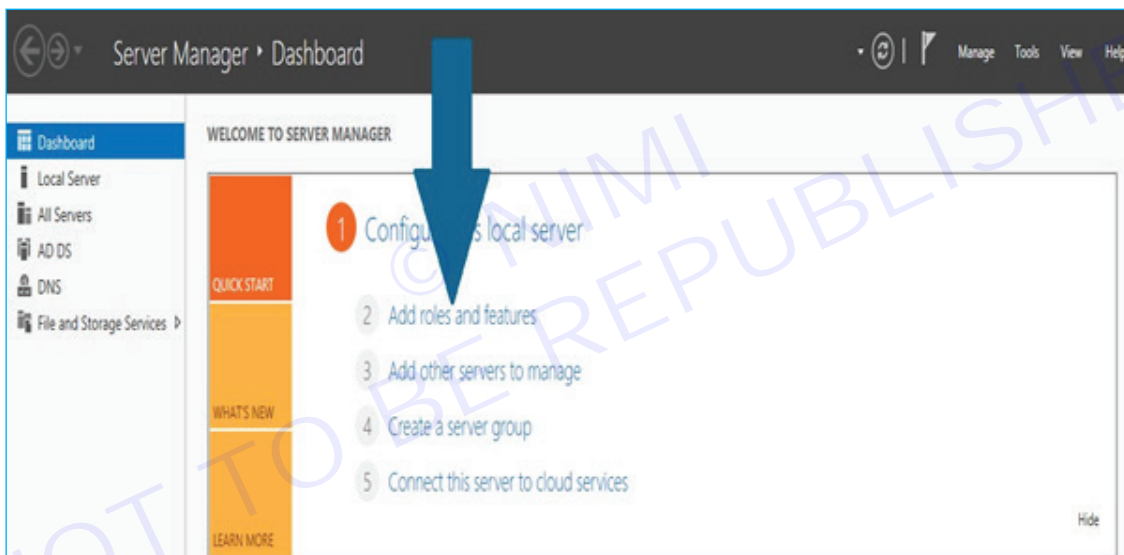
Tools/Materials

- PC/Laptop
- Windows server
- Internet connectivity

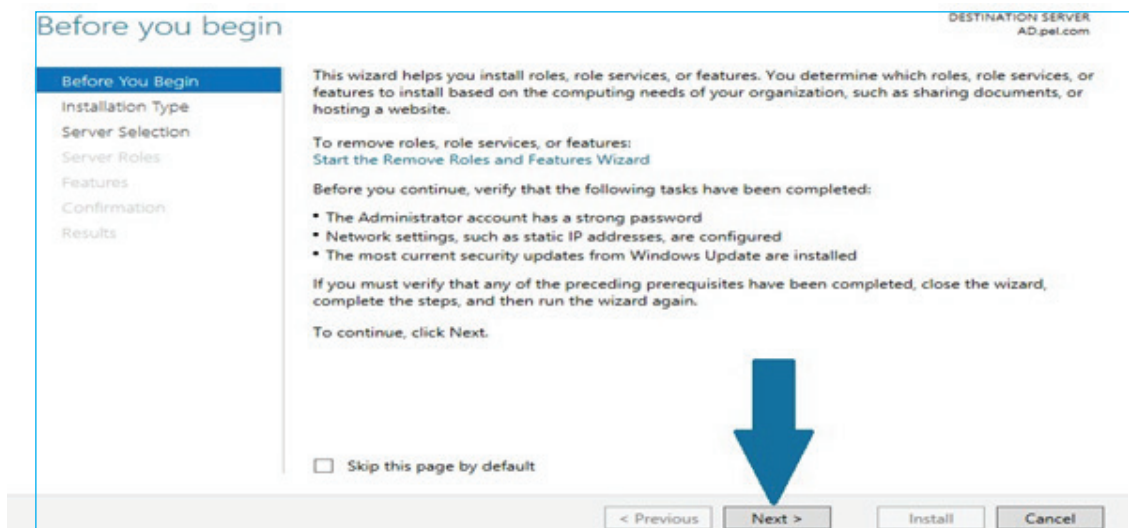
Procedure

TASK 1: Installing DHCP Server

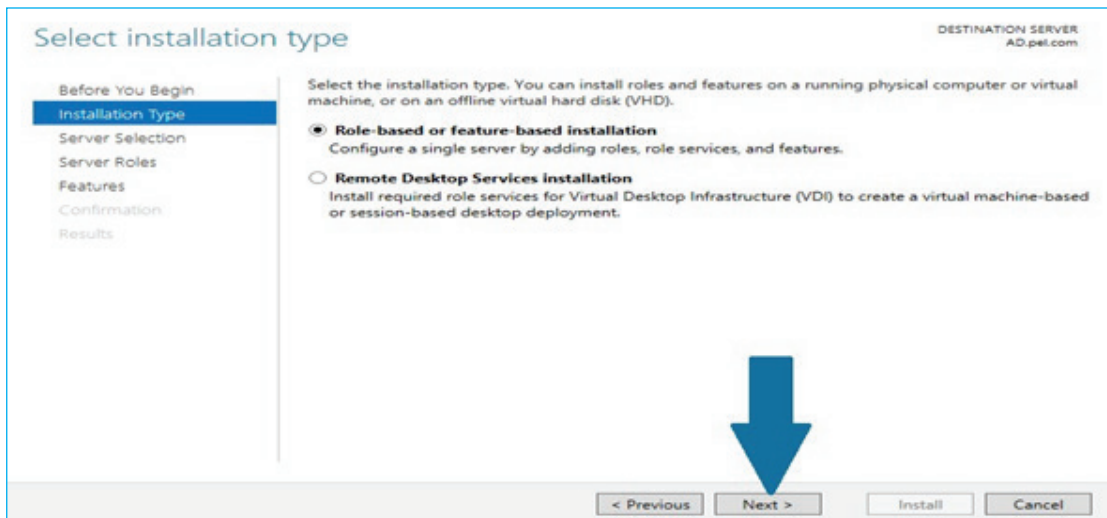
- 1 Open Server Manager from task bar and click Add roles and features



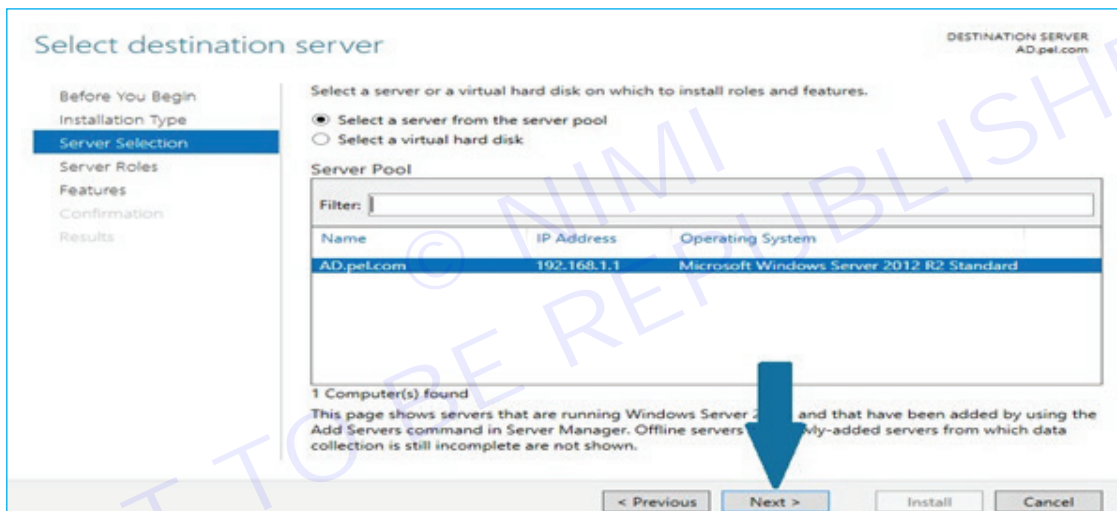
- 2 Before you run the installation wizard, make sure that an administrator account has a strong password, static IP is configured, and security updates from Windows updates are installed. When you are done, click Next



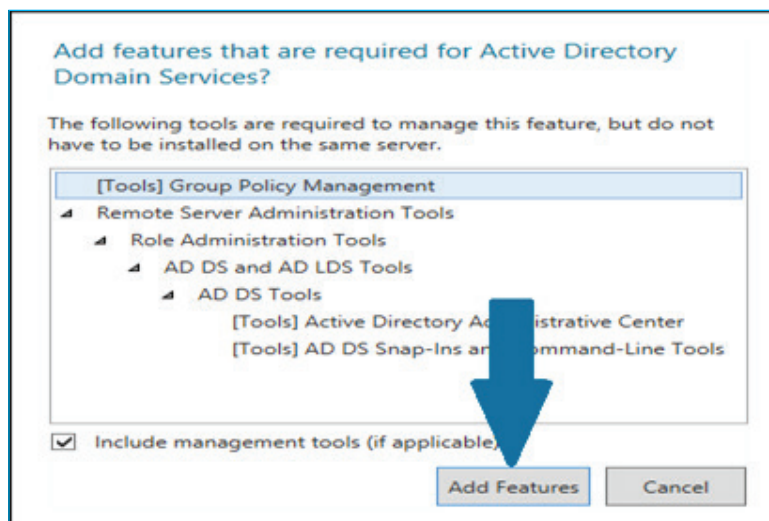
3 Select Role-based or feature-based installation and click Next



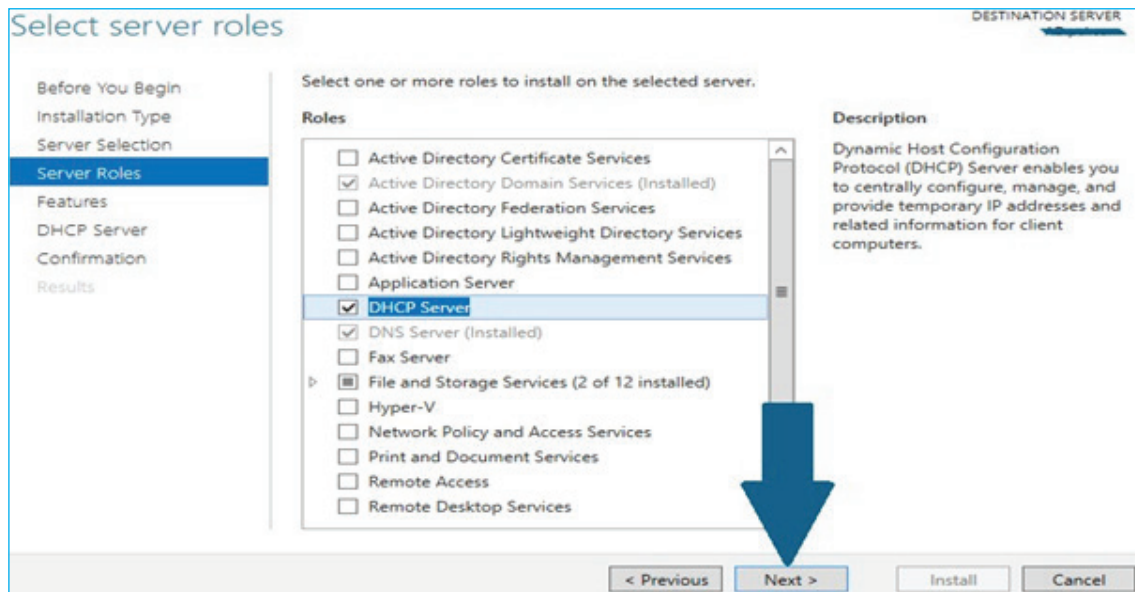
4 Select a destination server on which you want to install the DHCP server. In our case, there is only one server which is local server and it is selected by default. Click Next



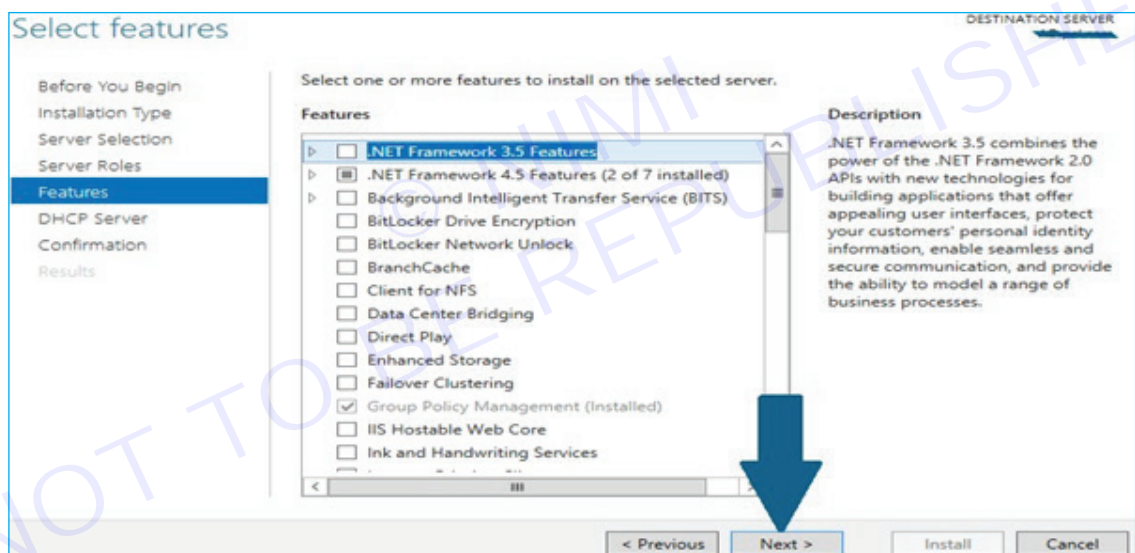
5 Select DHCP server role by checking the appropriate box. As soon as you check the box, a small window will pop up alerting you that there are some other features which are also required to be installed along with DHCP server. Click Add Features



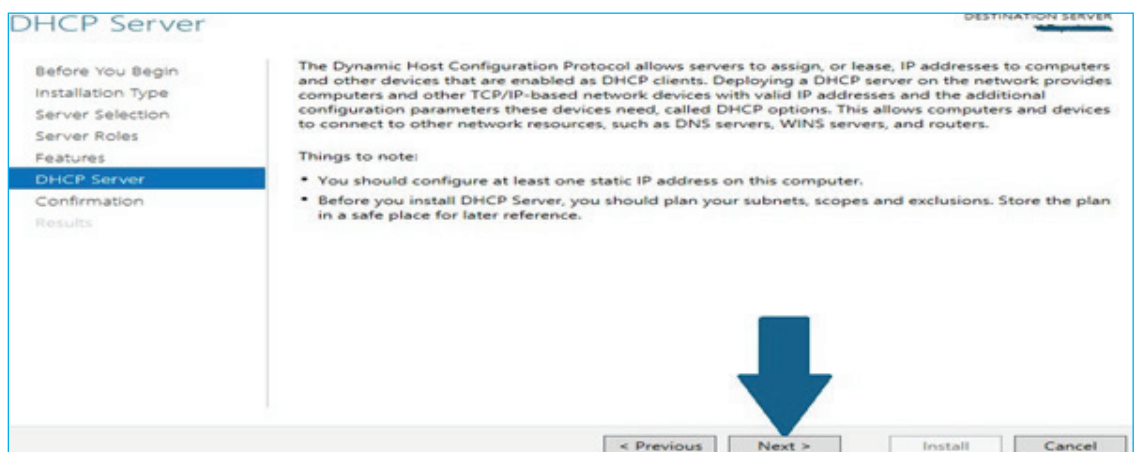
6 Click Next



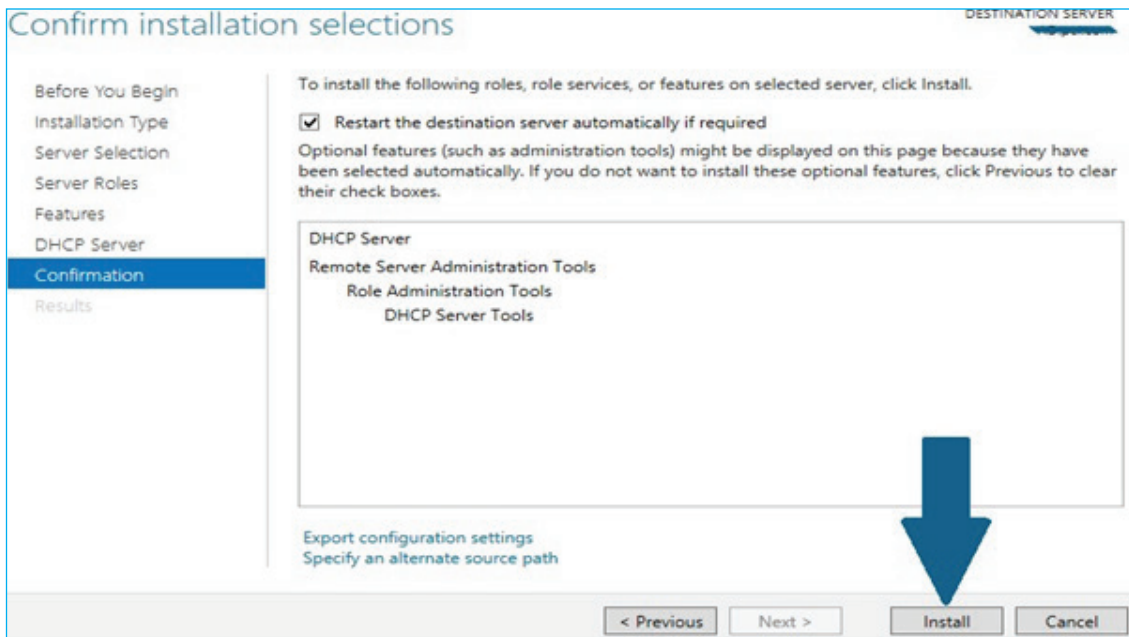
7 Click Next



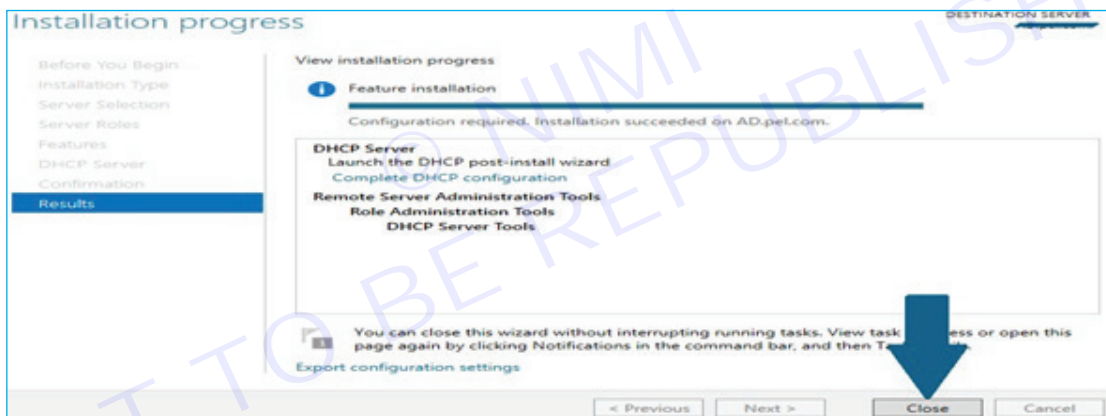
8 Note the things outlined in the screen and click Next



9 Confirm your installation selections and click Install

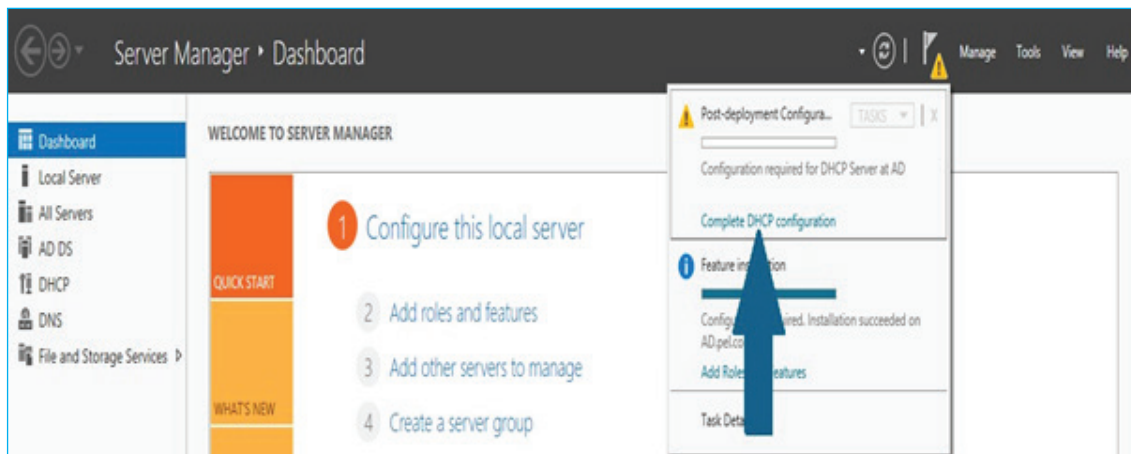


10 Click Close to finish the installation



TASK 2: Configuring DHCP Server and Creating Scope

1 Open Server Manager and click notifications icon. A small window will appear. Click Complete DHCP configuration



2 Click Next

Description

Description
Authorization
Summary

The following steps will be performed to complete the configuration of the DHCP Server on the target computer:

Create the following security groups for delegation of DHCP Server Administration.

- DHCP Administrators
- DHCP Users

Authorize DHCP server on target computer (if domain joined).

< Previous **Next >** Commit Cancel

3 Choose Skip AD authorization since we do not have any AD configured and click Commit

Authorization

Description
Authorization
Summary

Specify the credentials to be used to authorize this DHCP server in AD DS.

Use the following user's credentials
User Name: PEL\Administrator

Use alternate credentials
UserName: _____ Specify...

Skip AD authorization

< Previous Next > **Commit** Cancel

4 Read the summary and click Close

Summary

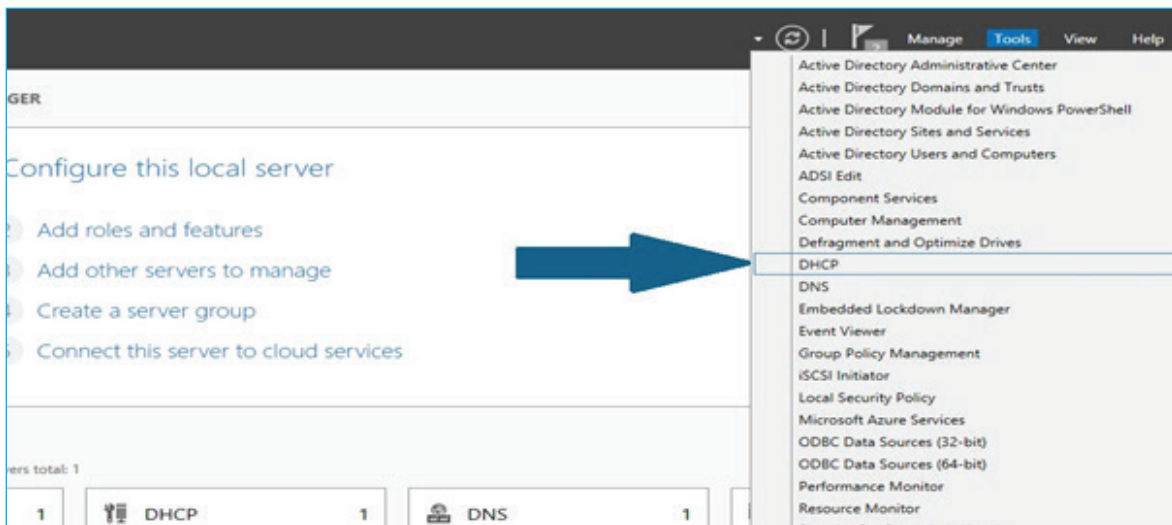
Description
Authorization
Summary

The status of the post install configuration steps are indicated below:

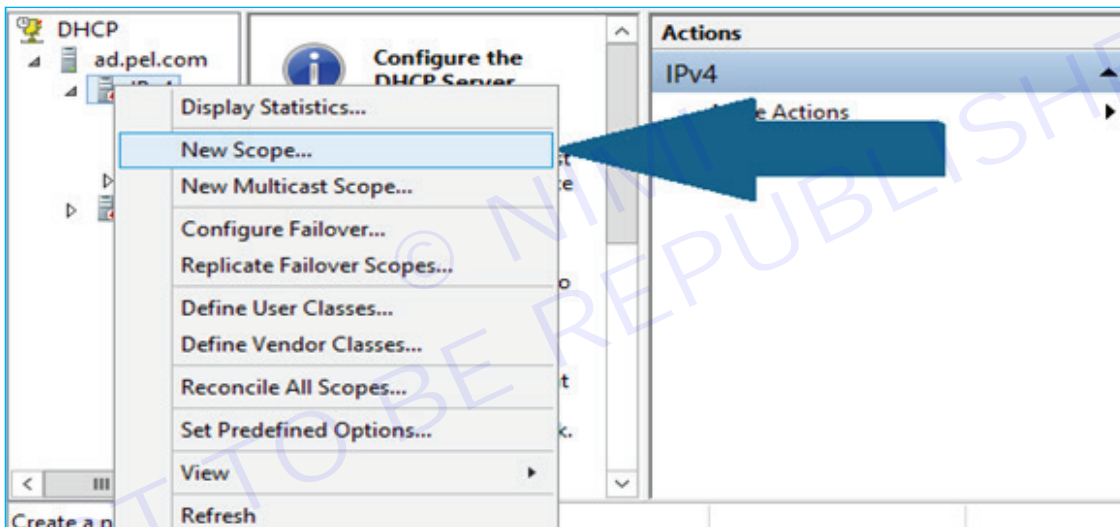
Creating security groups Done
Please restart the DHCP server service on the target computer for the security groups to be effective.

< Previous Next > **Close** Cancel

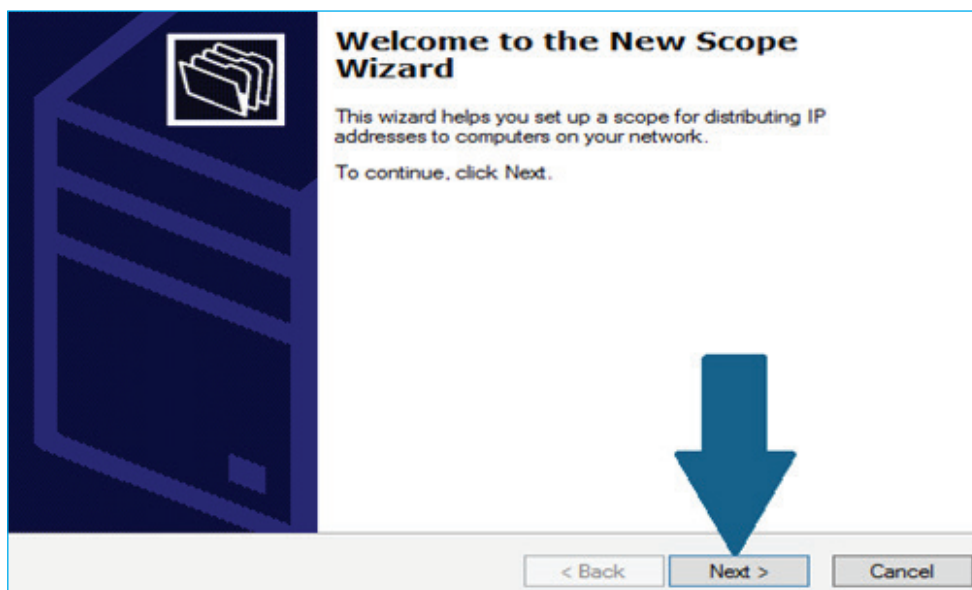
5 Open Server Manager and click on Tools. When a small window appear, scroll to DHCP and click it



6 In management console, right click on IPv4 and scroll to New Scope and click it.



7 Click Next

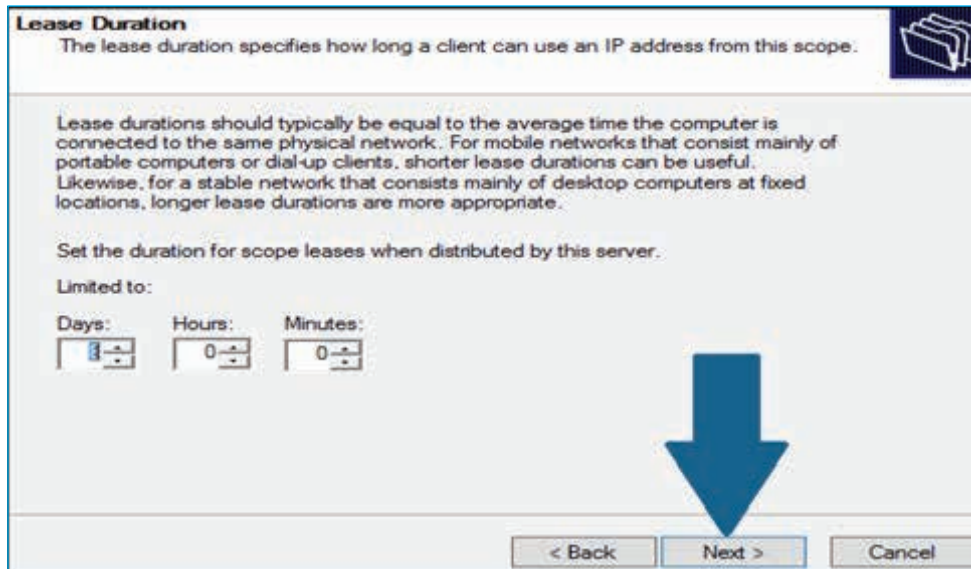


8 Provide name and meaningful description of this new scope and click Next

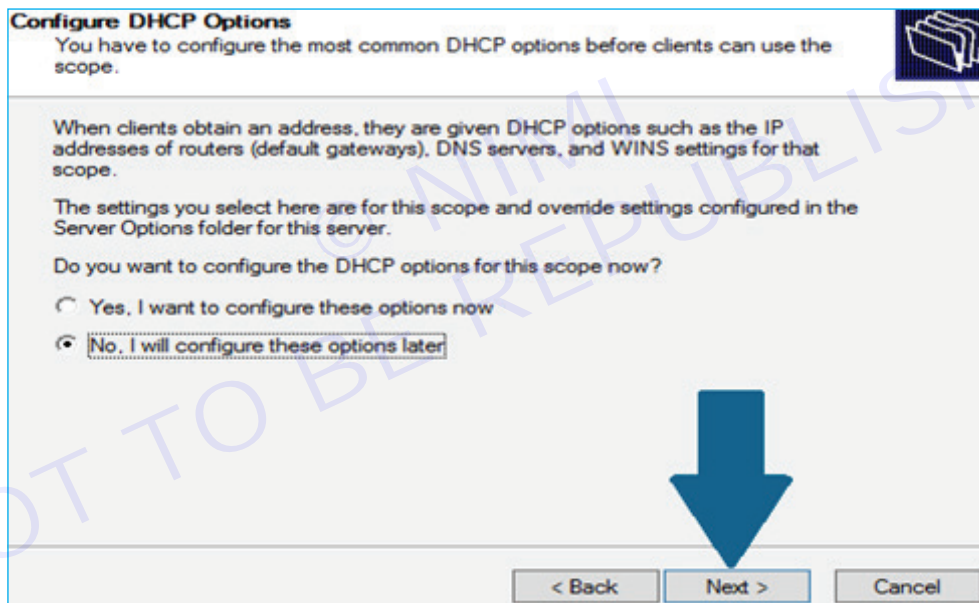
9 Provide IP address range along with sub net you need to distribute to client machines and click Next

10 Provide any IP addresses you need to exclude from pool and click Add. I have excluded a first IP address which is statically assigned to my DHCP server. Click Next

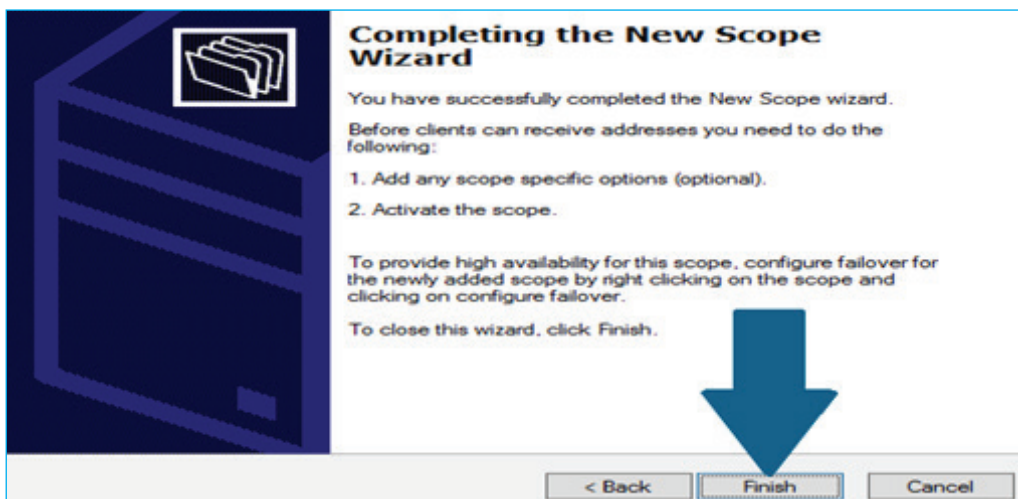
11 Keep lease duration as 8 days and click Next



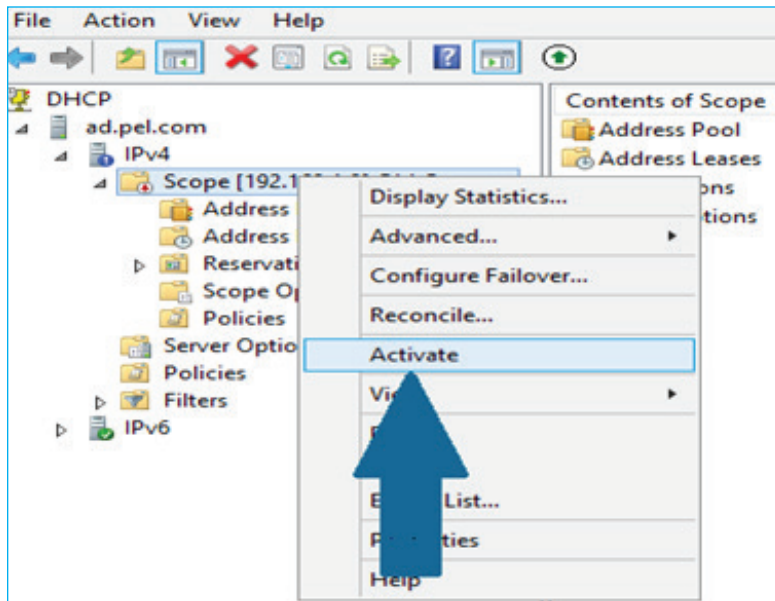
12 Choose No, I will configure these options later and click Next



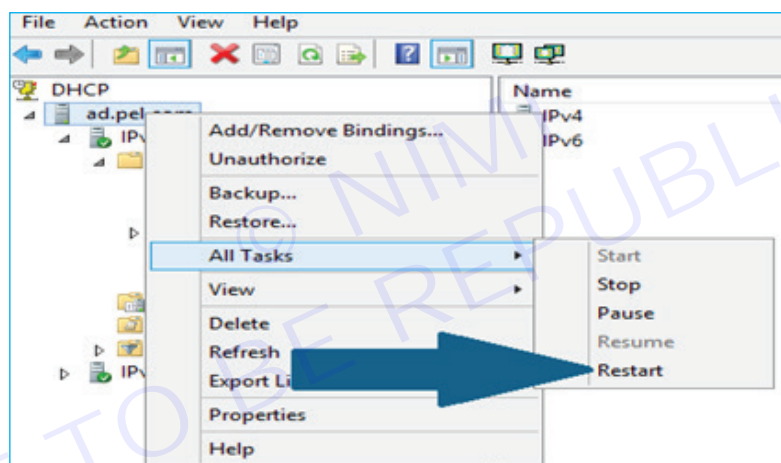
13 Click Finish to end the new scope wizard



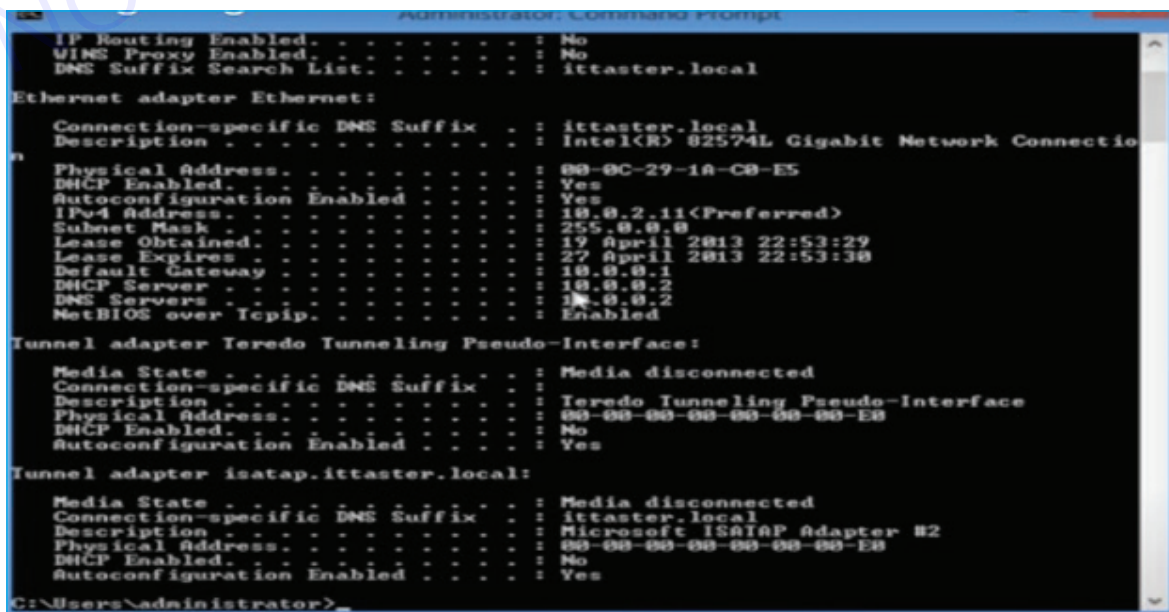
14. Right-click on new scope you just created in above step and click Activate



15 Right-click on your server, scroll to All Tasks and then click Restart to finish with configuration



16 Open Command Prompt and Type : ipconfig /all



EXERCISE 9 : Configuring Network Security for preventing Attacks

Objectives

At the end of this exercise you shall be able to

- implement a fire wall
- secure routers and wireless access point
- monitor network activity and security events.

Requirements

Tools/Materials

- PC laptop with network connectivity
- Firewall
- Routers

Procedure

TASK 1: Implement a Firewall

- 1 **Choose a firewall:** Select a firewall suitable for your network size and needs. Hardware firewalls offer robust protection, while software firewalls are more cost-effective for smaller networks.
- 2 **Configure firewall rules:** Define rules to allow only authorized traffic and block suspicious connections. Consider factors like source IP addresses, destination ports, and protocols.
- 3 **Enable logging and monitoring:** Regularly review firewall logs to identify potential threats and adjust rules accordingly.

TASK 2: Secure Your Routers and Wireless Access Points

- 1 **Change default passwords:** Replace factory-set passwords with strong, unique credentials for your router and wireless access points.
- 2 **Enable encryption:** Secure your Wi-Fi network with WPA2 or WPA3 encryption to prevent unauthorized access.
- 3 **Disable remote access:** If not necessary, disable remote management features on your router and access points to reduce attack surfaces.
- 4 **Update firmware regularly:** Install firmware updates promptly to patch vulnerabilities and keep your devices secure.

TASK 3: Employ Strong Passwords and User Authentication

- 1 **Enforce strong password policies:** Mandate complex passwords with a mix of uppercase and lowercase letters, numbers, and symbols. Consider implementing multi-factor authentication (MFA) for added security.
- 2 **Limit user privileges:** Assign users the minimum level of access required for their tasks, minimizing potential damage in case of compromised accounts.
- 3 **Educate users on cyber security:** Train employees and users on cyber hygiene practices like phishing awareness, avoiding suspicious links, and reporting suspicious activity.

TASK 4: Keep Software and Systems Updated

- 1 **Update operating systems and applications regularly:** Apply software updates promptly to patch known vulnerabilities and security holes.
- 2 **Disable unused software and services:** Remove unnecessary software and services to reduce potential attack vectors.

TASK 5: Monitor Network Activity and Security Events

- 1 **Implement a Security Information and Event Management (SIEM) system:** SIEM tools provide centralized logs and real-time monitoring of security events across your network, helping you detect and respond to threats promptly.
- 2 **Regularly review security logs:** Monitor logs for suspicious activity like failed login attempts, unauthorized access attempts, and malware signatures.

TASK 6: Conduct Regular Security Assessments and Penetration Testing

- 1 **Schedule regular vulnerability scans:** Employ vulnerability scanners to identify weaknesses in your network infrastructure and applications.
- 2 **Perform penetration testing:** Engage professional penetration testers to simulate real-world attacks and identify exploitable vulnerabilities.
- 3 **Address identified vulnerabilities:** Prioritize and address identified vulnerabilities based on their severity and potential impact.

EXERCISE 10 : Setting password policy

Objectives

At the end of this exercise you shall be able to

- implement a password policy.

Requirements

Tools/Materials

- PC laptop with network connectivity

Procedure

Setting up a password policy in a network is an essential security measure to protect sensitive information and resources from unauthorized access.

Implement a password policy

- 1 **Assessment:** Begin by assessing the current state of password security within your network. Understand the existing password practices, weaknesses, and areas that need improvement.
- 2 **Define Password Requirements:** Determine the password requirements that users must adhere to. These requirements typically include:
 - **Minimum password length:** Suggest a minimum length of 8-12 characters.
 - **Complexity:** Require a combination of uppercase letters, lowercase letters, numbers, and special characters.
 - **Expiry:** Set a policy for password expiration, such as every 90 days.
 - **History:** Enforce a rule that prevents users from reusing old passwords.
 - **Lockout:** Establish a threshold for failed login attempts before an account is locked out temporarily.
 - **Account Inactivity:** Consider disabling or prompting for password change after a certain period of inactivity.
 - **Two-Factor Authentication (2FA):** Encourage or mandate the use of 2FA where possible for an added layer of security.
- 3 **Communicate Policy:** Clearly communicate the password policy to all users within the network. Explain the rationale behind each requirement and the importance of adhering to them.
- 4 **Implement Policy:** Utilize the network's administrative tools or security software to enforce the password policy. This may involve configuring settings in:
 - Active Directory (for Windows networks)
 - Group Policy (for Windows networks)
 - LDAP (Lightweight Directory Access Protocol)
 - RADIUS (Remote Authentication Dial-In User Service)
 - IAM (Identity and Access Management) solutions
 - Password management tools
- 5 **Enforcement:** Regularly monitor adherence to the password policy. Implement mechanisms to enforce the policy automatically, such as system prompts for password changes when they expire, or locking out accounts after multiple failed login attempts.
- 6 **Education and Training:** Conduct training sessions or provide resources to educate users about the importance of strong passwords, how to create them securely, and the consequences of weak password practices.
- 7 **Periodic Review and Update:** Regularly review the password policy to ensure it remains effective and up-to-date with evolving security threats and best practices. Make necessary adjustments based on feedback, security incidents, or changes in regulations.
- 8 **Testing:** Periodically conduct security audits or penetration tests to evaluate the effectiveness of the password policy and identify any vulnerabilities that need to be addressed.

EXERCISE 11 : Sniffing on Switched Networks

Objectives

At the end of this exercise you shall be able to

- use wireshark, the packet sniffing tool.

Requirements

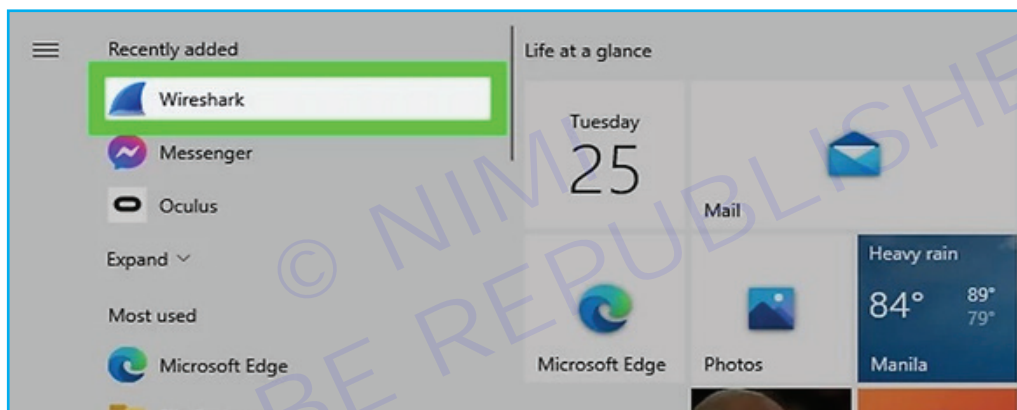
Tools/Materials

- Windows PC/Laptop connected to a network
- Wireshark network setup

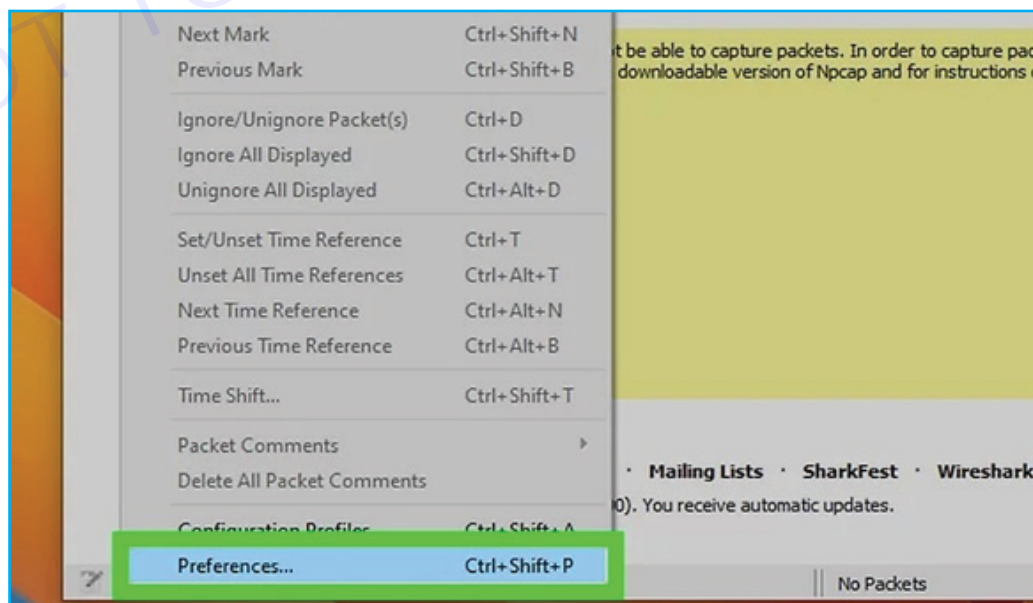
Procedure

TASK 1: Using Wireshark , a packet-sniffing tool

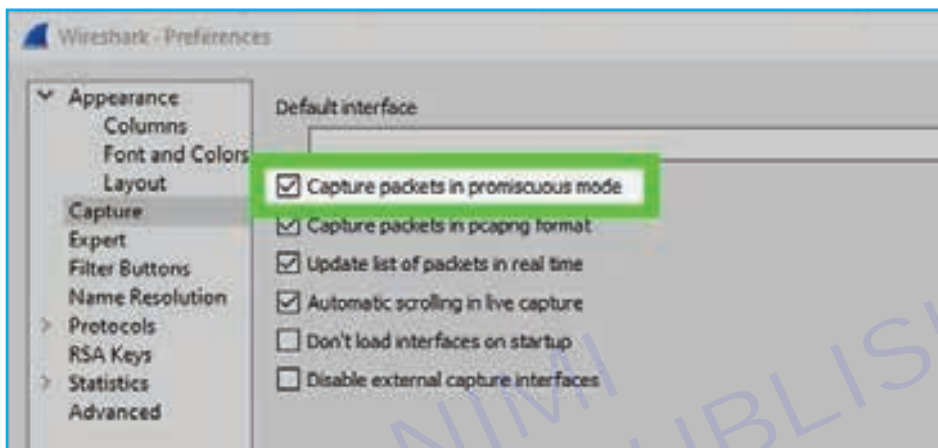
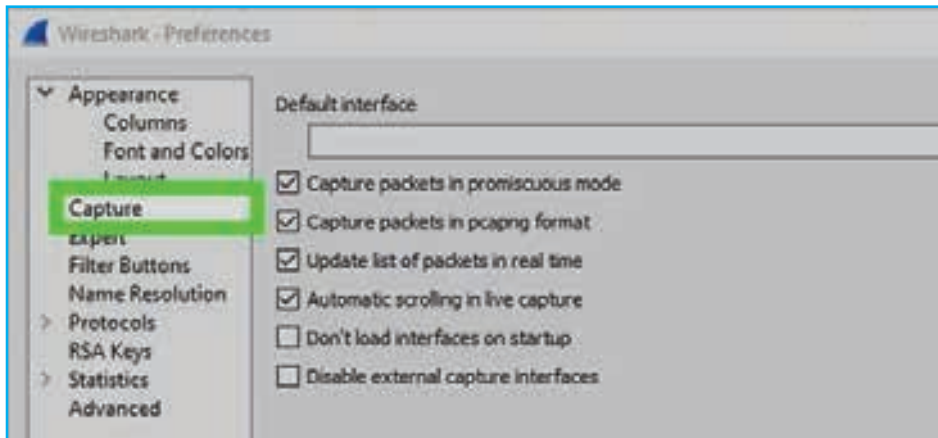
- 1 Open Wireshark. If you don't have the free program for Windows, Mac, and Linux, you can download it from their website.



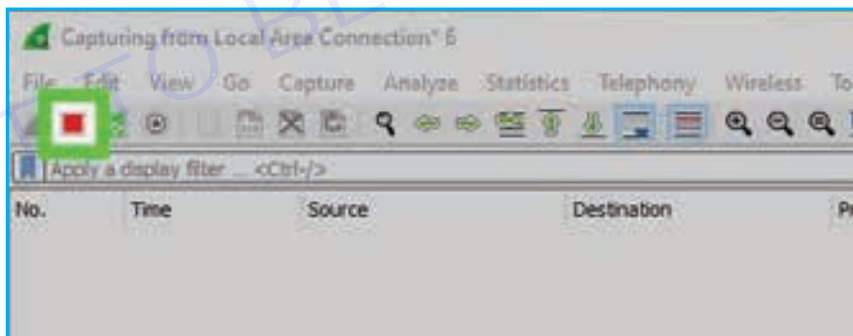
- 2 Navigate to "Edit" and click Preferences. You'll find "Edit" in the menu bar along the top of the Wireshark window.



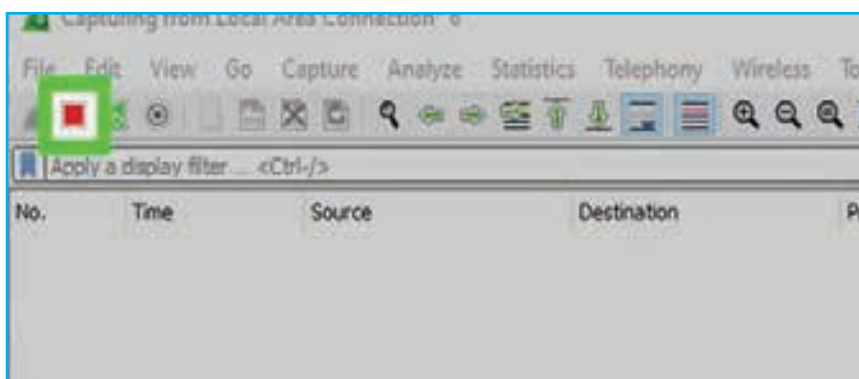
- 3 Click the Capture tab. It's in the panel on the left side of the window.
- 4 Ensure "Capture packets in promiscuous mode" is selected and press OK.



- 5 Close the Wireshark Preferences window if it doesn't close automatically.
- 6 Click the blue fin icon to start recording. You can also press Ctrl + E. You'll see a graph that indicates your network activity.



- 7 Press the red fin icon to stop recording. You'll see the previous history in the window at the top of your screen.



- 8 Click on an instance to see the IP addresses it was going to and coming from as well as additional data.

EXERCISE 12 : IP Address Spoofing

Objectives

At the end of this exercise you shall be able to

- do IP spoofing using Hide.me

Requirements

Tools/Materials

- Windows PC/Laptop connected to internet

Procedure

TASK 1: IP Spoofing using Hide.me

- 1 Download Hide.me VPN Software and install it.
- 2 Click "Enable VPN" button to enable the Virtual private network and your ip address will change.



- 3 Click on "Change" button in the right side of the window to set the country location for the IP Address.
- 4 Goto Cmd Prompt and type the command "ipconfig" then we will get the latest IP Address.

```

C:\Windows\system32\cmd.exe
Default Gateway . . . . . :
Ethernet adapter VMware Network Adapter VMnet8:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::4d38:179d:5c2f:7ad8%33
Autoconfiguration IPv4 Address. . : 169.254.226.216
Subnet Mask . . . . . : 255.255.0.0
Default Gateway . . . . . :

Ethernet adapter Ethernet 4:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :

```

- 5 On Click "Details" button in the left side of the window. We are able to see the location is changed to Netherlands. But our actual location is India.



EXERCISE 13 : DNS Spoofing

Objectives

At the end of this exercise you shall be able to

- do DNS spoofing using Ettercap.

Requirements

Tools/Materials

- PC/Laptop
- Kali Linux OS
- Ettercap

Procedure

TASK 1: DNS Spoofing using Ettercap

- 1 Start by booting up Kali Linux

Note: before you continue and make sure that you are on the same network as your target.

- 2 Open the file /etc/ettercap/etter.conf with a text editor like gedit and edit the file.

```
root@kali:~# gedit /etc/ettercap/etter.conf
```

- 3 Edit the uid and gid values at the top to make them 0.

```
# (at your option) any later version.
#
#
#####
[privs]
ec_uid = 0          # nobody is the default
ec_gid = 0          # nobody is the default

[mitm]
arp_storm_delay = 10      # milliseconds
arp_poison_smart = 0      # boolean
arp_poison_warm_up = 1   # seconds
arp_poison_delay = 10    # seconds
arp_poison_icmp = 1      # boolean
arp_poison_reply = 1     # boolean
arp_poison_request = 0   # boolean
arp_poison_equal_mac = 1 # boolean
dhcp_lease_time = 1800   # seconds
port_steal_delay = 10    # seconds
port_steal_send_delay = 2000 # microseconds
```

- 4 Remove both the # signs below where it says "if you use iptables".

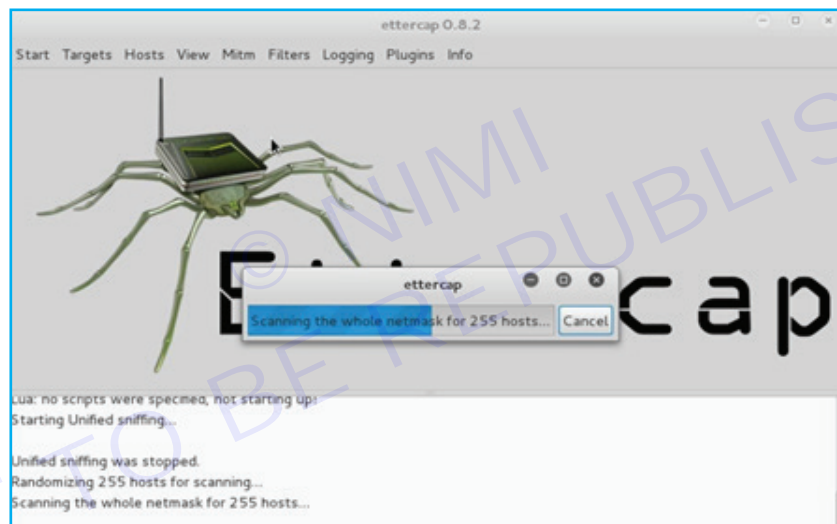
```
#-----
#   Linux
#-----
# if you use ipchains:
#redir_command_on = "ipchains -A input -i %iface -p tcp -s 0/0 -d 0/0 %port"
#redir_command_off = "ipchains -D input -i %iface -p tcp -s 0/0 -d 0/0 %port"
# if you use iptables:
#redir_command_on = "iptables -t nat -A PREROUTING -i %iface -p tcp --dport %port"
#redir_command_off = "iptables -t nat -D PREROUTING -i %iface -p tcp --dport %port"
#-----
#   Mac Os X
#-----
# quick and dirty way:
#redir_command_on = "ipfw -q add set %set fwd 127.0.0.1,%rport tcp from any"
#redir_command_off = "ipfw -q delete set %set"
# a better solution is to use a script that keeps track of the rules inserted
```


5 Run Ettercap by using Terminal.



6 Select Sniff > Unified sniffing... > (Select the interface connected to the internet) > OK it automatically starts sniffing after we press OK

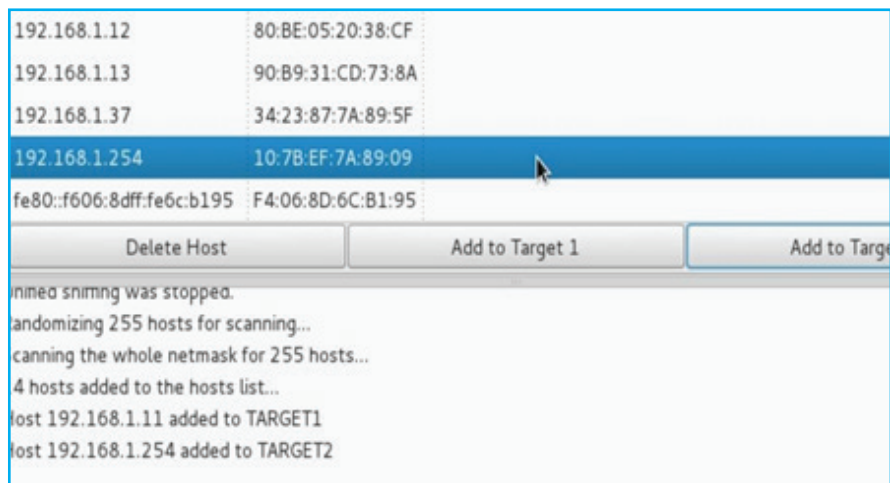
7 To scan for targets on your network go to Hosts > Scan for hosts and wait until it does the scan.



8 Go back to Hosts and select Host list to see all the targets that Ettercap has found.

9 Select IP address from the host list in Ettercap and choose Add to Target 1

10 Now select the gateway IP from the host list and choose Add to Target 2.



- 11 Go to the MITM tab and select ARP poisoning, choose Sniff remote connections and press OK.
- 12 go to Plugins > Manage the plugins and double click dns_spoof to activate that plugin.
- 13 Edit etter.dns file, This is the hosts file and is responsible for redirecting specific DNS requests.

```
root@Kali:~# gedit /etc/ettercap/etter.dns
```

- 14 Add another line, use whatever website you would like and change the IP address to your IP address.(Here example facebook is used).

```
# or for TXT query (value must be wrapped in double quotes):
# google.com TXT "v=spf1 ip4:192.168.0.3/32 ~all"
#
# NOTE: the wildcarded hosts can't be used to poison the PTR requests
# so if you want to reverse poison you have to specify a plain
# host. (look at the www.microsoft.com example)
#
#####

# microsoft sucks :)
# redirect it to www.linux.org
#
microsoft.com A 107.170.40.56
*.microsoft.com A 107.170.40.56
www.microsoft.com PTR 107.170.40.56 # Wildcards in PTR are not allowed
facebook.com A 192.168.1.39
*.facebook.com A 192.168.1.39

#####
# no one out there can have our domains...
#
www.alor.org A 127.0.0.1
www.naga.org A 127.0.0.1
www.naga.org AAAA 2001:db8::2

#####
# dual stack enabled hosts does not make life easy
# force them back to single stack
www.ietf.org A 127.0.0.1
www.ietf.org AAAA ::

www.example.org A 0.0.0.0
www.example.org AAAA ::1
```

- 15 Start Apache to accept incoming traffic.

```
root@Kali:~# service apache2 start
```

- 16 Go to /var/www/html folder and alter index.html page for your needs and save the page.
- 17 Go back to Ettercap and select Start > Start sniffing and that to start the attack.

— — — — —

EXERCISE 14 : Password Cracking: Dictionary vs Brute-Force vs Hybrid methods

Objectives

At the end of this exercise you shall be able to

- crack password by using Dictionary Attack method
- crack password by using Brute-Force Attack method
- crack password by using Hybrid Attack method.

Requirements

Tools/Materials

- Linux PC/Laptop with internet connection
- Password cracking tools John the Ripper, Hashcat, Hydra

Procedure

Password cracking involves attempting to gain unauthorized access to a system or an account by trying to decipher or guess the password. There are several methods for password cracking, including dictionary attacks, brute-force attacks, and hybrid attacks. Here are practical steps for each method:

TASK 1: Dictionary Attack

- 1 **Gather Wordlists:** Obtain a comprehensive wordlist or dictionary containing commonly used passwords, phrases, and combinations.
- 2 **Select Tools:** Choose a password cracking tool that supports dictionary attacks, such as John the Ripper, Hashcat, or Hydra.
- 3 **Configure Tool:** Set up the password cracking tool to use the selected wordlist as input.
- 4 **Execute Attack:** Run the tool against the target system or account, attempting to log in with each password in the dictionary.
- 5 **Analyze Results:** Review the output to identify successful password guesses and gain access to the target account.

TASK 2: Brute-Force Attack

- 1 **Determine Password Complexity:** Assess the complexity of the target password, including length and character set.
- 2 **Select Tools:** Choose a password cracking tool capable of brute-force attacks, such as John the Ripper, Hashcat, or Hydra.
- 3 **Configure Tool:** Set up the password cracking tool to systematically generate and try all possible combinations of characters within the specified parameters.
- 4 **Execute Attack:** Run the tool against the target system or account, attempting to guess the password through exhaustive trial and error.
- 5 **Monitor Progress:** Monitor the progress of the brute-force attack, as it may take significant time and computational resources.
- 6 **Analyze Results:** Review the output to identify successful password guesses and gain access to the target account.

TASK 3: Hybrid Attack

- 1 **Combine Wordlists and Brute Force:** Create a hybrid wordlist by combining common words, phrases, and patterns with brute-force-generated strings.
- 2 **Select Tools:** Choose a password cracking tool that supports hybrid attacks, such as John the Ripper, Hashcat, or Hydra.
- 3 **Configure Tool:** Set up the password cracking tool to use the hybrid wordlist as input, along with parameters for brute-force generation.
- 4 **Execute Attack:** Run the tool against the target system or account, attempting to guess the password using both dictionary-based and brute-force methods.
- 5 **Adjust Parameters:** Fine-tune the attack parameters based on initial results and feedback to optimize the cracking process.
- 6 **Analyze Results:** Review the output to identify successful password guesses and gain access to the target account.
- 7 **Additional Considerations:**
 - a **Resource Requirements:** Password cracking can be resource-intensive, requiring significant computational power and time, especially for brute-force attacks.
 - b **Legal and Ethical Considerations:** Ensure that password cracking activities comply with applicable laws, regulations, and ethical guidelines.
 - c **Defensive Measures:** Implement strong password policies, multi-factor authentication, and other security measures to protect against password cracking attempts.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 15 : Handling Denial-of-Service (DoS)

Objectives

At the end of this exercise you shall be able to

- handle DoS during a attack
- prevent DoS during the attack.

Requirements

Tools/Materials

- PC/Laptop with network connectivity
- Firewall & VPN

Procedure

TASK 1: Handling DoS During a Attack

- 1 **Identify the Attack:** The first step is to recognize a DoS attack. Signs include unusually slow network performance, website outages, or specific applications being unavailable.
- 2 **Isolate the Attack:** Try to isolate the affected system or service to minimize the impact on the entire network. Firewalls and traffic filtering rules can be helpful here.
- 3 **Activate DDoS Defense Mechanisms:** Many network devices like routers have built-in DDoS protection features. These may include rate limiting, which restricts incoming traffic volume, or blackholing malicious IP addresses.
- 4 **Contact Your Service Provider:** If you suspect a large-scale DDoS attack, especially for businesses or organizations, your internet service provider (ISP) can offer more advanced mitigation techniques and resources.
- 5 **Analyze and Adapt:** Once the immediate attack subsides, analyze logs and traffic patterns to understand the attack type. This will help refine your mitigation strategies for future attacks.

TASK 2: Preventing DoS Attacks

- 1 **Plan and Prepare:** Develop a DoS incident response plan that outlines roles, responsibilities, and communication protocols during an attack.
- 2 **Strengthen Network Security:** Regularly update firewalls, intrusion detection/prevention systems (IDS/IPS), and software on all devices.
- 3 **Utilize DDoS Mitigation Services:** Consider subscribing to DDoS protection services offered by security vendors. These services can filter malicious traffic before it reaches your network.
- 4 **Content Delivery Networks (CDNs):** CDNs can absorb large traffic spikes and distribute legitimate traffic more efficiently.
- 5 **Educate Users:** Train employees on potential social engineering tactics hackers use to launch DoS attacks via phishing emails or malware.

NOTE: DoS mitigation strategies depend on the specific attack type and the resources available.

EXERCISE 16 : Using Tools like John the Ripper, Cain & Abeletc

Objectives

At the end of this exercise you shall be able to

- use Cain&Abel test
- install John the Ripper pass and cracker tool
- crack password protected zip file using John the Ripper tool.

Requirements

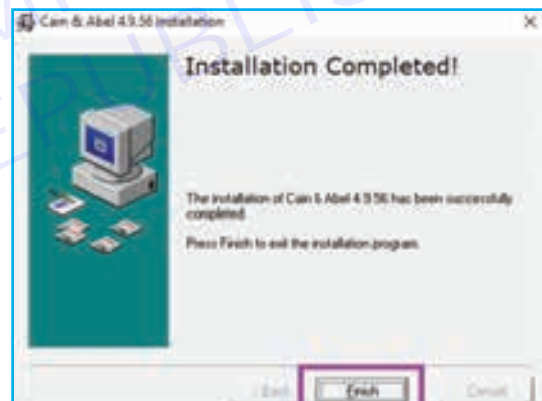
Tools/Materials

- PC with network connectivity
- Operating system, Windows/Linux Cain & Abel tool setup
- John the Ripper Password cracker tool

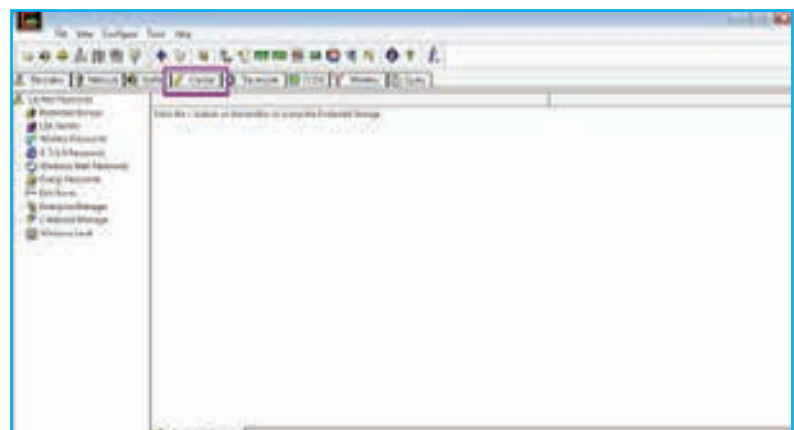
Procedure

TASK 1: Using tool Cain & Abel

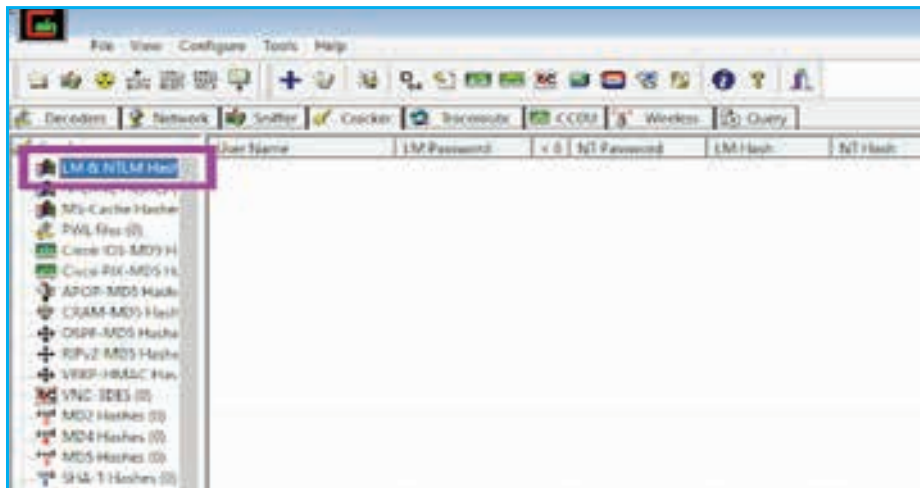
- 1 Download the Cain&Abel zip file from any browser.
- 2 Follow the installation instructions, proceeding by clicking “Next,” accepting the terms, and concluding by selecting “Finish” at the end.



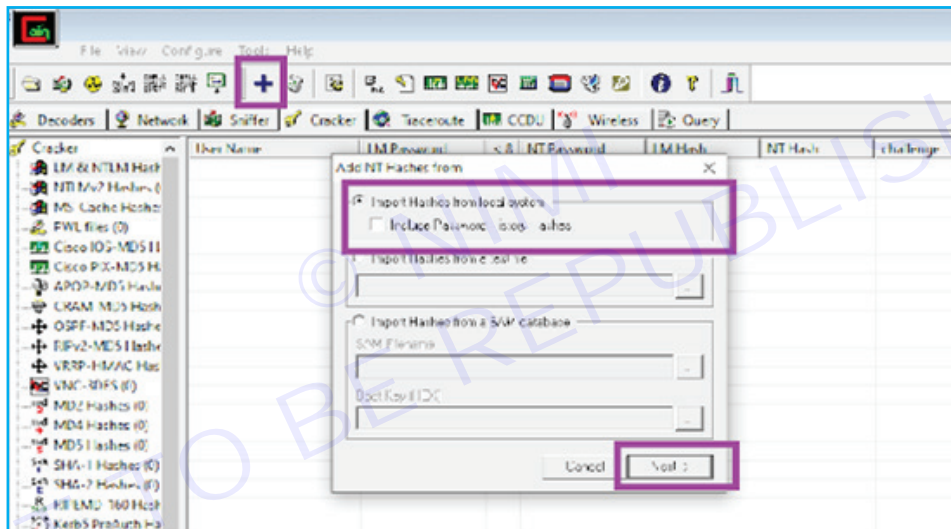
- 3 To install this tool, WinPcap also need. Click “Install” and then follow the setup steps as instructed by the wizard.
- 4 Click on “Cracker” in Cain&Abel.



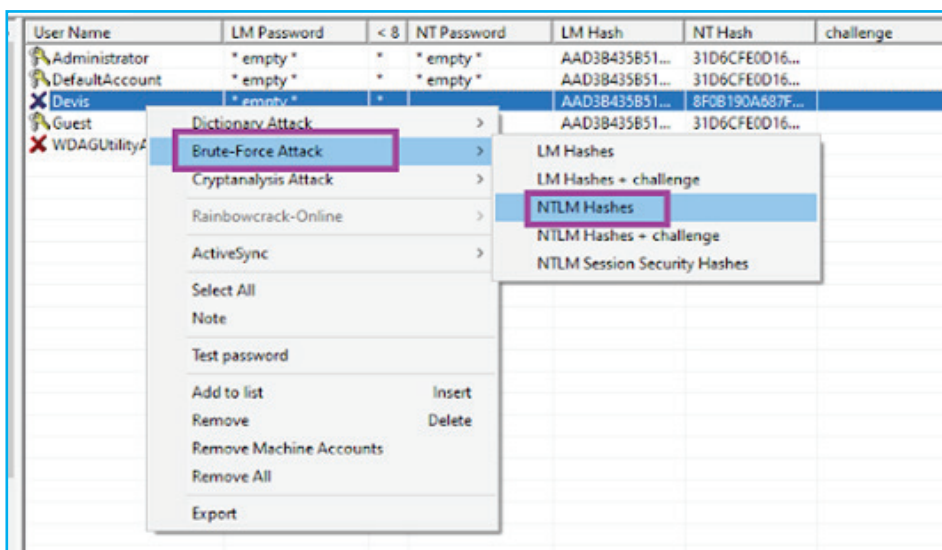
5 Click the option “LM & NTLM Hashes”.



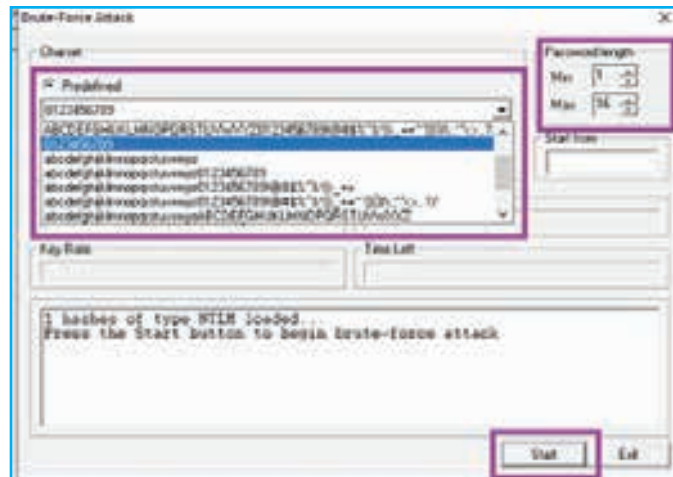
6 Click on the “+” symbol, then select “Import Hashes.” Check the corresponding box and proceed by clicking “Next.”



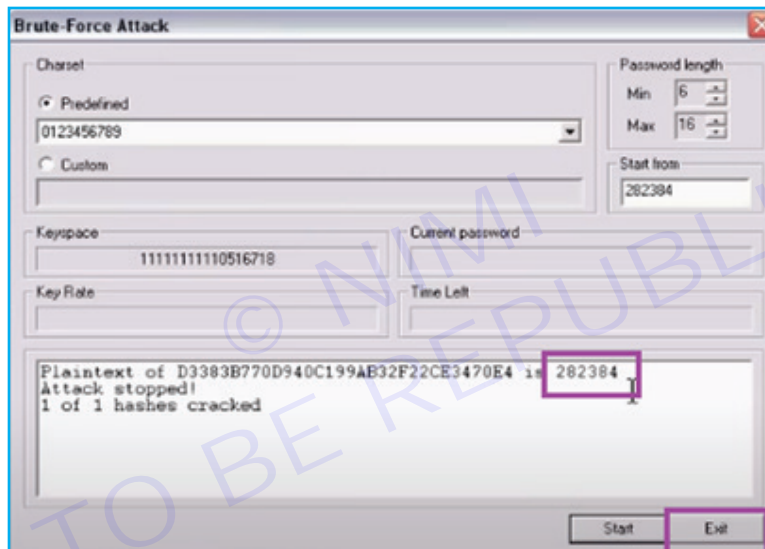
7 Choose the account for which you want to crack the password. Right-click on it, then opt for a brute force attack. Within the attack options, select NTLM hashes.



- Specify the Charset, Password length and then click on "Start". It will start making combinations and take some time to crack hashes.

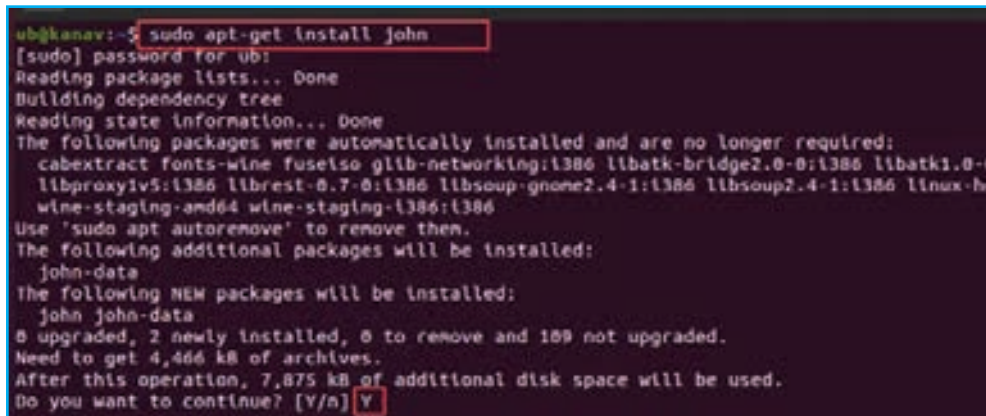


- Password cracked successfully.



TASK 2: Install John the Ripper Password Cracker

- Open the terminal in Linux distributions like Ubuntu, Fedora, Arch etc.
- Run the command `sudo apt-get install john`.
- Type Y and press enter to continue the installation.



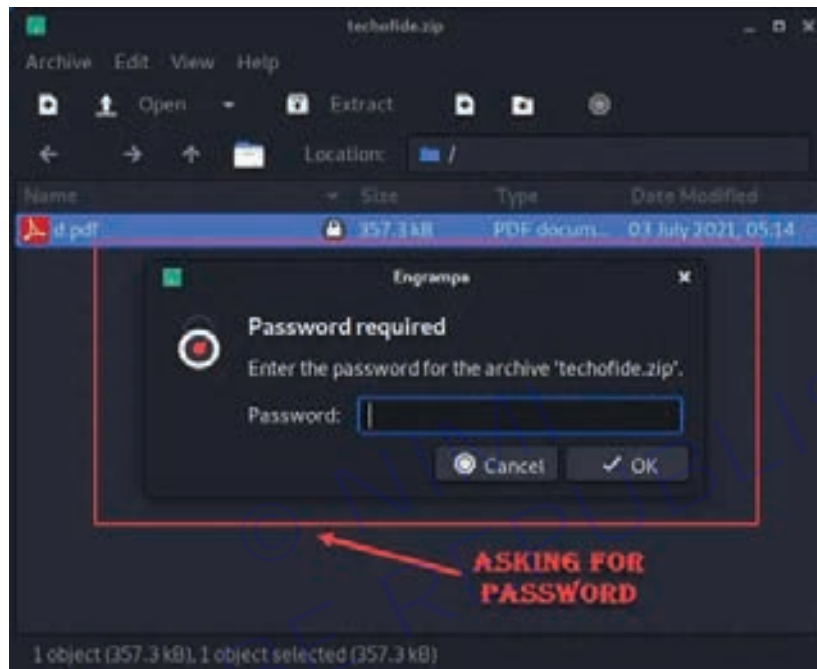
TASK 3: Use John the Ripper

- 1 Open the terminal
- 2 Type the command `john hash.txt --format=RAW-MD5` and hit enter

Note: type john followed by the hash file that you want to crack and then just define the format of the hash.

TASK 4: Cracking Zip File using John the Ripper

- 1 Select a zip folder (techofide.zip) which is password protected and asking for a password to open it.



- 2 Generate a hash of our zip file using the command `sudo zip2john techofide.zip > hash.txt`.

```
(kanav@Techofide)-[~/Desktop]
└─$ sudo zip2john techofide.zip > hash.txt
[sudo] password for kanav:
ver 2.0 efh 9901 techofide.zip/d.pdf PKZIP Encr: cmlen=340694, decmlen=357334, crc=CA455687
```

- 3 Use hash of our zip file and apply the command `sudo john --format=zip hash.txt` to crack the password.

```
(kanav@Techofide)-[~/Desktop]
└─$ sudo john --format=zip hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (ZIP, WinZip [PBKDF2-SHA1 256/256 AVX2 8x])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 3 candidates buffered for the current salt, minimum 16 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 16 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 6 candidates buffered for the current salt, minimum 16 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
54321 (techofide.zip/d.pdf)
lg 0:00:00:01 DONE 2/3 (2021-07-07 12:26) 0.9615g/s 34925p/s 34925c/s 34925C/s 123456..Peter
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

EXERCISE 17 : Configuring Firewalls

Objectives

At the end of this exercise you shall be able to

- setup the windows firewall
- crack firewall settings on the PC.

Requirements

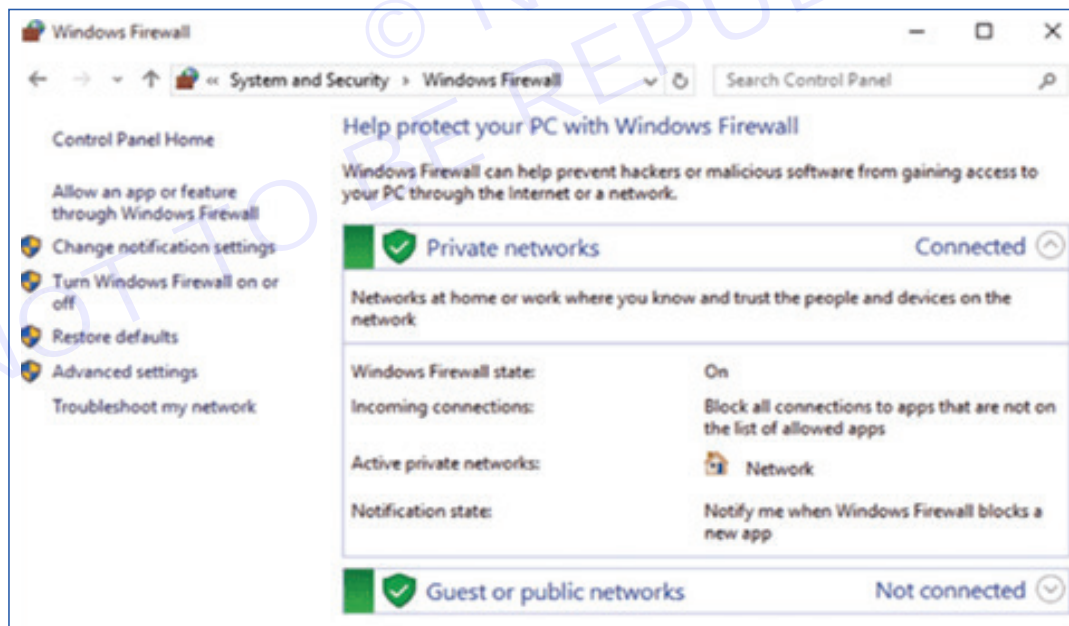
Tools/Materials

- Windows PC/Laptop with internet connectivity

Procedure

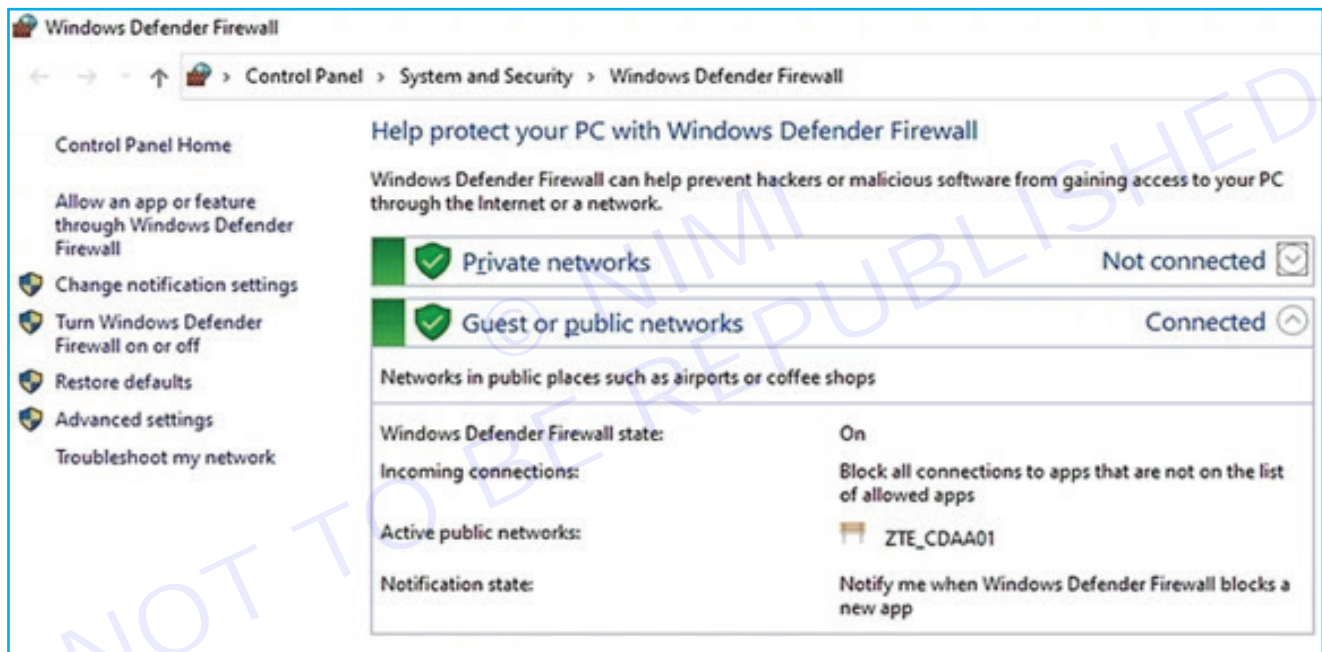
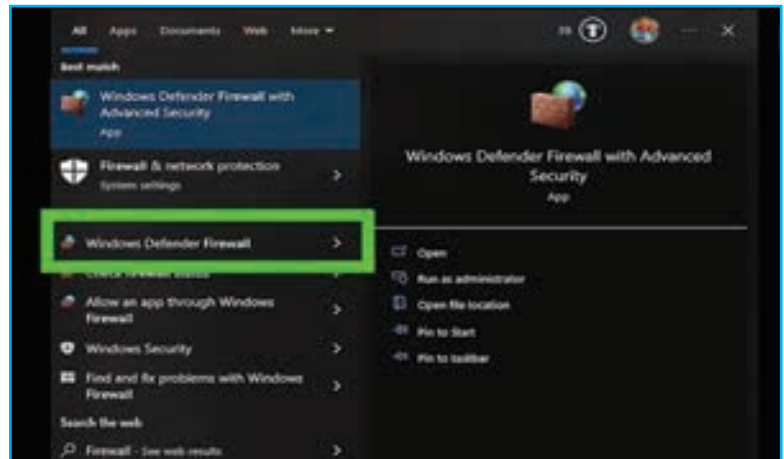
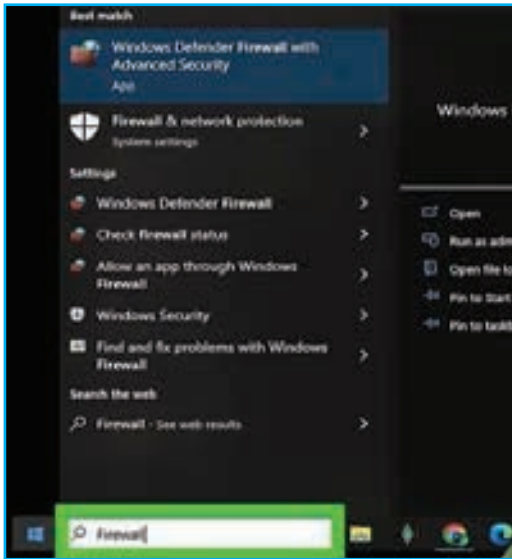
TASK 1: Set Up the Windows Firewall

- 1 Open the Control Panel.
- 2 Click the System and Security heading.
- 3 Click the Windows Firewall heading.
- 4 The Windows Firewall window appears.
- 5 To change the setting, click the Turn Windows Firewall “On” or “Off” link on the left side of the Windows Firewall window.

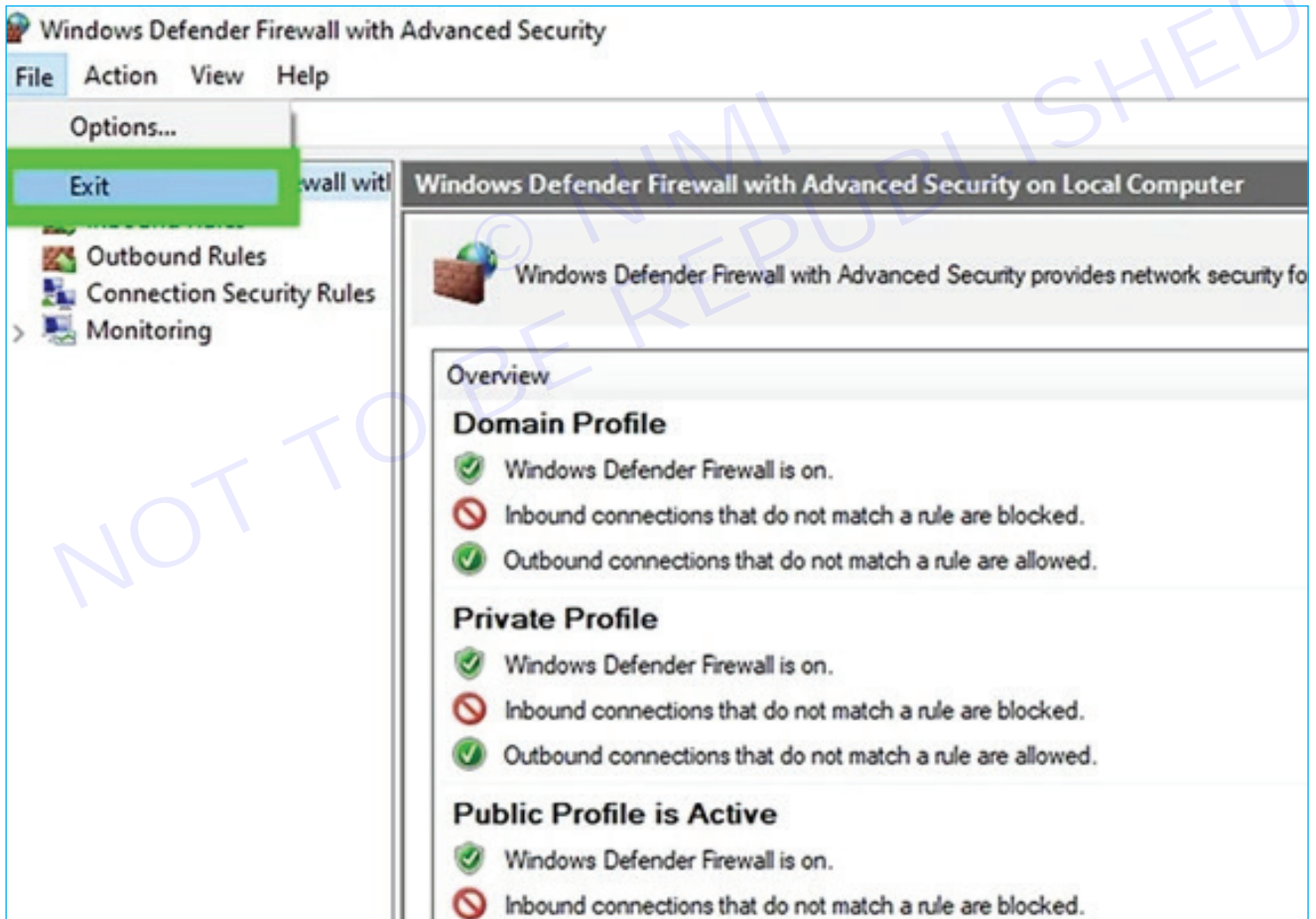
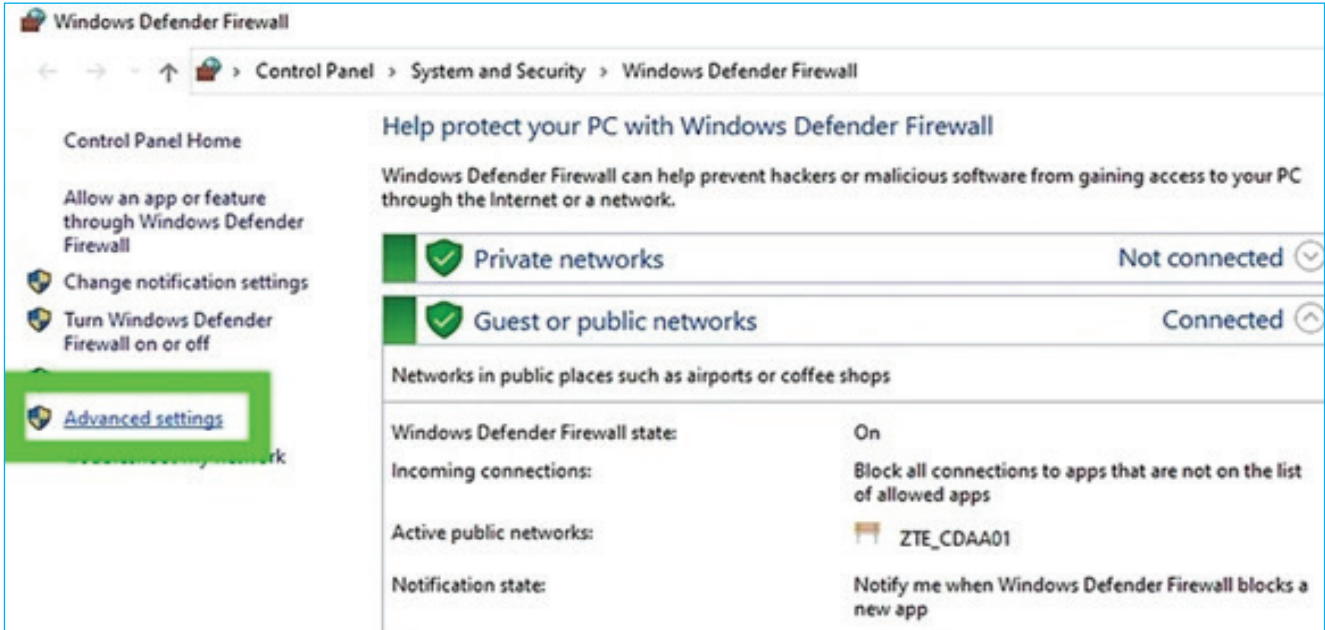


TASK 2: Checking Firewall Settings on a PC

- 1 Open your Start menu.
- 2 Type “firewall” into the search bar.
- 3 Click the “Windows Firewall” option on the top of the search window.
- 4 “Private networks” and “Guest or public networks” with green shields to the left of them, signifying that the firewall is active.



- 5 Click the "Advanced Settings" option to alter the following:
 - a "Inbound Rules" - Which incoming connections are automatically allowed.
 - b "Outbound Rules" - Which outgoing connections are automatically allowed.
 - c "Connection Security Rules" - Baselines for which connections your computer will allow and which ones it will block.
 - d "Monitoring" - An overview of your firewall's basic monitoring guidelines.
- 6 Exit the Advanced Settings menu when you're finished.



◆ Module 2 : Data base concepts ◆

EXERCISE 18 : Installation Steps of MySQL

Objectives

At the end of this exercise you shall be able to

- download MySQL installer
- know about the MySQL installation steps & procedure
- check and verify MySQL by using command prompt.

Requirements

Tools/Materials

- Operating System: Windows 10 or 11 (64-bit)
- MySQL Setup Software
- Desktop/Laptop with latest configuration

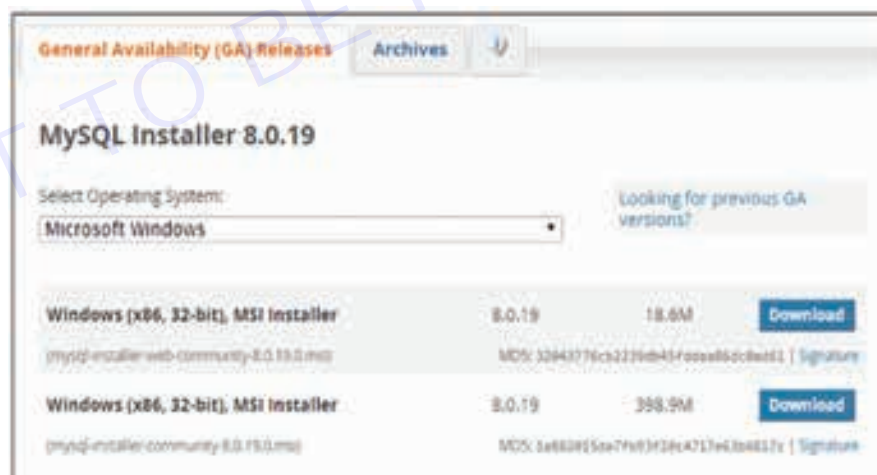
Procedure

Installation Methods

TASK 1 : Download MySQL

Step 1: Go to the official website of MySQL and download the community server edition software. Here, you will see the option to choose the Operating System, such as Windows.

Step 2: Next, there are two options available to download the setup. Choose the version number for the MySQL community server, which you want. If you have good internet connectivity, then choose the mysql-installer-web-community. Otherwise, choose the other one.

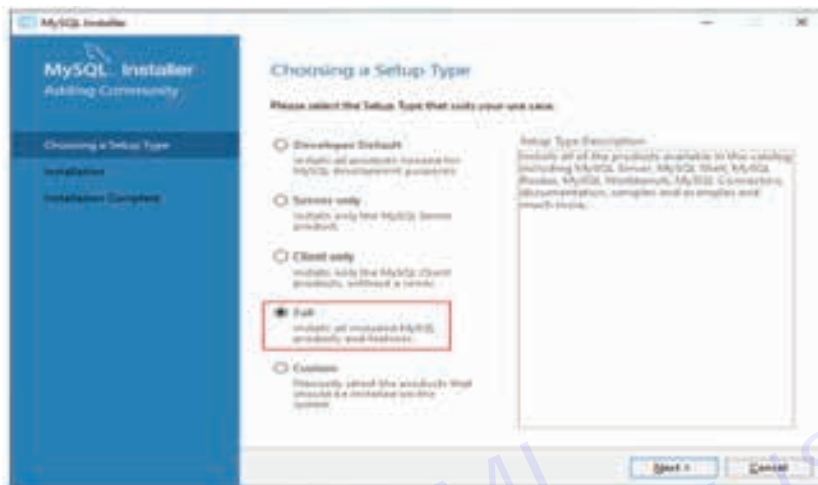
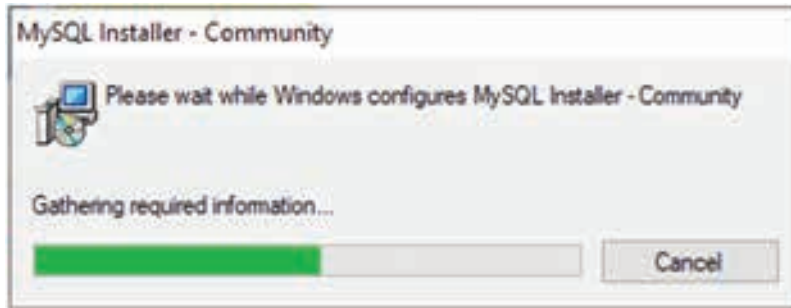


TASK 2 :Installing MySQL on Windows

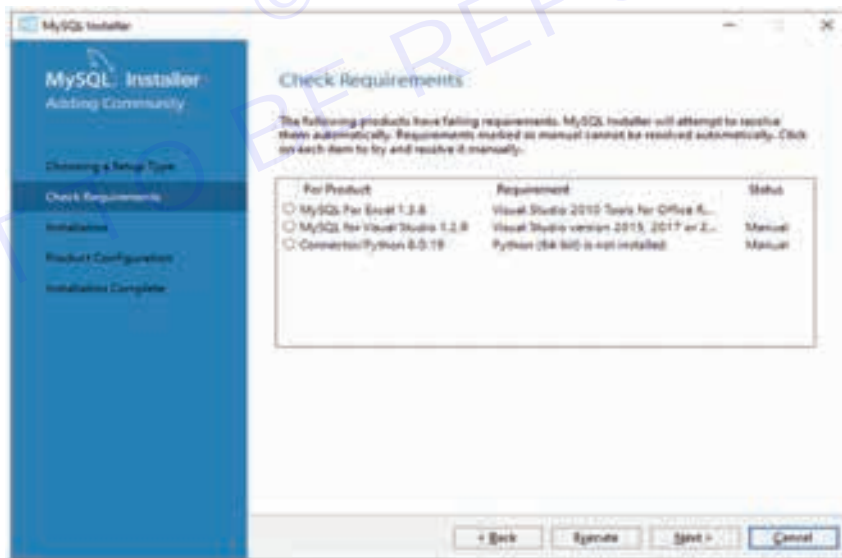
Step 1: After downloading the setup, unzip it anywhere and double click the MSI installer .exe file. It will give the following screen:

Step 2: In the next wizard, choose the Setup Type. There are several types available, and you need to choose the appropriate option to install MySQL product and features. Here, we are going to select the Full option and click on the Next button.

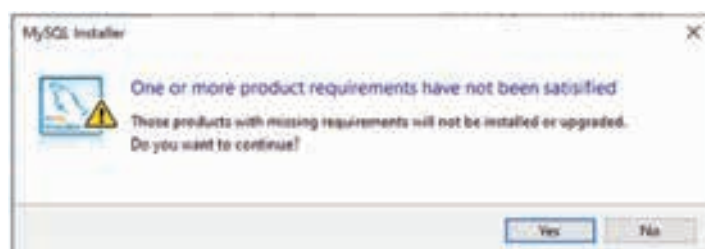
This option will install the following things: MySQL Server, MySQL Shell, MySQL Router, MySQL Workbench, MySQL Connectors, documentation, samples and examples, and many more.



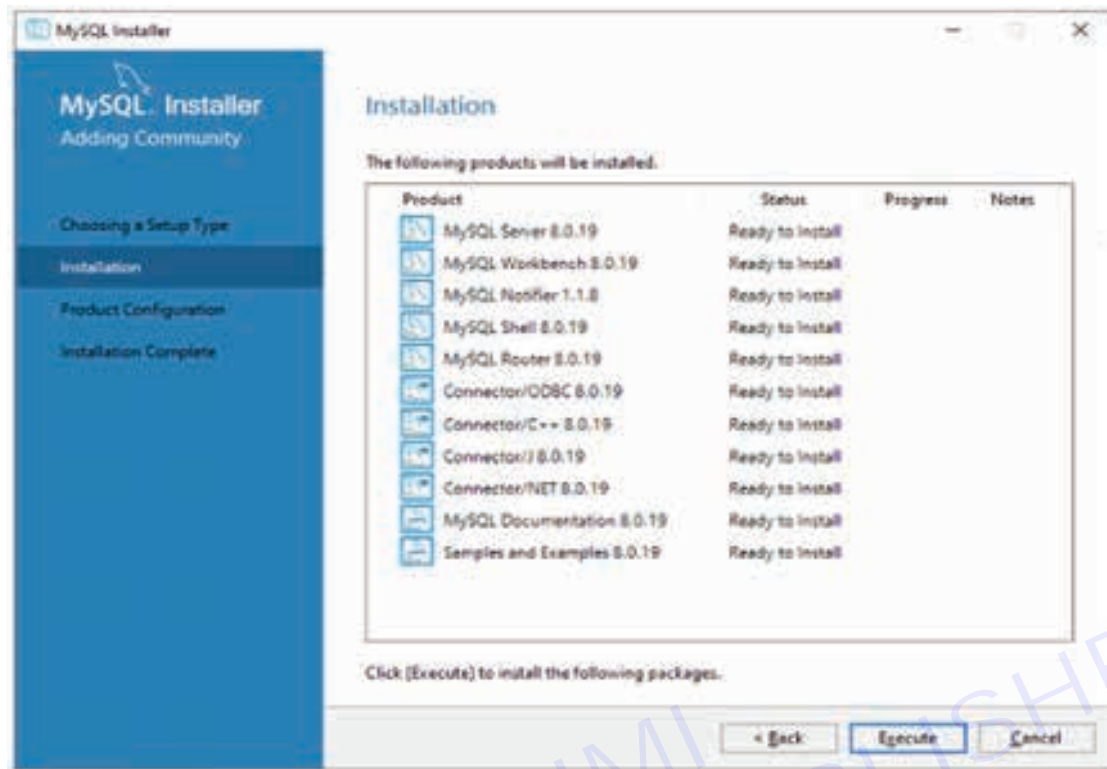
Step 3: Once we click on the Next button, it may give information about some features that may fail to install on your system due to a lack of requirements. We can resolve them by clicking on the Execute button that will install all requirements automatically or can skip them. Now, click on the Next button.



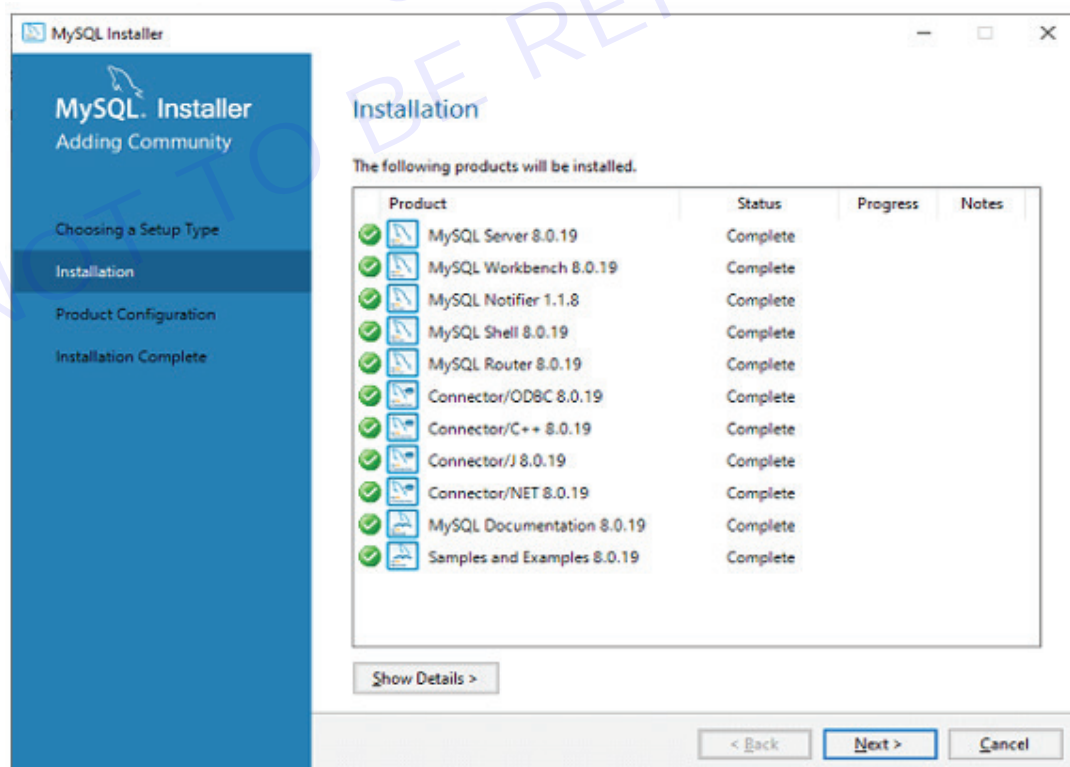
Step 4: In the next wizard, we will see a dialog box that asks for our confirmation of a few products not getting installed. Here, we have to click on the Yes button.



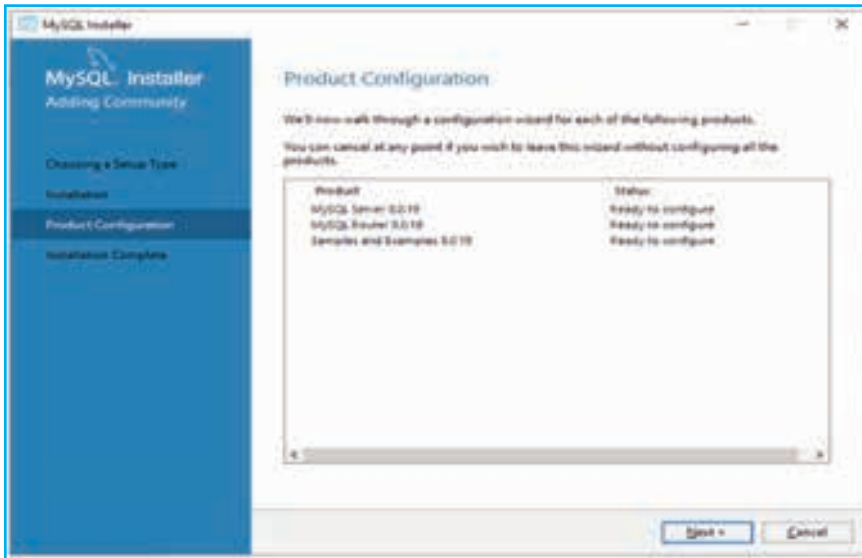
After clicking on the Yes button, we will see the list of the products which are going to be installed. So, if we need all products, click on the Execute button.



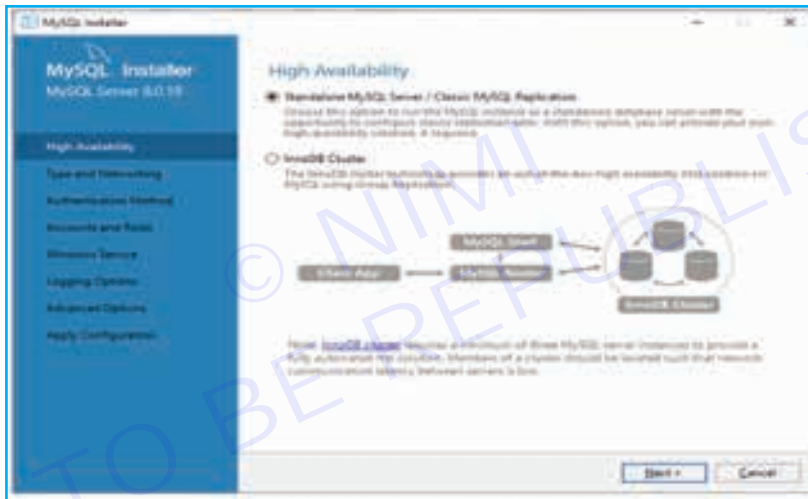
Step 5: Once we click on the Execute button, it will download and install all the products. After completing the installation, click on the Next button.



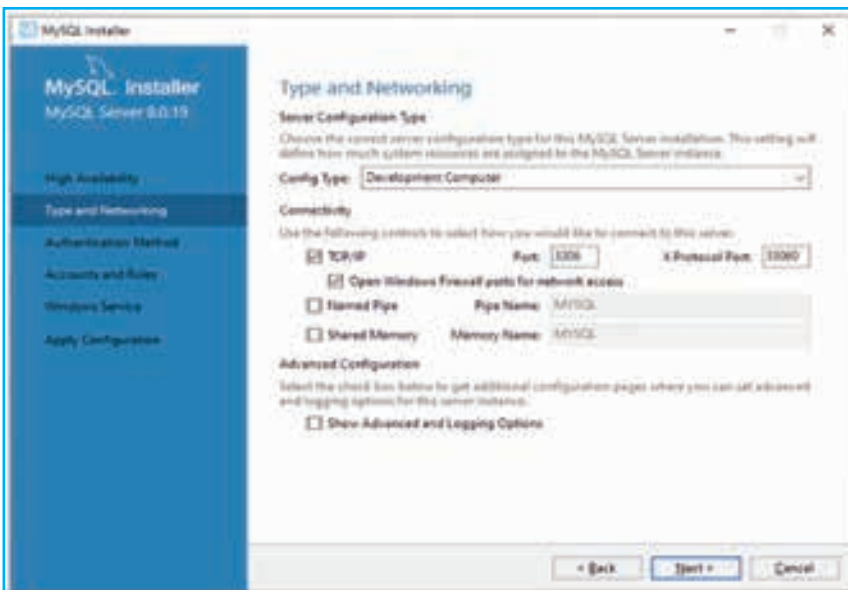
Step 6: In the next wizard, we need to configure the MySQL Server and Router. Here, I am not going to configure the Router because there is no need to use it with MySQL. We are going to show you how to configure the server only. Now, click on the Next button.



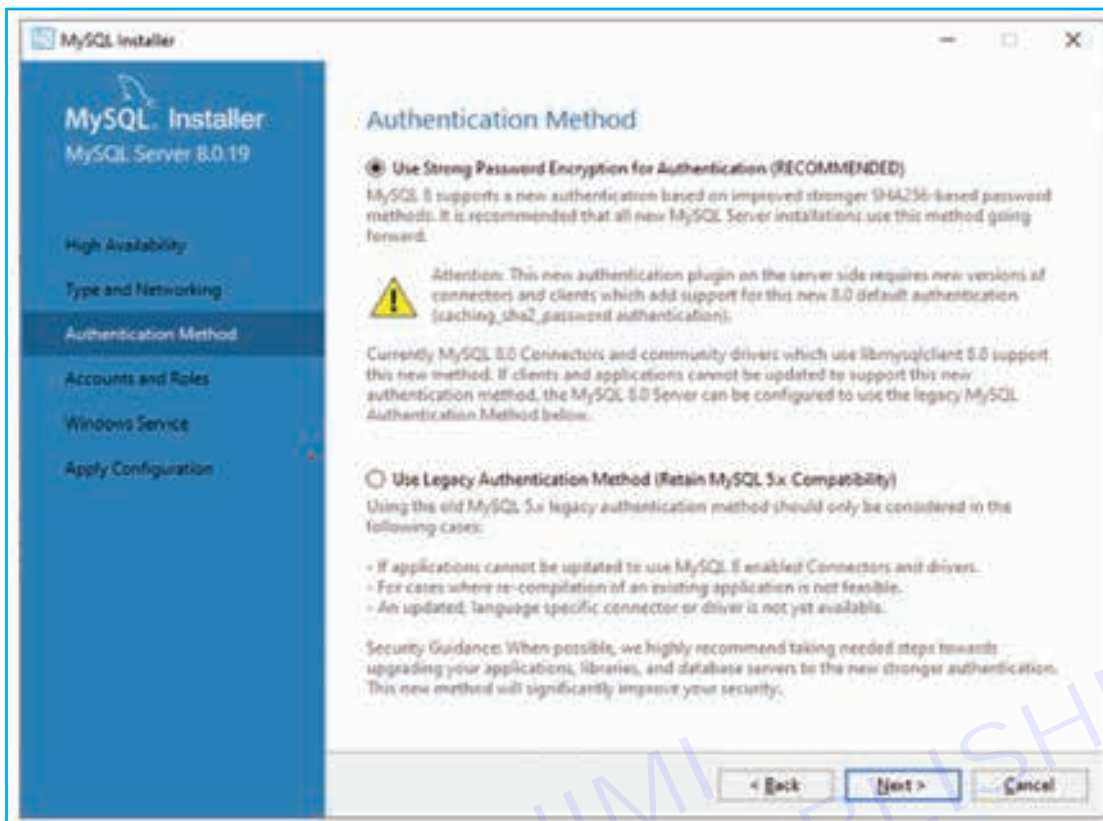
Step 7: As soon as you will click on the Next button, you can see the screen below. Here, we have to configure the MySQL Server. Now, choose the Standalone MySQL Server/Classic MySQL Replication option and click on Next. Here, you can also choose the InnoDB Cluster based on your needs.



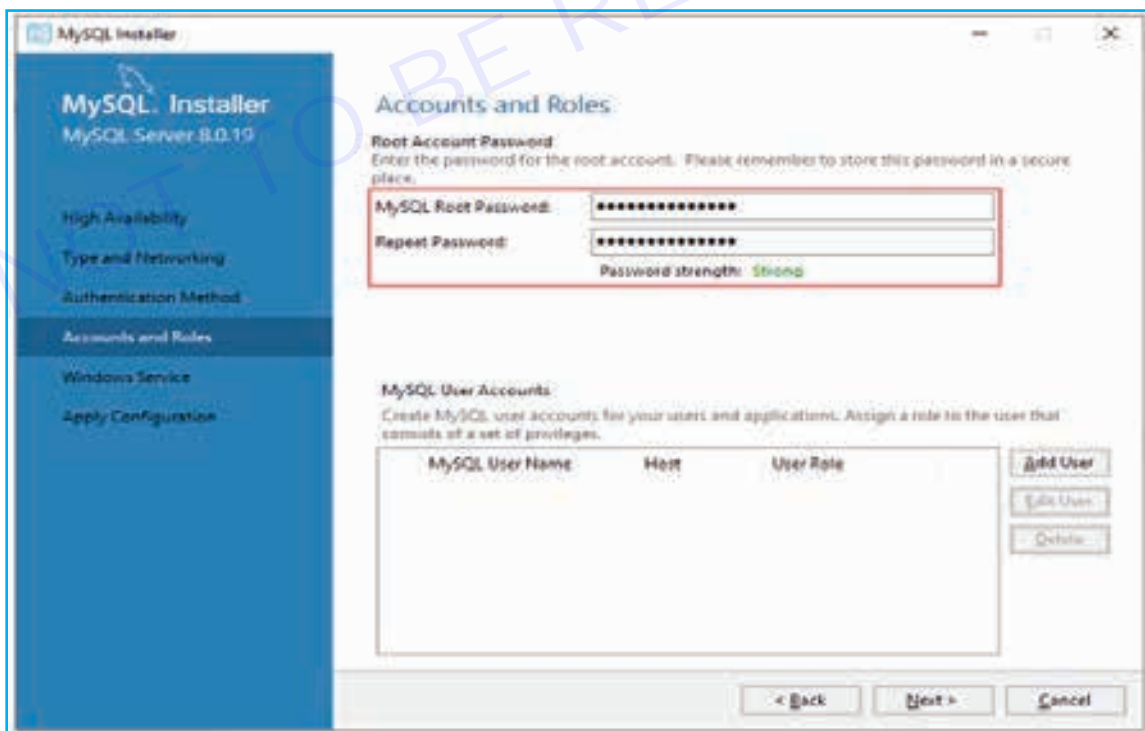
Step 8: In the next screen, the system will ask you to choose the Config Type and other connectivity options. Here, we are going to select the Config Type as 'Development Machine' and Connectivity as TCP/IP, and Port Number is 3306, then click on Next.



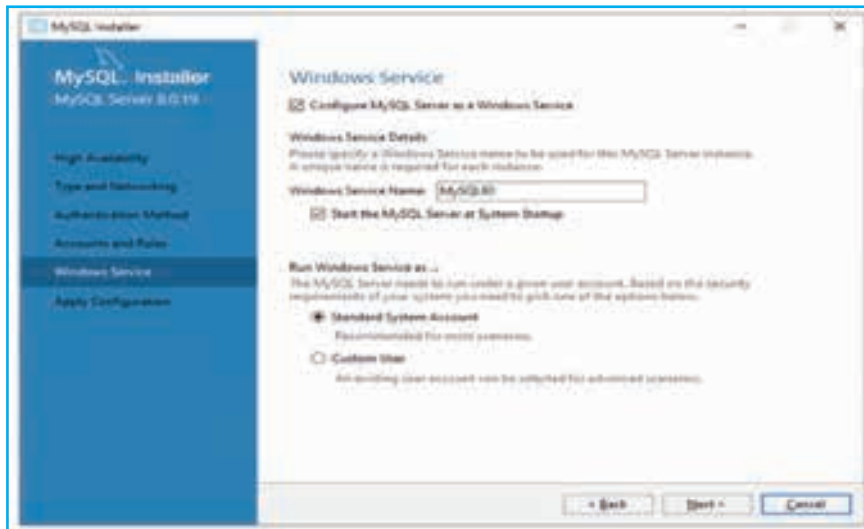
Step 9: Now, select the Authentication Method and click on Next. Here, I am going to select the first option.



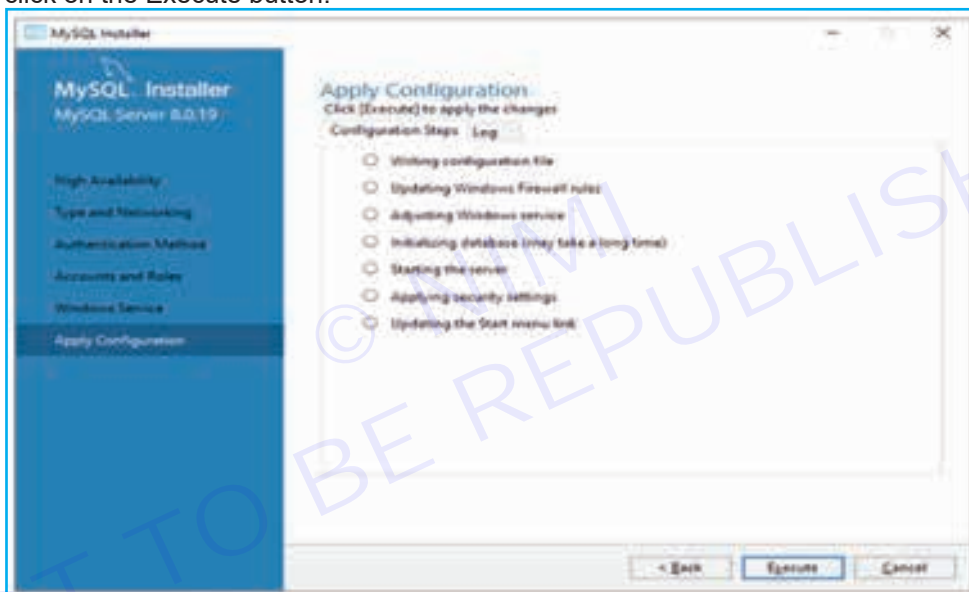
Step 10: The next screen will ask you to mention the MySQL Root Password. After filling the password details, click on the Next button.



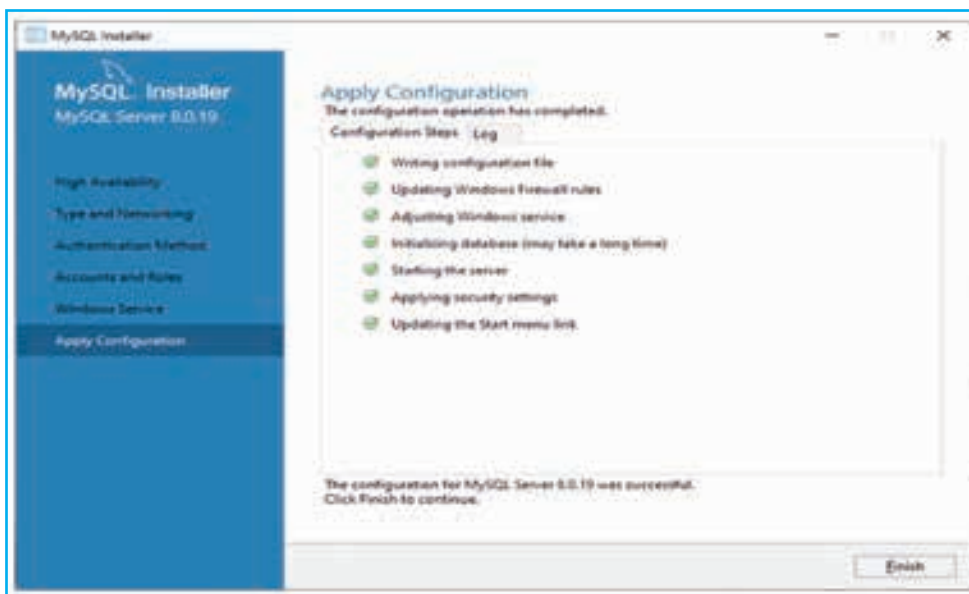
Step 11: The next screen will ask you to configure the Windows Service to start the server. Keep the default setup and click on the Next button.



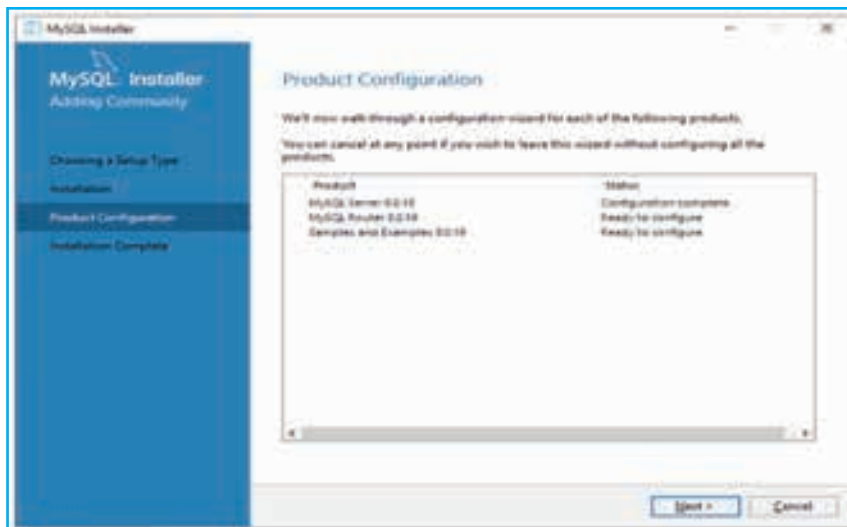
Step 12: In the next wizard, the system will ask you to apply the Server Configuration. If you agree with this configuration, click on the Execute button.



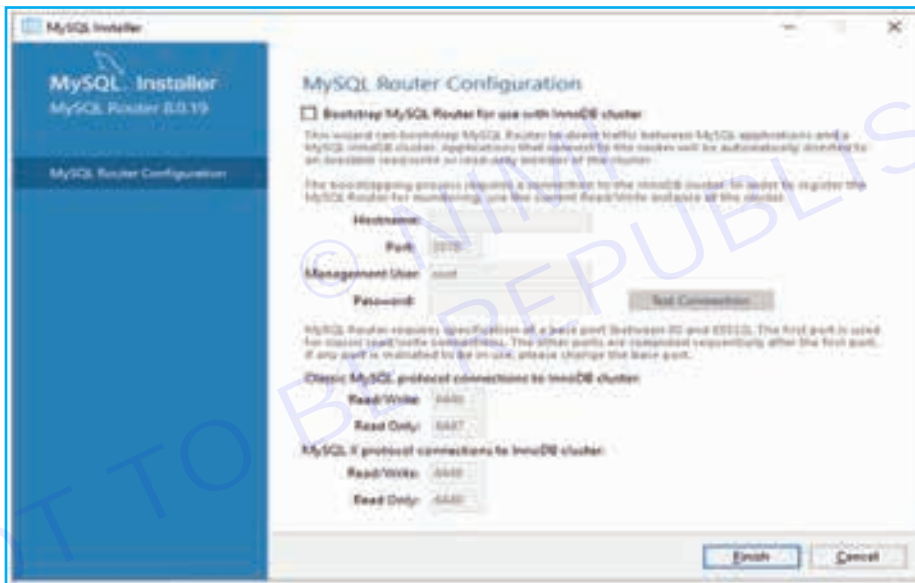
Step 13: Once the configuration has completed, you will get the screen below. Now, click on the Finish button to continue.



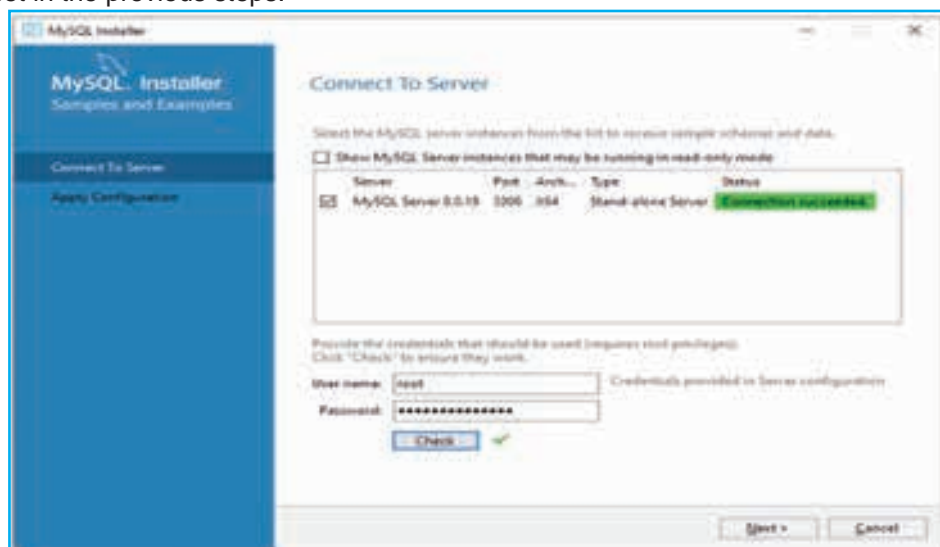
Step 14: In the next screen, you can see that the Product Configuration is completed. Keep the default setting and click on the Next-> Finish button to complete the MySQL package installation.



Step 15: In the next wizard, we can choose to configure the Router. So click on Next->Finish and then click the Next button.

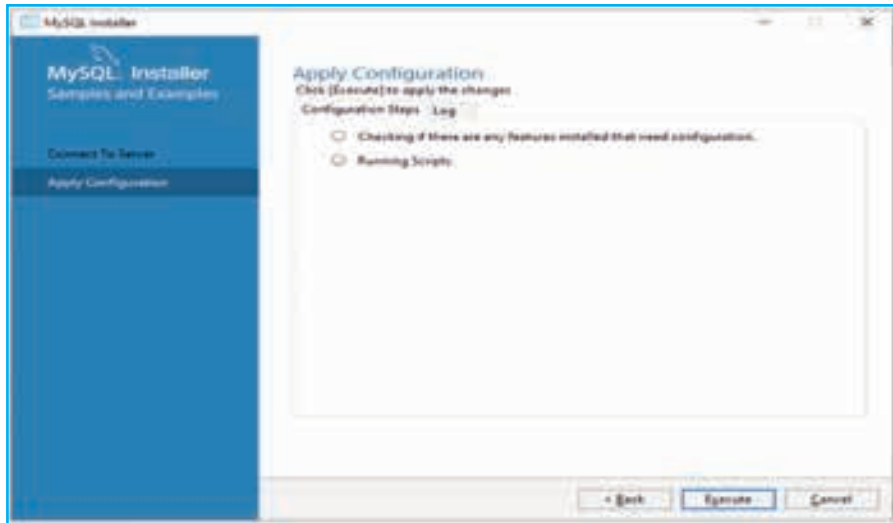


Step 16: In the next wizard, we will see the Connect to Server option. Here, we have to mention the root password, which we had set in the previous steps.

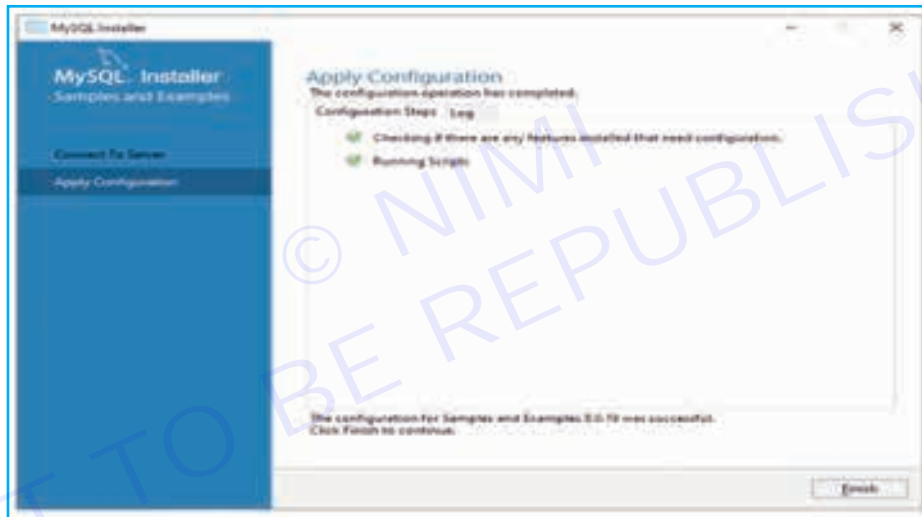


It is also required to check about the connection is successful or not by clicking on the Check button. If the connection is successful, click on the Execute button. Now, the configuration is complete, click on Next.

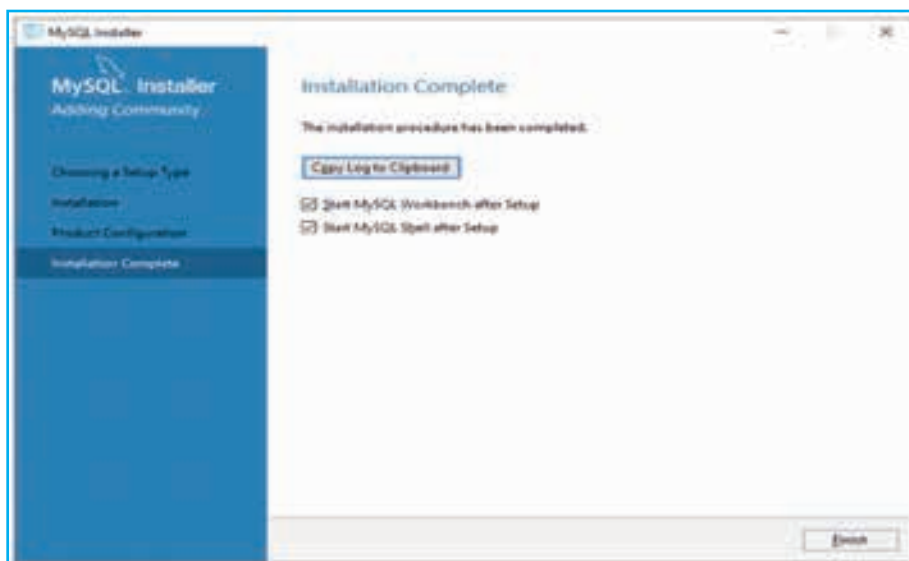
Step 17: In the next wizard, select the applied configurations and click on the Execute button.



Step 18: After completing the above step, we will get the following screen. Here, click on the Finish button.



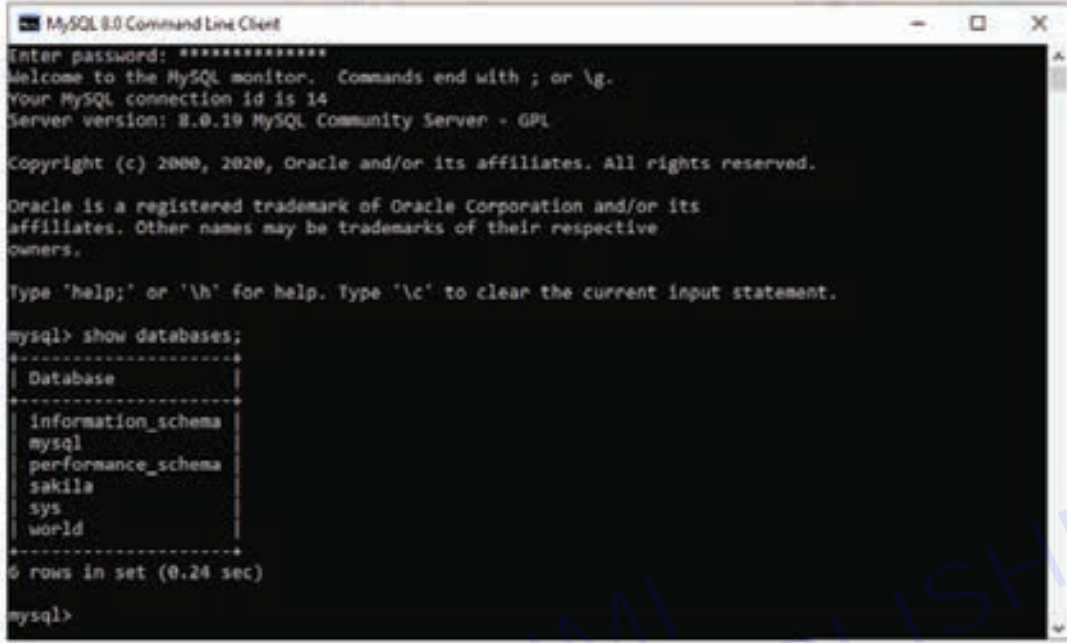
Step 19: Now, the MySQL installation is complete. Click on the Finish button.



TASK 3 : Verify MySQL

After Installation, Open your MySQL Command Line Client; it should have appeared with a `mysql>` prompt. If you have set any password, write your password here. Now, you are connected to the MySQL server, and you can execute all the SQL command at `mysql>` prompt as follows:

For example: Check the already created databases with `show databases` command:



```
MySQL 8.0 Command Line Client
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.19 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sakila |
| sys |
| world |
+-----+
6 rows in set (0.24 sec)

mysql>
```

MY SQL Installation & Verification Complete.

EXERCISE 19 : Troubleshooting Basic Installation Issues

Objectives

At the end of this exercise you shall be able to

- troubleshoot basic installed issues in MySQL
- verify installation status
- do advance trouble shooting

Requirements

Tools/Materials

- Operating System: Windows 10 or 11 (64-bit)
- MySQL Setup Software
- Desktop/Laptop will latest configuration

Procedure

Troubleshooting Basic Installation Issues in MySQL on Windows:

TASK 1 : Check Error Logs

- **Location:** The default location for the error log is C:\ProgramData\MySQL\MySQL Server 8.0\data\mysql.err.
- **Analysis:** Open the error log and search for specific error messages. Here are some common errors and their meanings:
- **Access denied:** Check the root password or permissions related to data directories.
- **Port conflicts:** Another program might be using the default port (3306). Disable conflicting programs or change the MySQL port.
- **Missing files or directories:** Verify that all installation files are present and in the correct locations.
- **Configuration errors:** Double-check your configuration settings in my.ini based on your specific setup.

TASK 2 : Verify Installation Status

- Open the “Services” app: On Windows 10,
- press Win+R, type services.msc, and press Enter. On Windows 11,
- right-click the Start button, select “Settings”, then “Apps & features”, and scroll down to “Optional features”.
- Under “More Windows features”, click “Add features”. Look for “MySQL Server” and install it if not already present.
- Check the status: In the “Services” app, locate “MySQL80” (or your version number) and check its status. If it’s not running, try starting it manually.

TASK 3 : Address Common Issues

- **Insufficient Permissions:** Ensure the MySQL service has the necessary permissions to access its data directories and files.
- **Antivirus/Firewall Interference:** Temporarily disable antivirus or firewall software to see if it’s blocking MySQL.
- **Incorrect Data Directory:** Verify the data directory path in the my.ini file matches the actual location.
- **Missing Visual C++ Redistributables:** Install the correct version of Visual C++ Redistributables, downloadable from Microsoft’s website.

TASK 4 : Advanced Troubleshooting

- **Use mysqldump to check data integrity:** After successful installation, consider using mysqldump to back up existing data if possible. If data corruption occurred during installation, you can potentially restore it after resolving the issue.
- **Reinstall MySQL:** If all else fails, consider reinstalling MySQL using the latest installer from the official website. Make sure to back up any existing data you may have.

EXERCISE 20 : Creating and Using a Database in MySQL

Objectives

At the end of this exercise you shall be able to

- create the database & use it
- create the table and use it & perform insert ,Select,update and delete queries

Requirements

Tools/Materials

- Operating System: Windows 10 or 11 (64-bit)
- MySQL Setup Software
- Desktop/Laptop with latest configuration

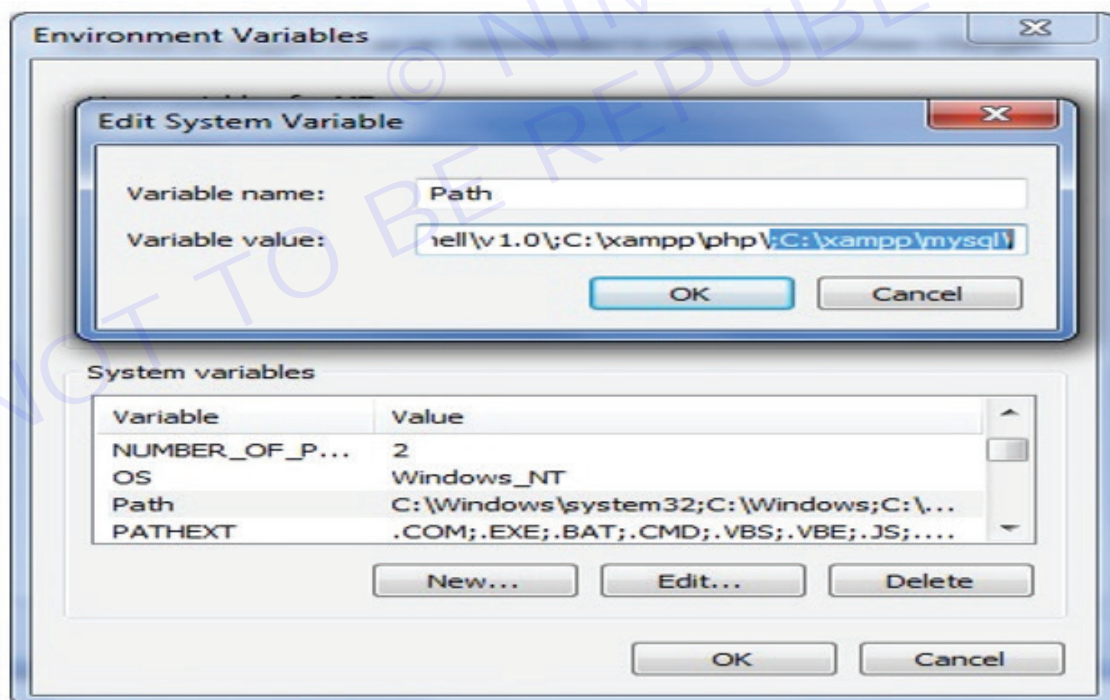
Procedure

TASK 1 : Creating and Using a Database in MySQL

1 Open the MySQL Command Line Client:

Windows: Search for “MySQL Shell” or “mysql.exe” in the Start menu.

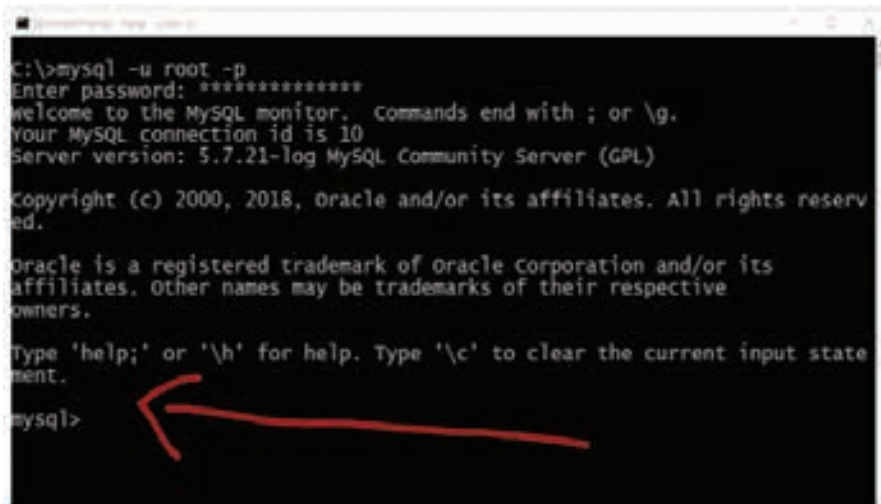
macOS/Linux: Open a terminal and type mysql (if installed globally) or path/to/your/mysql/bin/mysql.



2 Connect to the MySQL Server:

Type `mysql -u root -p`

Enter your root password when prompted.



```

C:\>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input state
ment.

mysql>

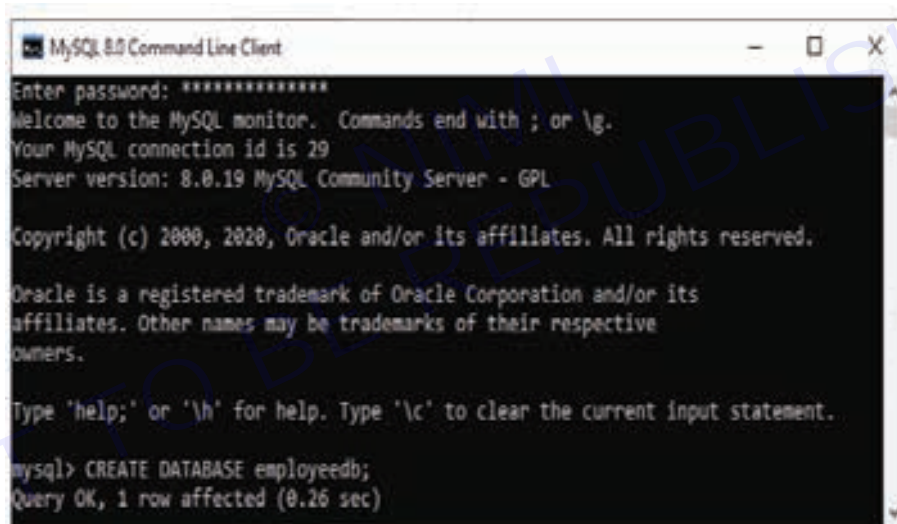
```

3 Create a Database:

Use the CREATE DATABASE command:

CREATE DATABASE database name;

Replace database name with the desired name for your database (e.g., employeedb).



```

MySQL 8.0 Command Line Client
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 29
Server version: 8.0.19 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE employeedb;
Query OK, 1 row affected (0.26 sec)

```

4 Select the Database:

Use the USE command:

USE database name;

This switches the focus to the newly created database.

5 Create a Table:

Use the CREATE TABLE command:

SQL

```
CREATE TABLE table name (
  column1 name data type PRIMARY KEY,
  column2 name data type,
  column3 name data type,
  ...
);
```

Replace:

table name with the desired name for your table (e.g., employee table).

column name with the names of each data column.

data type with the appropriate data type for each column (e.g., INT, VARCHAR, DATE).

PRIMARY KEY specifies the unique identifier for each row (usually an auto-incrementing INT).



```
MySQL 8.0 Command Line Client
Database
-----+-----
employeeeb
information_schema
myemployeeeb
mysql
mysqltestdb
mystudentdb
mytestdb_copy
performance_schema
sakila
sys
testdb
world
-----+-----
12 rows in set (0.01 sec)

mysql> USE employeeeb;
Database changed
mysql> CREATE TABLE employee_table(
-> id int NOT NULL AUTO_INCREMENT,
-> name varchar(45) NOT NULL,
-> occupation varchar(35) NOT NULL,
-> age int NOT NULL,
-> PRIMARY KEY (id)
-> );
Query OK, 0 rows affected (1.47 sec)
```

6 Insert Data:

Use the INSERT INTO command:

```
INSERT INTO table name (column1 name, column2 name, column3 name)
VALUES (value1, value2, value3);
```

Replace:

table name with the name of your table.

column name with the names of the columns you want to insert data into.

value with the actual data for each column (e.g., numbers, strings, dates).

7 Retrieve Data:

Use the SELECT command:

```
SELECT * FROM table name;
```

This retrieves all data from the table.

You can modify the SELECT command to specify specific columns or filter data based on conditions.

```

mysql> CREATE TABLE Student (
-> Stud_ID int AUTO_INCREMENT PRIMARY KEY,
-> Name varchar(45) DEFAULT NULL,
-> Email varchar(45) NOT NULL UNIQUE,
-> City varchar(25) DEFAULT NULL
-> );
Query OK, 0 rows affected (0.88 sec)

mysql> INSERT INTO Student(Stud_ID, Name, Email, City)
-> VALUES (1,'Stephen', 'stephen@javatpoint.com', 'Texax'),
-> (2, 'Joseph', 'Joseph@javatpoint.com', 'Alaska'),
-> (3, 'Peter', 'Peter@javatpoint.com', 'california');
Query OK, 3 rows affected (0.18 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> SELECT * FROM Student;
+-----+-----+-----+-----+
| Stud_ID | Name   | Email                               | City   |
+-----+-----+-----+-----+
| 1       | Stephen | stephen@javatpoint.com             | Texax  |
| 2       | Joseph  | Joseph@javatpoint.com              | Alaska |
| 3       | Peter   | Peter@javatpoint.com               | california |
+-----+-----+-----+-----+
    
```

8 Modify Data:

Use the UPDATE command:

UPDATE table name SET column name = new value WHERE condition;

Replace:

table name with the name of your table.

column name with the name of the column you want to update.

new value with the new data for the column.

condition specifies which rows to update (e.g., WHERE id = 1).

```

mysql> UPDATE trainer
-> SET email = 'mike@tutorialandexamples.com'
-> WHERE course_name = 'Java';
Query OK, 1 row affected (0.26 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> SELECT * FROM trainer;
+-----+-----+-----+
| course_name | trainer | email                               |
+-----+-----+-----+
| Java        | Mike    | mike@tutorialandexamples.com       |
| Python     | James   | james@javatpoint.com               |
| Android    | Robin   | robin@javatpoint.com               |
| Hadoop     | Stephen | stephen@javatpoint.com             |
| Testing    | Micheal | micheal@javatpoint.com             |
+-----+-----+-----+
    
```

9 Delete Data:

Use the DELETE FROM command:

DELETE FROM table name WHERE condition;

Replace:

table name with the name of your table.

condition specifies which rows to delete (e.g., WHERE age > 30).

```

mysql> SELECT * FROM City WHERE ID = 500;
+-----+-----+-----+-----+-----+
| ID | Name | CountryCode | District | Population |
+-----+-----+-----+-----+-----+
| 500 | Bolton | GBR | England | 139020 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> DELETE FROM City WHERE ID=500;
Query OK, 1 row affected (0.02 sec)

mysql> SELECT * FROM City WHERE ID = 500;
Empty set (0.01 sec)

mysql> MySQL Tutorials - www.TestingDocs.com

```

10 Exit MySQL:

Type exit or quit to disconnect from the server.

```

Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Mike>cd \
C:\>cd C:\Windows\Server\cores\mysql\bin
C:\Windows\Server\cores\mysql\bin>mysql -h127.0.0.1 -uroot -proot
Warning! Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.6.31 MySQL Community Server (GPL)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> exit
Bye

```

EXERCISE 21 : Designing of Tables

Objectives

At the end of this exercise you shall be able to

- connect the MySQL database server
- create database and table
- perform insert,select,delete queries

Requirements

Tools/Materials

- Operating System: Windows 10 or 11 (64-bit)
- MySQL Setup Software
- Desktop/Laptop with latest configuration

Procedure

TASK 1 : Creating Table, Insert values and retrieve data from that table

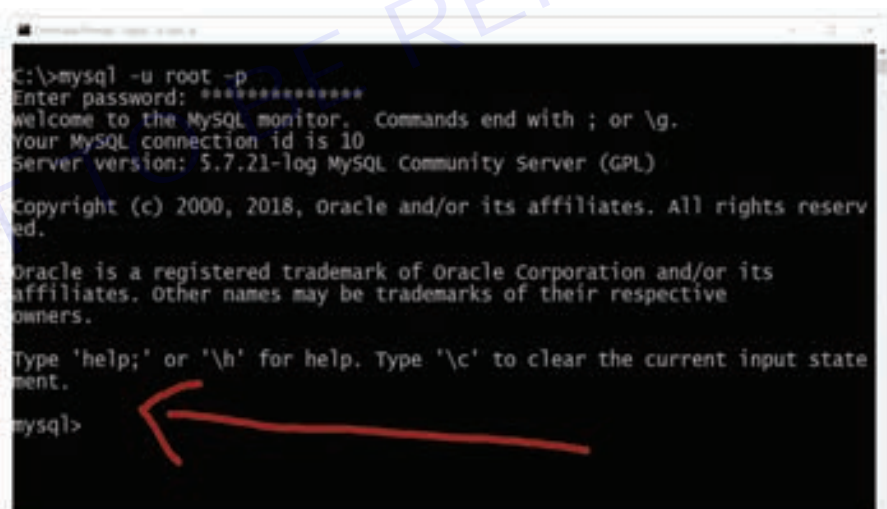
Step 1: Open MySQL Command Line:

Go to the Start menu, find MySQL, and open the MySQL Command Line client.

Step 2: Login to MySQL Server:

Enter the following command and press Enter. Replace username with your MySQL username, and you'll be prompted to enter your password.

```
mysql -u username -p
```



```
C:\>mysql -u root -p
Enter password: *****
welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

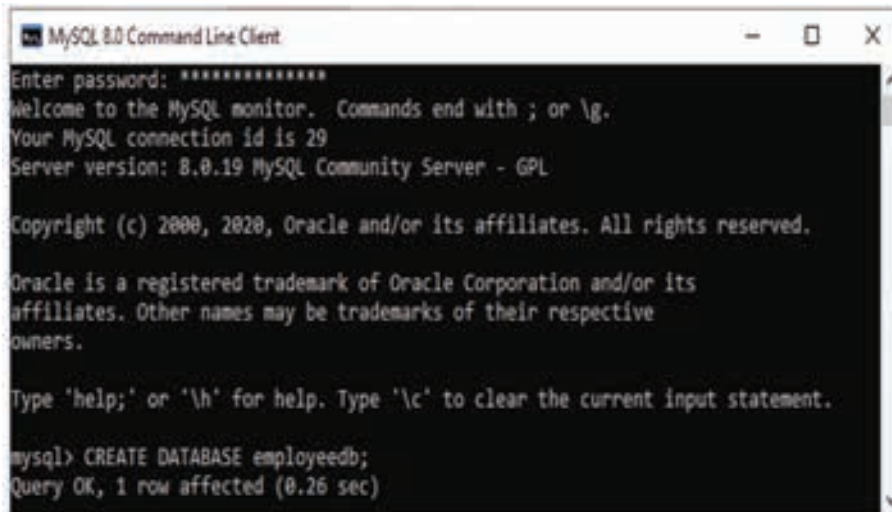
Type 'help;' or '\h' for help. Type '\c' to clear the current input state
ment.

mysql>
```

Step 3: Create a Database:

If you haven't created a database yet, create one using the following command. Replace yourdatabase with the desired name for your database.

```
CREATE DATABASE employeedb;
```



```

MySQL 8.0 Command Line Client
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 29
Server version: 8.0.19 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE employeeedb;
Query OK, 1 row affected (0.26 sec)

```

Step 4: Switch to the New Database:

Use the following command to switch to the newly created database.

USE employeeedb;

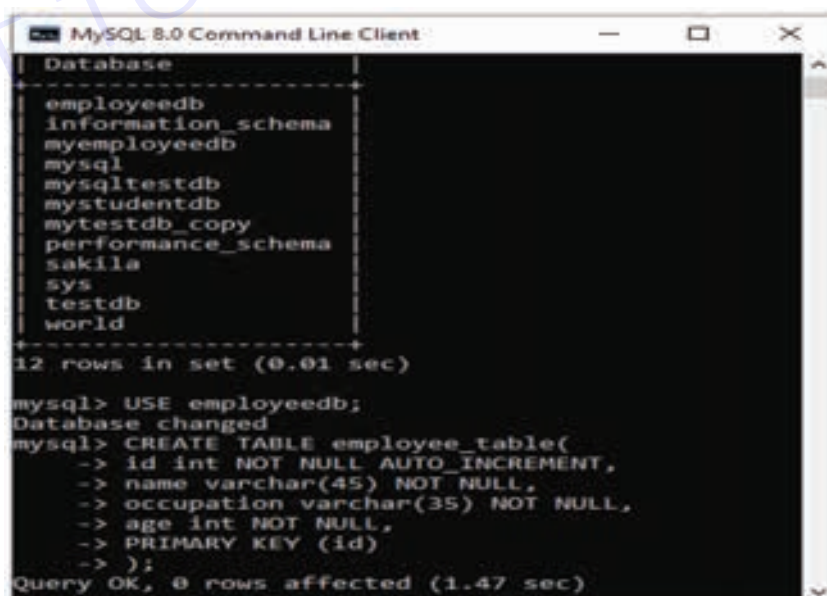
Step 5: Design Tables:

Create tables with the desired columns and data types. Here's an example for a simple "users" table:

```

CREATE TABLE users (
    user id INT AUTO INCREMENT PRIMARY KEY,
    username VARCHAR(50) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    password VARCHAR(255) NOT NULL,
    created at TIMESTAMP DEFAULT CURRENT TIMESTAMP
);

```



```

MySQL 8.0 Command Line Client
Database
+-----+
| employeeedb |
| information_schema |
| myemployeeedb |
| mysql |
| mysqltestdb |
| mystudentdb |
| mytestdb_copy |
| performance_schema |
| sakila |
| sys |
| testdb |
| world |
+-----+
12 rows in set (0.01 sec)

mysql> USE employeeedb;
Database changed
mysql> CREATE TABLE employee_table(
-> id int NOT NULL AUTO_INCREMENT,
-> name varchar(45) NOT NULL,
-> occupation varchar(35) NOT NULL,
-> age int NOT NULL,
-> PRIMARY KEY (id)
-> );
Query OK, 0 rows affected (1.47 sec)

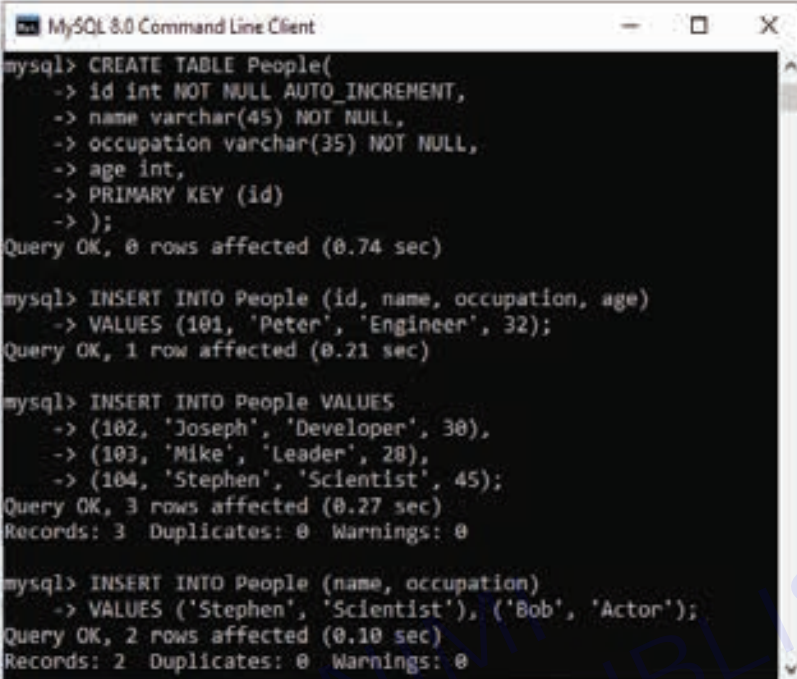
```

Step 6: Insert Data into the Table:

Once the table is created, you can insert data using the INSERT INTO statement:

```
INSERT INTO people (id, name, occupation, age)
```

```
VALUES (101, 'Peter', 'Engineer', 32);
```



```
mysql> CREATE TABLE People(
  -> id int NOT NULL AUTO_INCREMENT,
  -> name varchar(45) NOT NULL,
  -> occupation varchar(35) NOT NULL,
  -> age int,
  -> PRIMARY KEY (id)
  -> );
Query OK, 0 rows affected (0.74 sec)

mysql> INSERT INTO People (id, name, occupation, age)
  -> VALUES (101, 'Peter', 'Engineer', 32);
Query OK, 1 row affected (0.21 sec)

mysql> INSERT INTO People VALUES
  -> (102, 'Joseph', 'Developer', 30),
  -> (103, 'Mike', 'Leader', 28),
  -> (104, 'Stephen', 'Scientist', 45);
Query OK, 3 rows affected (0.27 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> INSERT INTO People (name, occupation)
  -> VALUES ('Stephen', 'Scientist'), ('Bob', 'Actor');
Query OK, 2 rows affected (0.10 sec)
Records: 2 Duplicates: 0 Warnings: 0
```

Step 7: Retrieve Data from the Table:

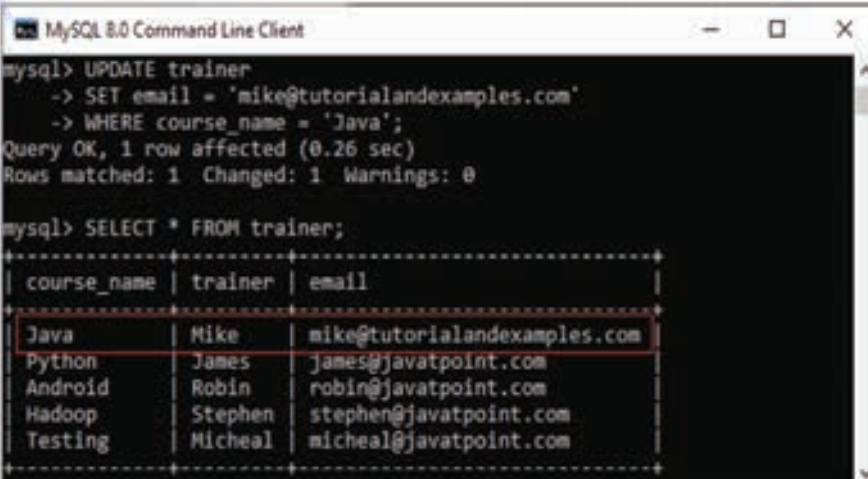
Use the SELECT statement to retrieve data from the table:

```
SELECT * FROM users;
```

Step 8: Update and Delete Data:

Modify existing data using the UPDATE statement and remove data using the DELETE statement:

```
UPDATE trainer SET email = 'mike@tutorialandexamples.com' WHERE course name = 'Java';
```



```
mysql> UPDATE trainer
  -> SET email = 'mike@tutorialandexamples.com'
  -> WHERE course_name = 'Java';
Query OK, 1 row affected (0.26 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> SELECT * FROM trainer;
+-----+-----+-----+
| course_name | trainer | email |
+-----+-----+-----+
| Java        | Mike   | mike@tutorialandexamples.com |
| Python     | James  | james@javatpoint.com |
| Android    | Robin  | robin@javatpoint.com |
| Hadoop     | Stephen | stephen@javatpoint.com |
| Testing    | Micheal | micheal@javatpoint.com |
+-----+-----+-----+
```

```
DELETE FROM city WHERE ID = 500;
```

```

mysql> SELECT * FROM City WHERE ID = 500;
+----+-----+-----+-----+-----+
| ID | Name  | CountryCode | District | Population |
+----+-----+-----+-----+-----+
| 500 | Bolton | GBR         | England | 139020     |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> DELETE FROM City WHERE ID=500;
Query OK, 1 row affected (0.02 sec)

mysql> SELECT * FROM City WHERE ID = 500;
Empty set (0.01 sec)

mysql> MySQL Tutorials - www.TestingDocs.com

```

Step 9: Managing Tables:

To see a list of tables in the current database, use the following command:

SHOW TABLES;

To get information about a specific table, you can use:

DESCRIBE yourtable;

```

C:\Windows\system32\cmd.exe - mysql -u root -p -h 127.0.0.1
mysql> use userlogin;
Database changed
mysql> show tables
+-----+
| Tables_in_userlogin |
+-----+
| user                 |
+-----+
1 row in set (0.00 sec)

mysql> describe user;
+----+-----+-----+-----+-----+-----+
| Field      | Type                | Null | Key | Default | Extra          |
+----+-----+-----+-----+-----+-----+
| id         | int(11) unsigned    | NO   | PRI | NULL    | auto_increment |
| username   | varchar(25)         | NO   |     | NULL    |                |
| passud     | varchar(25)         | NO   |     | NULL    |                |
| email      | varchar(40)         | NO   |     | NULL    |                |
+----+-----+-----+-----+-----+-----+
4 rows in set (0.05 sec)

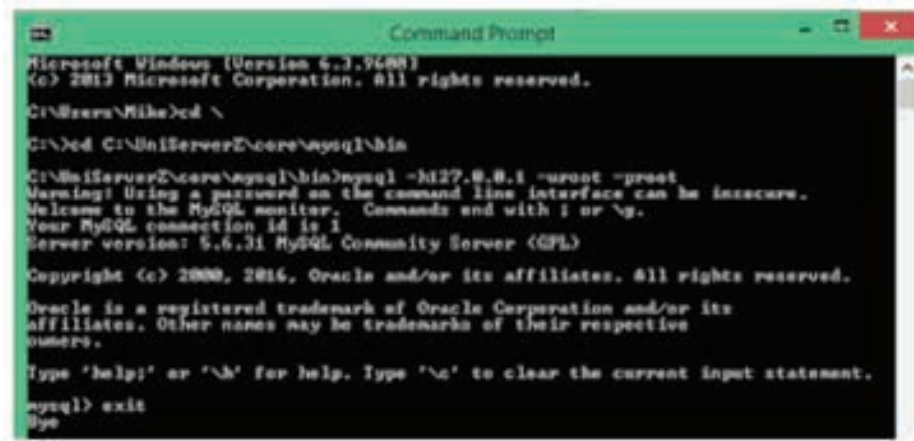
mysql> insert into user(username, passud, email)
-> values('prinotuber', '12345', '

```

Step 10: Exit MySQL Command Line:

When you're done working in the MySQL Command Line, you can exit by typing:

EXIT;



```
Microsoft Windows [Version 6.0.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Niki>cd \
C:\>cd C:\Bn\Server\care\mysql\bin
C:\Bn\Server\care\mysql\bin>mysql -h127.0.0.1 -uroot -proot
Warning! Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.6.31 MySQL Community Server (GPL)

Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> exit
bye
```

Create a table named CSA (Roll No., Stu name, Father Name, DOB, Adress, Phone No., Email) and insert 10 values in it.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 22 : Applying Data Integrity Rules

Objectives

At the end of this exercise you shall be able to

- start MySQL server and access MySQL
- design constraints
- check and verify integrity rules.

Requirements

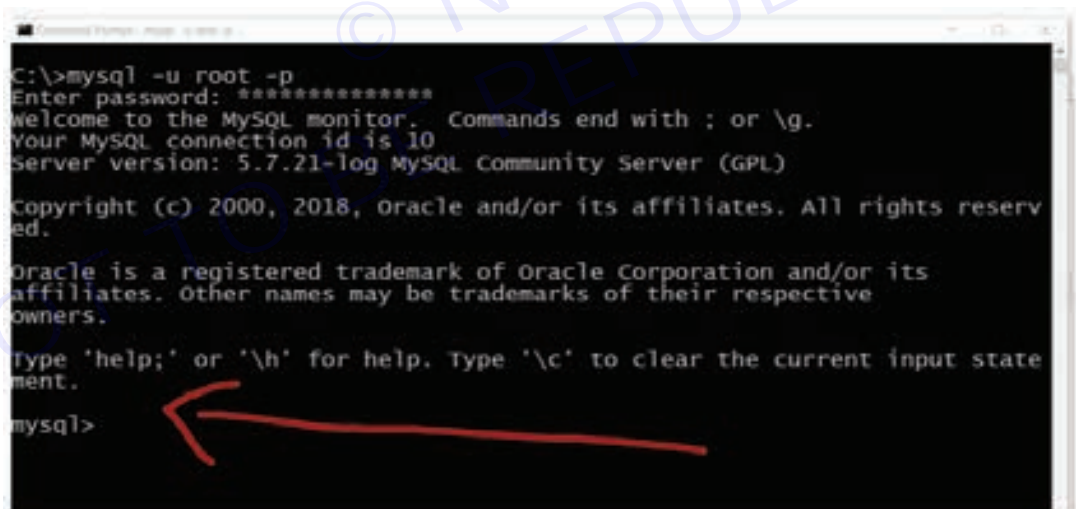
Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: Windows 10/11
- XAMPP server 3.3.0

Procedure

TASK 1 : Applying data integrity rules

- 1 **Start MySQL Server:** start the MySQL server. You can do this by going to the Services application in Windows, finding MySQL service, and starting it.
- 2 **Access MySQL:** Open the MySQL command line client or a GUI tool like MySQL Workbench to access your MySQL server.



```

C:\>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 5.7.21-log MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input state
ment.

mysql>
  
```

3 Define Your Rules

Start by understanding the data relationships and desired constraints. Consider:

Primary Keys: Identify unique identifiers for each table (e.g., product id in a products table).

Foreign Keys: Create references between related tables (e.g., order id in an orders table referencing product id in the products table).

Data Types: Specify allowed data types for each column (e.g., INT for numbers, VARCHAR for text).

NOT NULL: Mark columns that cannot be empty.

UNIQUE: Ensure specific values appear only once within a column.

CHECK: Define custom conditions for valid data (e.g., age must be positive).

4 Create a Database: Use the following SQL command to create a new database:

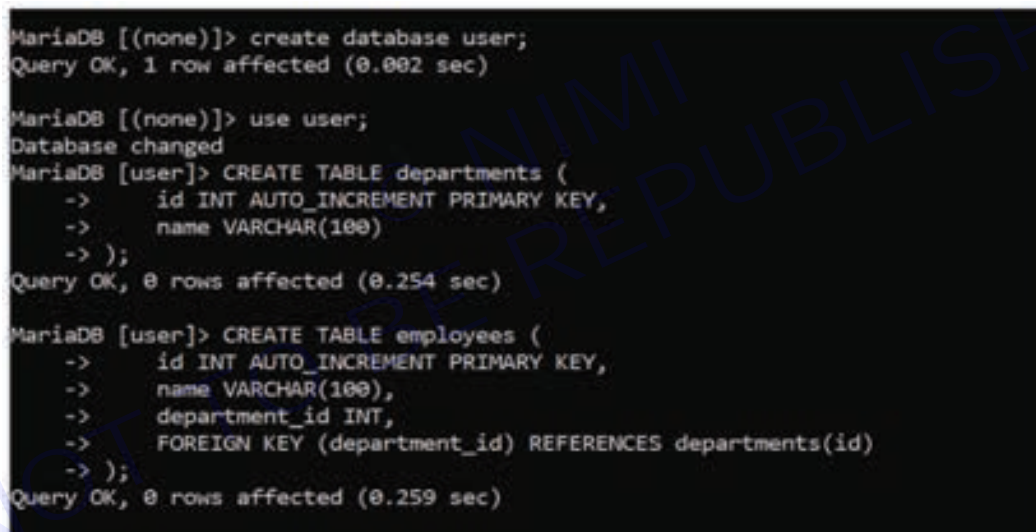
```
CREATE DATABASE your database name;
```

Use the Database: Switch to the newly created database using the following command:

```
USE your database name;
```

5 Create Tables: Create the tables for which you want to apply data integrity rules. For example:

```
CREATE TABLE departments (
    id INT AUTO INCREMENT PRIMARY KEY,
    name VARCHAR(100)
);
CREATE TABLE employees (
    id INT AUTO INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    department id INT,
    FOREIGN KEY (department id) REFERENCES departments(id)
);
```



```
MariaDB [(none)]> create database user;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> use user;
Database changed
MariaDB [user]> CREATE TABLE departments (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   name VARCHAR(100)
  -> );
Query OK, 0 rows affected (0.254 sec)

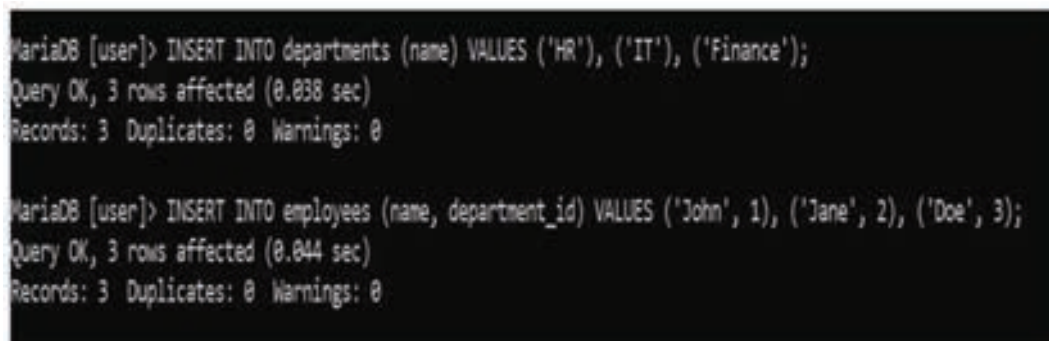
MariaDB [user]> CREATE TABLE employees (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   name VARCHAR(100),
  ->   department_id INT,
  ->   FOREIGN KEY (department_id) REFERENCES departments(id)
  -> );
Query OK, 0 rows affected (0.259 sec)
```

In this example, **the employees** table has a foreign key constraint referencing the **id** column of the **departments** table.

6 Insert Data: Insert some data into the tables you've created. Make sure to maintain referential integrity:

```
INSERT INTO departments (name) VALUES ('HR'), ('IT'), ('Finance');
```

```
INSERT INTO employees (name, department_id) VALUES ('John', 1), ('Jane', 2), ('Doe', 3);
```



```
MariaDB [user]> INSERT INTO departments (name) VALUES ('HR'), ('IT'), ('Finance');
Query OK, 3 rows affected (0.038 sec)
Records: 3 Duplicates: 0 Warnings: 0

MariaDB [user]> INSERT INTO employees (name, department_id) VALUES ('John', 1), ('Jane', 2), ('Doe', 3);
Query OK, 3 rows affected (0.044 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

7 Test Data Integrity: Try to insert data that violates the integrity rules you've defined. For example, try to insert an employee with a department id that doesn't exist:

```
INSERT INTO employees (name, department_id) VALUES ('Adam', 10);
```

```
MariaDB [user]> INSERT INTO employees (name, department_id) VALUES ('Adam', 10);  
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails ('user`.`employees`, CONSTRAINT `employees_ibfk_1` FOREIGN KEY (`department_id`) REFERENCES `departments` (`id`))  
MariaDB [user]>
```

You should get an error because the department id 10 doesn't exist in the departments table.

Update and Delete Data: When updating or deleting data, make sure to maintain referential integrity by updating or deleting related records appropriately.

Review and Modify Rules: Regularly review your data integrity rules and modify them as necessary to ensure they continue to meet the requirements of your application.

We can use the graphical interface of phpMyAdmin to manage tables and constraints.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 23 : Using the DDL, DCL, DML statements

Objectives

At the end of this exercise you shall be able to

- use DDL statements create, alter and drop
- use DCL statements grant & revoke
- use DML statements insert update delete

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 :Accessing MySQL:

MySQL Workbench: Download and install the graphical user interface (GUI) for MySQL management.

Command Prompt: Open the Command Prompt and navigate to the MySQL installation directory (e.g., C:\Program Files\MySQL\MySQL Server 8.0\bin).

Connecting to the Database:

MySQL Workbench: Provide your server hostname, username, password, and database name to connect.

Command Prompt: Use the mysql command followed by your credentials (e.g., mysql -h localhost -u root -p).

TASK 2 :Using DDL (Data Definition Language) Statements:

1 Create a Database:

```
CREATE DATABASE your database name;
```

Use the Database:

```
USE your database name;
```

Create a Table:

```
2 CREATE TABLE employees (
        id INT AUTO INCREMENT PRIMARY KEY,
        name VARCHAR(100),
        salary DECIMAL(10, 2)
);
```

3 Alter Table (Add a Column)

```
ALTER TABLE employees
```

```
ADD COLUMN hire date DATE;
```

```

MariaDB [(none)]> create database user;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> use user;
Database changed
MariaDB [user]> CREATE TABLE departments (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   name VARCHAR(100)
  -> );
Query OK, 0 rows affected (0.254 sec)

MariaDB [user]> CREATE TABLE employees (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   name VARCHAR(100),
  ->   department_id INT,
  ->   FOREIGN KEY (department_id) REFERENCES departments(id)
  -> );
Query OK, 0 rows affected (0.259 sec)

MariaDB [user]> select * from employees;
+----+-----+-----+
| id | name | department_id |
+----+-----+-----+
|  1 | John |             1 |
|  2 | Jane |             2 |
|  3 | Doe  |             3 |
+----+-----+-----+
3 rows in set (0.000 sec)

MariaDB [user]> alter table employees add column Hire_date date after name;
Query OK, 0 rows affected (0.079 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [user]> select * from employees;
+----+-----+-----+-----+
| id | name | Hire_date | department_id |
+----+-----+-----+-----+
|  1 | John | NULL     |             1 |
|  2 | Jane | NULL     |             2 |
|  3 | Doe  | NULL     |             3 |
+----+-----+-----+-----+
3 rows in set (0.000 sec)

```

4 Drop Table:

DROP TABLE employees;

TASK 3 : DCL (Data Control Language) Statements:

1 **GRANT:** Assigns permissions to users on databases and objects.

GRANT SELECT, INSERT ON customers TO user1@localhost;

2 **REVOKE:** Removes permissions from users.

If we want to revoke all privileges assign to the user, execute the following statement:

REVOKE ALL, GRANT OPTION FROM john@localhost;

REVOKE INSERT ON customers FROM user1@localhost;

```

MariaDB [user]> GRANT SELECT, UPDATE, INSERT ON user.* TO john@localhost;
Query OK, 0 rows affected (0.025 sec)

MariaDB [user]> SHOW GRANTS FOR john@localhost;
+-----+
| Grants for john@localhost |
+-----+
| GRANT USAGE ON *.* TO `john`@`localhost` IDENTIFIED BY PASSWORD '*81F5E21E35407D084A6CD4A731AEBFB6AF209E1B' |
| GRANT SELECT, INSERT, UPDATE ON `user`.* TO `john`@`localhost` |
+-----+
2 rows in set (0.000 sec)

MariaDB [user]> REVOKE UPDATE, INSERT ON user.* FROM john@localhost;
Query OK, 0 rows affected (0.066 sec)

MariaDB [user]> SHOW GRANTS FOR john@localhost;
+-----+
| Grants for john@localhost |
+-----+
| GRANT USAGE ON *.* TO `john`@`localhost` IDENTIFIED BY PASSWORD '*81F5E21E35407D084A6CD4A731AEBFB6AF209E1B' |
| GRANT SELECT ON `user`.* TO `john`@`localhost` |
+-----+
2 rows in set (0.000 sec)

```

3 Create User:

CREATE USER 'new user'@'localhost' IDENTIFIED BY 'user password';

```

MariaDB [user]> CREATE USER john@localhost IDENTIFIED BY 'root';
Query OK, 0 rows affected (0.028 sec)

MariaDB [user]> SHOW GRANTS FOR john@localhost;
+-----+
| Grants for john@localhost |
+-----+
| GRANT USAGE ON *.* TO `john`@`localhost` IDENTIFIED BY PASSWORD '*81F5E21E35407D084A6CD4A731AEBFB6AF209E1B' |
+-----+
1 row in set (0.000 sec)

MariaDB [user]> GRANT ALL ON mystudentdb.* TO john@localhost;
Query OK, 0 rows affected (0.020 sec)

MariaDB [user]> GRANT ALL ON user.* TO john@localhost;
Query OK, 0 rows affected (0.024 sec)

MariaDB [user]> REVOKE ALL, GRANT OPTION FROM john@localhost;
Query OK, 0 rows affected (0.024 sec)

```

4 Drop User:

DROP USER 'new user'@'localhost';

TASK 4 : Using DML (Data Manipulation Language) Statements:

1 INSERT: Adds new data to a table.

INSERT INTO departments (id, name) VALUES (4, 'Operation');

```

MariaDB [user]> select * from departments;
+----+-----+
| id | name  |
+----+-----+
| 1  | HR    |
| 2  | IT    |
| 3  | Finance |
+----+-----+
3 rows in set (0.065 sec)

MariaDB [user]> insert into departments(id, name) VALUES(4,'Operation');
Query OK, 1 row affected (0.044 sec)

MariaDB [user]> select * from departments;
+----+-----+
| id | name  |
+----+-----+
| 1  | HR    |
| 2  | IT    |
| 3  | Finance |
| 4  | Operation |
+----+-----+
4 rows in set (0.001 sec)

```

2 **UPDATE:** Modifies existing data in a table.

UPDATE departments SET name = 'Operation' WHERE id = 1;

```

MariaDB [user]> select * from departments;
+----+-----+
| id | name  |
+----+-----+
| 1  | HR    |
| 2  | IT    |
| 3  | Finance |
| 4  | Operation |
+----+-----+
4 rows in set (0.001 sec)

MariaDB [user]> UPDATE departments SET name = 'Operation' WHERE id = 1;
Query OK, 1 row affected (0.042 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [user]> select * from departments;
+----+-----+
| id | name  |
+----+-----+
| 1  | Operation |
| 2  | IT    |
| 3  | Finance |
| 4  | Operation |
+----+-----+
4 rows in set (0.000 sec)

```

3 **DELETE:** Removes data from a table.

DELETE FROM departments WHERE id = 4;

4 **Select Data:** Retrieve data from the table.

SELECT * FROM employees;

```
MariaDB [user]> select * from employees;
+----+-----+-----+-----+
| id | name | Hire_date | department_id |
+----+-----+-----+-----+
|  1 | John | NULL      | 1             |
|  2 | Jane | NULL      | 2             |
|  3 | Doe  | NULL      | 3             |
+----+-----+-----+-----+
3 rows in set (0.000 sec)
```

SELECT * FROM Departments;

```
MariaDB [user]> select * from departments;
+----+-----+
| id | name |
+----+-----+
|  1 | Operation |
|  2 | IT |
|  3 | Finance |
+----+-----+
3 rows in set (0.000 sec)
```

NOT TO BE REPRODUCED OR REPUBLISHED

EXERCISE 24 : Enforcing Constraints, Primary Key and Foreign Key

Objectives

At the end of this exercise you shall be able to

- connect the MySQL database server
- design the table
- create primary & foreign key

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 : Access MySQL:

MySQL Workbench: Download and install the user-friendly GUI for database management.

Command Prompt: Open the Command Prompt and navigate to the MySQL installation directory (e.g., C:\Program Files\MySQL\MySQL Server 8.0\bin).

Connect to your Database:

MySQL Workbench: Enter your server hostname, username, password, and database name to connect.

Command Prompt: Use the mysql command followed by your credentials (e.g., mysql -h localhost -u root -p).

TASK 2 : Design Your Tables:

1 Primary Key: Identify the unique identifier for each table. This column should not allow null values and should have a unique constraint applied:

```
product id INT PRIMARY KEY AUTO INCREMENT,
name VARCHAR(255) NOT NULL UNIQUE,
price DECIMAL(8,2) NOT NULL
);
```

Establish Relationships:

2 Foreign Key: Define references between related tables. The referencing column must match the data type of the referenced column:

```
CREATE TABLE orders (
order id INT PRIMARY KEY AUTO INCREMENT,
customer id INT NOT NULL,
product id INT NOT NULL,
FOREIGN KEY (customer id) REFERENCES customers(customer id),
FOREIGN KEY (product id) REFERENCES products(product id)
ON DELETE SET NULL
);
```

```
MariaDB [(none)]> create database user;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> use user;
Database changed
MariaDB [user]> CREATE TABLE departments (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   name VARCHAR(100)
  -> );
Query OK, 0 rows affected (0.254 sec)

MariaDB [user]> CREATE TABLE employees (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   name VARCHAR(100),
  ->   department_id INT,
  ->   FOREIGN KEY (department_id) REFERENCES departments(id)
  -> );
Query OK, 0 rows affected (0.259 sec)
```

Ensure that each table's primary key constraint is enforced. MySQL automatically enforces primary key constraints, preventing the insertion of duplicate keys.

- 3 **ON DELETE/UPDATE Actions:** Specify how the database reacts when data in the parent table is modified (e.g., ON DELETE SET NULL sets foreign key to null, ON DELETE CASCADE deletes related rows).
- 4 **CHECK Constraints:** Define custom conditions for valid data (e.g., age must be positive).

EXERCISE 25 : Adding indices to tables

Objectives

At the end of this exercise you shall be able to

- connect the MySQL database server
- create tables in database
- create index on tables.

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 : Access MySQL:

MySQL Workbench: Download and install the GUI for user-friendly management.

Command Prompt: Open it and navigate to the MySQL installation directory (e.g., C:\Program Files\MySQL\MySQL Server 8.0\bin).

Connect to your Database:

MySQL Workbench: Provide your server hostname, username, password, and database name.

Command Prompt: Use the mysql command followed by your credentials (e.g., mysql -h localhost -u root -p).

Identify Columns for Indexing:

Choose columns frequently used in WHERE clauses, ORDER BY, or JOINS. Consider data types and selectivities (percentage of unique values).

Use the Database:

USE your database name;

TASK 2 : Create a Table:

Let's assume you have a table called employee:

```
CREATE TABLE employee (
  id INT PRIMARY KEY,
  name VARCHAR(100),
  department id INT,
  salary DECIMAL(10, 2)
);
```

TASK 3 : Insert Data into the Table:

```
INSERT INTO employee (id, name, department id, salary)
VALUES (1, 'John Doe', 1, 50000.00), (2, 'Jane Smith', 2, 60000.00);
```

```

MariaDB [user]> CREATE TABLE employee (
->   id INT PRIMARY KEY,
->   name VARCHAR(100),
->   department_id INT,
->   salary DECIMAL(10, 2)
-> );
Query OK, 0 rows affected (0.278 sec)

MariaDB [user]> desc employee;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id             | int(11)       | NO   | PRI | NULL    |      |
| name           | varchar(100)  | YES  |     | NULL    |      |
| department_id  | int(11)       | YES  |     | NULL    |      |
| salary         | decimal(10,2) | YES  |     | NULL    |      |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.015 sec)

```

```

MariaDB [user]> INSERT INTO employee (id, name, department_id, salary)
-> VALUES (1, 'John Doe', 1, 50000.00), (2, 'Jane Smith', 2, 60000.00
-> );
Query OK, 2 rows affected (0.035 sec)
Records: 2 Duplicates: 0 Warnings: 0

MariaDB [user]> select * from employee;
+----+-----+-----+-----+
| id | name      | department_id | salary |
+----+-----+-----+-----+
|  1 | John Doe  |             1 | 50000.00 |
|  2 | Jane Smith |             2 | 60000.00 |
+----+-----+-----+-----+
2 rows in set (0.001 sec)

```

View the Table Structure:

DESC employees;

TASK 4 :Add Index:

Single Column Index-

CREATE INDEX idx_department_id ON employee (department_id);

```

MariaDB [user]> CREATE INDEX idx_department_id ON employee (department_id);
Query OK, 0 rows affected (0.160 sec)
Records: 0 Duplicates: 0 Warnings: 0

```

Multiple Column (Composite) Index-

CREATE INDEX idx_department_salary ON employee (department_id, salary);

View Index Information:

SHOW INDEX FROM employee;

```
MariaDB [user]> show index from employee;
```

```

+-----+-----+-----+-----+-----+-----+
| Table   | Non_unique | Key_name      | Seq_in_index | Column_name | Collation | Cardinality |
| Sub_part | Packed   | Null  | Index_type | Comment | Index_comment |
+-----+-----+-----+-----+-----+-----+
| employee |          0 | PRIMARY      |             1 | id         | A          |           2 |
| NULL    | NULL     |              |              |           |           |           |
| employee |          1 | idx_department_id |             1 | department_id | A          |           2 |
| NULL    | NULL     | YES         | BTREE      |           |           |           |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.001 sec)

```

TASK 5 :Test Query Performance:

Run queries and observe the performance improvement. For example:

```
SELECT * FROM employee WHERE department id = 1;
```

Drop Index (Optional):

If needed, you can drop an index:

```
DROP INDEX idx department id ON employee;
```

Questions

- 1 How do I check existing indices on a table?
- 2 How do I add a simple Index to a column?
- 3 How do I remove an Index from a table?
- 4 How can I check the performance improvement after adding an Index?
- 5 Can I add a unique Index to a column?

EXERCISE 26 : Simple Select Queries

Objectives

At the end of this exercise you shall be able to

- retrieve data from the table
- retrieve data from the table by using where clause
- retrieve data from the table by using aggregate functions.

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 :Using select queries:

1 Select all columns from a table:

SELECT * FROM your table name;

```
MariaDB [user]> select * from employee;
+-----+-----+-----+-----+
| id | name      | department_id | salary |
+-----+-----+-----+-----+
|  1 | John Doe  |             1 | 50000.00 |
|  2 | Jane Smith |             2 | 60000.00 |
+-----+-----+-----+-----+
2 rows in set (0.017 sec)
```

2 Select specific columns from a table:

SELECT column1, column2 FROM your table name;

```
MariaDB [user]> select name, salary from employee;
+-----+-----+
| name      | salary |
+-----+-----+
| John Doe  | 50000.00 |
| Jane Smith | 60000.00 |
+-----+-----+
2 rows in set (0.001 sec)
```

3 Select with a condition(WHERE clause):

SELECT * FROM your table name WHERE your column name = 'some value';

```

MariaDB [user]> select * from employee where id = 2;
+----+-----+-----+-----+
| id | name      | department_id | salary |
+----+-----+-----+-----+
|  2 | Jane Smith |           2   | 60000.00 |
+----+-----+-----+-----+
1 row in set (0.001 sec)

```

4 Select with multiple conditions:

SELECT * FROM your table name WHERE condition1 AND condition2;

```

MariaDB [user]> select * from employee where id = 1 and name = "John Doe";
+----+-----+-----+-----+
| id | name      | department_id | salary |
+----+-----+-----+-----+
|  1 | John Doe  |           1   | 50000.00 |
+----+-----+-----+-----+
1 row in set (0.001 sec)

```

5 Select with ordering(OBJECT BY clause):

SELECT * FROM your table name ORDER BY your column name DESC;

6 Select with a limit:

SELECT * FROM your table name LIMIT 10;

7 Select with a join:

SELECT t1.column1, t2.column2

FROM table1 t1

JOIN table2 t2 ON t1.common column = t2.common column;

8 Select with Aggregate functions (e.g. SUM, AVG):

SELECT AVG(your column name) AS average value

FROM your table name;

SELECT MIN(your column name) AS minimum value

FROM your table name;

- 1 How to retrieve maximum and minimum values in any particular column?
- 2 How to retrieve rows based on date range?
- 3 How to retrieve unique values in any column?

EXERCISE 27 : Insert, Update and Delete Queries

Objectives

At the end of this exercise you shall be able to

- create table and insert the value in the table
- update some values in the tables
- delete specific tuples from the table

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 :Using next query

1 Create database:

Create database database name:

Use database:

Use database name;

2 Create Table:

```
CREATE TABLE your table name (
    column1 datatype1,
    column2 datatype2,
    column3 datatype3,
    -- add more columns as needed
    PRIMARY KEY (column1)
```

);

Ex:

```
CREATE TABLE `employee` (
    `employee id` bigint unsigned NOT NULL AUTO INCREMENT,
    `first name` varchar(45) NOT NULL,
    `last name` varchar(45) NOT NULL,
    `last update` timestamp NOT NULL DEFAULT CURRENT TIMESTAMP,
    UNIQUE KEY `employee id` (`employee id`)
```

);


```

MariaDB [user]> CREATE TABLE `employee` (
  -> `employee_id` bigint unsigned NOT NULL AUTO_INCREMENT,
  -> `first_name` varchar(45) NOT NULL,
  -> `last_name` varchar(45) NOT NULL,
  -> `last_update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  -> UNIQUE KEY `employee_id` (`employee_id`)
  -> );
Query OK, 0 rows affected (0.165 sec)

```

To find the structure of a table called employee, you can use the MySQL DESCRIBE command:

DESCRIBE employee;

```

MariaDB [user]> describe employee;
+-----+-----+-----+-----+-----+-----+
| Field      | Type                | Null | Key | Default          | Extra          |
+-----+-----+-----+-----+-----+-----+
| employee_id | bigint(20) unsigned | NO   | PRI | NULL             | auto_increment |
| first_name  | varchar(45)         | NO   |     | NULL             |                |
| last_name   | varchar(45)         | NO   |     | NULL             |                |
| last_update | timestamp           | NO   |     | current_timestamp() |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.015 sec)

```

3 Insert Query: Adding new records to the table

INSERT INTO your table name (column1, column2, column3)

VALUES ('value1', 'value2', 'value3');

Ex: INSERT INTO employee(first name, last name)

VALUES ('Bob', 'Smith');

```

MariaDB [user]> INSERT INTO employee(first_name, last_name)
  -> VALUES ('Bob', 'Smith'
  -> );
Query OK, 1 row affected (0.042 sec)

```

For show the records

SELECT * FROM employee;

```

MariaDB [user]> select * from employee;
+-----+-----+-----+-----+
| employee_id | first_name | last_name | last_update |
+-----+-----+-----+-----+
|          1 | Bob       | Smith    | 2024-03-07 15:07:39 |
+-----+-----+-----+-----+
1 row in set (0.000 sec)

```

4 Using INSERT to add multiple rows at once

INSERT INTO my table(column name, column name 2)

VALUES

('value', 'value2'),
 ('value3', 'value4'),
 ('value5', 'value6');

Ex - INSERT INTO employee(first name, last name)

VALUES

('Abigail', 'Spencer'),
 ('Tamal', 'Wayne'),
 ('Katie', 'Singh'),
 ('Felipe', 'Espinosa');

```
MariaDB [user]> INSERT INTO employee(first_name, last_name)
-> VALUES
-> ('Abigail', 'Spencer'),
-> ('Tamal', 'Wayne'),
-> ('Katie', 'Singh'),
-> ('Felipe', 'Espinosa'
-> );
Query OK, 4 rows affected (0.038 sec)
Records: 4 Duplicates: 0 Warnings: 0
```

For show the records

SELECT * FROM employee;

```
MariaDB [user]> select * from employee;
+-----+-----+-----+-----+
| employee_id | first_name | last_name | last_update |
+-----+-----+-----+-----+
| 1 | Bob | Smith | 2024-03-07 15:07:39 |
| 2 | Abigail | Spencer | 2024-03-07 15:11:27 |
| 3 | Tamal | Wayne | 2024-03-07 15:11:27 |
| 4 | Katie | Singh | 2024-03-07 15:11:27 |
| 5 | Felipe | Espinosa | 2024-03-07 15:11:27 |
+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

TASK 2 : Using update query

Update Query:

UPDATE your table name

SET column1 = 'new value1', column2 = 'new value2'

WHERE condition;



```

MariaDB [user]> UPDATE employee set first_name = "John" WHERE employee_id = 1;
Query OK, 1 row affected (0.041 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [user]> select * from employee;
+-----+-----+-----+-----+
| employee_id | first_name | last_name | last_update |
+-----+-----+-----+-----+
|          1 | John      | Smith    | 2024-03-07 15:07:39 |
|          2 | Abigail   | Spencer  | 2024-03-07 15:11:27 |
|          3 | Tamal    | Wayne    | 2024-03-07 15:11:27 |
|          4 | Katie     | Singh    | 2024-03-07 15:11:27 |
|          5 | Felipe    | Espinosa | 2024-03-07 15:11:27 |
+-----+-----+-----+-----+
5 rows in set (0.000 sec)

```

TASK 3 :Using delete query:

Delete Query: to remove rows from tables

DELETE FROM your table name

WHERE condition;

EX - DELETE FROM employee

WHERE first name = 'Abigail';

```

MariaDB [user]> DELETE FROM employee
-> WHERE first_name = 'Abigail'
-> ;
Query OK, 1 row affected (0.041 sec)

```

For show the records

SELECT * FROM employee;

```

MariaDB [user]> select * from employee;
+-----+-----+-----+-----+
| employee_id | first_name | last_name | last_update |
+-----+-----+-----+-----+
|          1 | John      | Smith    | 2024-03-07 15:07:39 |
|          3 | Tamal    | Wayne    | 2024-03-07 15:11:27 |
|          4 | Katie     | Singh    | 2024-03-07 15:11:27 |
|          5 | Felipe    | Espinosa | 2024-03-07 15:11:27 |
+-----+-----+-----+-----+
4 rows in set (0.000 sec)

```

The return value here indicates that the DELETE command was processed with a single row being removed.

Using DELETE to remove multiple rows at once

2 You can remove multiple items at once with DELETE by manipulating the selection criteria specified in the WHERE clause.

For instance, to remove multiple rows by ID, you could type something like this:

Ex - DELETE FROM employee

WHERE employee id in (3,4);

```
MariaDB [user]> DELETE FROM employee
-> WHERE employee_id in (3,4
-> );
Query OK, 2 rows affected (0.043 sec)

MariaDB [user]> select * from employee;
+-----+-----+-----+-----+
| employee_id | first_name | last_name | last_update |
+-----+-----+-----+-----+
|          1 | John      | Smith    | 2024-03-07 15:07:39 |
|          5 | Felipe    | Espinosa | 2024-03-07 15:11:27 |
+-----+-----+-----+-----+
2 rows in set (0.000 sec)
```

3 You can even leave out the WHERE clause to remove all of the rows from a given table:

DELETE FROM employee;

```
MariaDB [user]> DELETE FROM employee;
Query OK, 2 rows affected (0.045 sec)

MariaDB [user]> select * from employee;
Empty set (0.000 sec)
```

Questions

- 1 How can you insert multiple records in a single query?
- 2 How can you update multiple columns in a single query?
- 3 How do you delete a specific record from a table?

EXERCISE 28: Using the Number, Date and Character Functions

Objectives

At the end of this exercise you shall be able to

- use the number functions in MySQL
- use the data functions in MySQL
- use the character functions in MySQL

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 :Using Number Functions:

1 ABS() – Absolute Value:

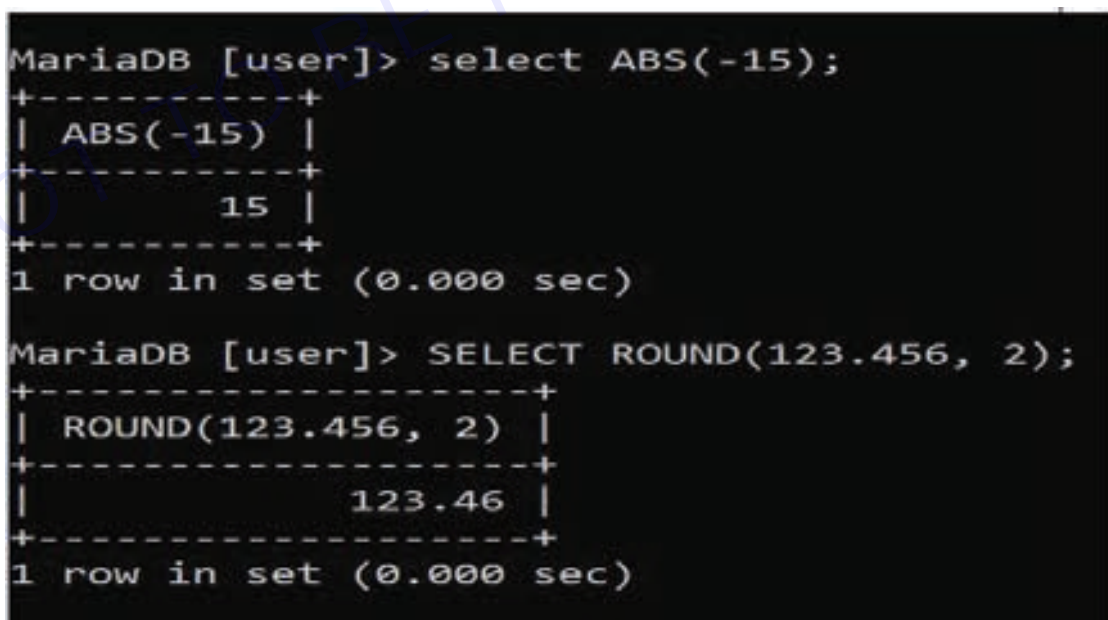
```
SELECT ABS(-15);
```

-- Result: 15

2 ROUND() - Round to specified numbers of decimal places.

```
SELECT ROUND(123.456, 2);
```

-- Result: 123.46



```

MariaDB [user]> select ABS(-15);
+-----+
| ABS(-15) |
+-----+
|         15 |
+-----+
1 row in set (0.000 sec)

MariaDB [user]> SELECT ROUND(123.456, 2);
+-----+
| ROUND(123.456, 2) |
+-----+
|           123.46 |
+-----+
1 row in set (0.000 sec)

```

3 SUM() – Calculate the sum of the values

```
SELECT SUM(column name) FROM your table;
```

```

MariaDB [user]> select * from employee;
+-----+-----+-----+-----+
| id | name      | department_id | salary |
+-----+-----+-----+-----+
|  1 | John Doe  |             1 | 50000.00 |
|  2 | Jane Smith |             2 | 60000.00 |
+-----+-----+-----+-----+
2 rows in set (0.149 sec)

MariaDB [user]> select sum(salary) from employee;
+-----+
| sum(salary) |
+-----+
| 110000.00 |
+-----+
1 row in set (0.001 sec)

```

TASK 2 : Using DATE Functions:

1 NOW() – Current date and time

```
SELECT NOW();
```

-- Result: Current date and time

2 DATE FORMAT() – Format a date

```
SELECT DATE FORMAT(NOW(), '%Y-%m-%d');
```

-- Result: Formatted date (YYYY-MM-DD)

```

MariaDB [user]> select now();
+-----+
| now() |
+-----+
| 2024-03-07 14:47:55 |
+-----+
1 row in set (0.000 sec)

MariaDB [user]> select date_format(now(), '%y-%m-%d');
+-----+
| date_format(now(), '%y-%m-%d') |
+-----+
| 24-03-07 |
+-----+
1 row in set (0.000 sec)

```

3 DATEDIFF() – Calculate the difference between two dates

```
SELECT DATEDIFF('2022-01-01', '2021-12-15');
```

-- Result: 17 (days)

```
MariaDB [user]> SELECT DATEDIFF('2022-01-01', '2021-12-15');
+-----+
| DATEDIFF('2022-01-01', '2021-12-15') |
+-----+
|                                     17 |
+-----+
1 row in set (0.001 sec)
```

TASK 3 :Using CHARACTER Functions:**1 CONCAT() – Concatenate strings**

```
SELECT CONCAT('Hello', ' ', 'World');
```

-- Result: Hello World

```
MariaDB [user]> SELECT CONCAT('Hello', ' ', 'World');
+-----+
| CONCAT('Hello', ' ', 'World') |
+-----+
| Hello World                    |
+-----+
1 row in set (0.001 sec)
```

2 LENGTH() – Length of a string

```
SELECT LENGTH('abcde');
```

-- Result: 5

3 SUBSTRING() – Extract part of a string

```
SELECT SUBSTRING('abcdef', 2, 3);
```

-- Result: bcd

1 How do you calculate the difference in days between two date columns?

```
SELECT DATEDIFF(end date, start date) FROM your table;
```

2 How can you concatenate two columns and add a separator between them?

```
SELECT CONCAT(column1, ' - ', column2) AS concatenated columns FROM your table;
```

3 How can you find the length of string column?

```
SELECT LENGTH(string column) FROM your table;
```

```
MariaDB [user]> SELECT LENGTH('abcde');
+-----+
| LENGTH('abcde') |
+-----+
|                5 |
+-----+
1 row in set (0.000 sec)

MariaDB [user]> SELECT SUBSTRING('abcdef', 2, 3);
+-----+
| SUBSTRING('abcdef', 2, 3) |
+-----+
| bcd                       |
+-----+
1 row in set (0.000 sec)
```

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 29 : Joins, Group by, Having, Sub query

Objectives

At the end of this exercise you shall be able to

- use joins in queries
- use group by in queries
- use having & subquery in queries

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 :Setup & Connect To MySQL:

Make sure you have MySQL installed on your Windows machine. You can use a tool like MySQL Command-Line Client or a graphical interface like MySQL Workbench.

mysql -u your username -p

```
# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.4.32-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> use user;
Database changed
```

Enter your password when prompted.

TASK 2 :Create sample tables and insert values

1 Let's create two simple tables -

```
CREATE TABLE employee (
    employee id INT PRIMARY KEY,
    employee name VARCHAR(50),
    department id INT
);
```

```

MariaDB [user]> CREATE TABLE employee (
  ->   employee_id INT PRIMARY KEY,
  ->   employee_name VARCHAR(50),
  ->   department_id INT
  -> );

```

Query OK, 0 rows affected (0.133 sec)

```

MariaDB [user]> desc employee;

```

Field	Type	Null	Key	Default	Extra
employee_id	int(11)	NO	PRI	NULL	
employee_name	varchar(50)	YES		NULL	
department_id	int(11)	YES		NULL	

3 rows in set (0.019 sec)

```

2 CREATE TABLE department (
  department id INT PRIMARY KEY,
  department name VARCHAR(50)
);

```

```

MariaDB [user]> CREATE TABLE department (
  ->   department_id INT PRIMARY KEY,
  ->   department_name VARCHAR(50)
  -> );
Query OK, 0 rows affected (0.138 sec)

```

3 Insert the values in the tables:

Insert values in Employees table -

```
INSERT INTO employee VALUES (1, 'John Doe', 1);
```

```
INSERT INTO employee VALUES (2, 'Jane Smith', 2);
```

```
INSERT INTO employee VALUES (3, 'Bob Johnson', 1);
```

```

MariaDB [user]> INSERT INTO employee VALUES (1, 'John Doe', 1);
Query OK, 1 row affected (0.051 sec)

```

```

MariaDB [user]> INSERT INTO employee VALUES (2, 'Jane Smith', 2);
Query OK, 1 row affected (0.038 sec)

```

```

MariaDB [user]> INSERT INTO employee VALUES (3, 'Bob Johnson', 1);
Query OK, 1 row affected (0.043 sec)

```

Insert values in Department table -

```
INSERT INTO department VALUES (1, 'HR');
```

```
INSERT INTO department VALUES (2, 'IT');
```

```
MariaDB [user]> INSERT INTO department VALUES (1, 'HR');
Query OK, 1 row affected (0.049 sec)

MariaDB [user]> INSERT INTO department VALUES (2, 'IT');
Query OK, 1 row affected (0.048 sec)
```

4 Show the records of both the tables :

SELECT * from employee;

```
MariaDB [user]> select * from employee;
+-----+-----+-----+
| employee_id | employee_name | department_id |
+-----+-----+-----+
|          1 | John Doe      |             1 |
|          2 | Jane Smith    |             2 |
|          3 | Bob Johnson   |             1 |
+-----+-----+-----+
3 rows in set (0.001 sec)
```

SELECT * from department;

```
MariaDB [user]> select * from department;
+-----+-----+
| department_id | department_name |
+-----+-----+
|             1 | HR              |
|             2 | IT              |
+-----+-----+
2 rows in set (0.000 sec)
```

TASK 3 :Using Joins:

Inner Join - Retrieve rows where there is a match in both tables.

SELECT * FROM employee

INNER JOIN department ON employee.department id = department.department id;

```
MariaDB [user]> SELECT * FROM employee
-> INNER JOIN department ON employee.department_id = department.department_id
-> ;
+-----+-----+-----+-----+-----+
| employee_id | employee_name | department_id | department_id | department_name |
+-----+-----+-----+-----+-----+
|          1 | John Doe      |             1 |             1 | HR              |
|          2 | Jane Smith    |             2 |             2 | IT              |
|          3 | Bob Johnson   |             1 |             1 | HR              |
+-----+-----+-----+-----+-----+
3 rows in set (0.001 sec)
```

2 Left Join - Retrieve all rows from the left table and the matching rows from the right table.

SELECT * FROM employee

LEFT JOIN department ON employee.department id = department.department id;

```
MariaDB [user]> SELECT * FROM employee
-> LEFT JOIN department ON employee.department_id = department.department_id;
+-----+-----+-----+-----+-----+
| employee_id | employee_name | department_id | department_id | department_name |
+-----+-----+-----+-----+-----+
| 1 | John Doe | 1 | 1 | HR |
| 2 | Jane Smith | 2 | 2 | IT |
| 3 | Bob Johnson | 1 | 1 | HR |
| 4 | Tom | 3 | 3 | FINANCE |
| 5 | Cruze | 3 | 3 | FINANCE |
+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

3 Right Join – Retrieve all rows from the right table and the matching rows from the left table.

SELECT * FROM employee

RIGHT JOIN department ON employee.department id = department.department id;

```
MariaDB [user]> SELECT * FROM employee
-> RIGHT JOIN department ON employee.department_id = department.department_id;
+-----+-----+-----+-----+-----+
| employee_id | employee_name | department_id | department_id | department_name |
+-----+-----+-----+-----+-----+
| 1 | John Doe | 1 | 1 | HR |
| 2 | Jane Smith | 2 | 2 | IT |
| 3 | Bob Johnson | 1 | 1 | HR |
| 4 | Tom | 3 | 3 | FINANCE |
| 5 | Cruze | 3 | 3 | FINANCE |
| NULL | NULL | NULL | 4 | HR |
+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)
```

4 Full Outer Join(Using UNION) - Retrieve all rows when there is a match in either the left or the right table.

SELECT * FROM employee

LEFT JOIN department ON employee.department id = department.department id

UNION

SELECT * FROM employee

RIGHT JOIN department ON employee.department id = department.department id;

```
MariaDB [user]> SELECT * FROM employee
-> LEFT JOIN department ON employee.department_id = department.department_id
-> UNION
-> SELECT * FROM employee
-> RIGHT JOIN department ON employee.department_id = department.department_id;
+-----+-----+-----+-----+-----+
| employee_id | employee_name | department_id | department_id | department_name |
+-----+-----+-----+-----+-----+
| 1 | John Doe | 1 | 1 | HR |
| 2 | Jane Smith | 2 | 2 | IT |
| 3 | Bob Johnson | 1 | 1 | HR |
| 4 | Tom | 3 | 3 | FINANCE |
| 5 | Cruze | 3 | 3 | FINANCE |
| NULL | NULL | NULL | 4 | HR |
+-----+-----+-----+-----+-----+
6 rows in set (0.003 sec)
```

5 Self Join - Perform a join on a table with itself.

SELECT e1.employee id, e1.employee name, e2.employee name AS manager name

FROM employee e1

INNER JOIN employee e2 ON e1.department id = e2.department id AND e1.employee id != e2.employee id;

```

MariaDB [user]> SELECT e1.employee_id, e1.employee_name, e2.employee_name AS manager_name
-> FROM employee e1
-> INNER JOIN employee e2 ON e1.department_id = e2.department_id AND e1.employee_id != e2.employee_
id
-> ;

```

employee_id	employee_name	manager_name
3	Bob Johnson	John Doe
1	John Doe	Bob Johnson
5	Cruze	Tom
4	Tom	Cruze

```

4 rows in set (0.001 sec)

```

TASK 4 : Using GROUP BY:

The MYSQL GROUP BY Clause is used to collect data from multiple records and group the result by one or more column. It is generally used in a SELECT statement.

SELECT product, SUM(amount) as total sales

FROM sales

GROUP BY product;

This query selects the product and the total sales (SUM(amount)) for each product from the sales table, grouped by the product column.

You can use other aggregate functions like COUNT, AVG, MIN, MAX, etc., depending on your requirements.

1 SELECT product, COUNT(*) as total orders, AVG(amount) as avg sales

FROM sales

GROUP BY product;

```

MariaDB [user]> SELECT COUNT(department_id),employee_name from employee GROUP BY department_id;

```

COUNT(department_id)	employee_name
2	John Doe
1	Jane Smith
2	Tom

```

3 rows in set (0.001 sec)

```

```

MariaDB [user]> select * from employee;

```

employee_id	employee_name	department_id
1	John Doe	1
2	Jane Smith	2
3	Bob Johnson	1
4	Tom	3
5	Cruze	3

```

5 rows in set (0.000 sec)

```

When you're done, you can exit the MySQL Command Line Client:

EXIT;

This will close the connection and return you to the regular command prompt.

TASK 5 :Using Having Clause:

Assume you have a table named sales with columns product and amount. You want to know the total amount of sales per product, but only for products with a total sales greater than a certain threshold, e.g., 100.

```
1 SELECT product, SUM(amount) as total sales
FROM sales
GROUP BY product
HAVING total sales > 100;
```

This query selects the product and the total sales (SUM(amount)) for each product from the sales table, grouped by the product column. The HAVING clause filters the results to include only those with a total sales greater than 100.

You can add more conditions in the HAVING clause based on your requirements.

```
2 SELECT product, SUM(amount) as total sales
FROM sales
GROUP BY product
HAVING total sales > 100 AND COUNT(*) > 2;
```

Example-

```
MariaDB [user]> SELECT COUNT(department_id),employee_name from employee GROUP BY department_id HAVING C
COUNT(department_id)<2;
+-----+-----+
| COUNT(department_id) | employee_name |
+-----+-----+
| 1 | Jane Smith |
+-----+-----+
1 row in set (0.001 sec)
```

When you're done, you can exit the MySQL Command Line Client:

EXIT;

TASK 6 : Using Sub Query:

If you haven't already created tables and inserted sample data, you can use the following SQL statements:

```
1 CREATE TABLE Departments:
CREATE TABLE departments (
    id INT PRIMARY KEY,
    name VARCHAR(50)
);
2 CREATE TABLE Employees:
CREATE TABLE employees (
    id INT PRIMARY KEY,
```

```

name VARCHAR(50),
department id INT,
FOREIGN KEY (department id) REFERENCES departments(id)
);

```

3 INSERT SOME VALUES in the Tables:

```

INSERT INTO departments VALUES (1, 'IT');
INSERT INTO employees VALUES (1, 'John Doe', 1);
INSERT INTO employees VALUES (2, 'Jane Smith', 1);
INSERT INTO employees VALUES (3, 'Bob Johnson', 2);

```

Write the Sub Query:

4 write a subquery to find the employees who work in the 'IT' department.

```

SELECT name
FROM employee
WHERE department id = (SELECT id FROM department WHERE name = 'IT');

```

This subquery (SELECT id FROM departments WHERE name = 'IT') retrieves the department ID for the 'IT' department.

Execute the Query:

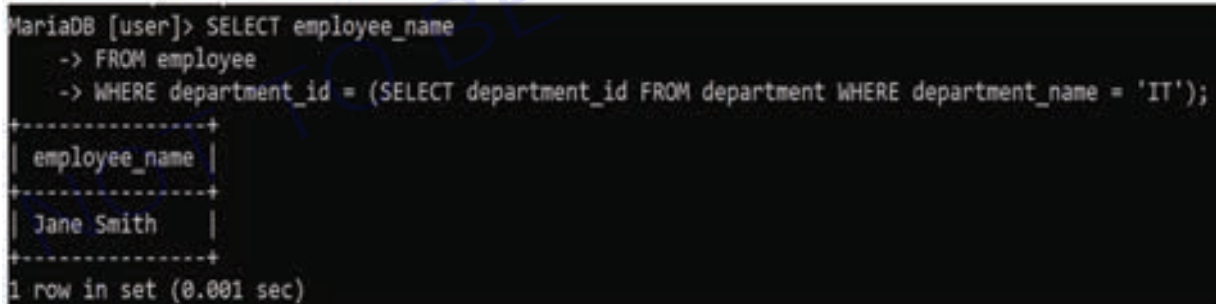
Run the entire query to get the result.

```

SELECT employee name
FROM employee
WHERE department id = (SELECT department id FROM department WHERE department name = 'IT');

```

The output should be:



```

MariaDB [user]> SELECT employee_name
  -> FROM employee
  -> WHERE department_id = (SELECT department_id FROM department WHERE department_name = 'IT');
+-----+
| employee_name |
+-----+
| Jane Smith    |
+-----+
1 row in set (0.001 sec)

```

This example demonstrates how to use a sub query to retrieve information based on conditions from another table.

Questions

1 How to retrieve a list of employees with their department names?

```

SELECT employees.name, departments.name AS department
FROM employees
JOIN departments ON employees.department id = departments.id;

```

2 How to get a count of employees in each department?

```

SELECT departments.name AS department, COUNT(*) AS employee count
FROM employees

```

JOIN departments ON employees.department id = departments.id

GROUP BY departments.name;

3 How to retrieve the average salary of employees in each department?

SELECT departments.name AS department, AVG(salary) AS average salary

FROM employees

JOIN departments ON employees.department id = departments.id

GROUP BY departments.name;

4 How to find the employees who earn more than the average salary in their department?

SELECT name, salary

FROM employees

WHERE salary > (

 SELECT AVG(salary)

 FROM employees AS e2

 WHERE e2.department id = employees.department id

);

5 How to retrieve departments with the highest average salary?

SELECT name AS department, AVG(salary) AS average salary

FROM employees

JOIN departments ON employees.department id = departments.id

GROUP BY departments.name

HAVING AVG(salary) = (

 SELECT MAX(avg salary)

 FROM (

 SELECT AVG(salary) AS avg salary

 FROM employees

 GROUP BY department id

) AS department avg

);

EXERCISE 30 : Indexing and Optimizing query

Objectives

At the end of this exercise you shall be able to

- create and use the index
- identify slow queries
- optimize query executive

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 : INDEXING:

1 Open MySQL Command Line-

Open the MySQL Command Line or any MySQL client that allows you to execute SQL commands.

2 Connect to your database-

```
mysql -u your username -p
```

Enter your password when prompted

3 Select Your Database-

USE your database name;

Identify the column to be indexed

Let's assume you want to create an index on a column named **example column** in a table named **your table**.

4 Create the Index-

```
CREATE INDEX idx example column ON your table(example column);
```

5 Verify the Index-

```
SHOW INDEX FROM your table;
```

This command displays information about the indexes on the specified table. You should see the newly created index in the output

Example Output –

```
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+
| Table      | Non unique | Key name      | Seq in index | Column name | Collation | Cardinality | Sub part | Packed
| Null      | Index type | Comment      | Index comment |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| your table |      1 | idx example column |      1 | example column | A      |      0 | NULL | NULL |
YES | BTREE      |      |      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+

```

In this output, you can see information about the index, including its name (idx example column), the indexed column (example column), and the index type (BTREE).

Test the query performance

After creating the index, you can test the performance of queries involving the indexed column to see if there is an improvement in retrieval speed.

TASK 2 : QUERY OPTIMIZATION

Optimizing MySQL queries is crucial for improving database performance. Here are steps you can take to optimize queries in MySQL on a Windows environment:

1 Identify slow queries-

Open MySQL Command Line or a MySQL client.

Execute the following command to enable the slow query log in your MySQL configuration file (my.ini):

```
SET GLOBAL slow query log = 'ON';
```

```
SET GLOBAL slow query log file = 'C:\path\to\your\log\file.log';
```

```
SET GLOBAL long query time = 1;
```

Replace 'C:\path\to\your\log\file.log' with the desired path for your slow query log file. The long query time is set to 1 second in this example, meaning queries taking longer than 1 second will be logged.

2 Analyze the slow query log:

Execute slow queries in your application or environment.

After some time, check the slow query log for potential issues:

```
SHOW VARIABLES LIKE 'slow query log';
```

```
SHOW VARIABLES LIKE 'slow query log file';
```

3 Use EXPLAIN to analyze queries-

Before optimizing, use the EXPLAIN statement to analyze the execution plan of a slow query:

```
EXPLAIN SELECT * FROM your table WHERE your condition;
```

Review the output to understand how MySQL is executing the query and identify potential bottlenecks.

4 Add Indexes:

Identify columns used in WHERE, JOIN, or ORDER BY clauses.

Create indexes on these columns to speed up query execution:

```
CREATE INDEX idx your column ON your table(your column);
```

Replace idx your column, your table, and your column with appropriate names.

5 Optimize WHERE Clause:

Use appropriate indexing to optimize WHERE clauses.

Avoid using functions or operations in the WHERE clause that prevent the use of indexes.

EXERCISE 31 : Creating & Using Stored procedure

Objectives

At the end of this exercise you shall be able to

- Create a stored procedure
- execute the stored procedure
- drop the stored procedure

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 :Create and use stored procedure

1 Connect to Your Database:

```
mysql -u your username -p
```

Enter your password when prompted.

Select Your Database:

```
USE your database name;
```

Replace your database name with the name of your actual database.

```
MariaDB [(none)]> create database user;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> use user;
Database changed
```

2 Create a simple Stored Procedure:

In this example, the stored procedure will concatenate "Hello, " with the input parameter and return the result.

```
DELIMITER //
```

```
CREATE PROCEDURE GetGreeting(IN input name VARCHAR(50), OUT greeting result VARCHAR(100))
```

```
BEGIN
```

```
SET greeting result = CONCAT('Hello, ', input name);
```

```
END //
```

```
DELIMITER ;
```

```
MariaDB [user]> DELIMITER //
MariaDB [user]> CREATE PROCEDURE GetGreeting(IN input_name VARCHAR(50), OUT greeting_result VARCHAR(100))
)
-> BEGIN
-> SET greeting_result = CONCAT('Hello, ', input_name);
-> END
-> //
Query OK, 0 rows affected (0.105 sec)
```

This stored procedure, named GetGreeting, takes an input parameter input name and returns an output parameter greeting result.

3 Call the Stored Procedure:

Now, you can call the stored procedure and capture the output. You'll use the CALL statement to execute the stored procedure.

```
SET @name = 'John';
CALL GetGreeting(@name, @greeting);
SELECT @greeting AS GreetingOutput;
```

In this example, @name is the input parameter, and @greeting is the output parameter. Replace 'John' with the desired input value.

4 View the Output:

After executing the above commands, you should see the output of the stored procedure.

```
MariaDB [user]> DELIMITER ;
MariaDB [user]> SET @name = 'John';
Query OK, 0 rows affected (0.000 sec)

MariaDB [user]> CALL GetGreeting(@name, @greeting);
Query OK, 0 rows affected (0.001 sec)

MariaDB [user]> SELECT @greeting AS GreetingOutput;
+-----+
| GreetingOutput |
+-----+
| Hello, John    |
+-----+
1 row in set (0.000 sec)
```

The output shows the result of the stored procedure, which is the greeting message based on the provided input.

5 Drop the Stored Procedure:

If you want to remove the stored procedure, you can use the following command:

```
DROP PROCEDURE IF EXISTS GetGreeting;
```

This step is optional and can be done if you no longer need the stored procedure.

```
MariaDB [user]> DROP PROCEDURE IF EXISTS GetGreeting;
Query OK, 0 rows affected (0.114 sec)

MariaDB [user]> CALL GetGreeting(@name, @greeting);
ERROR 1305 (42000): PROCEDURE user.GetGreeting does not exist
```

EXERCISE 32 : Creating and executing MySQL table level Triggers

Objectives

At the end of this exercise you shall be able to

- create the table level triggers
- use the table level triggers

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 :Create & execute triggers

1 Connect to Your Database –

```
mysql -u your username -p
```

Enter your password when prompted.

Select Your Database –

USE your database name;

Replace **your database name** with the name of your actual database.

```
MariaDB [(none)]> create database user;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> use user;
Database changed
```

2 Create a table –

Let's create a simple table for this example:

```
CREATE TABLE example table (
  id INT PRIMARY KEY AUTO INCREMENT,
  data VARCHAR(255),
  created at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

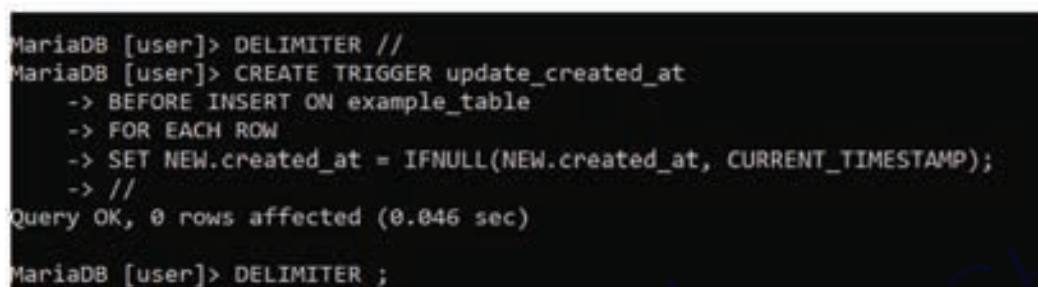
```
MariaDB [user]> CREATE TABLE example_table (
->   id INT PRIMARY KEY AUTO_INCREMENT,
->   data VARCHAR(255),
->   created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
-> )
-> ;
Query OK, 0 rows affected (0.103 sec)
```

This table has an id column as the primary key, a data column, and a created at column with a default value set to the current timestamp.

3 Create a Table Level Trigger –

Now, let's create a trigger that updates the created at timestamp whenever a new record is inserted:

```
DELIMITER //
CREATE TRIGGER update_created_at
BEFORE INSERT ON example_table
FOR EACH ROW
SET NEW.created_at = IFNULL(NEW.created_at, CURRENT_TIMESTAMP);
//
DELIMITER ;
```



```
MariaDB [user]> DELIMITER //
MariaDB [user]> CREATE TRIGGER update_created_at
-> BEFORE INSERT ON example_table
-> FOR EACH ROW
-> SET NEW.created_at = IFNULL(NEW.created_at, CURRENT_TIMESTAMP);
-> //
Query OK, 0 rows affected (0.046 sec)
MariaDB [user]> DELIMITER ;
```

This trigger is named **update created at** and is set to execute **BEFORE INSERT** on the **example table**. It updates the **created at** column with the current timestamp if the value is NULL.

4 Insert a Record –

Now, let's insert a record into the table:

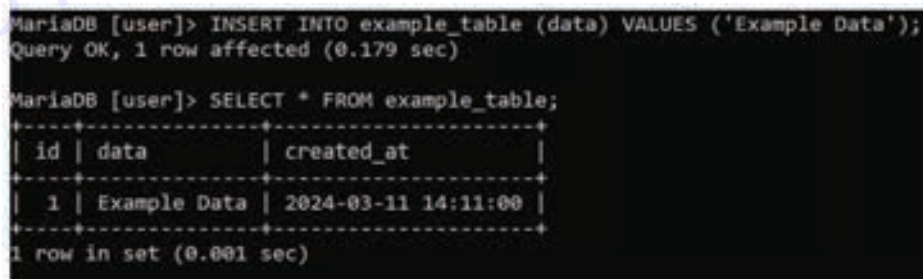
```
INSERT INTO example_table (data) VALUES ('Example Data');
```

View The Updated Record –

Now, retrieve the record to see the effect of the trigger:

```
SELECT * FROM example_table;
```

You should see the created at column automatically updated with the current timestamp.



```
MariaDB [user]> INSERT INTO example_table (data) VALUES ('Example Data');
Query OK, 1 row affected (0.179 sec)

MariaDB [user]> SELECT * FROM example_table;
+----+-----+-----+
| id | data      | created_at          |
+----+-----+-----+
| 1  | Example Data | 2024-03-11 14:11:00 |
+----+-----+-----+
1 row in set (0.001 sec)
```

5 DROP the Trigger:

If you want to remove the trigger, you can use the following command:

```
DROP TRIGGER IF EXISTS update_created_at;
```

This step is optional and can be done if you no longer need the trigger.

EXERCISE 33 : Creating Cursors in MYSQL

Objectives

At the end of this exercise you shall be able to

- create stored procedure with cursor

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 :Create cursors in MySQL

1 CONNECT TO YOUR DATABASE –

```
mysql -u your username -p
```

Enter your password when prompted.

2 SELECT YOUR DATABASE –

USE your database name;

Replace your database name with the name of your actual database

3 CREATE A TABLE –

Let's create a simple table for this example:

```
MariaDB [(none)]> create database user;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> use user;
Database changed
```

CREATE TABLE example table (

id INT PRIMARY KEY AUTO INCREMENT,

data VARCHAR(255)

);

```
MariaDB [user]> CREATE TABLE example_table (
-> id INT PRIMARY KEY AUTO_INCREMENT,
-> data VARCHAR(255)
-> );
Query OK, 0 rows affected (0.149 sec)
```

4 INSERT SAMPLE DATA –

Insert some sample data into the table:

```
INSERT INTO example table (data) VALUES ('Data 1'), ('Data 2'), ('Data 3');
```

5 Create A Stored Procedure With A Cursor –

Now, create a stored procedure that uses a cursor to iterate through the rows and display the data:

```
DELIMITER //
```

```
CREATE PROCEDURE IterateExampleTable()
```

```
BEGIN
```

```
    DECLARE done BOOLEAN DEFAULT FALSE;
```

```
    DECLARE data value VARCHAR(255);
```

```
    -- Declare a cursor for the table
```

```
    DECLARE cursor example CURSOR FOR
```

```
        SELECT data FROM example table;
```

```
    -- Declare an exit handler to close the cursor when no more rows are found
```

```
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
```

```
    -- Open the cursor
```

```
    OPEN cursor example;
```

```
    -- Start looping through the rows
```

```
    cursor loop: LOOP
```

```
        -- Fetch the next row into data value
```

```
        FETCH cursor example INTO data value;
```

```
        -- Check if we have reached the end of the cursor
```

```
        IF done THEN
```

```
            LEAVE cursor loop;
```

```
        END IF;
```

```
        -- Output the current row data
```

```
        SELECT data value AS OutputData;
```

```
    END LOOP;
```

```
    -- Close the cursor
```

```
    CLOSE cursor example;
```

```
END //
```

```
DELIMITER ;
```



```

MariaDB [user]> DELIMITER //
MariaDB [user]> CREATE PROCEDURE IterateExampleTable()
-> BEGIN
->     DECLARE done BOOLEAN DEFAULT FALSE;
->     DECLARE data_value VARCHAR(255);
->     DECLARE cursor_example CURSOR FOR
->         SELECT data FROM example_table;
->     DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
->     OPEN cursor_example;
->     cursor_loop: LOOP
->     FETCH cursor_example INTO data_value;
->     IF done THEN
->         LEAVE cursor_loop;
->     END IF;
->     SELECT data_value AS OutputData;
->     END LOOP;
->     CLOSE cursor_example;
-> END//
Query OK, 0 rows affected (0.053 sec)

```

6 Call The Stored Procedure

Now, call the stored procedure to execute the cursor:

CALL IterateExampleTable();

You should see the data from each row of the example table printed as output.

```

MariaDB [user]> DELIMITER ;
MariaDB [user]> CALL IterateExampleTable();
+-----+
| OutputData |
+-----+
| Data 1     |
+-----+
1 row in set (0.001 sec)

+-----+
| OutputData |
+-----+
| Data 2     |
+-----+
1 row in set (0.008 sec)

+-----+
| OutputData |
+-----+
| Data 3     |
+-----+
1 row in set (0.016 sec)

Query OK, 0 rows affected (0.019 sec)

```

Question

- 1 Explain the key components of a cursor declaration in MySQL.
- 2 When would you use a SCROLL cursor in MySQL, and provide an example?

EXERCISE 34 : Using Cursors in MySQL

Objectives

At the end of this exercise you shall be able to

- create a stored procedure with cursor
- use the cursors in MySQL

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 : Using a cursor in MySQL

1 OPEN MYSQL COMMAND LINE –

Open the MySQL Command Line or any MySQL client that allows you to execute SQL commands.

2 CONNECT YOUR DATABASE –

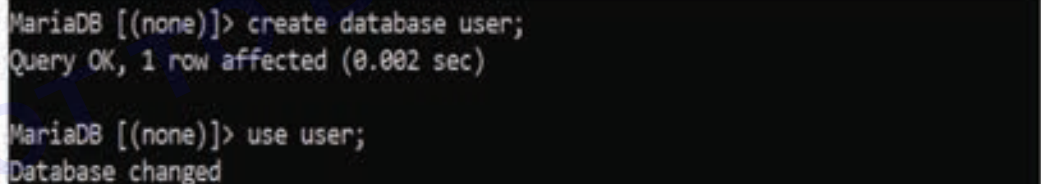
```
mysql -u your username -p
```

Enter your password when prompted.

3 SELECT YOUR DATABASE –

USE your database name;

Replace **your database name** with the name of your actual database.



```
MariaDB [(none)]> create database user;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> use user;
Database changed
```

4 Create a table –

Let's create a simple table for this example:

```
CREATE TABLE employee (
  emp id INT PRIMARY KEY AUTO INCREMENT,
  emp name VARCHAR(255),
  emp salary DECIMAL(10, 2)
);
```

```

MariaDB [user]> CREATE TABLE employee (
  ->     emp_id INT PRIMARY KEY AUTO_INCREMENT,
  ->     emp_name VARCHAR(255),
  ->     emp_salary DECIMAL(10, 2)
  -> );
Query OK, 0 rows affected (0.133 sec)

```

5 Insert sample data –

Insert some sample data into the table:

```
INSERT INTO employee (emp name, emp salary) VALUES
```

```
('John Doe', 50000.00),
```

```
('Jane Smith', 60000.00),
```

```
('Bob Johnson', 75000.00);
```

```

MariaDB [user]> INSERT INTO employee (emp_name, emp_salary) VALUES
  ->     ('John Doe', 50000.00),
  ->     ('Jane Smith', 60000.00),
  ->     ('Bob Johnson', 75000.00
  -> );
Query OK, 3 rows affected (0.041 sec)
Records: 3 Duplicates: 0 Warnings: 0

```

6 Create a stored procedure with a cursor

Now, create a stored procedure that uses a cursor to iterate through the rows and display the employee names:

```
DELIMITER //
```

```
CREATE PROCEDURE DisplayEmployeeNames()
```

```
BEGIN
```

```
    DECLARE done BOOLEAN DEFAULT FALSE;
```

```
    DECLARE emp name value VARCHAR(255);
```

```
    -- Declare a cursor for the table
```

```
    DECLARE cursor employee CURSOR FOR
```

```
        SELECT emp name FROM employee;
```

```
    -- Declare an exit handler to close the cursor when no more rows are found
```

```
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
```

```
    -- Open the cursor
```

```
    OPEN cursor employee;
```

```
    -- Start looping through the rows
```

```
cursor loop: LOOP
```

```
    -- Fetch the next row into emp name value
```

```
    FETCH cursor employee INTO emp name value;
```

```
    -- Check if we have reached the end of the cursor
```

```
    IF done THEN
```

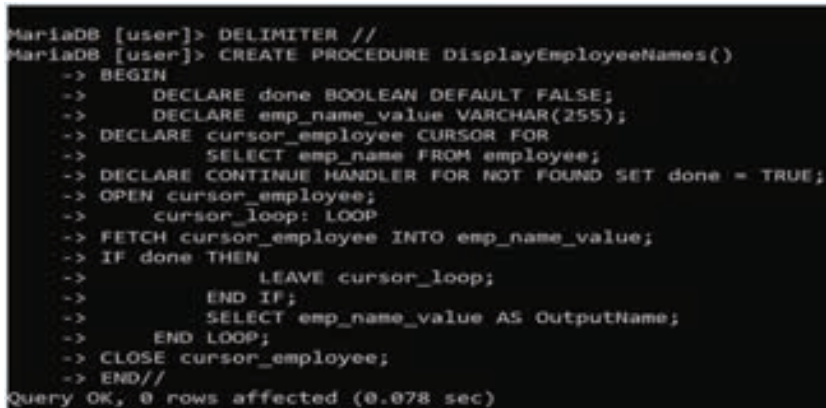
```
        LEAVE cursor loop;
```

```
    END IF;
```

```

-- Output the current row data
SELECT emp name value AS OutputName;
END LOOP;
-- Close the cursor
CLOSE cursor employee;
END //
DELIMITER ;

```



```

MariaDB [user]> DELIMITER //
MariaDB [user]> CREATE PROCEDURE DisplayEmployeeNames()
-> BEGIN
->   DECLARE done BOOLEAN DEFAULT FALSE;
->   DECLARE emp_name_value VARCHAR(255);
->   DECLARE cursor_employee CURSOR FOR
->     SELECT emp_name FROM employee;
->   DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
->   OPEN cursor_employee;
->   cursor_loop: LOOP
->     FETCH cursor_employee INTO emp_name_value;
->     IF done THEN
->       LEAVE cursor_loop;
->     END IF;
->     SELECT emp_name_value AS OutputName;
->   END LOOP;
->   CLOSE cursor_employee;
-> END//
Query OK, 0 rows affected (0.078 sec)

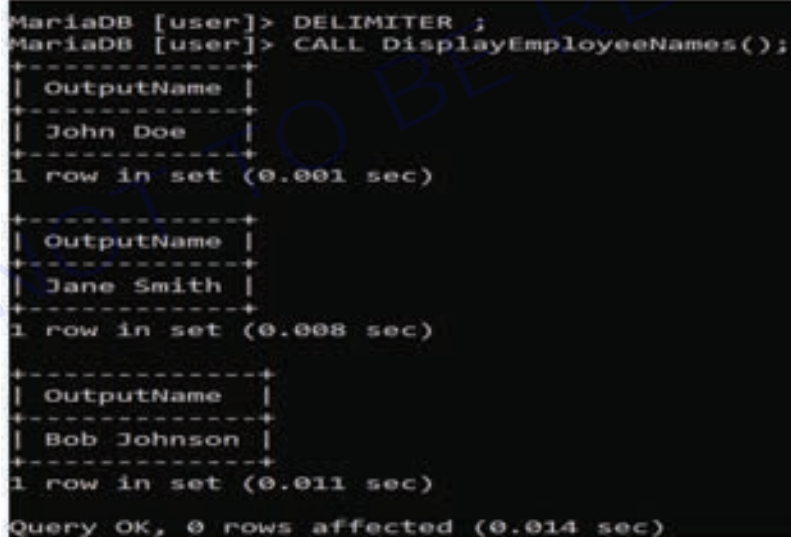
```

7 CALL THE STORED PROCEDURE –

Now, call the stored procedure to execute the cursor

CALL DisplayEmployeeNames();

You should see the employee names printed as output.



```

MariaDB [user]> DELIMITER ;
MariaDB [user]> CALL DisplayEmployeeNames();
+-----+
| OutputName |
+-----+
| John Doe   |
+-----+
1 row in set (0.001 sec)

+-----+
| OutputName |
+-----+
| Jane Smith |
+-----+
1 row in set (0.008 sec)

+-----+
| OutputName |
+-----+
| Bob Johnson |
+-----+
1 row in set (0.011 sec)

Query OK, 0 rows affected (0.014 sec)

```

- 1 Explain the concept of cursor parameters and provide an example of a stored procedure that uses input parameters to filter the result set retrieved by a cursor.

EXERCISE 35 : Implementing MySQL Security

Objectives

At the end of this exercise you shall be able to

- secure MySQL database

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 : The process of securing MySQL.

1 Change ROOT Password if needed

ALTER USER 'root'@'localhost' IDENTIFIED BY 'new password';

```
MariaDB [user]> ALTER USER 'root'@'localhost' IDENTIFIED BY '12345'
-> ;
Query OK, 0 rows affected (0.033 sec)
```

2 Create the New User

Create a new user with restricted privileges:

CREATE USER 'your username'@'localhost' IDENTIFIED BY 'your password';

```
MariaDB [user]> CREATE USER 'user1'@'localhost' IDENTIFIED BY '123';
Query OK, 0 rows affected (0.015 sec)
```

3 Grant Permissions

Grant specific permissions to your new user.

Replace your database with the name of the database you want to grant access to.

GRANT ALL PRIVILEGES ON your database.* TO 'your username'@'localhost';

FLUSH PRIVILEGES;

```
MariaDB [user]> GRANT ALL PRIVILEGES ON user.* TO 'user1'@'localhost';
Query OK, 0 rows affected (0.020 sec)
```

4 Remove Anonymous User

Remove anonymous users for security:

DROP USER ''@'localhost';

FLUSH PRIVILEGES;

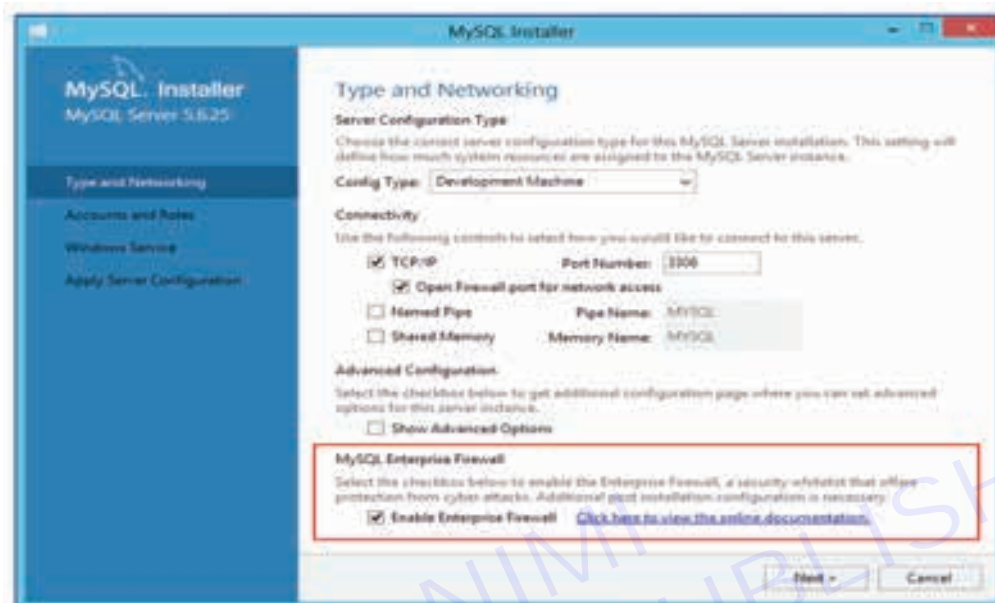
5 Bind MySQL to LocalHost

Edit the MySQL configuration file (my.ini or my.cnf) to bind MySQL to localhost:

bind-address = 127.0.0.1

6 Enable the Firewall

Open the Windows Firewall and create an inbound rule to allow traffic on the MySQL port (default is 3306).



7 Install and Use SSL/TLS

For additional security, consider configuring MySQL to use SSL/TLS. This involves obtaining SSL certificates and modifying the MySQL configuration.

8 Regularly Update MySQL

Stay updated with the latest MySQL releases to ensure that you have the latest security patches.

Questions

- 1 How do you secure the MySQL root account?
- 2 How do you grant SELECT privileges to a user for a specific database?

EXERCISE 36 : Simple application on database using SP, Triggers, Cursors & Indexing

Objectives

At the end of this exercise you shall be able to

- create simple application on database using SP Triggers cursors & indexing

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- Operating system: window 10:11
- XAMPP server r3.3.0

Procedure

TASK 1 : Creating a simple application using stored procedures, triggers, cursors, and indexing in MySQL

1 CREATE DATABASE –

```
CREATE DATABASE library db;
```

```
USE library db;
```

2 CREATE TABLES –

Create table books and users -

```
CREATE TABLE books (
    book id INT PRIMARY KEY AUTO INCREMENT,
    title VARCHAR(255) NOT NULL,
    author VARCHAR(255) NOT NULL,
    quantity INT NOT NULL
);
```

```
CREATE TABLE users (
    user id INT PRIMARY KEY AUTO INCREMENT,
    name VARCHAR(255) NOT NULL,
    email VARCHAR(255) NOT NULL
);
```

3 INSERT VALUES –

```
INSERT INTO books (title, author, quantity) VALUES
```

```
('Book 1', 'Author 1', 5),
```

```
('Book 2', 'Author 2', 8),
```

```
('Book 3', 'Author 3', 3);
```

```
INSERT INTO users (name, email) VALUES
```

```
('User 1', 'user1@example.com'),
```

```
('User 2', 'user2@example.com');
```

4 STORED PROCEDURE –

```

DELIMITER //
CREATE PROCEDURE GetBooksByAuthor(IN authorName VARCHAR(255))
BEGIN
    SELECT * FROM books WHERE author = authorName;
END //
DELIMITER ;
TRIGGER –
DELIMITER //
CREATE TRIGGER AfterBookBorrowed
AFTER INSERT ON borrowed books
FOR EACH ROW
BEGIN
    UPDATE books
    SET quantity = quantity - 1
    WHERE book id = NEW.book id;
END //
DELIMITER ;

```

5 CURSOR –

```

DELIMITER //
CREATE PROCEDURE DisplayBooksAndQuantities()
BEGIN
    DECLARE done INT DEFAULT FALSE;
    DECLARE book title VARCHAR(255);
    DECLARE book quantity INT;
    DECLARE books cursor CURSOR FOR
        SELECT title, quantity FROM books;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
    OPEN books cursor;
    read loop: LOOP
        FETCH books cursor INTO book title, book quantity;
        IF done THEN
            LEAVE read loop;
        END IF;
        SELECT CONCAT(book title, ': ', book quantity) AS BookInfo;
    END LOOP;
    CLOSE books cursor;
END //
DELIMITER ;

```


6 INDEXING –

```
CREATE INDEX idx_email ON users(email);
```

7 TESTING –

```
-- Test Stored Procedure
```

```
CALL GetBooksByAuthor('Author 1');
```

```
-- Test Trigger (Assuming you have a 'borrowed books' table)
```

```
INSERT INTO borrowed_books (user_id, book_id) VALUES (1, 1);
```

```
-- Test Cursor
```

```
CALL DisplayBooksAndQuantities();
```

```
-- Test Indexing
```

```
SELECT * FROM users WHERE email = 'user1@example.com';
```

Question

- 1 How would you call the InsertUser stored procedure to add a new user with the username "nsti" and email "nsti@example.com" to the database?
- 2 Explain the purpose of the before insert user trigger. How does it modify the incoming data before insertion into the users table?
- 3 What does the DisplayUsers procedure do, and how would you execute it to see the details of all users in the users table?
- 4 Describe the purpose of the idx_username index on the users table. How does it improve query performance?

◆ Module 3 : Introduction to JAVA Script ◆

EXERCISE 37 : Using the JAVA Script Syntax

Objectives

At the end of this exercise you shall be able to

- create javascriptcode as in line style
- use external javascript files in HTML.

Requirements

Tools/Materials

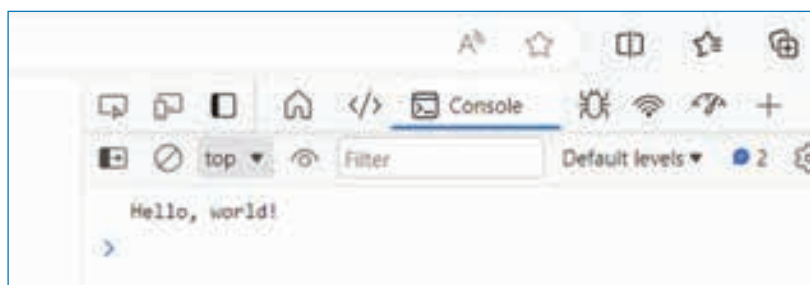
- Desktop / Laptop with latest configuration
- Text editor
- Web browser

Procedure

TASK 1: Inline JavaScript

- 1 Open the text editor
- 2 Write the following codes


```
<html >
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>JavaScript Example</title>
</head>
<body>
<script>
// JavaScript code goes here
console.log("Hello, world!");
</script>
</body>
</html>
```
- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



TASK 2: External JavaScript**I Create JavaScript file**

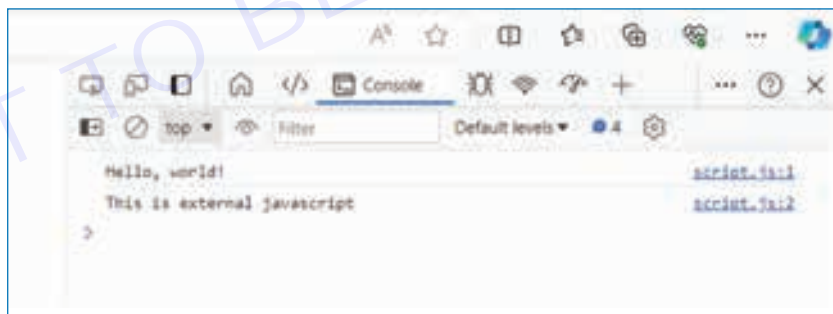
- 1 Open the text editor
- 2 Write the following codes

```
console.log("Hello, world!");
console.log("This is external javascript");
```
- 3 Save the program as 'script.js'

II Link the External JavaScript File to HTML

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>JavaScript Example</title>
<!-- Link to external JavaScript file -->
<script src="script.js"></script>
</head>
<body>
<!-- HTML content -->
</body>
</html>
```
- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



- 1 What is inline JavaScript, and how does it differ from external JavaScript files?
- 2 What are some advantages and disadvantages of using inline JavaScript in web development?
- 3 What are the different ways to include inline JavaScript in HTML elements
- 4 How do you include an external JavaScript file in your HTML document, and what are the different ways to specify the path to the file?

EXERCISE 38 : Using Variables, Operators and Writing Expressions

Objectives

At the end of this exercise you shall be able to

- create variables
- use operators
- use operators

Requirements

Tools/Materials

- Desktop / Laptop with latest configuration
- Text editor
- Web browser

Procedure

TASK 1: Using Variables

I Declaring variables and assigning values

1 Open the text editor

2 Write the following codes

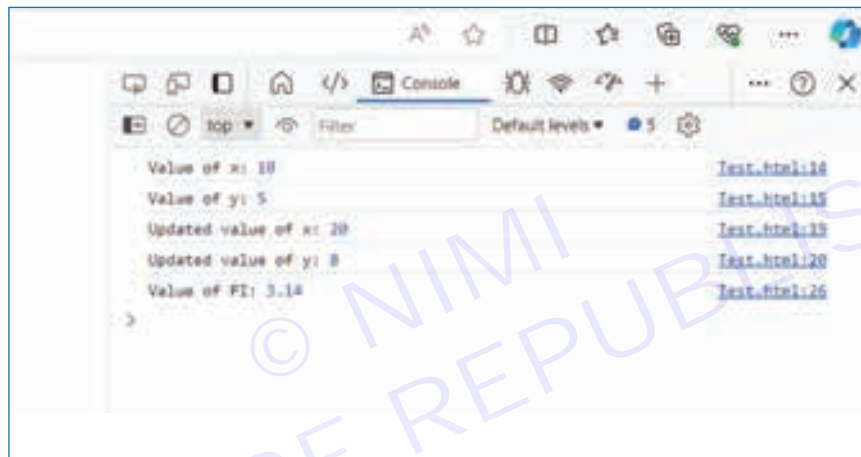
```
<html >
<head>
<title> Declaring variables and assigning values </title>
</head>
<body>
<script>
// Declaring variables
let x; // Declaration without initialization
let y = 5; // Declaration with initialization
// Assigning values to variables
x = 10;
// Printing variables
console.log("Value of x:", x); // Output: 10
console.log("Value of y:", y); // Output: 5
// Reassigning variables
x = 20;
y = 8;
console.log("Updated value of x:", x); // Output: 20
```

```

console.log("Updated value of y:", y); // Output: 8
// Constants
const PI = 3.14;
// PI = 3.14159; // Trying to reassign a constant will result in an error
console.log("Value of PI:", PI); // Output: 3.14
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



II Using different types of variable

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Different types variable </title>
</head>
<body>
<script>
// Numbers:
let num = 123.45;
// Strings:
let text = "This is a string.";
// Boolean (true/false):
let isTrue = true;
// Arrays (lists of values):

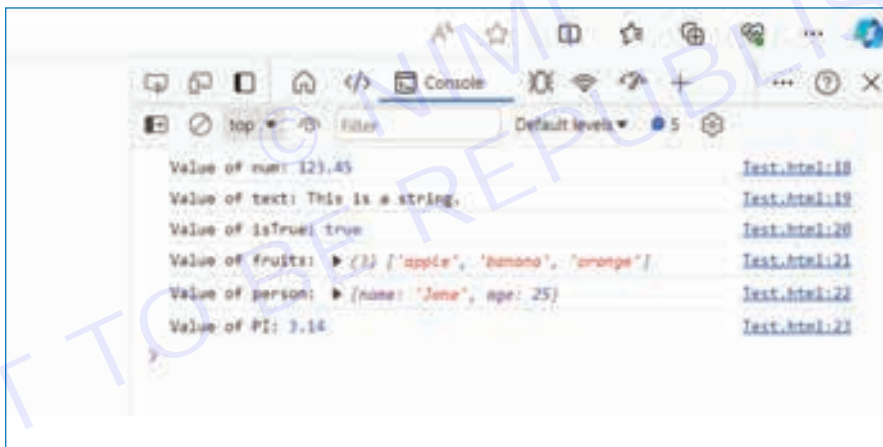
```

```

let fruits = ["apple", "banana", "orange"];
// Objects (collections of key-value pairs):
let person = { name: "Jane", age: 25 };
const PI=3.14;
console.log("Value of num:", num);
console.log("Value of text:", text);
console.log("Value of isTrue:", isTrue);
console.log("Value of fruits:", fruits);
console.log("Value of person:", person);
console.log("Value of PI:", PI); // Output: 3.14
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



TASK 2: Using Operators

I Arithmetic Operators

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Arithmetic Operators </title>
</head>
<body>
<script>
let a = 10;

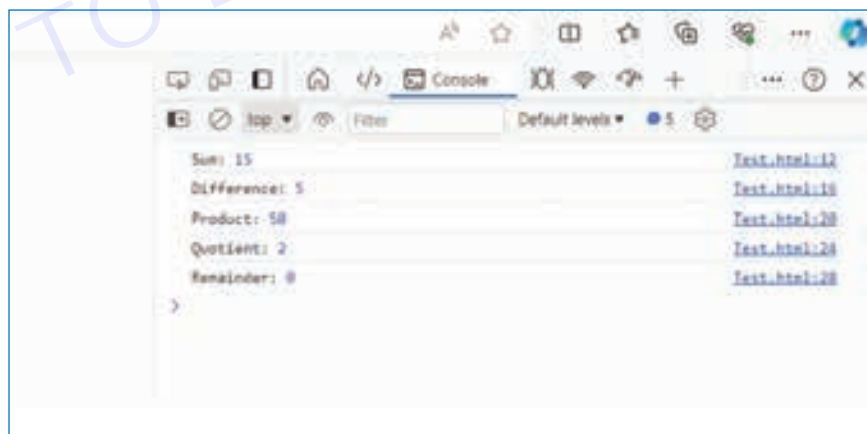
```

```

let b = 5;
// Addition
let sum = a + b;
console.log("Sum:", sum); // Output: 15
// Subtraction
let difference = a - b;
console.log("Difference:", difference); // Output: 5
// Multiplication
let product = a * b;
console.log("Product:", product); // Output: 50
// Division
let quotient = a / b;
console.log("Quotient:", quotient); // Output: 2
// Modulus
let remainder = a % b;
console.log("Remainder:", remainder);
// Output: 0 (10 divided by 5 leaves no remainder)
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



II Comparison operators

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Comparison operators </title>

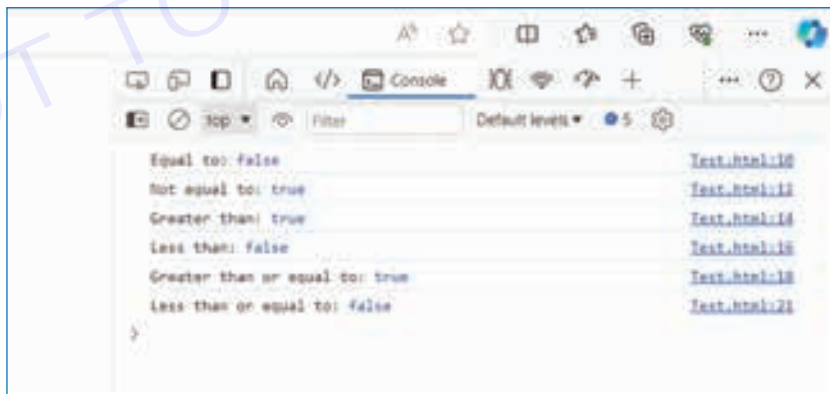
```

```

</head>
<body>
<script>
let a = 10;
let b = 5;
// Equal to
console.log("Equal to:", a == b); // Output: false
// Not equal to
console.log("Not equal to:", a != b); // Output: true
// Greater than
console.log("Greater than:", a > b); // Output: true
// Less than
console.log("Less than:", a < b); // Output: false
// Greater than or equal to
console.log("Greater than or equal to:", a >= b); // Output: true
// Less than or equal to
console.log("Less than or equal to:", a <= b); // Output: false
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



III Logical operators

- 1 Open the text editor
- 2 Write the following codes


```

<html >
<head>
<title> Logical operators </title>

```



```

</head>
<body>
<script>
let x = true;
let y = false;
// Logical AND (&&)
console.log("Logical AND:", x && y); // Output: false
// Logical OR (||)
console.log("Logical OR:", x || y); // Output: true
// Logical NOT (!)
console.log("Logical NOT for x:", !x); // Output: false
console.log("Logical NOT for y:", !y); // Output: true
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



IV Assignment Operators

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Assignment Operators </title>
</head>
<body>
<script>
let a = 10;

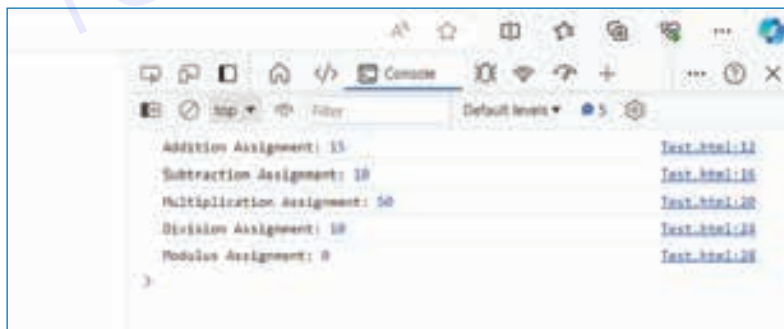
```

```

let b = 5;
// Addition assignment (+=)
a += b; // equivalent to: a = a + b;
console.log("Addition Assignment:", a); // Output: 15
// Subtraction assignment (-=)
a -= b; // equivalent to: a = a - b;
console.log("Subtraction Assignment:", a); // Output: 10
// Multiplication assignment (*=)
a *= b; // equivalent to: a = a * b;
console.log("Multiplication Assignment:", a); // Output: 50
// Division assignment (/=)
a /= b; // equivalent to: a = a / b;
console.log("Division Assignment:", a); // Output: 10
// Modulus assignment (%=)
a %= b; // equivalent to: a = a % b;
console.log("Modulus Assignment:", a);
// Output: 0 (since 10 divided by 5 leaves no remainder)
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



V. Ternary Operator

- 1 Open the text editor
- 2 Write the following codes


```

<html >
<head>
<title> Ternary Operator </title>

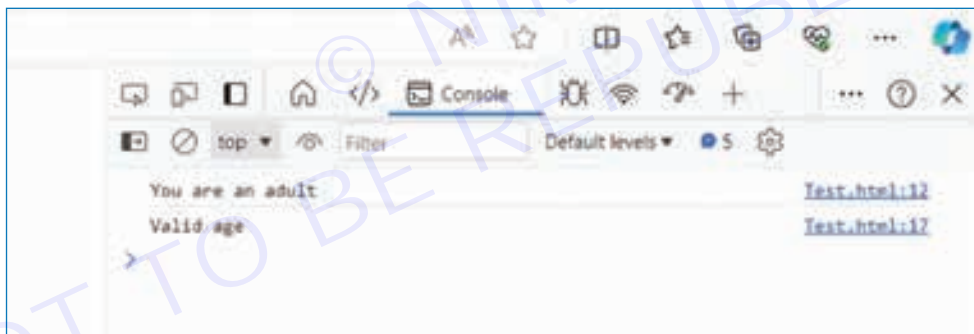
```

```

</head>
<body>
<script>
let age = 20;
// Ternary Operator (conditional operator)
let message = (age >= 18) ? "You are an adult" : "You are a minor";
console.log(message);
// Output: You are an adult (since age is 20, which is greater than or equal to 18)
// Another example
let result = (age < 0) ? "Invalid age" : "Valid age";
console.log(result); // Output: Valid age (since age is 20, which is not less than 0)
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



TASK 3: Writing Expressions

- 1 Open the text editor
- 2 Write the following codes

```

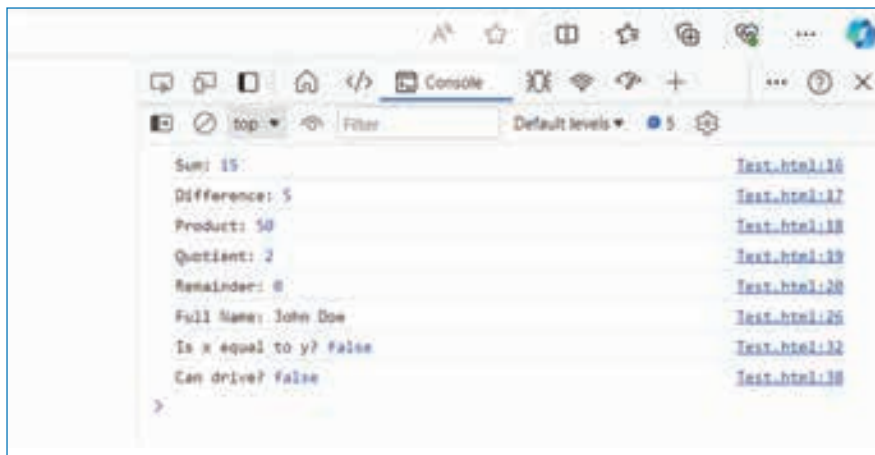
<html >
<head>
<title> Different types variable </title>
</head>
<body>
<script>
// Arithmetic Expressions
let a = 10;
let b = 5;

```

```

let sum = a + b;
let difference = a - b;
let product = a * b;
let quotient = a / b;
let remainder = a % b;
console.log("Sum:", sum); // Output: 15
console.log("Difference:", difference); // Output: 5
console.log("Product:", product); // Output: 50
console.log("Quotient:", quotient); // Output: 2
console.log("Remainder:", remainder); // Output: 0
// String Concatenation Expression
let firstName = "John";
let lastName = "Doe";
let fullName = firstName + " " + lastName;
console.log("Full Name:", fullName); // Output: John Doe
// Comparison Expressions
let x = 10;
let y = 5;
let isEqual = x === y;
console.log("Is x equal to y?", isEqual); // Output: false
// Logical Expressions
let isAdult = true;
let hasLicense = false;
let canDrive = isAdult && hasLicense;
console.log("Can drive?", canDrive); // Output: false
</script>
</body>
</html>

```



- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.
 - 1 How do you declare and initialize variables in JavaScript, and what are some best practices for naming conventions and variable declaration?
 - 2 Declare a variable without initializing it in JavaScript? If so, what value does it hold by default check it?
 - 3 Provide examples demonstrating the difference of var, let, and const type of variable declaration.
 - 4 Write JavaScript code that calculates the area of a rectangle and a circle. Declare appropriate variables for the length, width (rectangle), and radius (circle). Use appropriate formulas to calculate the area and display the results with clear labels (e.g., "Area of rectangle: ", "Area of circle: ").
 - 5 Write JavaScript code that converts a temperature value from Celsius to Fahrenheit. Declare a variable for the temperature in Celsius. Use the formula $(\text{Celsius} * 9/5) + 32$ to convert the temperature and store the result in a new variable. Display the original temperature in Celsius and the converted temperature in Fahrenheit with appropriate labels.
 - 6 Create a JavaScript program that converts time between different units (e.g., seconds to minutes, hours to days).
 - 7 Use the ternary operator to check if the entered age is greater than or equal to 18. If it is, display "You are eligible for voting," otherwise display "You are not eligible for voting."
 - 8 Develop a JavaScript program, in which enter the total purchase amount and whether they have a discount coupon (true/false). Use logical operators to calculate and display the final amount after applying the discount.
 - 9 Write a JavaScript program that prompts the user to enter their score for two different exams. Use comparison operators to determine which exam score is higher and display a message indicating which exam the user performed better in.
 - 10 Provide examples demonstrating modulus operators (%) usage and practical applications.
 - 11 Provide examples of unary operators commonly used in arithmetic expressions.

EXERCISE 39 : Programming with Control Flow statements

Objectives

At the end of this exercise you shall be able to

- use different types of conditional statements
- use different types of loop statements.

Requirements

Tools/Materials

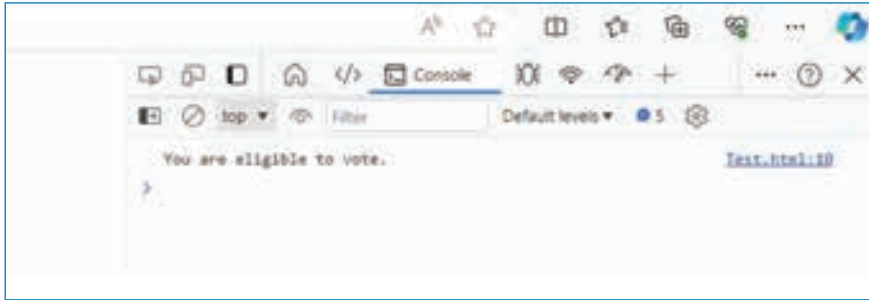
- Desktop / Laptop with latest configuration
- Text editor
- Web browser

Procedure

TASK 1: Using if statement

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
<title> if statement </title>
</head>
<body>
<script>
let age = 20;
if (age >= 18)
{
console.log("You are eligible to vote.");
}
</script>
</body>
</html>
```
- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



TASK 2: Using if-else statement

1 Open the text editor

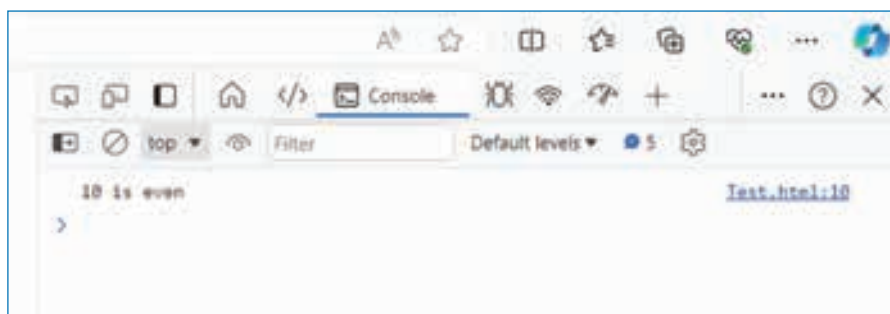
2 Write the following codes

```
<html >
<head>
<title> if statement </title>
</head>
<body>
<script>
const number = 10;
if (number % 2 === 0)
{
console.log(number+ " is even.");
}
else
{
console.log(number+ " is odd.");
}
</script>
</body>
</html>
```

3 Save the program as a .html file

4 Open the html file with a web browser

5 Go to the browsers console tab and verify the output.

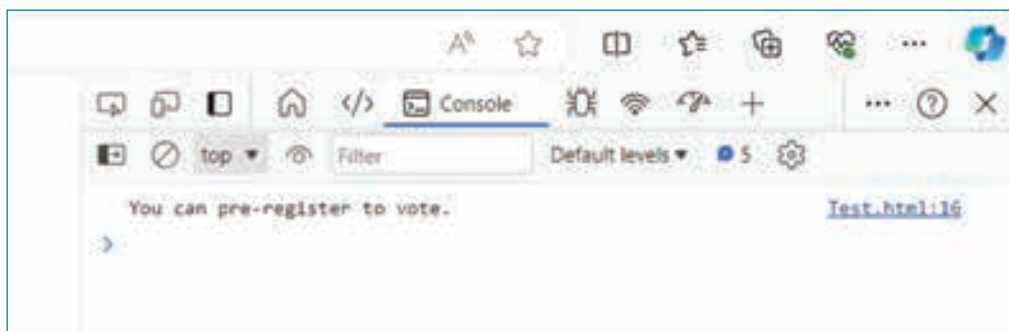


TASK 3 : Using if-else-if statement

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
<title> if-else-if statement </title>
</head>
<body>
<script>
const age = 17;
const votingAge = 18;
if (age >= votingAge)
{
console.log("You are eligible to vote.");
}
else if (age >= 16)
{
console.log("You can pre-register to vote.");
}
else
{
console.log("You are not eligible to vote yet.");
}
</script>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



TASK 4 : Using Nested if statements

1 Open the text editor

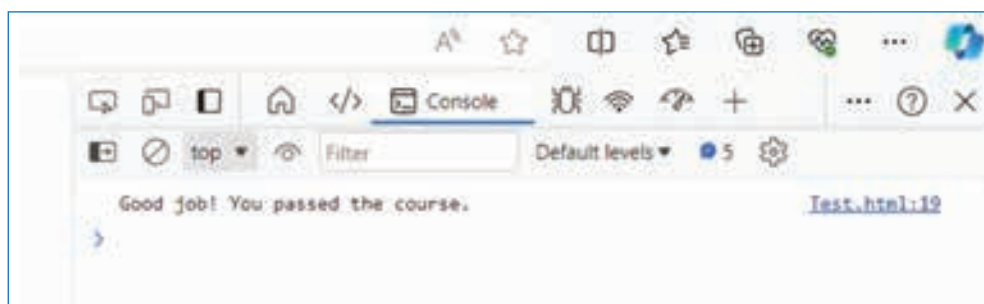
2 Write the following codes

```
<html >
<head>
<title> Nested if statement </title>
</head>
<body>
<script>
const grade = 85;
const passingGrade = 70;
const honorRoll = 90;
if (grade >= passingGrade)
{
if (grade >= honorRoll)
{
console.log("Excellent! You made the honor roll.");
}
else
{
console.log("Good job! You passed the course.");
}
}
else
{
console.log("You need to study more and retake the course.");
}
}
</script>
</body>
</html>
```

3 Save the program as a .html file

4 Open the html file with a web browser

5 Go to the browsers console tab and verify the output.



TASK 5 : Using switch statement

1 Open the text editor

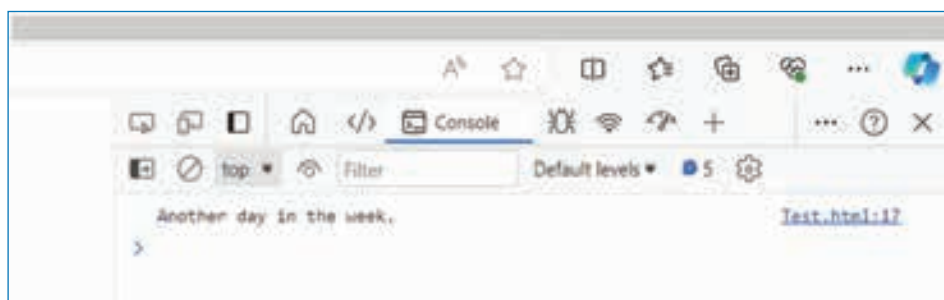
2 Write the following codes

```
<html >
<head>
<title> switch statement </title>
</head>
<body>
<script>
const day = "Wednesday";
switch (day)
{
case "Monday": console.log("Start of the week!");
break;
case "Friday": console.log("TGIF!");
break;
case "Weekend": console.log("Time to relax!");
break;
default: console.log("Another day in the week.");
}
</script>
</body>
</html>
```

3 Save the program as a .html file

4 Open the html file with a web browser

5 Go to the browsers console tab and verify the output.

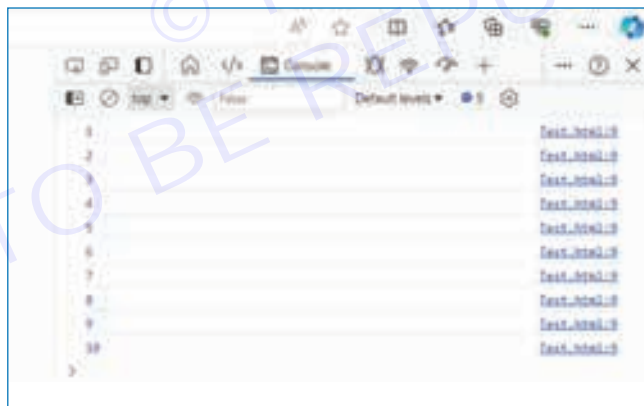


TASK 6 : Using for loop

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
<title> For loop </title>
</head>
<body>
<script>
for (let i = 1; i <= 10; i++)
{
console.log(i);
}
</script>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.

**TASK 7 : Using for loop and Iterating with conditions**

- 1 Open the text editor
- 2 Write the following codes

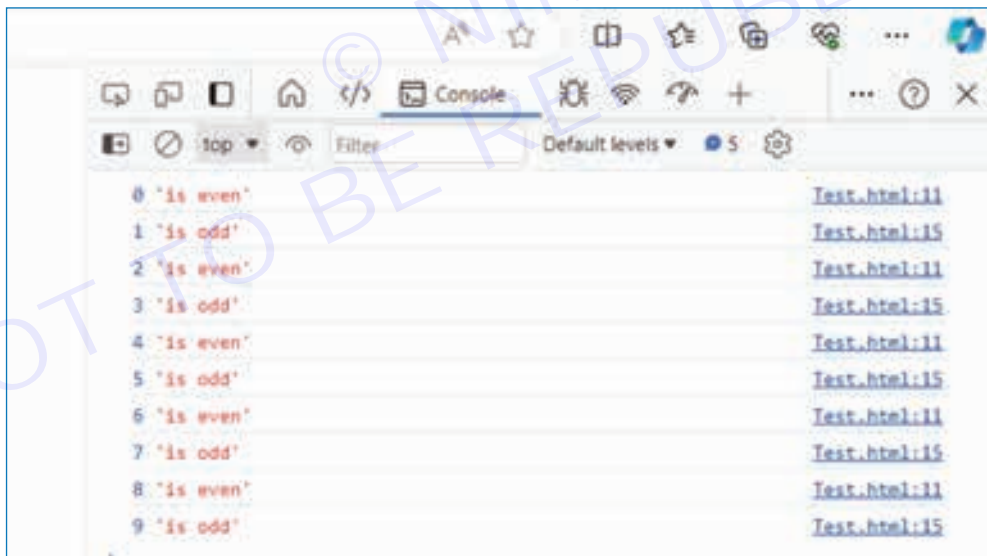
```
<html >
<head>
<title> . For loop Iterating with conditions </title>
</head>
<body>
```

```

<script>
for (let i = 0; i < 10; i++)
{
if (i % 2 === 0)
{
console.log(i, "is even");
}
else
{
console.log(i, "is odd");
}
}
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



TASK 8 : Using for loop and - Iterate through properties of an object

- 1 Open the text editor
- 2 Write the following codes


```

html >
<head>
<title> for loop </title>

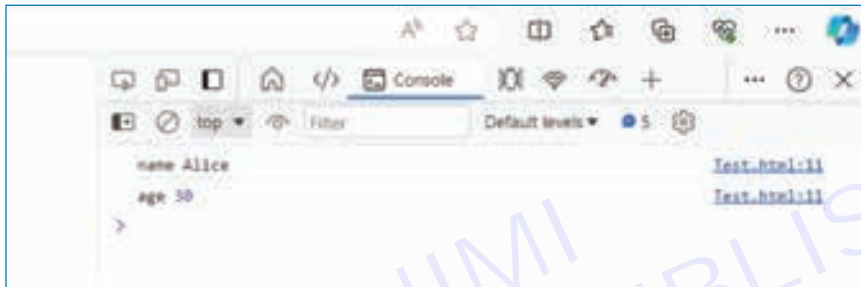
```

```

</head>
<body>
<script>
const person = { name: "Alice", age: 30 };
for (const key in person)
{
console.log(key, person[key]);
}
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



TASK 9 : Using for loop and -Iterate through values of iterable objects (arrays, strings etc.)

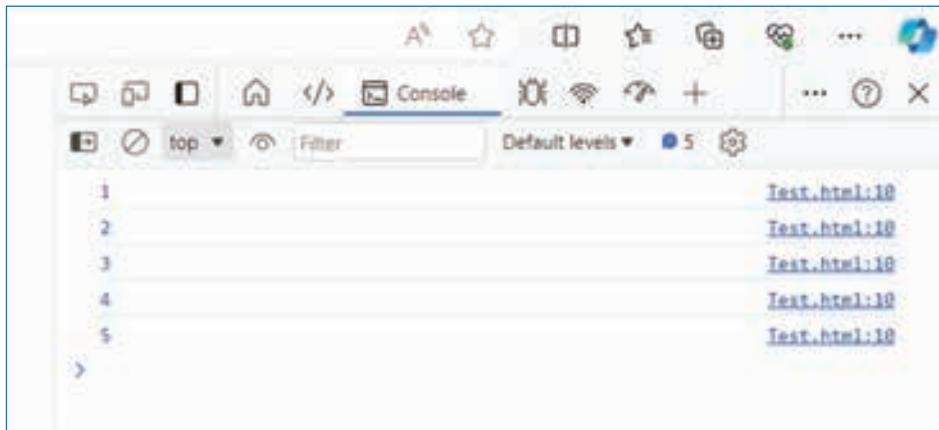
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> for loop </title>
</head>
<body>
<script>
const numbers = [1, 2, 3, 4, 5];
for (const number of numbers)
{
console.log(number);
}
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



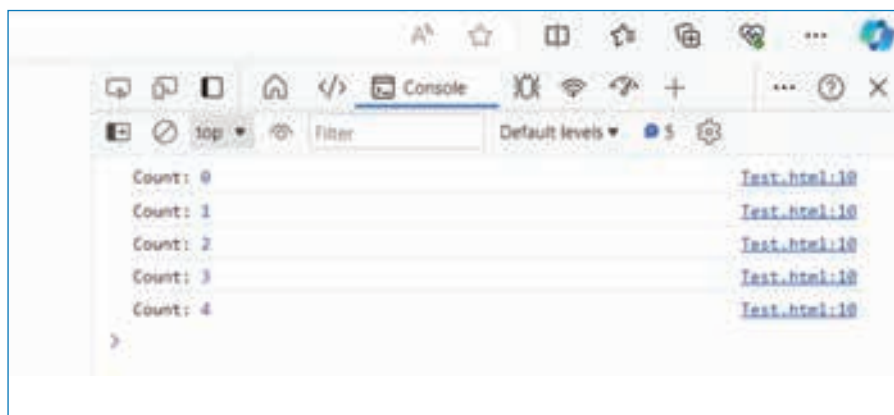
TASK 10 : Using while loop

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> while loop </title>
</head>
<body>
<script>
let count = 0;
while (count < 5)
{
console.log("Count:", count);
count++;
}
</script>
</body>
</html>

```



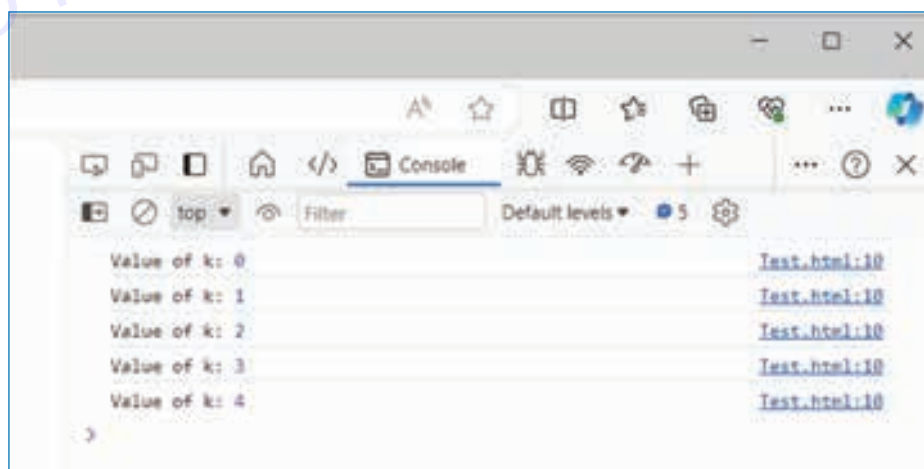
- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.

TASK 11 : Using do-while loop

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
<title> do-while </title>
</head>
<body>
<script>
let k = 0;
do
{
console.log("Value of k:", k);
k++;
} while (k < 5);
</script>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



- 1 Write a JavaScript program that takes a person's age as input and determines whether they are eligible for voting. If the age is 18 or above, the function should return "Eligible for voting," otherwise, it should return "Not eligible for voting." Use if-else statements to implement the logic
- 2 Develop a JavaScript program that takes a year as input and determines whether it is a leap year or not. If the year is divisible by 4 but not by 100, or if it is divisible by 400, it is considered a leap year.
- 3 Write a JavaScript program that takes two numbers as input and display the maximum of the two. Use if-else statements to implement the comparison.
- 4 Create a JavaScript program that takes a number representing a month (1 for January, 2 for February, etc.) and prints the corresponding season. Consider the following mappings: 1-3 (Winter), 4-6 (Spring), 7-9 (Summer), 10-12 (Fall). Use if-else-if statements to implement this logic.
- 5 Create a JavaScript program that takes a student's score as input and determines their grade. If the score is greater than or equal to 90, assign the grade 'A'; if between 80 and 89, assign 'B'; if between 70 and 79, assign 'C'; if between 60 and 69, assign 'D'; otherwise, assign 'F'. By Using if-else-if statements.
- 6 Write a JavaScript program that takes three numbers as input and determines the largest among them. However, if two or more numbers are equal, display a message indicating that there is a tie, using nested if-else statements.
- 7 Write a JavaScript program that takes a grade (A, B, C, D, F) as input and shows message indicating the corresponding description (excellent, good, average, poor, fail). Use a switch statement to implement this logic.
- 8 Create a JavaScript program that takes a month's number (1 for January, 2 for February, etc.) as input and returns the name of the month. Use a switch statement to implement this logic.
- 9 Develop a JavaScript program that generates the Fibonacci sequence up to a given number of terms using a for loop. The Fibonacci sequence starts with 0 and 1, and each subsequent number is the sum of the two preceding ones.
- 10 Write a JavaScript program that takes a number and then prints its multiplication table up to 10 using a for loop.
- 11 Create a JavaScript program that takes a positive integer as input and determines whether it is a prime number or not. Use a while loop to implement the prime number checking algorithm.
- 12 Write a JavaScript program that calculates the sum of digits of a given number. Use a while loop to extract digits one by one until the number becomes zero, and accumulate the sum.
- 13 Develop a JavaScript program that calculates the factorial of a given number using a do-while loop. The factorial of a number n is the product of all positive integers less than or equal to n .

EXERCISE 40 : Creating and using Objects in JavaScript

Objectives

At the end of this exercise you shall be able to

- create objects
- use document object model.

Requirements

Tools/Materials

- Desktop / Laptop with latest configuration
- Text editor
- Web browser

Procedure

TASK 1: Creating an Object

I Using Object Literal Notation

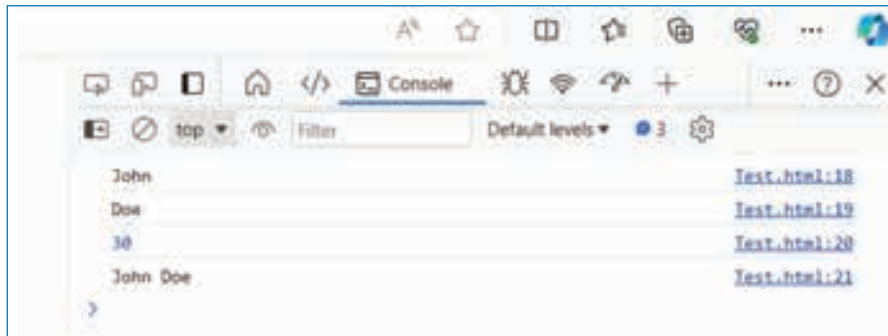
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Object Literal Notation </title>
</head>
<body>
<script>
// Creating an object using object literal notation
var person = {
  firstName: "John",
  lastName: "Doe",
  age: 30,
  fullName: function() {
    return this.firstName + " " + this.lastName;
  }
};
// Accessing object properties and methods
console.log(person.firstName); // Output: John
console.log(person.lastName); // Output: Doe
console.log(person.age); // Output: 30
console.log(person.fullName()); // Output: John Doe
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



II Using Constructor Functions

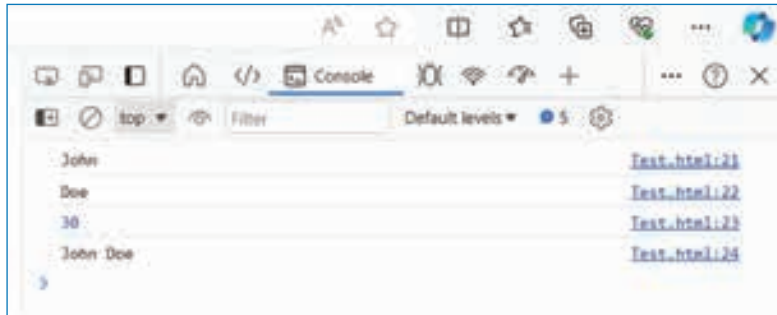
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Constructor Functions </title>
</head>
<body>
<script>
// Constructor function for creating person objects
function Person(firstName, lastName, age) {
this.firstName = firstName;
this.lastName = lastName;
this.age = age;
this.fullName = function() {
return this.firstName + " " + this.lastName;
};
}
// Creating a new instance of Person
var person1 = new Person("John", "Doe", 30);
// Accessing object properties and methods
console.log(person1.firstName); // Output: John
console.log(person1.lastName); // Output: Doe
console.log(person1.age); // Output: 30
console.log(person1.fullName()); // Output: John Doe
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



III Using Classes

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Classes </title>
</head>
<body>
<script>
// ES6 class for creating person objects
class Person {
  constructor(firstName, lastName, age) {
    this.firstName = firstName;
    this.lastName = lastName;
    this.age = age;
  }
  fullName() {
    return this.firstName + " " + this.lastName;
  }
}
// Creating a new instance of Person
let person2 = new Person("Jane", "Smith", 25);
// Accessing object properties and methods
console.log(person2.firstName); // Output: Jane
console.log(person2.lastName); // Output: Smith
console.log(person2.age); // Output: 25

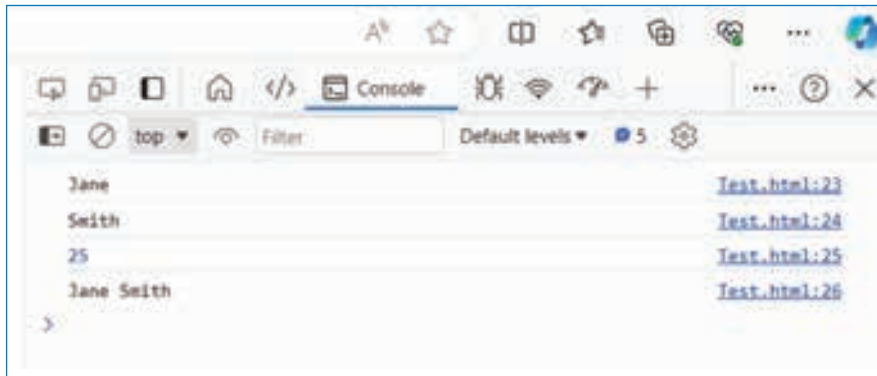
```

```

console.log(person2.fullName()); // Output: Jane Smith
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



TASK 2: Using Document Object Model

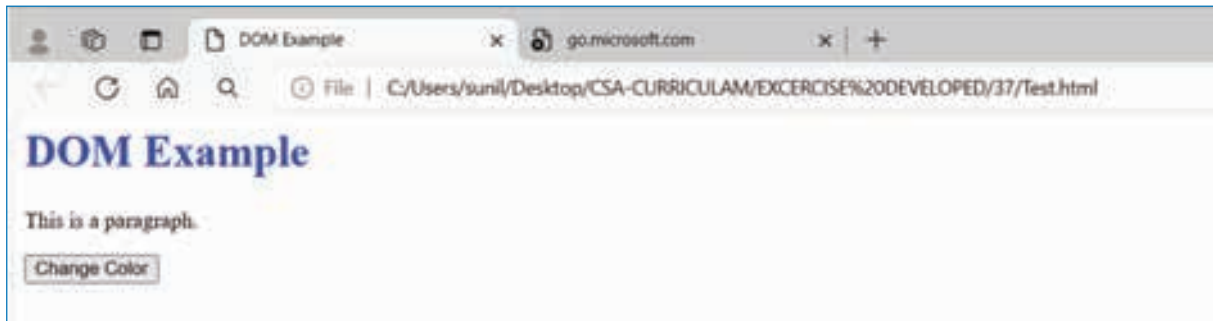
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title>DOM Example</title>
</head>
<body>
<h1 id="heading">DOM Example</h1>
<p id="paragraph">This is a paragraph.</p>
<button onclick="changeColor()">Change Color</button>
<script>
// Function to change the color of the heading
function changeColor() {
var heading = document.getElementById('heading');
heading.style.color = 'blue';
}
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



- 1 Create a JavaScript object named book with properties title, author, and pages. Initialize the title property with "The Great Gatsby", author with "F. Scott Fitzgerald", and pages with 180.
- 2 Define a JavaScript class named Circle with properties radius and calculateArea method that calculates the area of the circle. Create an instance of the Circle class with a radius of 7 and calculate its area.
- 3 Create a constructor function named Car that initializes the properties make, model, and year. Create an instance of the Car object with the make "Ford", model "Mustang", and year 2022.
- 4 Develop a JavaScript program that dynamically adds new elements (e.g., paragraphs, images) to a webpage when a button is clicked. Use DOM manipulation
- 5 Create a JavaScript program that generates a random number between 1 and 100. Display the generated number on the webpage inside a <div> element using the innerHTML property.
- 6 Develop a JavaScript program that calculates the factorial of a given number entered by the user. Display the result on the webpage using the innerHTML property.

EXERCISE 41 : Creating and using Functions

Objectives

At the end of this exercise you shall be able to

- create user defined functions
- create Anonymus functions

Requirements

Tools/Materials

- Desktop / Laptop with latest configuration
- Text editor
- Web browser

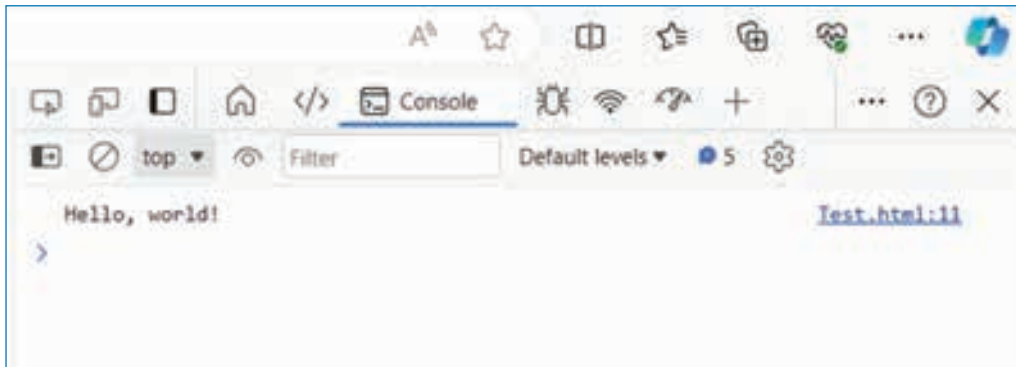
Procedure

TASK 1: Creating a Function

I Function without parameters

- 1 Open the text editor
- 2 Write the following codes


```
<html >
<head>
<title> Function without parameters </title>
</head>
<body>
<script>
// Defining a function named "greet" without any parameters
function greet()
{
// Inside the function, logging a greeting message to the console
console.log("Hello, world!");
}
// Calling the function
greet(); // Output: Hello, world!
</script>
</body>
</html>
```
- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



II. Function with parameters

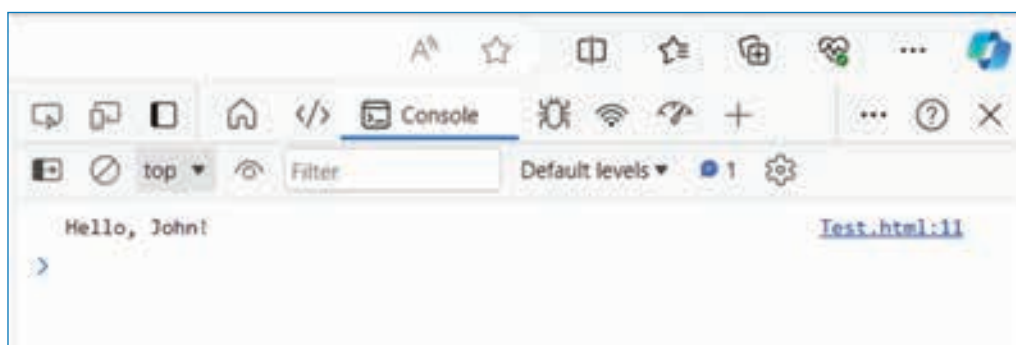
1. Open the text editor
2. Write the following codes

```

<html >
<head>
<title> Function with parameters </title>
</head>
<body>
<script>
// Defining a function named "greet" that takes a parameter named "name"
function greet(name)
{
// Inside the function, logging a greeting message to the console with the provided name
console.log("Hello, " + name + "!");
}
// Calling the function with an argument "John"
greet("John"); // Output: Hello, John!
</script>
</body>
</html>

```

3. Save the program as a .html file
4. Open the html file with a web browser
5. Go to the browsers console tab and verify the output.

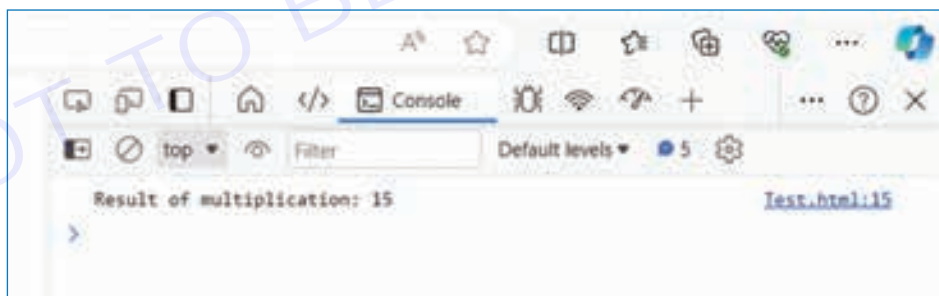


III. Function with return statement

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
<title> Function with return statement </title>
</head>
<body>
<script>
// JavaScript function with a return statement
function multiply(num1, num2) {
return num1 * num2;
}
// Example of using the function
var result = multiply(5, 3);
console.log("Result of multiplication: " + result);
</script> </script>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.

**TASK 2: Creating Anonymous Functions****I. Using Function Expression**

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
<title> </title>
</head>
<body>
```

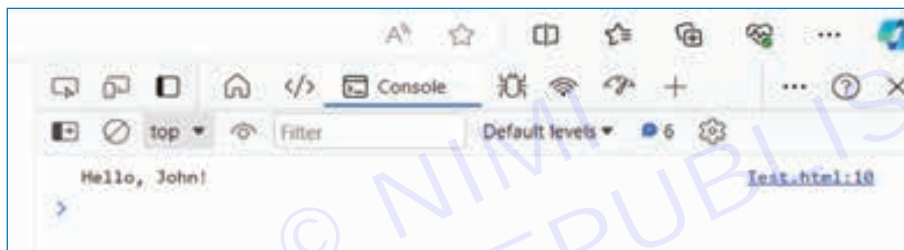


```

<script>
var greet = function(name)
{
// Inside the function, logging a greeting message to the console with the provided name
console.log("Hello, " + name + "!");
};
// Calling the function with an argument "John"
greet("John"); // Output: Hello, John!
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



II Using Arrow Function

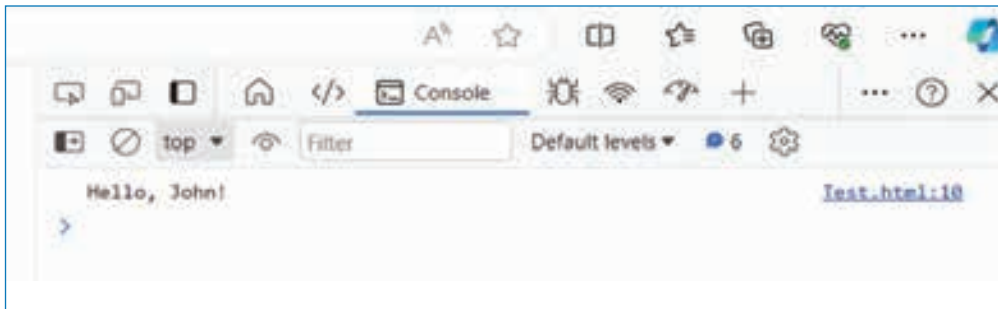
- 1 Open the text editor
- 2 Write the following codes


```

<html >
<head>
<title> </title>
</head>
<body>
<script>
const greet = () => {
console.log("Hello!");
};
greet(); // Call the function
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



- 1 Create a JavaScript function named `greetUser` that displays a greeting message on the webpage when called.
- 2 Write a JavaScript function named `printMessage` that takes a string parameter `message` and logs the message to the console.
- 3 Create a JavaScript function named `printMultiplicationTable` that takes a parameter `number` and prints the multiplication table of that number up to 10. Use `console.log()` to display each multiplication result.
- 4 Write a JavaScript function that takes two numbers as parameters and returns their sum. Store the result in a variable and log it to the console.
- 5 Create a JavaScript function named `calculateArea` that takes two parameters `length` and `width` and returns the area of a rectangle.
- 6 Write a JavaScript function named `calculatePower` that takes two parameters `base` and `exponent` and returns the result of raising the base to the exponent.
- 7 Create a function expression named `isEven` that takes a parameter `num` and returns `true` if the number is even, otherwise returns `false`.
- 8 Define an arrow function named `capitalizeString` that takes a parameter as a string and returns the string with the first letter capitalized.

EXERCISE 42 : Using Java Script with Forms

Objectives

At the end of this exercise you shall be able to

- use HTML forms with javascript
- validate HTML form using javascript
- create Dynamic form elements.

Requirements

Tools/Materials

- Desktop / Laptop with latest configuration
- Text editor
- Web browser

Procedure

TASK 1: Simple Form Submission

- 1 Open the text editor
- 2 Write the following codes


```
<html>
<head></head>
<body>
<form id="myForm">
<label for="username">Username:</label>
<input type="text" id="username" name="username">
<button type="submit">Submit</button>
</form>
<script>
const form = document.getElementById("myForm");
form.addEventListener("submit", handleSubmit);
function handleSubmit(event) {
event.preventDefault();

const username = document.getElementById("username").value;
console.log("Submitted username:", username);
}
</script>
</body>
</html>
```
- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Enter a text in the textbox and press submit button.
- 6 Go to the browsers console tab and verify the output.

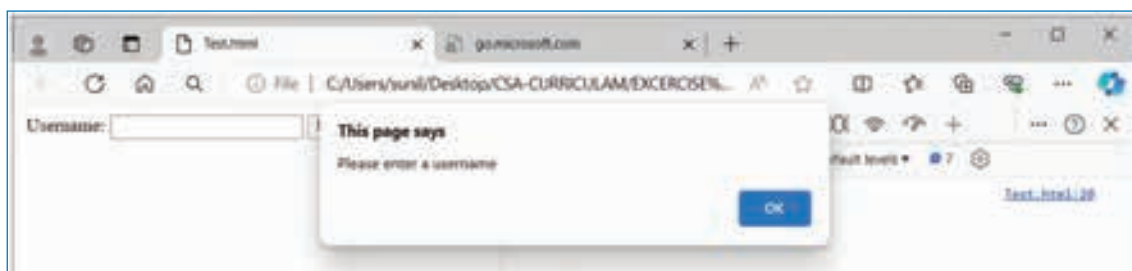


TASK 2: Form Validation

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head></head>
<body>
<form id="myForm">
<label for="username">Username:</label>
<input type="text" id="username" name="username">
<button type="submit">Submit</button>
</form>
<script>
const form = document.getElementById("myForm");
form.addEventListener("submit", handleSubmit);
function handleSubmit(event) {
event.preventDefault();
const username = document.getElementById("username").value;
if (username === "") {
alert("Please enter a username");
return; // Prevent form submission
}
console.log("Submitted username:", username);
}
</script>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browser's console tab and verify the output.



TASK 3: Create Dynamic Form Elements

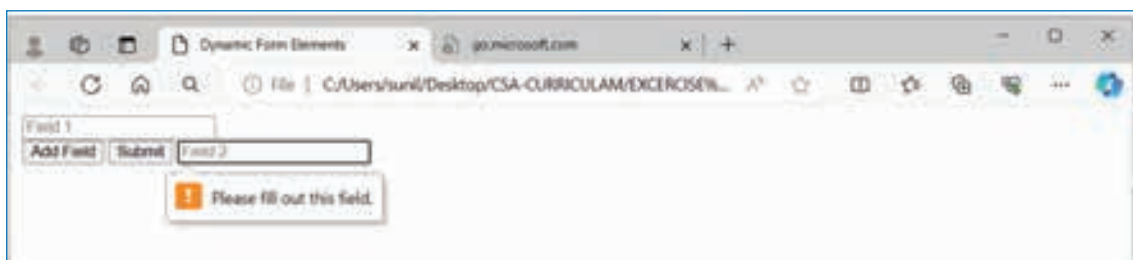
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title>Dynamic Form Elements</title>
</head>
<body>
<div id="container">
<form id="myForm">
<!-- Initially, the form has one input field -->
<input type="text" name="field1" placeholder="Field 1">
<br>
<button type="button" onclick="addField()">Add Field</button>
<button type="submit">Submit</button>
</form>
</div>
<script>
function addField() {
var form = document.getElementById("myForm");
var inputCount = form.getElementsByTagName("input").length + 1; // Get the number of existing input fields
and increment by 1
var newInput = document.createElement("input"); // Create a new input element
newInput.type = "text";
newInput.name = "field" + inputCount; // Assign a unique name to the new input field
newInput.placeholder = "Field " + inputCount; // Placeholder text
newInput.required = true; // Optionally, make the field required
form.appendChild(newInput); // Append the new input field to the form
}
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output..



- 1 Write a JavaScript program that validates an email input field in a form to ensure it is in a valid email format (e.g., contains "@" and "."). Display an error message if the email format is invalid.
- 2 Create a JavaScript program that implements form validation for a password input field. Validate the password to ensure it meets certain criteria, such as minimum length and inclusion of special characters. Display an error message if the password does not meet the criteria.
- 3 Write a JavaScript program that dynamically adds input fields (e.g., text input, select dropdown) to a form when a button is clicked.
- 4 Create a JavaScript program that dynamically removes input fields from a form when a button is clicked, based on user interaction or condition.
- 5 Create a JavaScript program that disables a form submit button until all required fields in the form are filled. Enable the submit button only when all required fields are filled.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 43 : Creating Cookies with JavaScript

Objectives

At the end of this exercise you shall be able to

- create cookies using javascript
- display cookie details using javascript
- delete a cookie using javascript.

Requirements

Tools/Materials

- Desktop / Laptop with latest configuration
- Text editor
- Web browser

Procedure

TASK 1: Create a cookie and display its value using an alert

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head>
<title> Cookies </title>
</head>
<body>
<script>
document.cookie="username=sun";
document.cookie="age=33";
document.cookie="country=india";
alert(document.cookie);
</script>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser and verify the output.

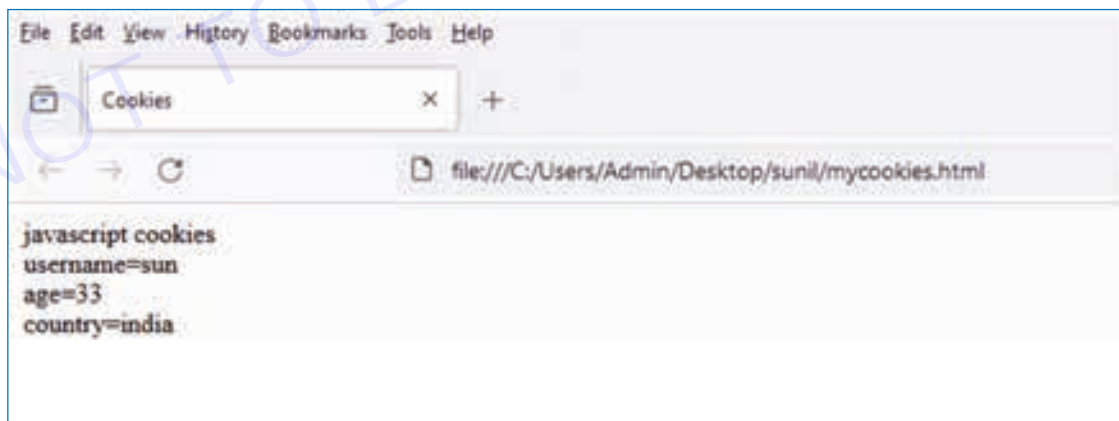


TASK 2: Create a cookie and display its value using innerHTML property

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head>
<title> Cookies </title>
</head>
<body>
<p id="para">javascript cookies</p>
<script>
document.cookie="username=sun";
document.cookie="age=33";
document.cookie="country=india";
var cookie=document.cookie.split(";");
for(i=0;i<cookie.length;i++)
{
document.getElementById("para").innerHTML+="<br />" +cookie[i];
}
</script>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser and verify the output.



TASK 3 : Create and display a cookie using a user defined function

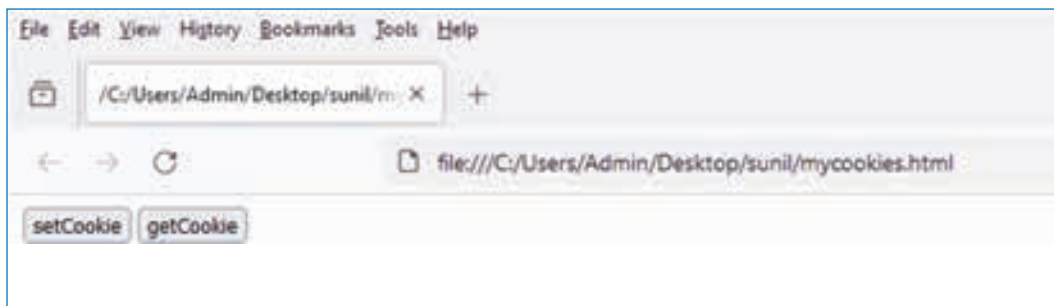
- 1 Open the text editor
- 2 Write the following codes

```

<html>
<head>
<script>
function setCookie()
{
document.cookie="username=Duke Martin";
}
function getCookie()
{
if(document.cookie.length!=0)
{
alert(document.cookie);
}
else
{
alert("Cookie not available");
}
}
</script>
</head>
<body>
<input type="button" value="setCookie" onclick="setCookie()">
<input type="button" value="getCookie" onclick="getCookie()">
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser and verify the output.





TASK 4 : Deleting cookie

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head>
<title> Cookies </title>
</head>
<body>
<p id="para">javascript cookies</p>
<script>
function deleteCookie(name)
{
// Set the cookie name with an empty value and an expired date in the past.
document.cookie = name + "=; expires=Thu, 01 Jan 1970 00:00:00 UTC; path=/;";
alert("Cookie not available");
}
// Example usage:
deleteCookie("username ");
</script>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser and verify the output.



- 1 Write a JavaScript program that creates a cookie named "username" with the value "John" and an expiration time of 7 days.
- 2 Write a JavaScript program that creates a cookie named "visited" with the value "true" when a user first visits a webpage. Check if the cookie exists when the user revisits the page and display a welcome message accordingly.
- 3 Create a JavaScript program that prompts the user to enter their preferred language and creates a cookie named "language" with the chosen language as its value. Ensure that the cookie expires after 30 days.

EXERCISE 44 : Creating CSS

Objectives

At the end of this exercise you shall be able to

- use css in HTML page
- manipulate the style property of HTML elements using javascript.

Requirements

Tools/Materials

- Desktop / Laptop with latest configuration
- Text editor
- Web browser

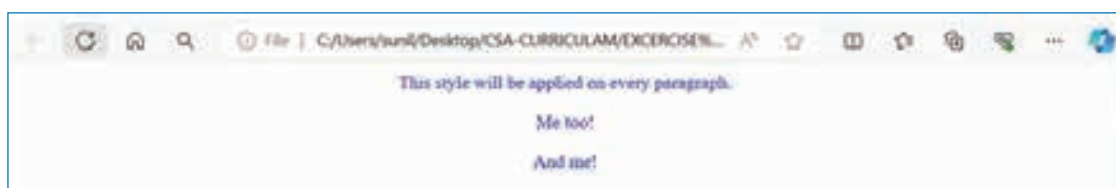
Procedure

TASK 1: Using CSS Element Selector

- 1 Using open the text editor
- 2 Write the following codes

```
<html>
<head>
<style>
p{
text-align: center;
color: blue;
}
</style>
</head>
<body>
<p>This style will be applied on every paragraph.</p>
<p id="para1">Me too!</p>
<p>And me!</p>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 2: Using CSS Id Selector

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head>
<style>
#para1 {
text-align: center;
color: blue;
}
</style>
</head>
<body>
<p id="para1">Hello Javatpoint.com</p>
<p>This paragraph will not be affected.</p>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.

**TASK 3: Using CSS Class Selector**

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head>
<style>
.center {
text-align:right;
color: blue;
}
h1.center {
text-align: center;
color: red;
}
</style>
```

```

</style>
</head>
<body>
<h1 class="center">This heading is red and center-aligned.</h1>
<p class="center">This paragraph is blue and right-aligned.</p>
<p class="center">This paragraph is blue and right-aligned.</p>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



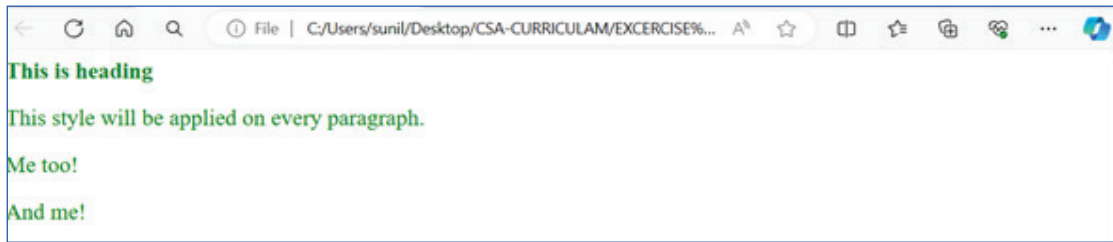
TASK 4: Using CSS Universal Selector

- 1 Open the text editor
- 2 Write the following codes


```

<html>
<head>
<style>
* {
color: green;
font-size: 20px;
}
</style>
</head>
<body>
<h2>This is heading</h2>
<p>This style will be applied on every paragraph.</p>
<p id="para1">Me too!</p>
<p>And me!</p>
</body>
</html>

```
- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 5: Using CSS Group Selector

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head>
<style>
h1, h2, p {
text-align: center;
color: blue;
}
</style>
</head>
<body>
<h1>Hello Javatpoint.com</h1>
<h2>Hello Javatpoint.com (In smaller font)</h2>
<p>This is a paragraph.</p>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 6: Using External Stylesheet**I create an external stylesheet**

- 1 Open the text editor
- 2 Write the following codes

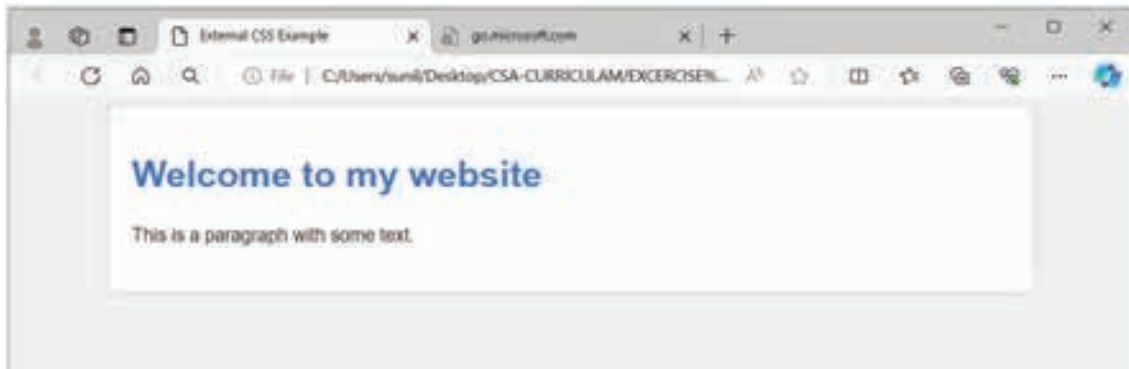

```
/* styles.css */
body {
font-family: Arial, sans-serif;
background-color: #f0f0f0;
color: #333;
}
h1 {
color: #007bff;
}
.container {
width: 80%;
margin: 0 auto;
padding: 20px;
background-color: #fff;
border-radius: 5px;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
p {
line-height: 1.6;
}
```

- 3 Save the coding as a style.css file

II Link the css file in HTML

- 1 Open the text editor
- 2 Write the following codes


```
<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>External CSS Example</title>
<link rel="stylesheet" href="styles.css">
</head>
<body>
<div class="container">
<h1>Welcome to my website</h1>
<p>This is a paragraph with some text.</p>
</div>
</body>
</html>
```
- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 7: Using Internal Stylesheet

- 1 Open the text editor
- 2 Write the following codes

```

<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Internal CSS Example</title>
<style>
/* CSS code here */
body {
font-family: Arial, sans-serif;
background-color: #f0f0f0;
color: #333;
}
h1 {
color: #007bff;
}
.container {
width: 80%;
margin: 0 auto;
padding: 20px;
background-color: #fff;
border-radius: 5px;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
p {
line-height: 1.6;
}
</style>
</head>
<body>

```

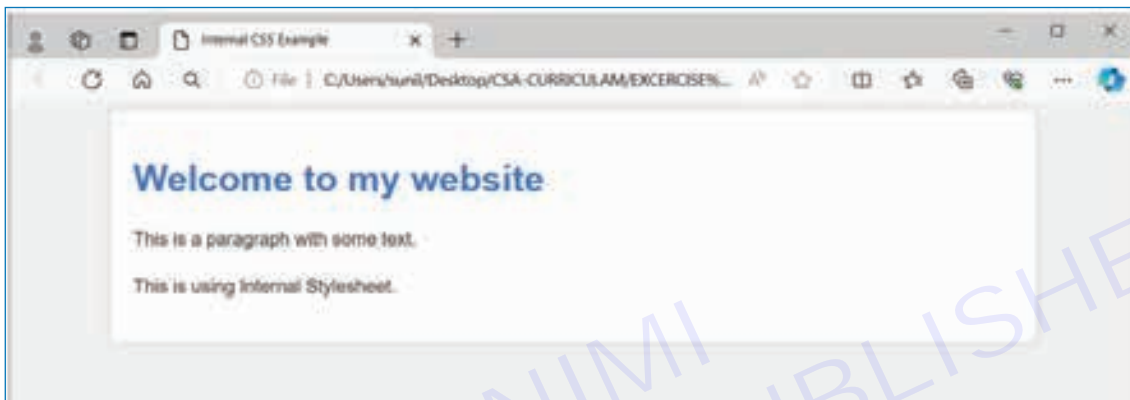


```

<div class="container">
<h1>Welcome to my website</h1>
<p>This is a paragraph with some text.</p>
<p>This is using Internal Stylesheet.</p>
</div>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 8: Using Inline CSS

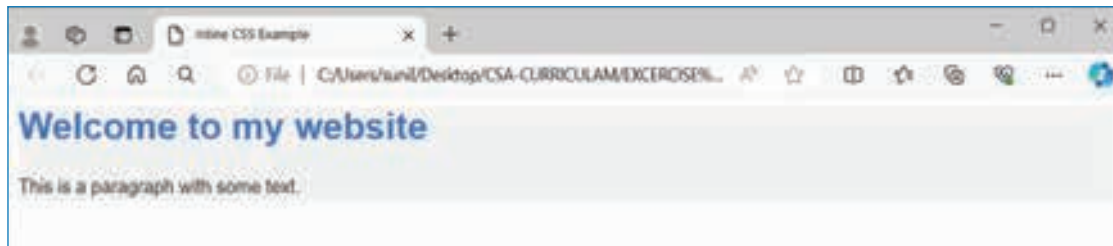
- 1 Open the text editor
- 2 Write the following codes

```

<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Inline CSS Example</title>
</head>
<body>
<div style="font-family: Arial, sans-serif; background-color: #f0f0f0; color: #333;">
<h1 style="color: #007bff;">Welcome to my website</h1>
<p style="line-height: 1.6;">This is a paragraph with some text.</p>
</div>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 9 : Directly manipulating the style property of HTML elements using Javascript

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Directly Manipulating Style Property</title>
<style>
/* CSS for demonstration purposes only */
.box {
width: 100px;
height: 100px;
background-color: blue;
color: white;
text-align: center;
line-height: 100px;
margin: 20px;
}
</style>
</head>
<body>
<div id="box1" class="box">Box 1</div>
<div id="box2" class="box">Box 2</div>
<script>
// Get references to the HTML elements
const box1 = document.getElementById('box1');
const box2 = document.getElementById('box2');
// Apply CSS styles programmatically
box1.style.backgroundColor = 'red';
box1.style.color = 'black';
box1.style.border = '2px solid black';
  
```

```

box2.style.backgroundColor = 'green';
box2.style.color = 'white';
box2.style.border = '2px solid white';
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 10 : creates a <style> element dynamically and appends it to the document

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Creating a Style Element</title>
</head>
<body>
<div id="content">This text will be styled.</div>
<script>
// Create a style element
const styleElement = document.createElement('style');
// Define your CSS rules
const cssRules = `
#content {
color: blue;
font-size: 20px;
font-weight: bold;

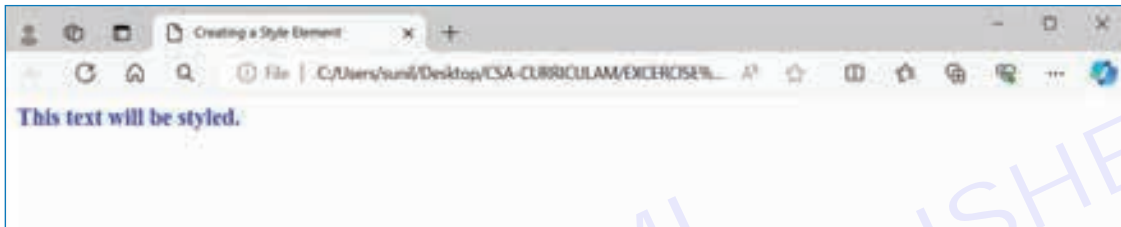
```

```

}
;
// Set the CSS rules to the style element
styleElement.innerHTML = cssRules;
// Append the style element to the document's head
document.head.appendChild(styleElement);
</script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



- 1 Style all <h1> elements on a webpage to have a font size of 24px and be displayed in bold by using CSS element Selector.
- 2 Apply a background color of yellow to all <div> elements with the class container by using CSS element Selector.
- 3 Style the font size of an element to be 24px and bold by using its id.
- 4 Style all elements with the class "btn" to have a background color of blue and white text color using CSS.
- 5 Style the color of all text on the webpage to be black using the universal selector.
- 6 Set the text alignment to center for all <h1>, <h2>, and <h3> elements on the webpage by using CSS group Selector.
- 7 Create an external stylesheet and link to an HTML document and apply different style to elements.
- 8 Provide an example of how an inline CSS is applied to an HTML element.

EXERCISE 45 : Error Handling in JavaScript

Objectives

At the end of this exercise you shall be able to

- handle error using try - catch
- handle error using try - catch and throw
- handle error using try - catch finally.

Requirements

Tools/Materials

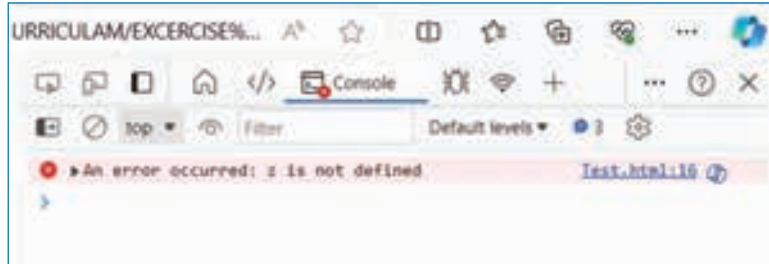
- Desktop / Laptop with latest configuration
- Text editor
- Web browser

Procedure

TASK 1: Error Handling using try-catch

- 1 Open the text editor
- 2 Write the following codes


```
<html >
<head>
<title> Object Literal Notation </title>
</head>
<body>
<script>
try
{
// Code that may throw an error
let x = 1;
let y = x + z; // z is not defined, this will throw an error
}
catch (error)
{
// Handle the error
console.error("An error occurred:", error.message);
}
</script>
</body>
</html>
```
- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



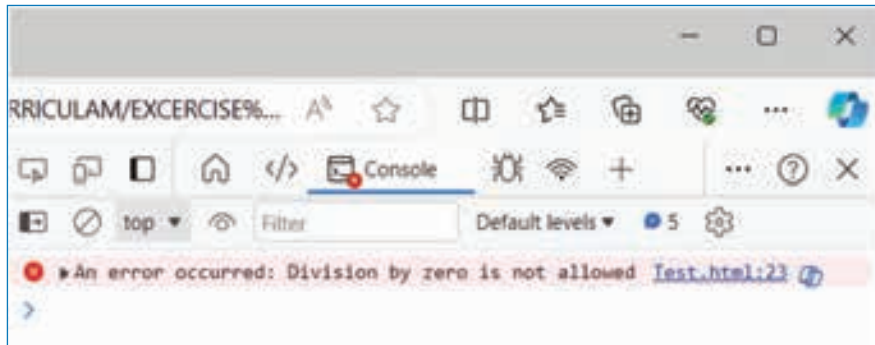
TASK 2: Error Handling using try-catch and throw

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Object Literal Notation </title>
</head>
<body>
<script>
function divide(a, b)
{
if (b === 0)
{
throw new Error("Division by zero is not allowed");
}
return a / b;
}
try
{
console.log(divide(10, 0));
}
catch (error)
{
console.error("An error occurred:", error.message);
}
</script>
</body>
</html>
  
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Go to the browsers console tab and verify the output.



TASK 3: Error Handling using try-catch-finally

1 Open the text editor

2 Write the following codes

```

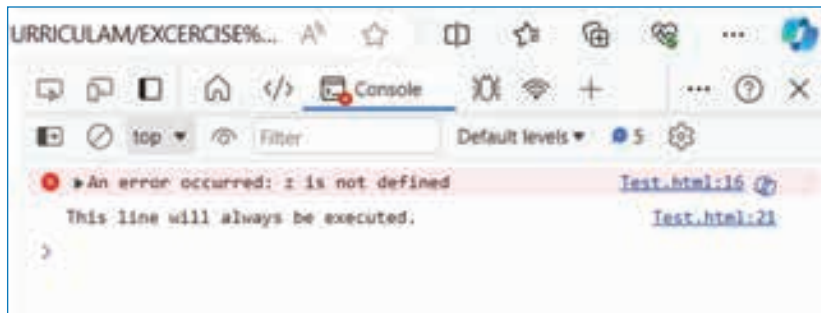
<html >
<head>
<title> Object Literal Notation </title>
</head>
<body>
<script>
try {
// Code that may throw an error
let x = 1;
let y = x + z; // ReferenceError: z is not defined
console.log("This line will not be executed if an error occurs.");
}
catch (error)
{
// Handle the error
console.error("An error occurred:", error.message); // Output: An error occurred: z is not defined
}
finally
{
// Code that runs regardless of whether an error occurred or not
console.log("This line will always be executed.");
}
</script>
</body>
</html>

```

3 Save the program as a .html file

4 Open the html file with a web browser

5 Go to the browsers console tab and verify the output.



- 1 Create a JavaScript function that takes two parameters, numerator and denominator, and attempts to divide numerator by denominator inside a try block. Use a catch block to handle any division by zero errors and log an appropriate error message.
- 2 Create a JavaScript program that attempts to execute a non-existent function inside a try block. Use a catch block to handle the error and log a message indicating that the function does not exist.
- 3 Create a JavaScript function named calculateFactorial that takes a parameter num and calculates the factorial of the number. Use a try-catch block to handle the scenario where num is a negative integer. If num is negative, throw an error with an appropriate message.
- 4 Write a JavaScript program that prompts the user to enter their age. Use a try-catch block to validate that the entered age is a positive integer. If the age is not valid, throw an error with a custom message.
- 5 Write a JavaScript program that attempts to access a property of an undefined object inside a try block. Use a catch block to handle the error and log a custom error message to the console. Include a finally block to show some message, regardless of whether an error occurred or not.

EXERCISE 46 : Implementing an AJAX application

Objectives

At the end of this exercise you shall be able to

- handle AJAX request
- update HTML page content dynamically

Requirements

Tools/Materials

- Desktop / Laptop with latest configuration
- Text editor
- Web browser
- Apache web server

Procedure

TASK 1: Implementation an AJAX application

I Create a JavaScript file to handle the AJAX request and update the content dynamically.

1 Open the text editor

2 Write the following codes

```
// Get references to the button and content container
const loadButton = document.getElementById('loadButton');
const contentContainer = document.getElementById('contentContainer');
// Function to load content asynchronously
function loadContent() {
  // Create a new XMLHttpRequest object
  const xhr = new XMLHttpRequest();
  // Define the callback function to handle the response
  xhr.onreadystatechange = function() {
    // Check if the request is complete and successful
    if (xhr.readyState === XMLHttpRequest.DONE && xhr.status === 200) {
      // Update the content container with the response text
      contentContainer.innerHTML = xhr.responseText;
    }
  };
  // Open a GET request to the desired URL
  xhr.open('GET', 'example.txt', true);
  // Send the request
  xhr.send();
}
// Add an event listener to the button to trigger the AJAX request
loadButton.addEventListener('click', loadContent);
```

3 Save the program in C:\Apache24\htdocs in a folder as script.js

II Create an HTML file with an empty container for displaying content and a button to trigger the AJAX request.

1 Open the text editor

2 Write the following codes

```
<html >
<head>
<title>AJAX Example</title>
</head>
<body>
<button id="loadButton">Load Content</button>
<div id="contentContainer"></div>
<script src="script.js"></script>
</body>
</html>
```

3 Save the program in C:\Apache24\htdocs in a folder as ajaxtest.html

III Test the application

1 Create a text file named example.txt in the same directory as your HTML file.

(This file will contain the content to load dynamically.)

2 Run the Apache services from windows services

3 Open your HTML file in a web browser and click the "Load Content" button.

4 verify the output



1 Create a JavaScript program that sends form data to a server using AJAX when the form is submitted. Display a success message to the user upon successful submission.

2 Write a JavaScript program that uses AJAX to fetch data from a JSON file and display it on a webpage.

◆ Module 4 : PHP (Hyper Text Pre Processor) ◆

EXERCISE 47 : Demonstrate on- Paginators, popovers, progress, spinner

Objectives

At the end of this exercise you shall be able to

- use paginators & popovers in HTML
- use progress bars in HTML
- use spinners in HTML.

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- web browser
- Internet connection

Procedure

TASK 1: Using Paginators

1 create a basic pagination

- Open the text editor
- Write the following codes

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h2>Pagination </h2>
<p>a basic pagination</p>
<ul class="pagination">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-link" href="#">1</a></li>
  <li class="page-item"><a class="page-link" href="#">2</a></li>
  <li class="page-item"><a class="page-link" href="#">3</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
</div>
</body>
</html>
```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



2 Pagination - Active State

- Open the text editor
- Write the following codes

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h2>Pagination </h2>
<p> Add class .active to let the user know which page he/she is on </p>
<ul class="pagination">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-link" href="#">1</a></li>
  <li class="page-item active"><a class="page-link" href="#">2</a></li>
  <li class="page-item"><a class="page-link" href="#">3</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
</div>
</body>
</html>
```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



3 Pagination - Disabled State

- Open the text editor
- Write the following codes

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h2>Pagination- Disabled </h2>
<p> Add class . disabled to disable the pagination </p>
<ul class="pagination">
  <li class="page-item disabled"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-link" href="#">1</a></li>
  <li class="page-item"><a class="page-link" href="#">2</a></li>
  <li class="page-item"><a class="page-link" href="#">3</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
</div>
</body>
</html>
```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



4 Pagination Sizing

- Open the text editor
- Write the following codes

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h2>Pagination-Sizing </h2>
<p> Add class '.pagination-lg' for larger blocks </p>
<ul class="pagination pagination-lg">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-link" href="#">1</a></li>
  <li class="page-item"><a class="page-link" href="#">2</a></li>
  <li class="page-item"><a class="page-link" href="#">3</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
<p> Add class '.pagination-sm' for smaller blocks </p>
<ul class="pagination pagination-sm">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-link" href="#">1</a></li>
  <li class="page-item"><a class="page-link" href="#">2</a></li>
  <li class="page-item"><a class="page-link" href="#">3</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
</div>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



5 Pagination Alignment

- Open the text editor
- Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">

```

```

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h2>Pagination Alignment </h2>
<p> Default (left-aligned)</p>
<!-- Default (left-aligned) -->
<ul class="pagination" style="margin:20px 0">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-link" href="#">1</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
<p> Center-aligned </p>
<!-- Center-aligned -->
<ul class="pagination justify-content-center" style="margin:20px 0">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-link" href="#">1</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
<p> Right-aligned</p>
<!-- Right-aligned -->
<ul class="pagination justify-content-end" style="margin:20px 0">
  <li class="page-item"><a class="page-link" href="#">Previous</a></li>
  <li class="page-item"><a class="page-link" href="#">1</a></li>
  <li class="page-item"><a class="page-link" href="#">Next</a></li>
</ul>
</div>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



TASK 2: Using Popovers

1 create a Popovers

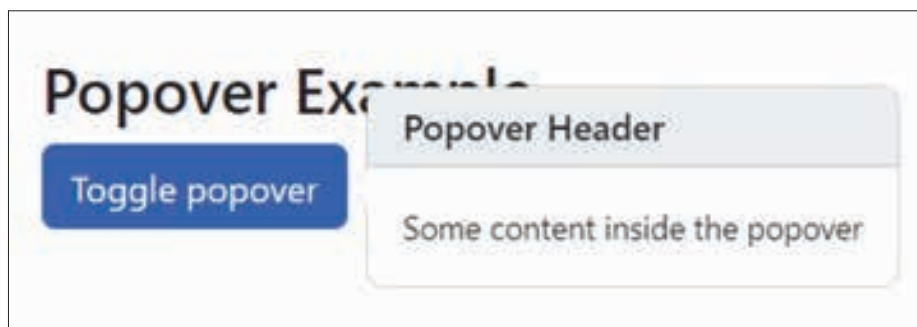
- Open the text editor
- Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h3>Popover Example</h3>
<button type="button" class="btn btn-primary" data-bs-toggle="popover" title="Popover Header" data-bs-content="Some content inside the popover">
Toggle popover
</button>
</div>
<script>
var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'))
var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {
return new bootstrap.Popover(popoverTriggerEl)
})
</script>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



2 Positioning Popovers

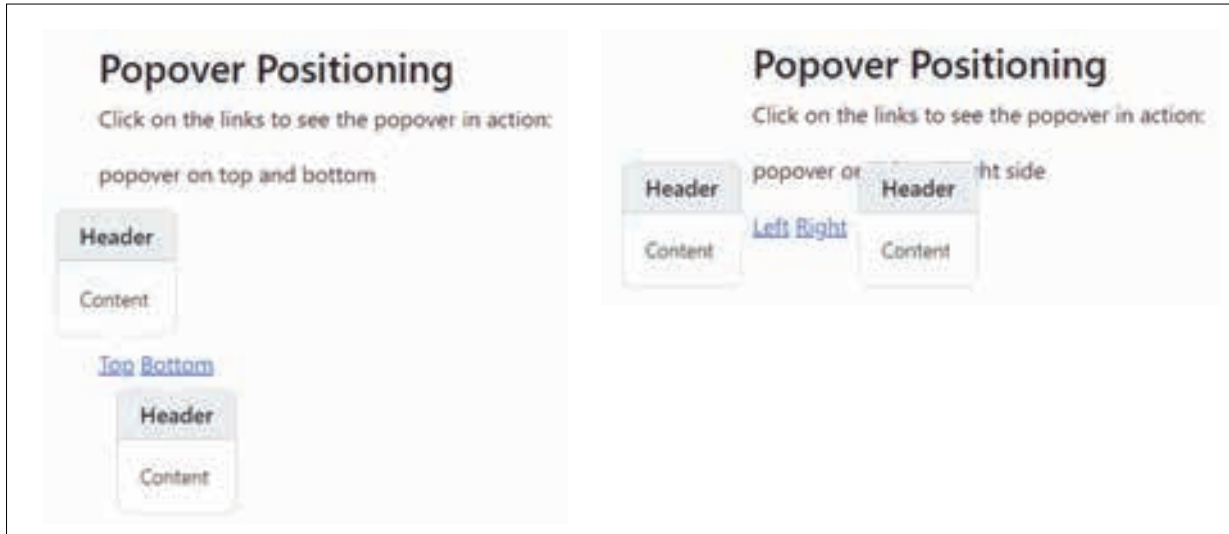
- Open the text editor
- Write the following codes

```

<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js">
</script>
</head>
<body>
<div class="container mt-5">
<h3>Popover Positioning</h3>
<p>Click on the links to see the popover in action:</p>
<p>popover on top and bottom</p>
<br/><br/><br/><br/>
<a href="#" title="Header" data-bs-toggle="popover" data-bs-placement="top"
data-bs-content="Content">Top</a>
<a href="#" title="Header" data-bs-toggle="popover" data-bs-placement="bottom"
data-bs-content="Content">Bottom</a>
</div>
<br/><br/><br/>
<div class="container mt-5">
<h3>Popover Positioning</h3>
<p>Click on the links to see the popover in action:</p>
<p>popover on left and right side </p>
<a href="#" title="Header" data-bs-toggle="popover" data-bs-placement="left"
data-bs-content="Content">Left</a>
<a href="#" title="Header" data-bs-toggle="popover" data-bs-placement="right"
data-bs-content="Content">Right</a>
</div>
<script>
var popoverTriggerList = [].slice.call(document.querySelectorAll("[data-bs-toggle="popover"]"))
var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {
return new bootstrap.Popover(popoverTriggerEl)
})
</script>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



3 Hoverable Popover

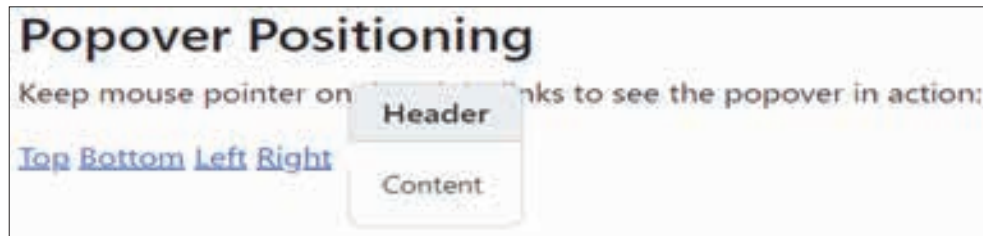
- Open the text editor
- Write the following codes

```

<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-5">
<h3>Popover Positioning</h3>
<p>Keep mouse pointer on the Right links to see the popover in action:</p>
<a href="#" title="Header" data-bs-toggle="popover" data-bs-placement="top" data-bs-content="Content">
Top</a>
<a href="#" title="Header" data-bs-toggle="popover" data-bs-placement="bottom"
data-bs-content="Content">Bottom</a>
<a href="#" title="Header" data-bs-toggle="popover" data-bs-placement="left" data-bs-content="Content">
Left</a>
<a href="#" title="Header" data-bs-toggle="popover" data-bs-placement="right" data-bs-content="Content"
data-bs-trigger="hover">Right</a>
</div>
<script>
var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'))
var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {
return new bootstrap.Popover(popoverTriggerEl)
})
</script>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



TASK 3: Using Progress Bars

1 create a progress bar

- Open the text editor
- Write the following codes

```
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>

<div class="container mt-3">
<h2>Progress Bar With Label</h2>
<div class="progress" style="height:20px">
<div class="progress-bar" style="width:50%">50%</div>
</div>
</div>

</body>
</html>
```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.

Progress Bar With Label



2 Coloured Progress Bars

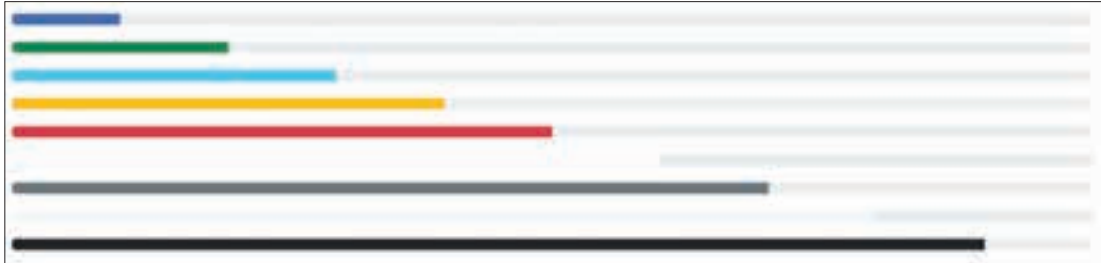
- Open the text editor
- Write the following codes

```

<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<!-- Blue -->
<div class="progress">
<div class="progress-bar" style="width:10%"></div>
</div><br>
<!-- Green -->
<div class="progress">
<div class="progress-bar bg-success" style="width:20%"></div>
</div><br>
<!-- Turquoise -->
<div class="progress">
<div class="progress-bar bg-info" style="width:30%"></div>
</div><br>
<!-- Orange -->
<div class="progress">
<div class="progress-bar bg-warning" style="width:40%"></div>
</div><br>
<!-- Red -->
<div class="progress">
<div class="progress-bar bg-danger" style="width:50%"></div>
</div><br>
<!-- White -->
<div class="progress border">
<div class="progress-bar bg-white" style="width:60%"></div>
</div><br>
<!-- Grey -->
<div class="progress">
<div class="progress-bar bg-secondary" style="width:70%"></div>
</div><br>
<!-- Light Grey -->
<div class="progress border">
<div class="progress-bar bg-light" style="width:80%"></div>
</div><br>
<!-- Dark Grey -->
<div class="progress">
<div class="progress-bar bg-dark" style="width:90%"></div>
</div><br>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.

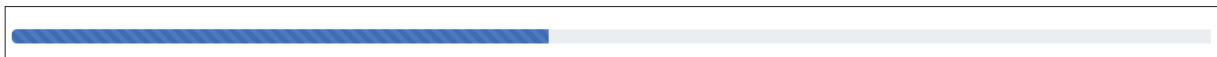


3 Animated Progress Bar

- Open the text editor
- Write the following codes

```
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<br/>
<div class="progress">
<div class="progress-bar progress-bar-striped" style="width:40%"></div>
</div>
</body>
</html>
```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



4 Multiple Progress Bars

- Open the text editor
- Write the following codes

```
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<br/>
<div class="progress">
<div class="progress-bar bg-success" style="width:40%">
Free Space
</div>
<div class="progress-bar bg-warning" style="width:10%">
Warning
</div>
<div class="progress-bar bg-danger" style="width:20%">
Danger
</div>
</div>
</body>
</html>
```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



TASK 4: Using Spinners

1 Create a spinner/loader

- Open the text editor
- Write the following codes

```
<!DOCTYPE html>
<html>
<head>
<title>Bootstrap Example</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
```

```

<h2>Colored Spinners</h2>
<div class="spinner-border"></div>
</div>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



2 Colored Spinners

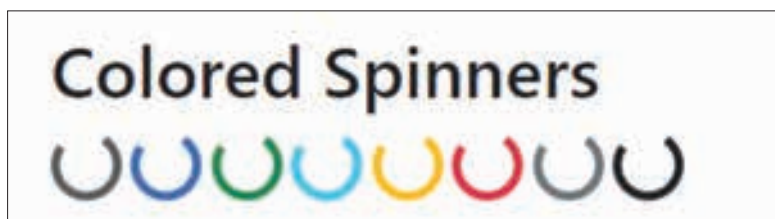
- Open the text editor
- Write the following codes

```

<!DOCTYPE html>
<html>
<head>
<title>Bootstrap Example</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h2>Colored Spinners</h2>
<div class="spinner-border text-muted"></div>
<div class="spinner-border text-primary"></div>
<div class="spinner-border text-success"></div>
<div class="spinner-border text-info"></div>
<div class="spinner-border text-warning"></div>
<div class="spinner-border text-danger"></div>
<div class="spinner-border text-secondary"></div>
<div class="spinner-border text-dark"></div>
<div class="spinner-border text-light"></div>
</div>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.

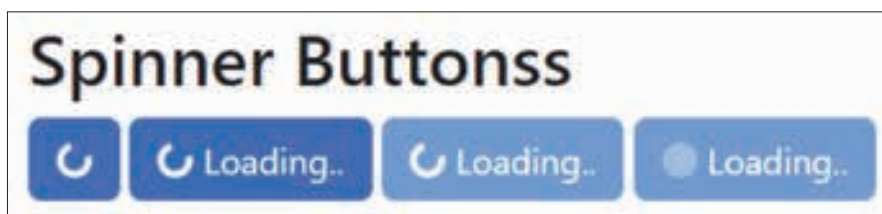


3 Spinner Buttons

- Open the text editor
- Write the following codes

```
<!DOCTYPE html>
<html>
<head>
<title>Bootstrap Example</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h2>Spinner Buttonss</h2>
<button class="btn btn-primary">
<span class="spinner-border spinner-border-sm"></span>
</button>
<button class="btn btn-primary">
<span class="spinner-border spinner-border-sm"></span>
Loading..
</button>
<button class="btn btn-primary" disabled>
<span class="spinner-border spinner-border-sm"></span>
Loading..
</button>
<button class="btn btn-primary" disabled>
<span class="spinner-grow spinner-grow-sm"></span>
Loading..
</button>
</div>
</body>
</html>
```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



- implement a basic paginator using Bootstrap classes and components.
- Customize the appearance of the pagination links (e.g., font size, color, hover effects).

— — — — —

EXERCISE 48 : Demonstrate Table, toasts, tooltips

Objectives

At the end of this exercise you shall be able to

- use bootstrap tables in HTML
- use bootstrap toasts in HTML
- use bootstrap tooltips in HTML.

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- web browser
- Internet connection

Procedure

TASK 1: Using Table

1 Create a basic Bootstrap 5 table

- Open the text editor
- Write the following codes

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h2>Table</h2>
<table class="table">
<thead>
<tr>
<th>Firstname</th>
<th>Lastname</th>
<th>Email</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Doe</td>
<td>john@example.com</td>
```

```

</tr>
<tr>
  <td>Mary</td>
  <td>Moe</td>
  <td>mary@example.com</td>
</tr>
</tbody>
</table>
</div>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.

Table		
Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com

2 Table with Striped Rows

- Open the text editor
- Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
  <h2>Table-striped</h2>
  <table class="table table-striped">
    <thead>
      <tr>
        <th>Firstname</th>
        <th>Lastname</th>
        <th>Email</th>
      </tr>
    </thead>

```

```

<tbody>
<tr>
  <td>John</td>
  <td>Doe</td>
  <td>john@example.com</td>
</tr>
<tr>
  <td>Mary</td>
  <td>Moe</td>
  <td>mary@example.com</td>
</tr>
<tr>
  <td>July</td>
  <td>Dooley</td>
  <td>july@example.com</td>
</tr>
</tbody>
</table>
</div>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.

Table-striped		
Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

3 Bordered Table

- Open the text editor
- Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>

```

```

</head>
<body>
<div class="container mt-3">
  <h2>Table-striped</h2>
<table class="table table-bordered">
  <thead>
    <tr>
      <th>Firstname</th>
      <th>Lastname</th>
      <th>Email</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>John</td>
      <td>Doe</td>
      <td>john@example.com</td>
    </tr>
    <tr>
      <td>Mary</td>
      <td>Moe</td>
      <td>mary@example.com</td>
    </tr>
    <tr>
      <td>July</td>
      <td>Dooley</td>
      <td>july@example.com</td>
    </tr>
  </tbody>
</table>
</div>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

4 Table with Hover Rows

- Open the text editor
- Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
  <h2> Hover Rows </h2>
  <table class="table table-bordered">
    <thead>
      <tr>
        <th>Firstname</th>
        <th>Lastname</th>
        <th>Email</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>John</td>
        <td>Doe</td>
        <td>john@example.com</td>
      </tr>
      <tr>
        <td>Mary</td>
        <td>Moe</td>
        <td>mary@example.com</td>
      </tr>
      <tr>
        <td>July</td>
        <td>Dooley</td>
        <td>july@example.com</td>
      </tr>
    </tbody>
  </table>
</div>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.

Hover Rows

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com

5 Table with Contextual Classes

- Open the text editor
- Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h2>Contextual Classes</h2>
<table class="table">
<thead>
<tr>
<th>Firstname</th>
<th>Lastname</th>
<th>Email</th>
</tr>
</thead>
<tbody>
<tr class="table-primary">
<td>John</td>
<td>Doe</td>
<td>john@example.com</td>
</tr>
<tr class="table-success">
<td>Mary</td>
<td>Moe</td>
<td>mary@example.com</td>

```

```

</tr>
<tr class="table-danger">
  <td>July</td>
  <td>Dooley</td>
  <td>july@example.com</td>
</tr>
<tr class="table-info">
  <td>Mary</td>
  <td>Moe</td>
  <td>mary@example.com</td>
</tr>
<tr class="table-warning">
  <td>July</td>
  <td>Dooley</td>
  <td>july@example.com</td>
</tr>
<tr class="table-active">
  <td>Mary</td>
  <td>Moe</td>
  <td>mary@example.com</td>
</tr>
<tr class="table-secondary">
  <td>July</td>
  <td>Dooley</td>
  <td>july@example.com</td>
</tr>
<tr class="table-light">
  <td>July</td>
  <td>Dooley</td>
  <td>july@example.com</td>
</tr>
<tr class="table-dark">
  <td>Mary</td>
  <td>Moe</td>
  <td>mary@example.com</td>
</tr>
</tbody>
</table>
</div>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.

Contextual Classes

Firstname	Lastname	Email
John	Doe	john@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com
Mary	Moe	mary@example.com
July	Dooley	july@example.com
July	Dooley	july@example.com
Mary	Moe	mary@example.com

TASK 2: Using Toasts

1 Create a Bootstrap 5 toast

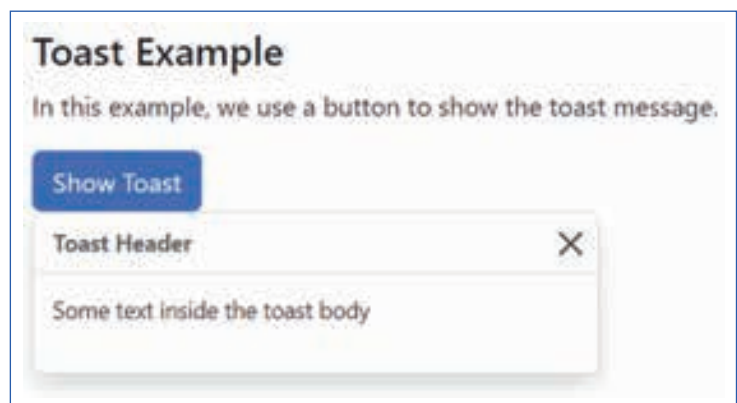
- Open the text editor
- Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h3>Toast Example</h3>
<p>In this example, we use a button to show the toast message.</p>
<button type="button" class="btn btn-primary" id="toastbtn">Show Toast</button>
<div class="toast">
<div class="toast-header">
<strong class="me-auto">Toast Header</strong>
<button type="button" class="btn-close" data-bs-dismiss="toast"></button>
</div>
<div class="toast-body">
<p>Some text inside the toast body</p>
</div>
</div>
</div>
</div>
<script>
document.getElementById("toastbtn").onclick = function() {
var toastEList = [].slice.call(document.querySelectorAll('.toast'))
var toastList = toastEList.map(function(toastEl) {
return new bootstrap.Toast(toastEl)
})
toastList.forEach(toast => toast.show())
}
</script>
</body>
</html>

```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



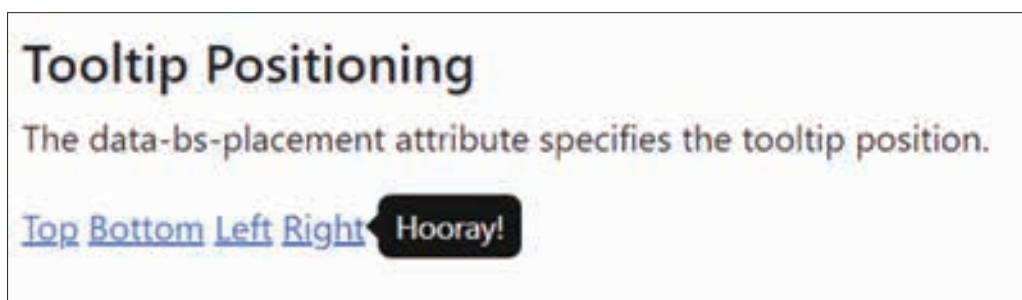
TASK 3: Using Tooltips

1 Create a Bootstrap 5 Tooltips

- Open the text editor
- Write the following codes

```
<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<div class="container mt-3">
<h3>Tooltip Positioning</h3>
<p>The data-bs-placement attribute specifies the tooltip position.</p>
<a href="#" data-bs-toggle="tooltip" data-bs-placement="top" title="Hooray!">Top</a>
<a href="#" data-bs-toggle="tooltip" data-bs-placement="bottom" title="Hooray!">Bottom</a>
<a href="#" data-bs-toggle="tooltip" data-bs-placement="left" title="Hooray!">Left</a>
<a href="#" data-bs-toggle="tooltip" data-bs-placement="right" title="Hooray!">Right</a>
</div>
<script>
  var tooltipTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="tooltip"]'))
  var tooltipList = tooltipTriggerList.map(function (tooltipTriggerEl) {
    return new bootstrap.Tooltip(tooltipTriggerEl)
  })
</script>
</body>
</html>
```

- Save the program as a .html file
- Open the html file with a web browser
- Verify the output.



EXERCISE 49 : Demonstrate on Bootstrap Styling essentials like Breakpoints for components, layouts and grid systems. Practice on typography, floats, flex, alignment, borders, position of elements, shadow and visibility

Objectives

At the end of this exercise you shall be able to

- use bootstrap break point components, layouts and grid systems in HTML
- use bootstrap typography, floats, flex, and alignment in HTML
- use bootstrap borders, position of elements and shadow & visibility in HTML.

Requirements

Tools/Materials

- Desktop/Laptop with latest configuration
- web browser
- Internet connection

Procedure

TASK 1: Applying Breakpoints to Components

1 Open the text editor

2 Write the following codes

```
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Bootstrap Breakpoints</title>
<!-- Bootstrap CSS -->
<link href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">
<style>
/* Additional custom styles */
.custom-component {
background-color: lightblue;
padding: 20px;
margin-bottom: 20px;
}
</style>
<!-- Bootstrap JS (optional) -->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</head>
<body>
```

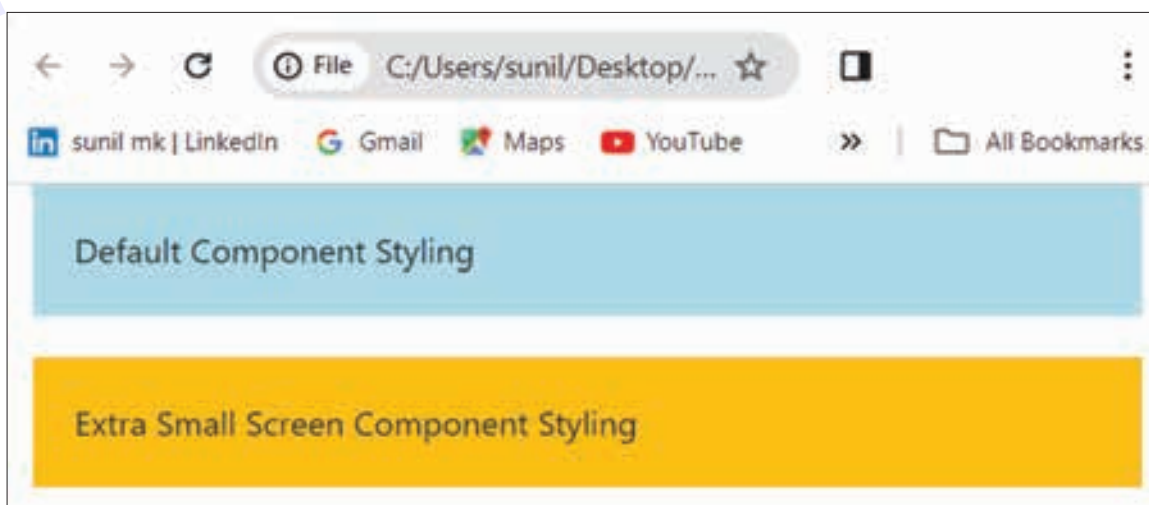
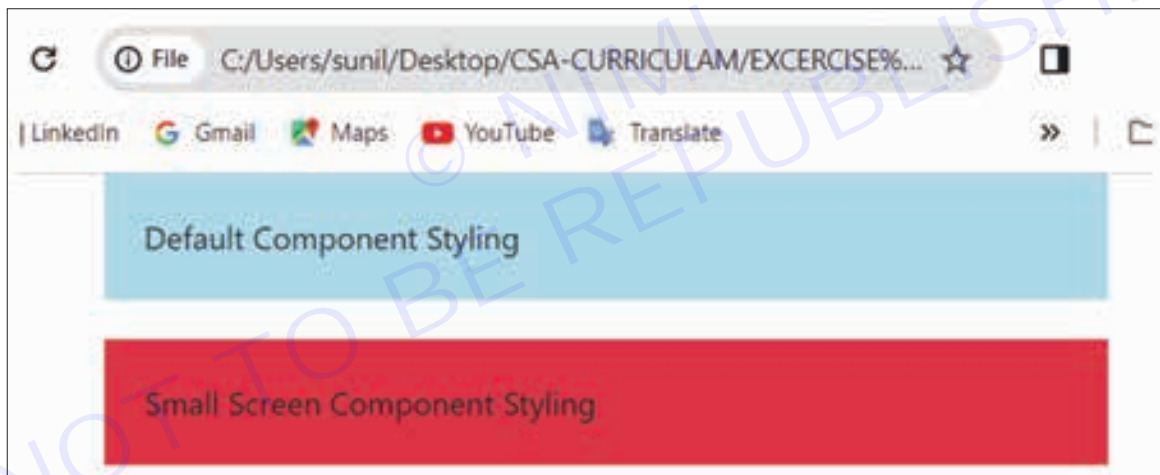
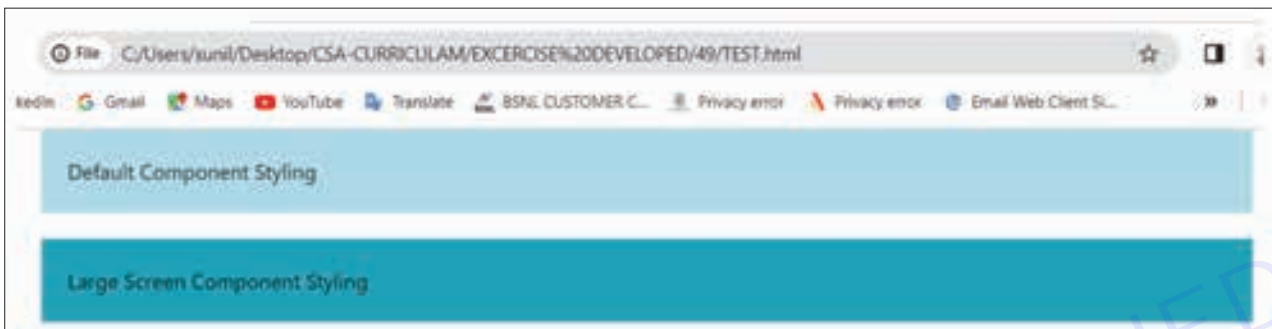
```

<div class="container">
<!-- Component with default styling -->
<div class="custom-component">
Default Component Styling
</div>
<!-- Component with styling for extra small screens -->
<div class="custom-component bg-warning d-block d-sm-none">
Extra Small Screen Component Styling
</div>
<!-- Component with styling for small screens -->
<div class="custom-component bg-danger d-none d-sm-block d-md-none">
Small Screen Component Styling
</div>
<!-- Component with styling for medium screens -->
<div class="custom-component bg-success d-none d-md-block d-lg-none">
Medium Screen Component Styling
</div>
<!-- Component with styling for large screens -->
<div class="custom-component bg-info d-none d-lg-block d-xl-none">
Large Screen Component Styling
</div>
<!-- Component with styling for extra large screens -->
<div class="custom-component bg-secondary d-none d-xl-block">
Extra Large Screen Component Styling
</div>
</div>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Change the screen size and Verify the output.





TASK 2: Applying layouts and grid systems

- 1 Open the text editor
- 2 Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Bootstrap v3 Layouts and Grid Systems Example</title>
<!-- Bootstrap CSS -->
<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css rel="stylesheet">
</head>
<body>
<div class="container">
<div class="row">
<div class="col-md-4">
<div class="well">Column 1</div>
</div>
<div class="col-md-4">
<div class="well">Column 2</div>
</div>
<div class="col-md-4">
<div class="well">Column 3</div>
</div>
</div>
<div class="row">
<div class="col-md-6">
<div class="well">Column 1</div>
</div>
<div class="col-md-6">
<div class="well">Column 2</div>
</div>
</div>
<div class="row">
<div class="col-md-8">
<div class="well">Column 1</div>
</div>
<div class="col-md-4">
<div class="well">Column 2</div>
</div>
</div>
<div class="row">
<div class="col-md-3">
<div class="well">Column 1</div>
</div>
<div class="col-md-6">
<div class="well">Column 2</div>
</div>
<div class="col-md-3">

```

```

<div class="well">Column 3</div>
</div>
</div>
<div class="row">
<div class="col-md-9">
<div class="well">Column 1</div>
</div>
<div class="col-md-3">
<div class="well">Column 2</div>
</div>
</div>
</div>
<!-- Bootstrap JS -->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 3: Practice on typography

- 1 Open the text editor
- 2 Write the following codes

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Bootstrap v5 Typography Example</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
<div class="container mt-5">
<h1>Typography Example</h1>
<h2>Headings</h2>
<h1 class="display-1">Display 1</h1>
<h2 class="display-2">Display 2</h2>
<h3 class="display-3">Display 3</h3>
<h4 class="display-4">Display 4</h4>
<h5 class="display-5">Display 5</h5>
<h6 class="display-6">Display 6</h6>
<h2>Text Styles</h2>
<p class="lead">This is a lead paragraph.</p>
<p>This is a regular paragraph.</p>
<p class="text-muted">This is a muted paragraph.</p>
<p class="text-primary">This is a primary-colored paragraph.</p>
<p class="text-secondary">This is a secondary-colored paragraph.</p>
<p class="text-success">This is a success-colored paragraph.</p>
<p class="text-danger">This is a danger-colored paragraph.</p>
<p class="text-warning">This is a warning-colored paragraph.</p>
<p class="text-info">This is an info-colored paragraph.</p>
<p class="text-light bg-dark">This is a light-colored paragraph on a dark background.</p>
<p class="text-dark">This is a dark-colored paragraph.</p>
<p class="text-body">This is a paragraph using the body text color.</p>
<p class="text-reset">This is a paragraph with reset color.</p>
<h2>Text Alignment</h2>
<p class="text-start">Left aligned text.</p>
<p class="text-center">Center aligned text.</p>
<p class="text-end">Right aligned text.</p>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0alpha1/dist/js/bootstrap.bundle.min.js">
</script>
</body>
</html>
```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 4: Using floats

- 1 Open the text editor
- 2 Write the following codes

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Bootstrap v5 Floats Example</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
<style>
.float-start {
float: left;
}
.float-end {
float: right;
}
```

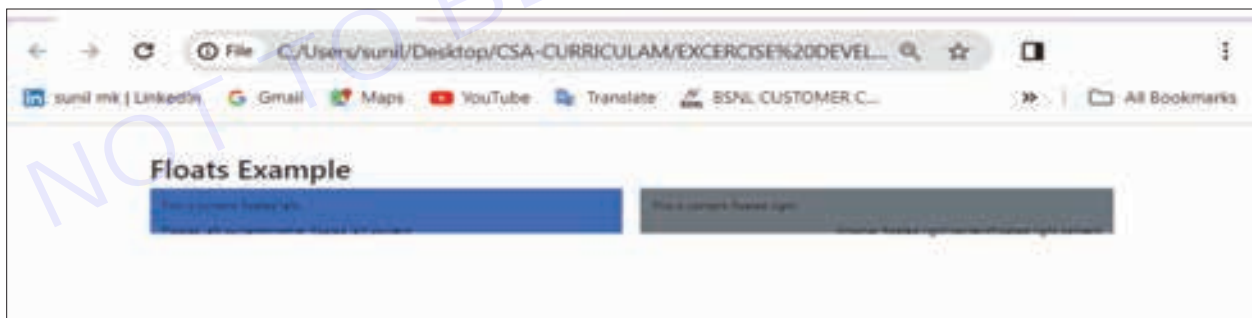


```

</style>
</head>
<body>
<div class="container mt-5">
<h1>Floats Example</h1>
<div class="row">
<div class="col-md-6">
<div class="bg-primary p-3 mb-3">
<p>This is content floated left.</p>
<p class="float-start">Floated left content</p>
<p class="float-start">Another floated left content</p>
</div>
</div>
<div class="col-md-6">
<div class="bg-secondary p-3 mb-3">
<p>This is content floated right.</p>
<p class="float-end">Floated right content</p>
<p class="float-end">Another floated right content</p>
</div>
</div>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 5: Using flexs

- 1 Open the text editor
- 2 Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Bootstrap v5 Flex Example</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">

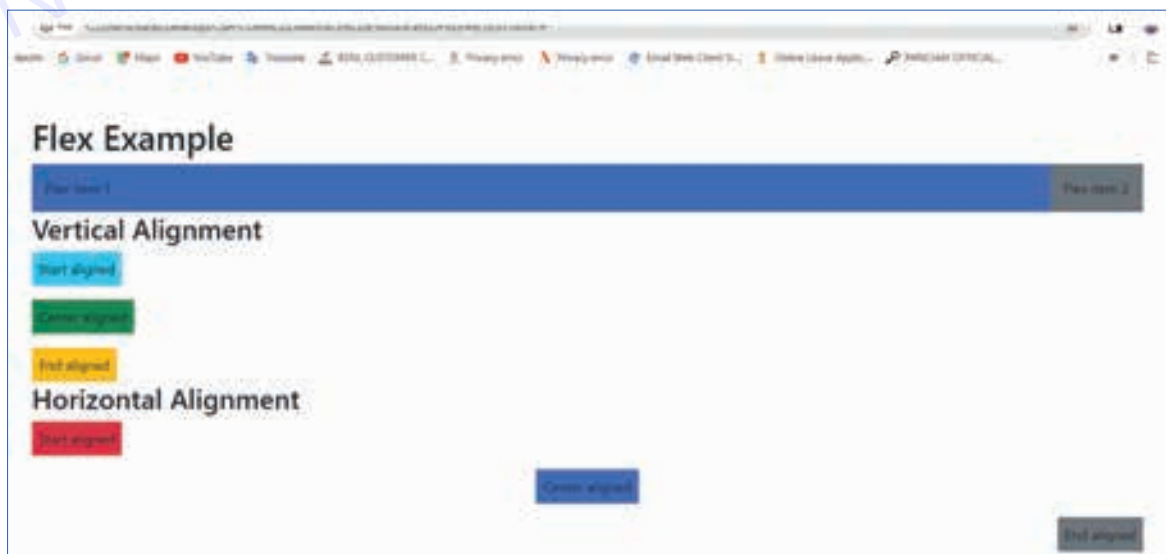
```

```

</head>
<body>
<div class="container mt-5">
<h1>Flex Example</h1>
<div class="d-flex">
<div class="flex-grow-1 bg-primary p-3 mr-3">Flex item 1</div>
<div class="bg-secondary p-3">Flex item 2</div>
</div>
<h2>Vertical Alignment</h2>
<div class="d-flex align-items-start mb-3">
<div class="p-2 bg-info">Start aligned</div>
</div>
<div class="d-flex align-items-center mb-3">
<div class="p-2 bg-success">Center aligned</div>
</div>
<div class="d-flex align-items-end">
<div class="p-2 bg-warning">End aligned</div>
</div>
<h2>Horizontal Alignment</h2>
<div class="d-flex justify-content-start mb-3">
<div class="p-2 bg-danger">Start aligned</div>
</div>
<div class="d-flex justify-content-center mb-3">
<div class="p-2 bg-primary">Center aligned</div>
</div>
<div class="d-flex justify-content-end">
<div class="p-2 bg-secondary">End aligned</div>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 6: Using alignment

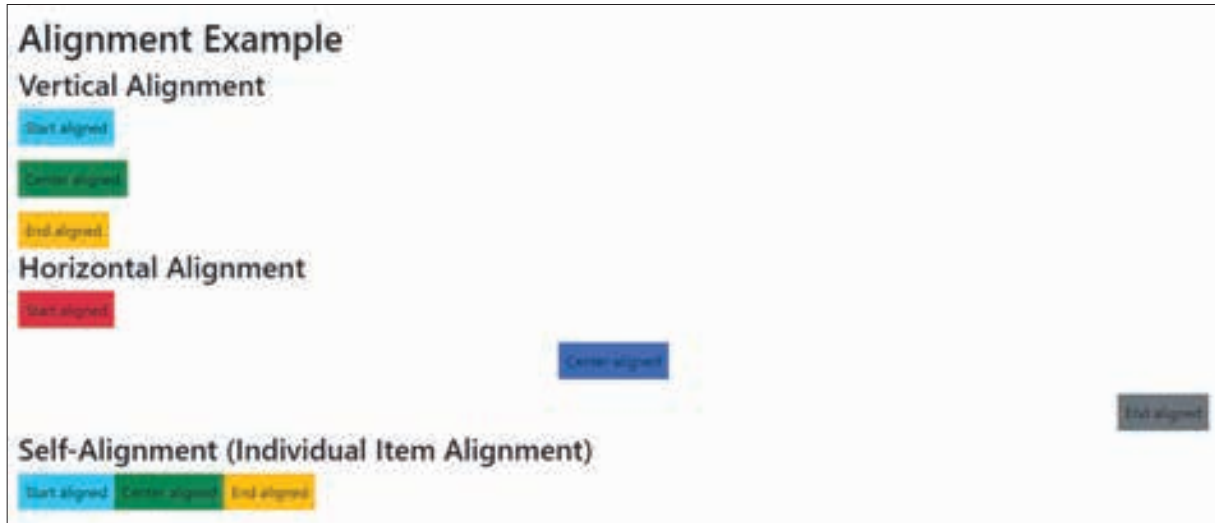
- 1 Open the text editor
- 2 Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Bootstrap v5 Alignment Example</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
<div class="container mt-5">
<h1>Alignment Example</h1>
<h2>Vertical Alignment</h2>
<div class="d-flex align-items-start mb-3">
<div class="p-2 bg-info">Start aligned</div>
</div>
<div class="d-flex align-items-center mb-3">
<div class="p-2 bg-success">Center aligned</div>
</div>
<div class="d-flex align-items-end">
<div class="p-2 bg-warning">End aligned</div>
</div>
<h2>Horizontal Alignment</h2>
<div class="d-flex justify-content-start mb-3">
<div class="p-2 bg-danger">Start aligned</div>
</div>
<div class="d-flex justify-content-center mb-3">
<div class="p-2 bg-primary">Center aligned</div>
</div>
<div class="d-flex justify-content-end">
<div class="p-2 bg-secondary">End aligned</div>
</div>
<h2>Self-Alignment (Individual Item Alignment)</h2>
<div class="d-flex align-items-start">
<div class="p-2 bg-info align-self-start">Start aligned</div>
<div class="p-2 bg-success align-self-center">Center aligned</div>
<div class="p-2 bg-warning align-self-end">End aligned</div>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 7: Using borders

- 1 Open the text editor
- 2 Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Bootstrap v5 Borders Example</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
<div class="container mt-5">
<h1>Borders Example</h1>
<div class="border p-3 mb-3">Default border</div>
<div class="border border-primary p-3 mb-3">Primary border</div>
<div class="border border-secondary p-3 mb-3">Secondary border</div>
<div class="border border-success p-3 mb-3">Success border</div>
<div class="border border-danger p-3 mb-3">Danger border</div>
<div class="border border-warning p-3 mb-3">Warning border</div>
<div class="border border-info p-3 mb-3">Info border</div>
<div class="border border-light p-3 mb-3">Light border</div>
<div class="border border-dark p-3 mb-3">Dark border</div>
<div class="border border-white p-3 mb-3">White border</div>
<h2>Rounded Borders</h2>
<div class="rounded border p-3 mb-3">Default rounded border</div>
<div class="rounded border border-primary p-3 mb-3">Primary rounded border</div>
<div class="rounded border border-secondary p-3 mb-3">Secondary rounded border</div>
<div class="rounded border border-success p-3 mb-3">Success rounded border</div>
<div class="rounded border border-danger p-3 mb-3">Danger rounded border</div>
<div class="rounded border border-warning p-3 mb-3">Warning rounded border</div>
<div class="rounded border border-info p-3 mb-3">Info rounded border</div>

```

```

<div class="rounded border border-light p-3 mb-3">Light rounded border</div>
<div class="rounded border border-dark p-3 mb-3">Dark rounded border</div>
<div class="rounded border border-white p-3 mb-3">White rounded border</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 8: Positioning of elements

- 1 Open the text editor
- 2 Write the following codes

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Bootstrap v5 Positioning Example</title>
<!-- Bootstrap CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>

```

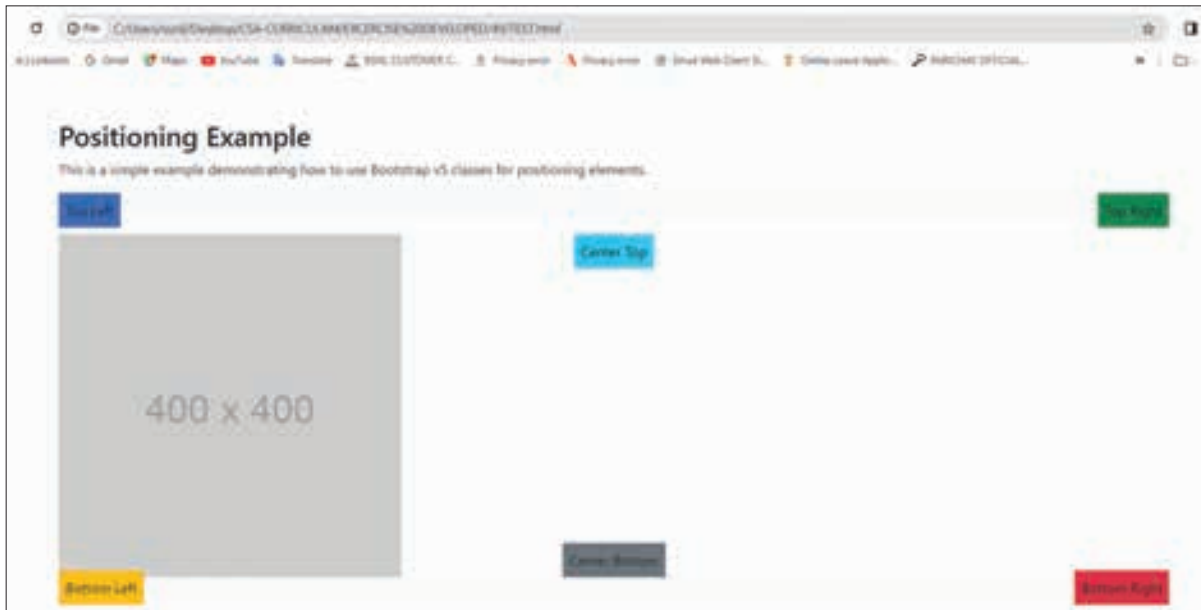
```

<div class="container">
<div class="row mt-5">
<div class="col">
<h2>Positioning Example</h2>
<p>This is a simple example demonstrating how to use Bootstrap v5 classes for positioning elements.</p>
<div class="position-relative bg-light p-3 mb-3">
<div class="position-absolute top-0 start-0 bg-primary p-2">
Top Left
</div>
<div class="position-absolute top-0 end-0 bg-success p-2">
Top Right
</div>
</div>
<div class="position-relative">

<div class="position-absolute top-0 start-50 translate-middle-x bg-info p-2">
Center Top
</div>
<div class="position-absolute bottom-0 start-50 translate-middle-x bg-secondary p-2">
Center Bottom
</div>
</div>
<div class="position-relative bg-light p-3 mb-3">
<div class="position-absolute bottom-0 start-0 bg-warning p-2">
Bottom Left
</div>
<div class="position-absolute bottom-0 end-0 bg-danger p-2">Bottom Right
</div>
</div>
</div>
</div>
</div>
</div>
<!-- Bootstrap Bundle with Popper -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



TASK 9: Shadow and Visibility in bootstrap v5

- 1 Open the text editor
- 2 Write the following codes

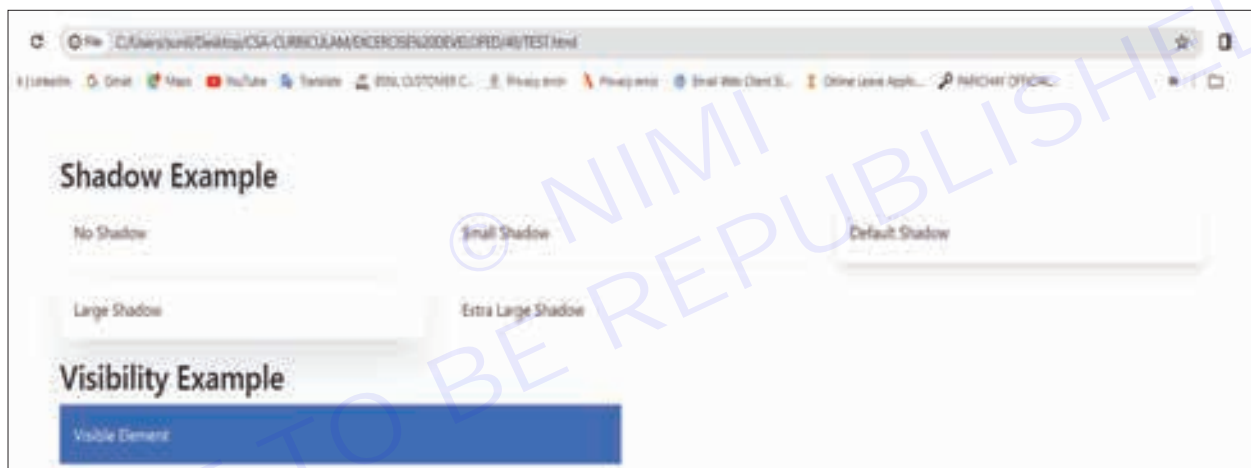
```
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Bootstrap v5 Shadow and Visibility Example</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
<div class="container mt-5">
<h2>Shadow Example</h2>
<div class="row">
<div class="col-md-4">
<div class="p-3 mb-3 bg-white shadow-none">No Shadow</div>
</div>
<div class="col-md-4">
<div class="p-3 mb-3 bg-white shadow-sm">Small Shadow</div>
</div>
<div class="col-md-4">
<div class="p-3 mb-3 bg-white shadow">Default Shadow</div>
</div>
<div class="col-md-4">
<div class="p-3 mb-3 bg-white shadow-lg">Large Shadow</div>
</div>
<div class="col-md-4">
<div class="p-3 mb-3 bg-white shadow-2xl">Extra Large Shadow</div>
</div>
```

```

</div>
<h2>Visibility Example</h2>
<div class="row">
<div class="col-md-6">
<div class="p-3 mb-3 bg-primary text-white visible">Visible Element</div>
</div>
<div class="col-md-6">
<div class="p-3 mb-3 bg-secondary invisible">Invisible Element</div>
</div>
</div>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

- 3 Save the program as a .html file
- 4 Open the html file with a web browser
- 5 Verify the output.



EXERCISE 50 : Perform Installation of Apache Web Server Practice simple PHP programs. Practicing on programming to test events

Objectives

At the end of this exercise you shall be able to

- perform installation of apache webserver
- perform installation of PHP
- configure apache webserver to run PHP as a module.

Requirements

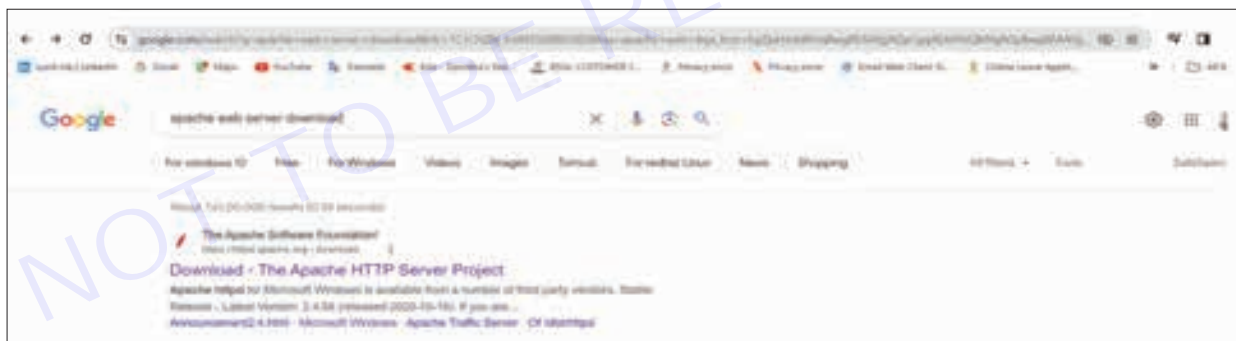
Tools/Materials

- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apssache web server
- PHP
- Text editor
- seb browser

Procedure

TASK 1: Perform Installation of Apache Web Server on a Windows system

- 1 Search 'Apache Web Server Download' in Google and click on the first link



2 Click on the 'number of third party vendors'

The screenshot shows the Apache HTTP Server Project website. The main heading is "APACHE HTTP SERVER PROJECT". Below it, there is a section titled "Downloading the Apache HTTP Server". A yellow arrow points to the text: "Apache httpd for Microsoft Windows is available from a number of third party vendors." Another yellow arrow points to the text: "Only current recommended releases are available on the main distribution site. Historical releases, including the 1.3, 2.0 and 2.2 families of releases, are available from the archive.apache.org site."

At the bottom left, there is a navigation menu with links: About, License, FAQ, and Security Reports. Below this menu, the text reads: "Stable Release - Latest Version:"

3 Select 'Apache Lounge'

The screenshot shows the Apache HTTP Server Project website, specifically the page titled "Using Apache HTTP Server on Microsoft Windows". The page content includes sections for "Operating System Requirements" and "Downloading Apache for Windows". A yellow arrow points to the text: "Popular options for deploying Apache httpd, and, optionally, PHP and MySQL, on Microsoft Windows, include:"

Below this text, there is a list of options:

- [Apache Lounge](#)
- [Bitnami WAMP Stack](#)
- [WampServer](#)
- [XAMPP](#)

A yellow arrow points to the first option, "Apache Lounge".

4 Click 'Apache 2.4.58 Win64' for downloading Apache Web Server software



- 03 July 2023
mod_wasm 0.12.1
- 03 June 2023
mod_qos 11.74
- 31 May 2023
httpd 2.4.57 Update OpenSSL
- 15 May 2023
httpd 2.4.57 Update OpenSSL

Apache 2.4.58 Win64

 [httpd-2.4.58-win64-VS17.zip](#) 19 Oct '23 11.386k
PGP Signature (Public [PGP key](#)), SHA1-SHA512 (checksums)

Apache 2.4.58 Win32

 [httpd-2.4.58-win32-VS17.zip](#) 19 Oct '23 10.239k
PGP Signature (Public [PGP key](#)), SHA1-SHA512 (checksums)

To be sure that a download is intact and has not been tampered with, use PGP, see [PGP Signature](#)

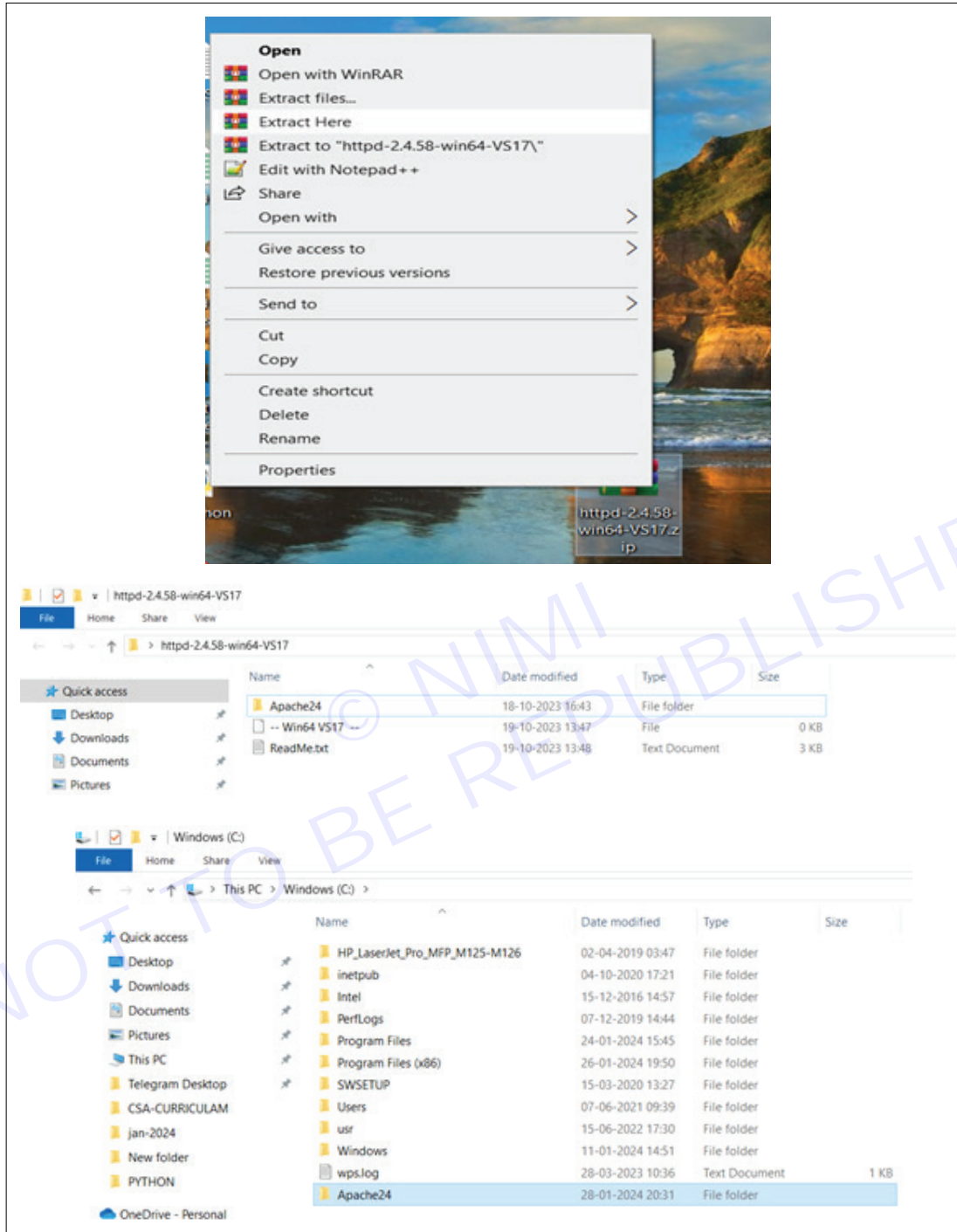
Recent download history

 **httpd-2.4.58-win64-VS17.zip**
6.3/11.1 MB • 44 seconds left

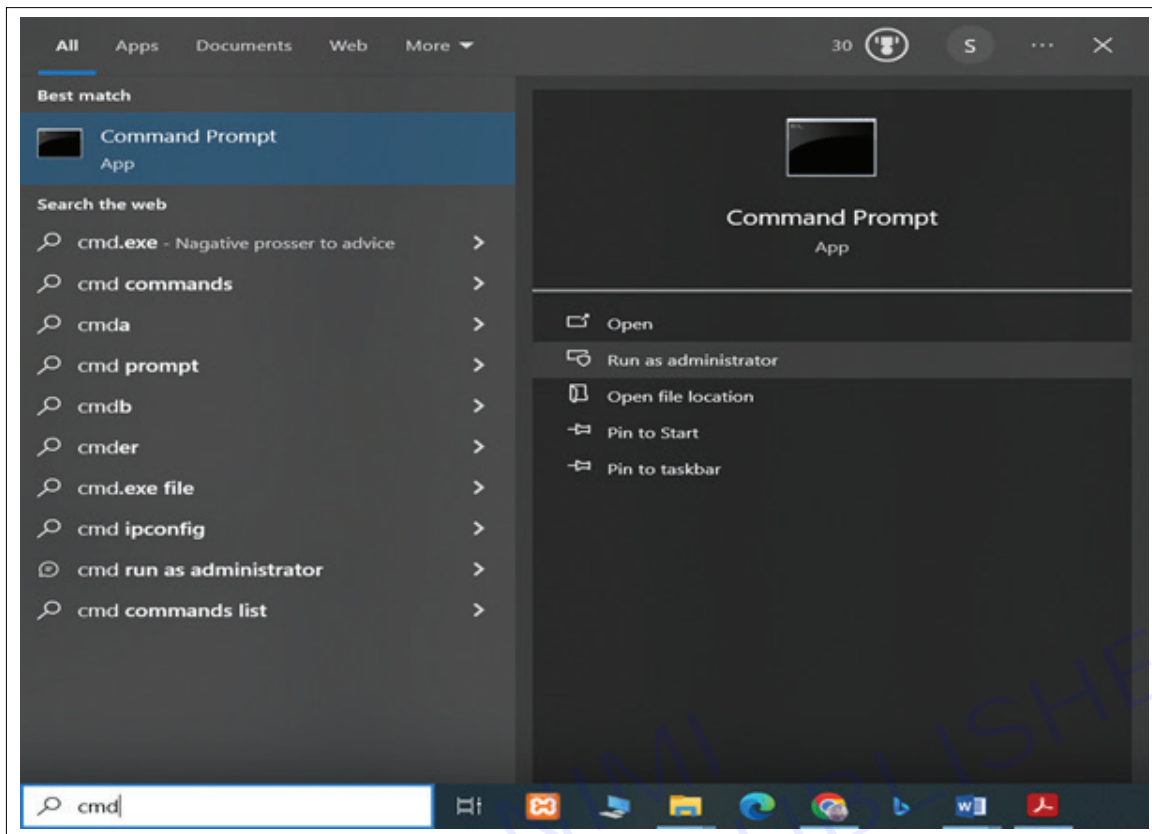
Full download history



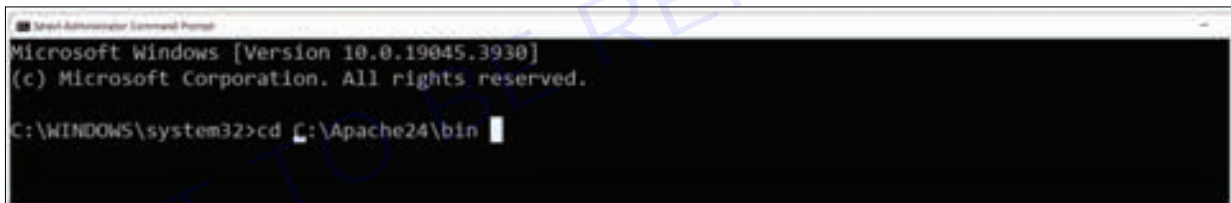
- Right click on the downloaded file and Unzip the Apache24 folder to c:/Apache24 (that is the ServerRoot in the config).



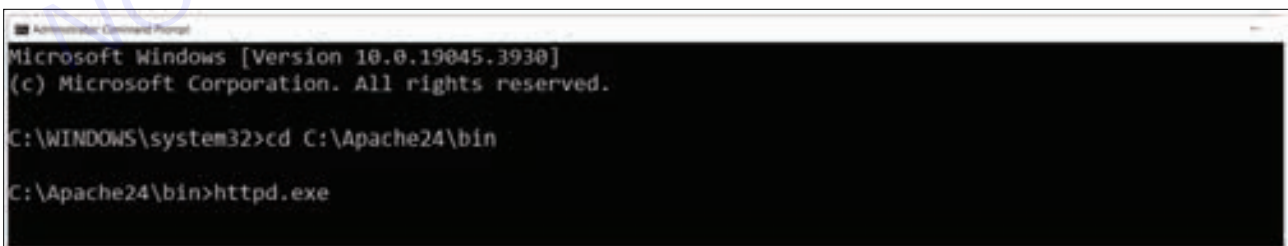
- 6 Open the command prompt as 'Run as Administrator'



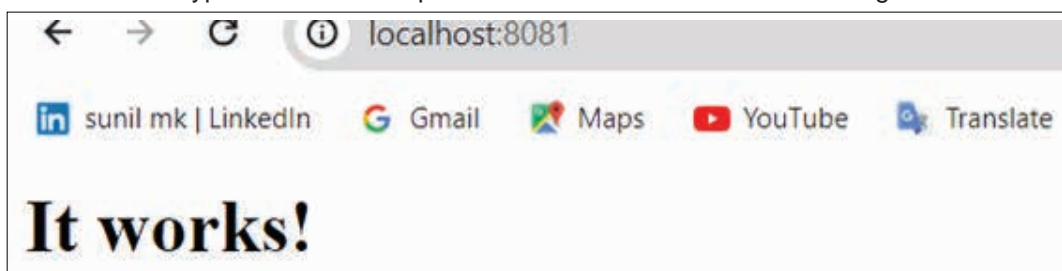
- 7 Change the command prompt to C:\Apache24\bin by using CD command



- 8 Run the httpd.exe file in the command prompt

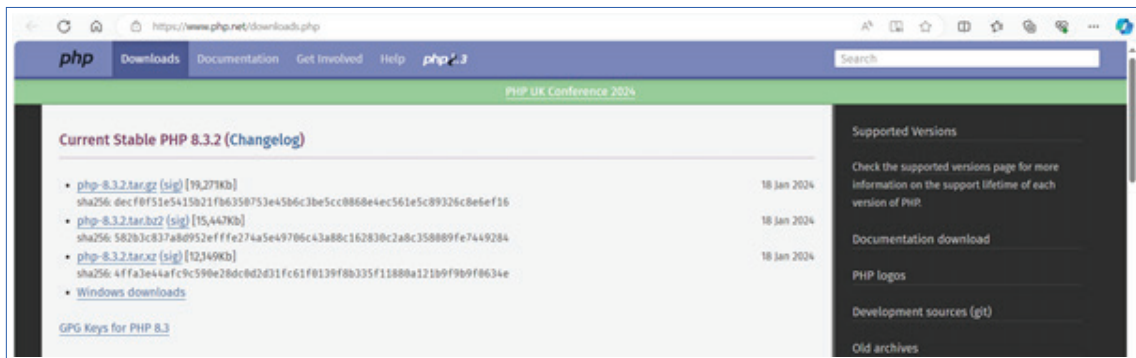


- 9 Open the browser and type the address http://localhost to check the server running

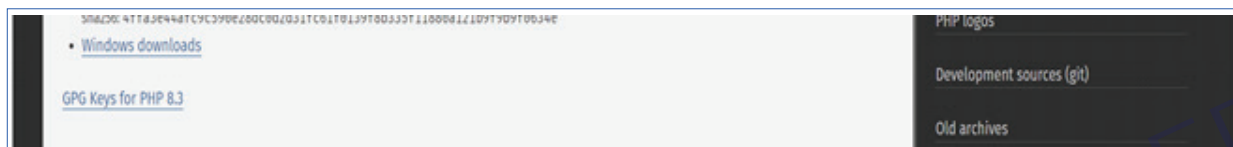


TASK 2: Perform Installation of PHP in windows system

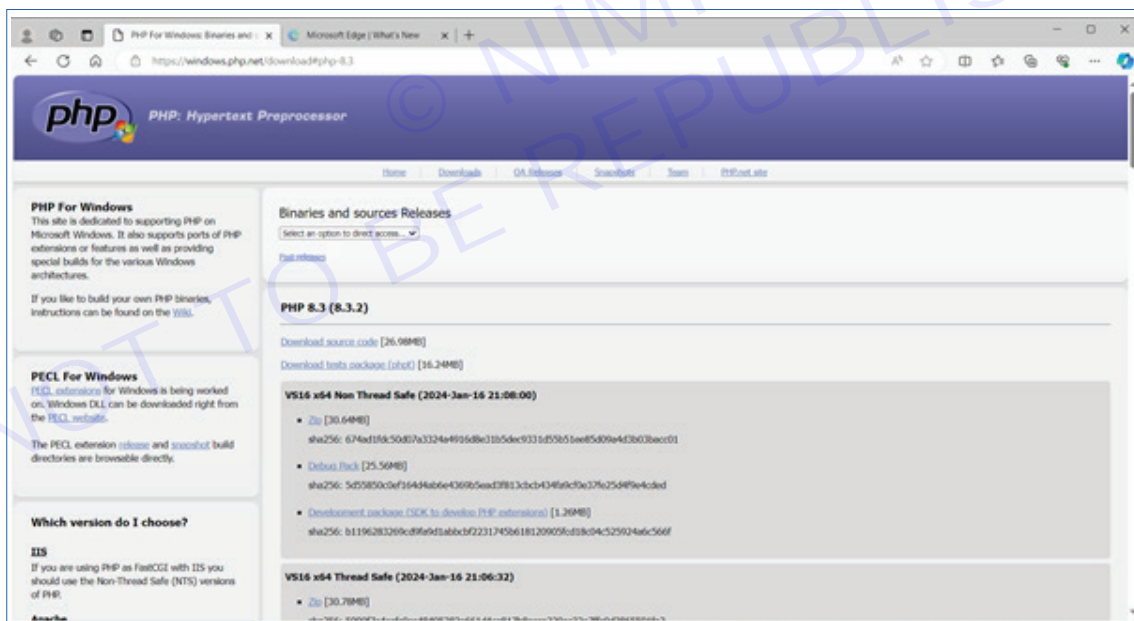
1 Open the link <http://www.php.net/downloads.php> in any browser

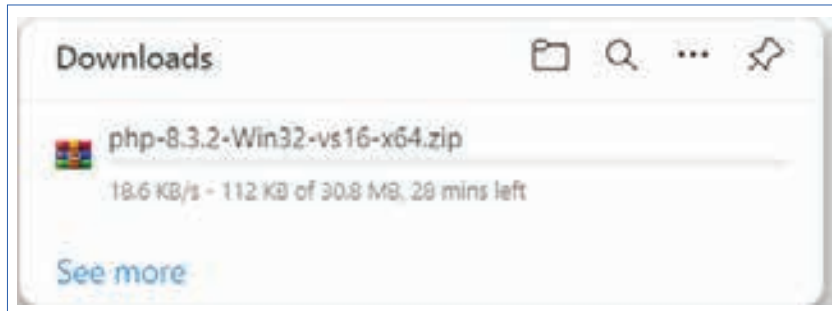


2 Click on 'Windows downloads'

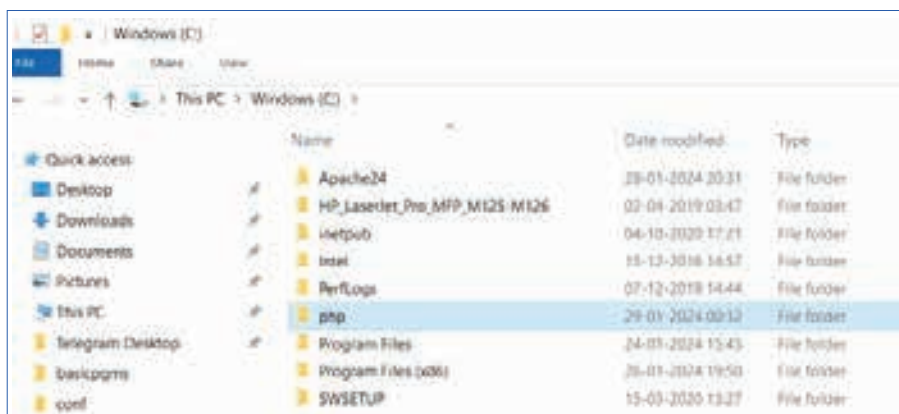
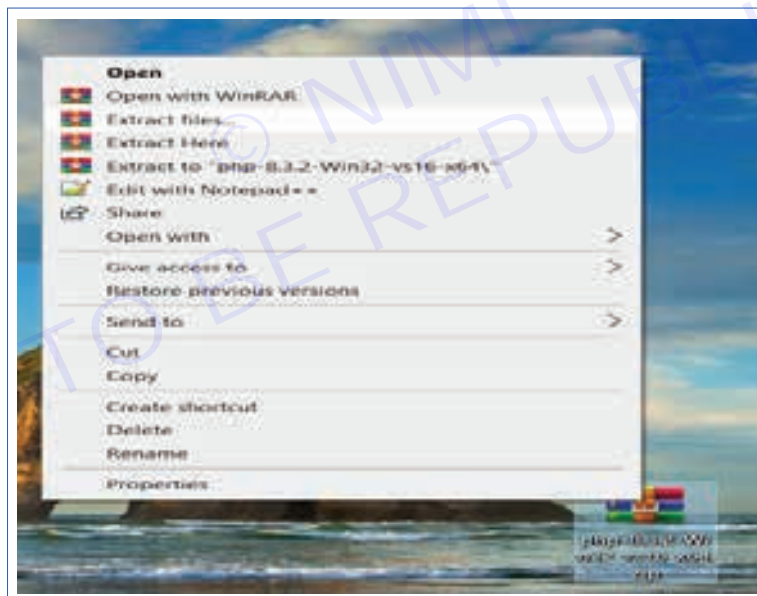


3 Click Zip link in 'VS16 x64 Non Thread Safe' under PHP 8.3 for downloading PHP Software

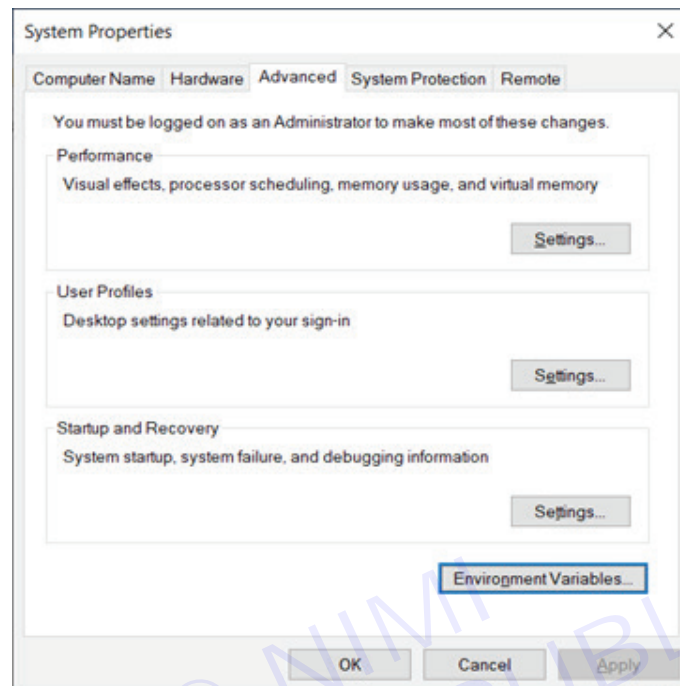




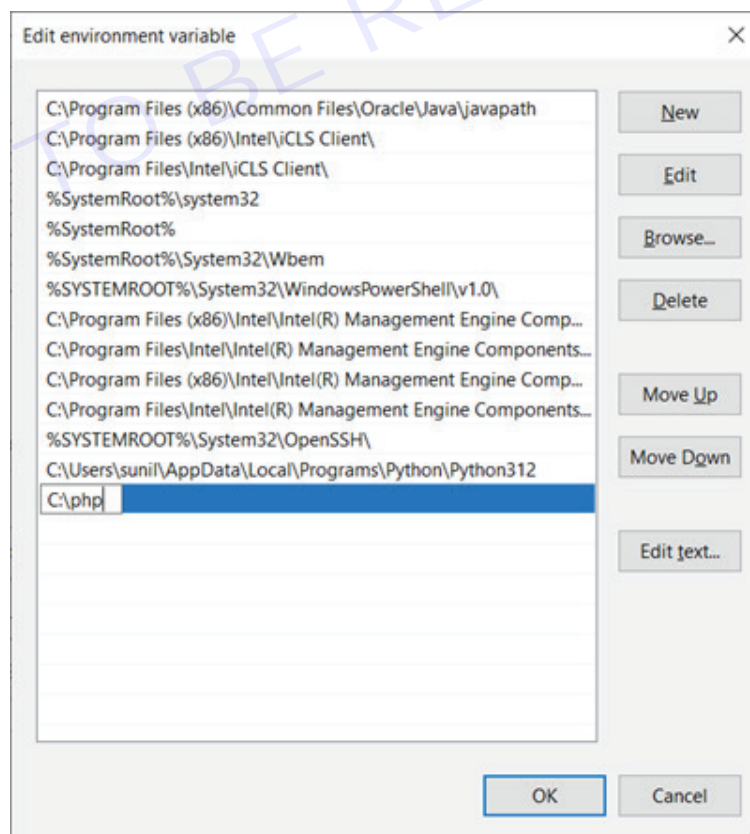
4 Right click on the downloaded file and Unzip the folder to c:/php



- 5 Open the environment variables and add php folder
 - Right-click on “This PC” or “Computer” and select “Properties.”
 - Click on “Advanced system settings”
 - Click the “Environment Variables” button.



- 6 Add php home folder by clicking 'New' button



- 7 Open the command prompt and check the PHP installed version by the command 'php -v'

```

Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

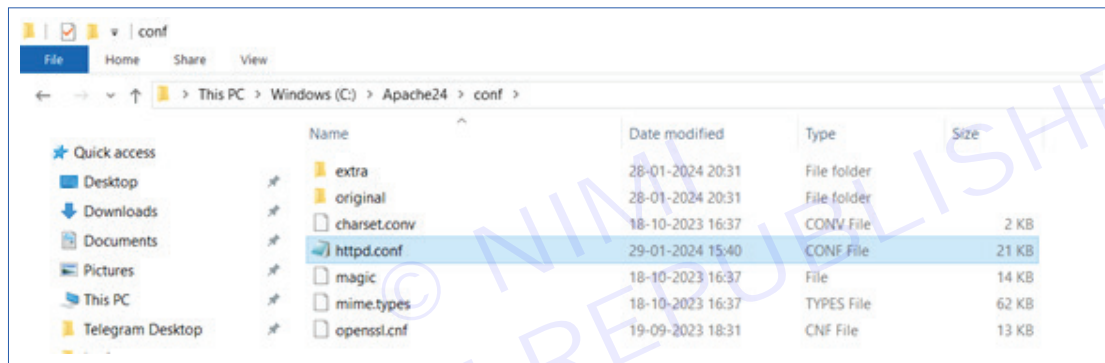
C:\Users\sunil>php -v
PHP 8.3.2 (cli) (built: Jan 16 2024 20:47:57) (ZTS Visual C++ 2019 x64)
Copyright (c) The PHP Group
Zend Engine v4.3.2, Copyright (c) Zend Technologies

C:\Users\sunil>

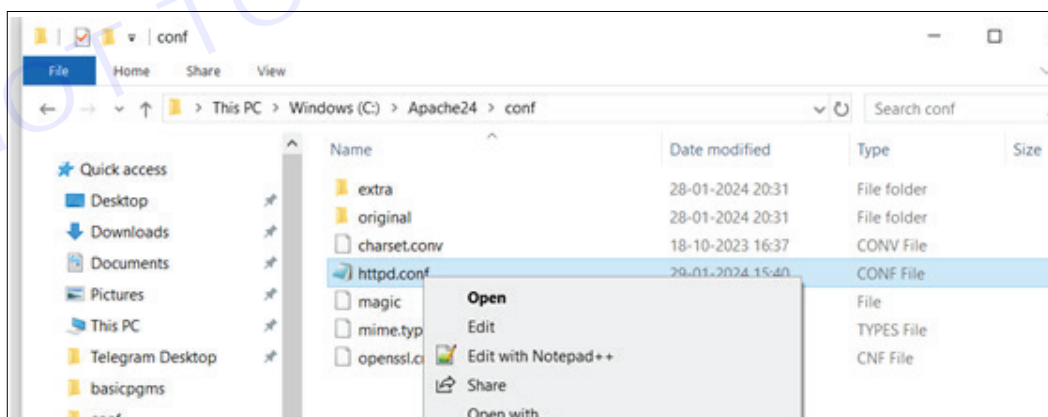
```

TASK 3: Configure Apache to run PHP as a Module

- 1 Open the apache configuration folder



- 2 Open 'httpd.conf' file



```

# This is the main Apache HTTP server configuration file. It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.4/> for detailed information.
# In particular, see
# <URL:http://httpd.apache.org/docs/2.4/mod/directives.html>
# for a discussion of each configuration directive.
#
# Do NOT simply read the instructions in here without understanding
# what they do. They're here only as hints or reminders. If you are unsure
# consult the online docs. You have been warned.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32), the
# server will use that explicit path. If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so "logs/access_log"
# with ServerRoot set to "/usr/local/apache2" will be interpreted by the
# server as "/usr/local/apache2/logs/access_log", whereas "/logs/access_log"
# will be interpreted as '/logs/access_log'.

```

- 3 Add the following codes in the file at the top of the file

```

AddHandler application/x-httpd-php .php
AddType application/x-httpd-php .php .html
LoadModule php_module "c:/php/php8apache2_4.dll"
PHPIniDir "c:/php"
LoadFile "c:/php/php8ts.dll"

```

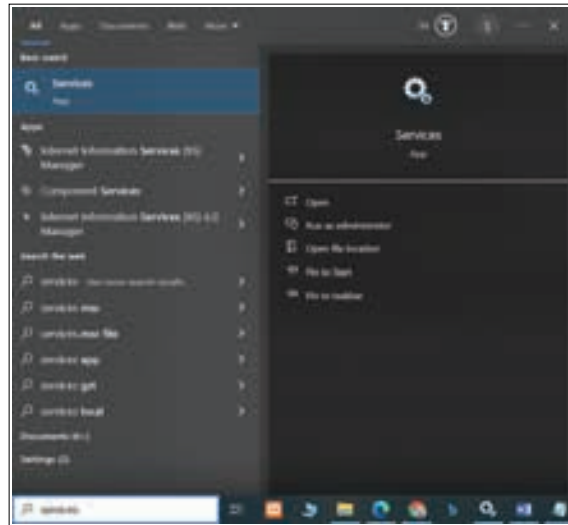
```

#
AddHandler application/x-httpd-php .php
AddType application/x-httpd-php .php .html
LoadModule php_module "c:/php/php8apache2_4.dll"
PHPIniDir "c:/php"
LoadFile "c:/php/php8ts.dll"

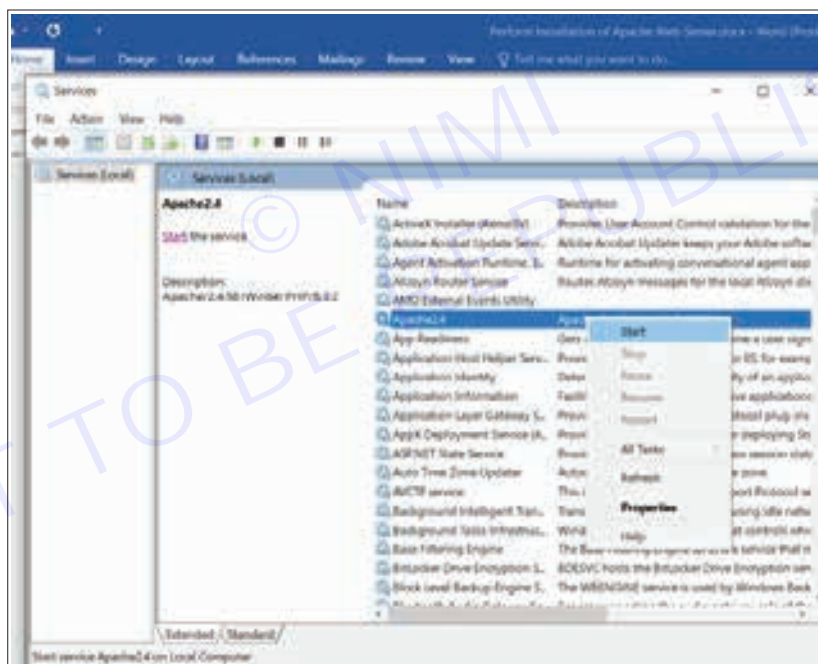
# This is the main Apache HTTP server configuration file. It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.4/> for detailed information.

```

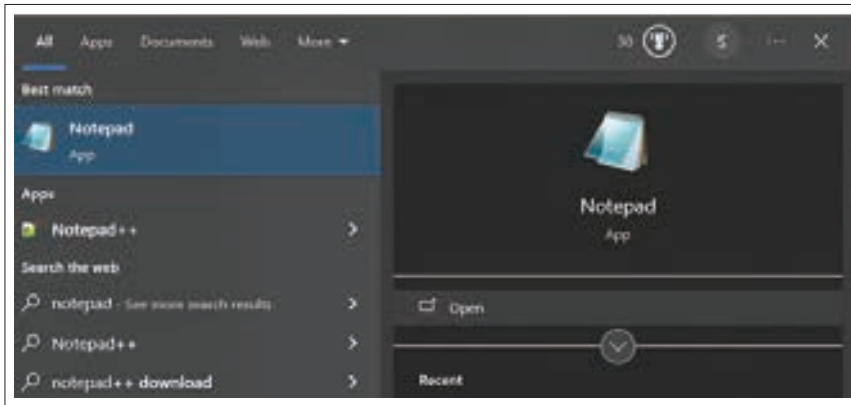
4 Type services in search menu to open windows services



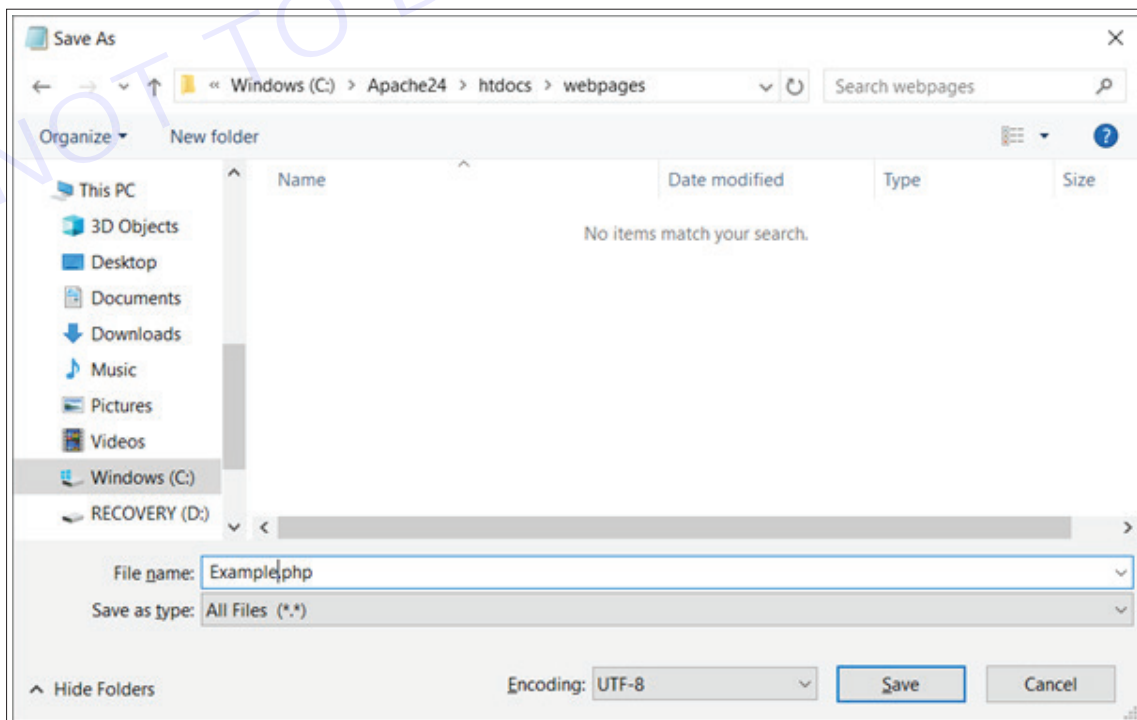
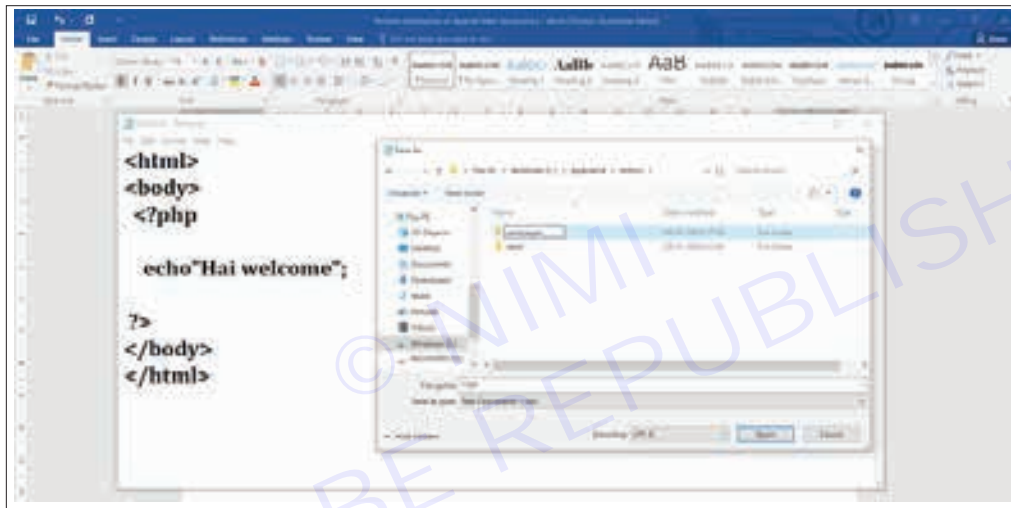
5 Start the Apache services from the windows services



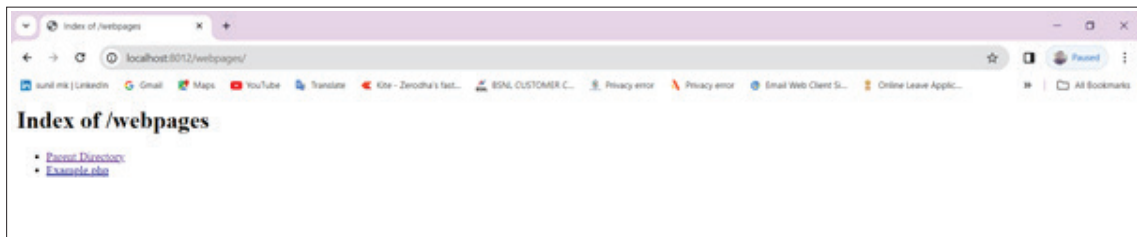
6 Open the notepad and write sample PHP



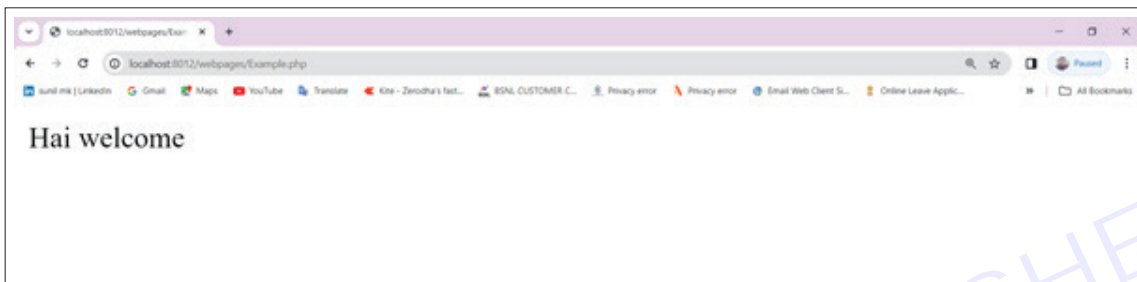
7 Save the in C:\Apache24\htdocs in a folder with .php extension



- 8 Open the browser and type the following address
`http://localhost/foldername/`



- 9 Click the php file to run the php sample script and verify the output



© NIMI
NOT TO BE REPUBLISHED

EXERCISE 51 : Demonstrate on if statement Using the else clause with if statement, switch statement Using the ? operator, while statement, do while statement, for statement, Breaking out of loops, Nesting loops

Objectives

At the end of this exercise you shall be able to

- use if-else and while statement in PHP
- use while and for loops in PHP
- use ? operator and nesting loops in PHP.

Requirements

Tools/Materials

- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apache web server
- PHP
- Text editor
- Web browser

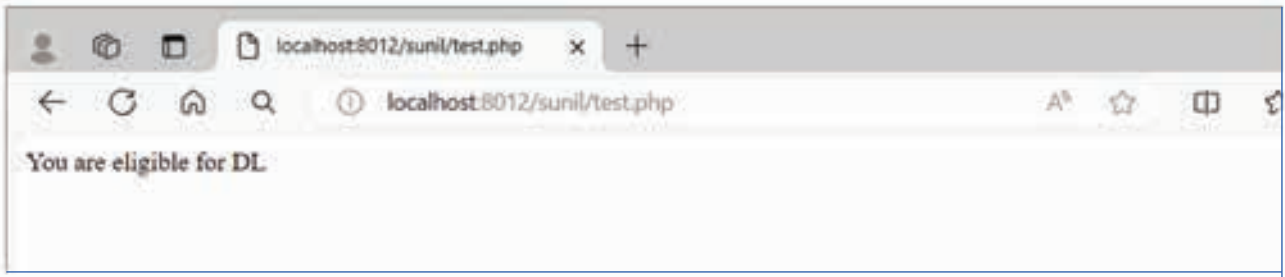
Procedure

TASK 1: Using If statement

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
  <?php
    $age = 21;
    if ($age>18)
    {
      echo "You are eligible for DL";
    }
  ?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

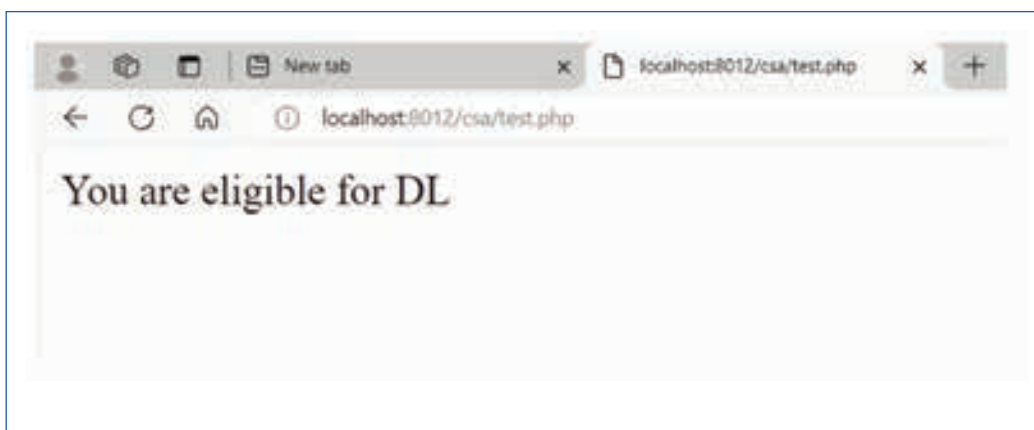


TASK 2: Using the else clause with if statement

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
  <?php
    $age = 21;
    if ($age >= 18)
    {
      echo "You are eligible for DL";
    }
    else
    {
      echo "You are NOT eligible for DL";
    }
  ?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 3: Using Switch statement

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head> <title>PHP </title></head>
<body>
<?php
    $favcolor = "brown";
    switch ($favcolor)
    {
        case "red": echo "Your favorite color is red!";
            break;
        case "blue": echo "Your favorite color is blue!";
            break;
        case "green": echo "Your favorite color is green!";
            break;
        default: echo "Your favorite color is neither red, blue, nor green!";
    }
?>
<p>END</p>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

**TASK 4: Using the ? Operator**

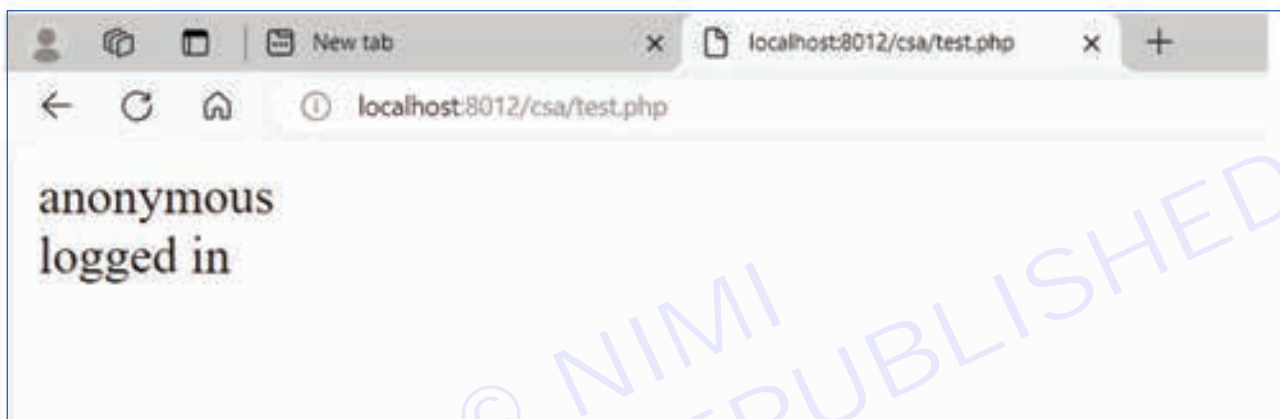
- 1 Open the text editor
- 2 Write the following codes
<?php


```

echo $status = (empty($user)) ? "anonymous" : "logged in";
echo("<br>");
$user = "John Doe";
echo $status = (empty($user)) ? "anonymous" : "logged in";
?>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 5: Using While statement

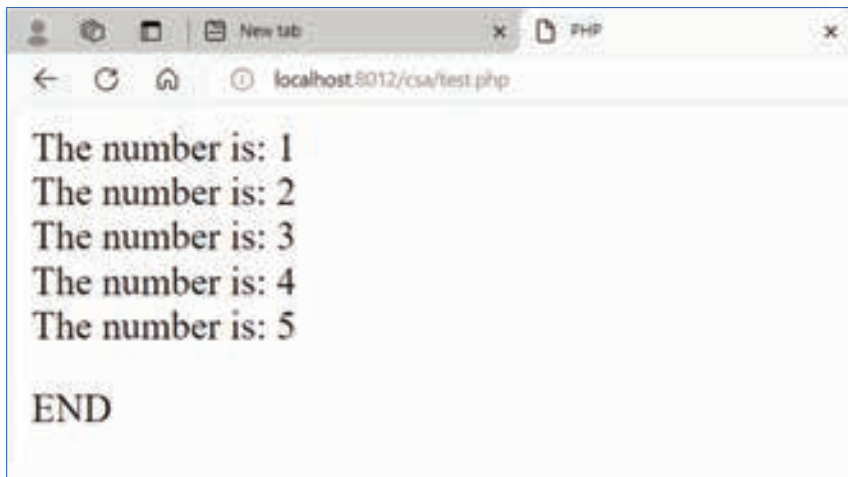
- 1 Open the text editor
- 2 Write the following codes


```

<html>
<head> <title>PHP </title></head>
<body>
<?php
    $x = 1;
    while($x <= 5)
    {
        echo "The number is: $x <br>";
        $x++;
    }
?>
<p>END</p>
</body>
</html>

```
- 3 Save the program in C:\Apache24\htdocs in a folder with php extension
- 4 Run the Apache services from windows services

- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

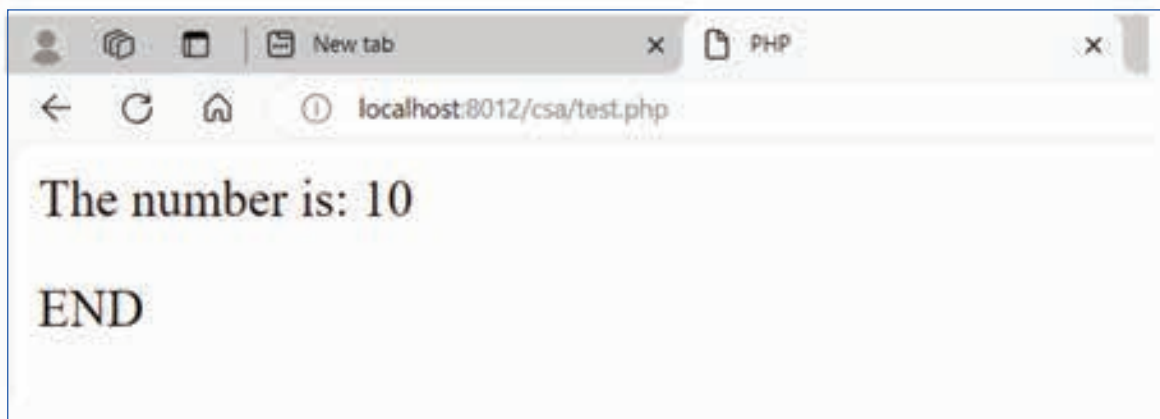


TASK 6: Using do-while statement

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head> <title>PHP </title></head>
<body>
  <?php
    $x = 10;
    do
    {
      echo "The number is: $x <br>";
      $x++;
    } while ($x <= 5);
  ?>
  <p>END</p>
</body>
</html>
```

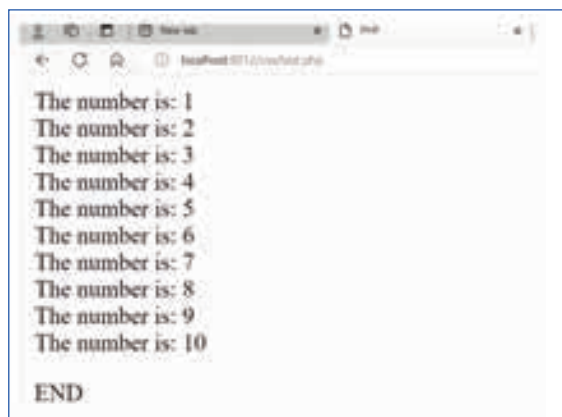
- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 7: Using for statement

- 1 Open the text editor
- 2 Write the following codes


```
<html>
<head> <title>PHP </title></head>
<body>
<?php
    for ($x =1; $x <=10; $x++)
    {
        echo "The number is: $x <br>";
    }
?>
<p>END</p>
</body>
</html>
```
- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
<http://localhost/foldername/>
- 6 Click the php file to run and verify the output

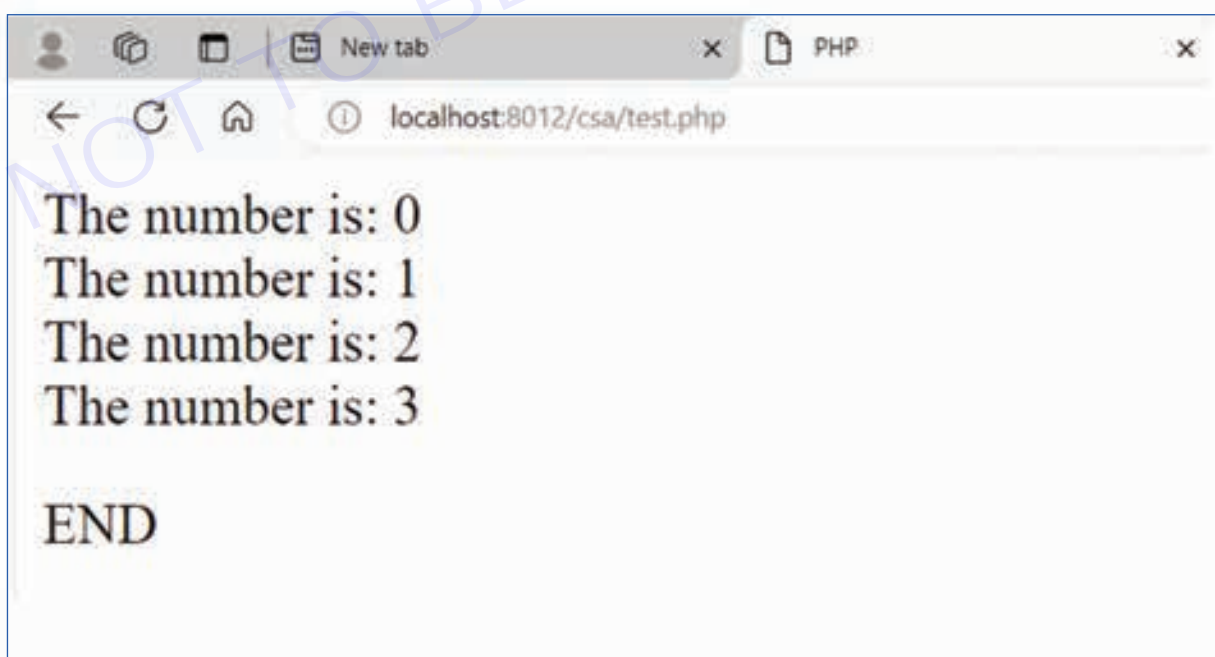


TASK 8: Breaking out of loops

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head> <title>PHP </title></head>
<body>
<?php
    for ($x = 0; $x < 10; $x++)
    {
        if ($x == 4)
        {
            break;
        }
        echo "The number is: $x <br>";
    }
?>
<p>END</p>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
<http://localhost/foldername/>
- 6 Click the php file to run and verify the output

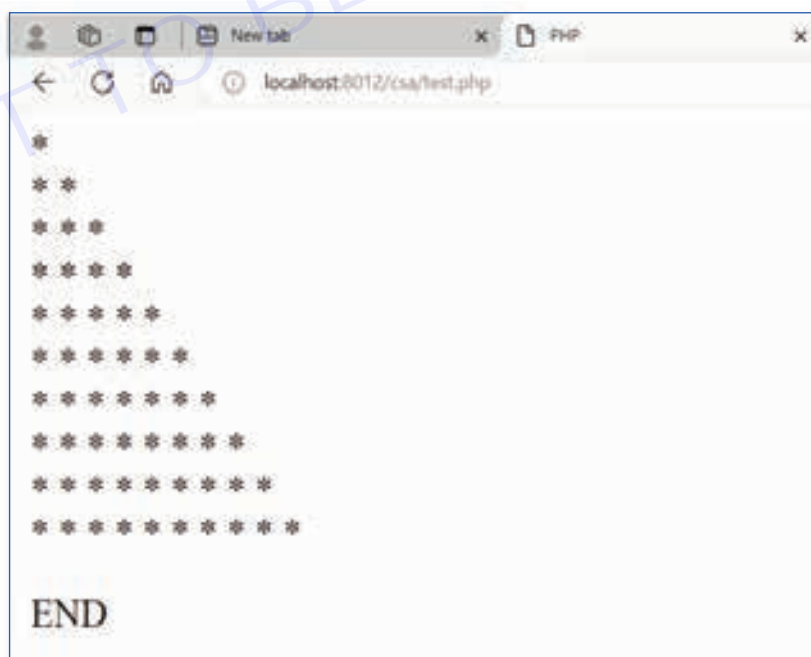


TASK 9: Nesting loops

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head> <title>PHP </title></head>
<body>
<?php
    for ($x = 1; $x <=10; $x++)
    {
        for ($y = 1; $y <=$x; $y++)
        {
            echo "** ";
        }
        echo "<br/>";
    }
?>
<p>END</p>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



- 1 Create a PHP if statement to determine whether a variable \$temperature is greater than 30 degrees Celsius then display "It's hot outside!"
- 2 Create a PHP program that determines whether a student has passed or failed an exam based on their score. If the score is greater than or equal to 60, display a "Pass" message; otherwise, display a "Fail" message.
- 3 Write a PHP if statement to determine if a variable \$num is divisible by 2 and 3. If it is, echo "The number is divisible by both 2 and 3", otherwise echo "The number is not divisible by both 2 and 3".
- 4 Write a PHP switch statement that checks the value of a variable \$dayOfWeek and echoes "It's a weekday" for Monday to Friday, and "It's a weekend" for Saturday and Sunday.
- 5 Write a PHP switch statement to determine the discount percentage based on the quantity of items purchased. If the quantity is 1-10, apply a 5% discount, if it's 11-20, apply a 10% discount, and if it's over 20, apply a 15% discount.
- 6 Write a PHP ternary operator statement to determine if a given number stored in \$num is even or odd, and echo "Even" if it's even, otherwise echo "Odd".
- 7 Write a PHP while loop that calculates the factorial of a given number \$n and echoes the result.
- 8 Create a PHP while loop that generates Fibonacci numbers until reaching a value greater than 1000, and echoes each Fibonacci number.
- 9 Create a PHP for loop to generate multiples of 2 and 3 in between 1 and 100, and echoes each number.
- 10 Write a PHP script using nested for loops to create a simple multiplication table (e.g., 1x1=1, 1x2=2, ..., 5x5=25).
- 11 Create a script with nested for loops. The outer loop iterates 3 times, and the inner loop iterates 5 times. Inside the inner loop, use break to exit the entire loop structure (both inner and outer) if the current iteration number in the inner loop reaches 3.

EXERCISE 52 : Demonstrate on Function and returning value from function, user defined functions, dynamic functions, variable scope, accessing variable with the global statement, Function calls with the static statement, setting default values for arguments, Passing arguments to a function by value, Passing arguments to a function by reference, Testing for function existence

Objectives

At the end of this exercise you shall be able to

- create and use user defined functions in PHP
- check different variable scope in PHP
- test the existence of function in PHP.

Requirements

Tools/Materials

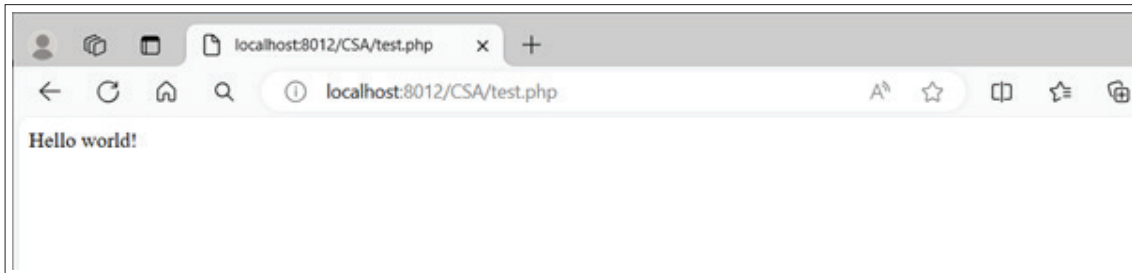
- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apache web server
- PHP
- Text editor
- Web browsers

Procedure

TASK 1: Create user defined functions

- 1 Open the text editor
- 2 Write the following codes


```
<html>
<body>
<?php
    function writeMsg()
    {
        echo "Hello world!";
    }
    writeMsg();
?>
</body>
</html>
```
- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 2: Passing arguments to a function by value

- 1 Open the text editor
- 2 Write the following codes

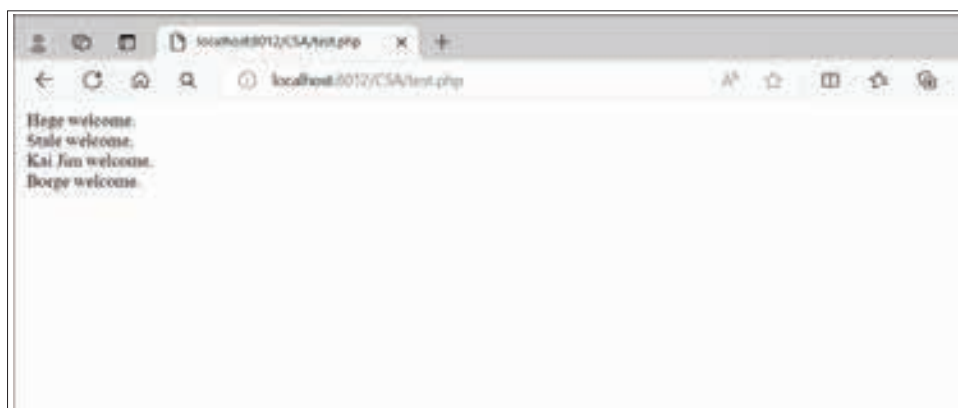
```

<html>
<body>

    <?php
    function familyName($fname)
    {
        echo "$fname welcome.<br>";
    }

    familyName("Hege");
    familyName("Stale");
    familyName("Kai Jim");
    familyName("Borge");
    ?>
</body>
</html>
  
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 3: Setting default values for arguments

- 1 Open the text editor
- 2 Write the following codes

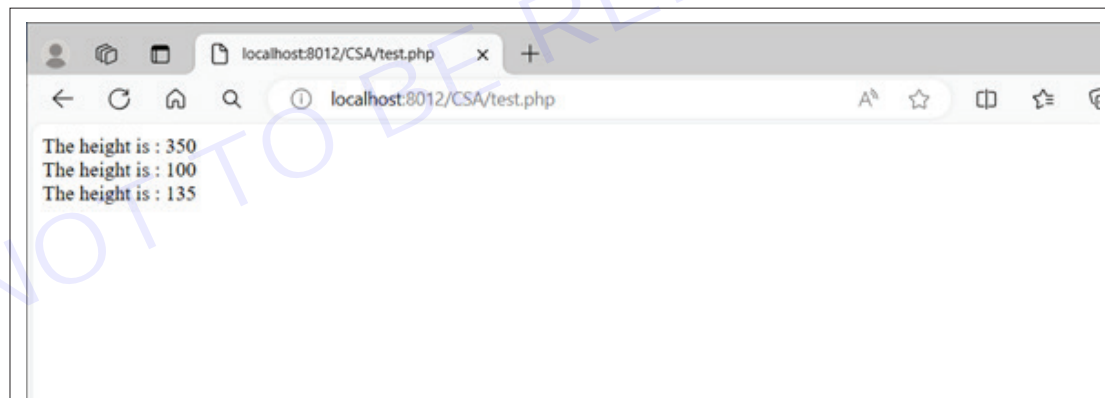
```

<html>
<body>
<?php
    function setHeight(int $minheight=100)
    {
    echo "The height is : $minheight <br>";
    }

    setHeight(350);
    setHeight();
    setHeight(135);
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

**TASK 4: Passing arguments to a function by reference**

By default, function arguments are passed by value (so that if the value of the argument within the function is changed, it does not get changed outside of the function). To allow a function to modify its arguments, they must be passed by reference.

To have an argument to a function always passed by reference, prepend an ampersand (&) to the argument name in the function definition:

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<head>

```

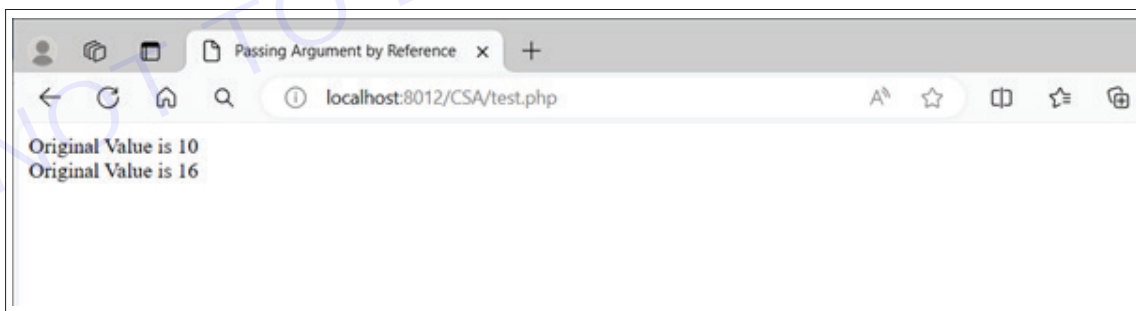
```

<title>Passing Argument by Reference</title>
</head>
<body>

    <?php
        function addFive($num)
        {
            $num += 5;
        }
        function addSix(&$num)
        {
            $num += 6;
        }
        $orignum = 10;
        addFive( $orignum );
        echo "Original Value is $orignum<br />";
        addSix( $orignum );
        echo "Original Value is $orignum<br />";
    ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



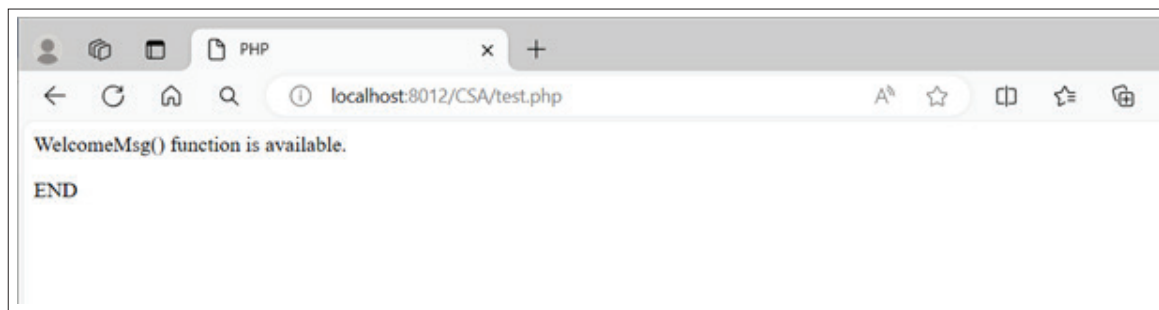
TASK 5: Testing for function existence

If you ever want to know whether you have a function available to you, use the `function_exists()` function. This takes one string parameter that is the name of a function and returns true if the function exists or false if it does not.

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head> <title>PHP </title></head>
<body>
<?php
    function WelcomeMsg()
    {
        echo "Welcome to GeeksforGeeks";
    }
    if (function_exists('WelcomeMsg'))
    {
        echo "WelcomeMsg() function is available.\n";
    }
    else
    {
        echo "WelcomeMsg() function is not available.\n";
    }
?>
<p>END</p>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
`http://localhost/foldername/`
- 6 Click the php file to run and verify the output



TASK 6: Returning value from function

- 1 Open the text editor
- 2 Write the following codes

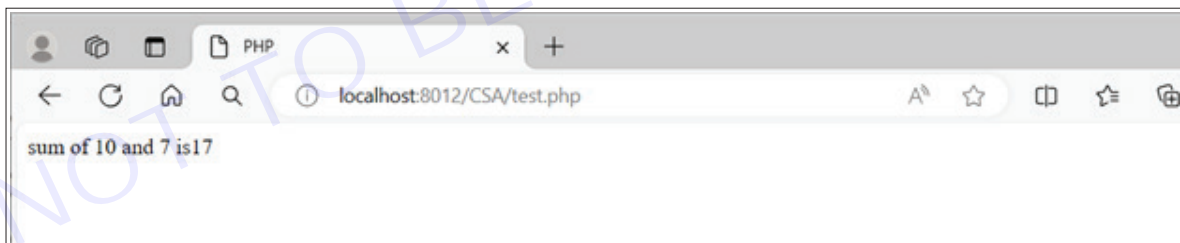
```

<html>
<head> <title>PHP </title></head>
<body>
  <?php
      function sum(int $x, int $y)
      {
          $z = $x + $y;
          return $z;
      }
      $a=10;
      $b=7;
      echo "sum of $a and $b is".sum(10,7);

  ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

**TASK 7: Create dynamic functions**

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<head>
<title>Dynamic Function Calls</title>
</head>
<body>
  <?php
      function sayHello()
      {

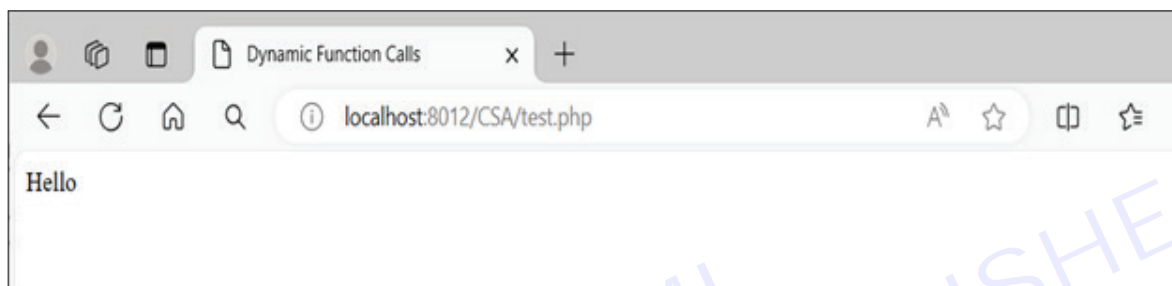
```

```

        echo "Hello<br />";
    }
    $function_holder = "sayHello";
    $function_holder();
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 8: Accessing variable with the global statement

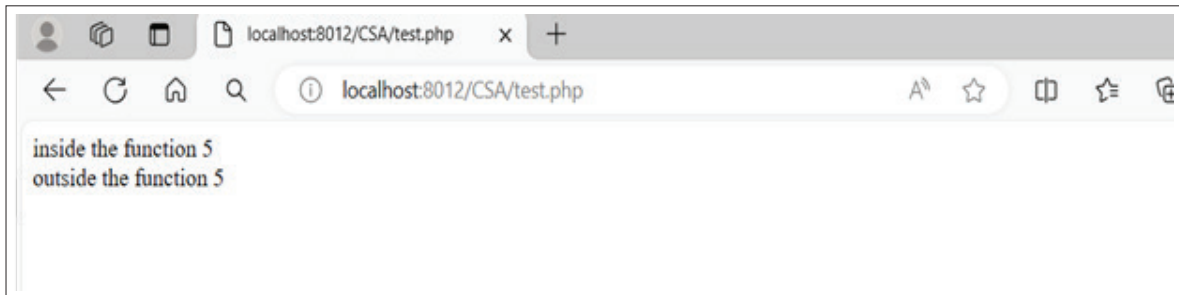
- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
    $count = 1;
    function keep_track()
    {
        global $count;
        $count=5;
        echo"inside the function ". $count;
        echo "<br />";
    }
    keep_track();
    echo "outside the function ".$count."<br />";
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

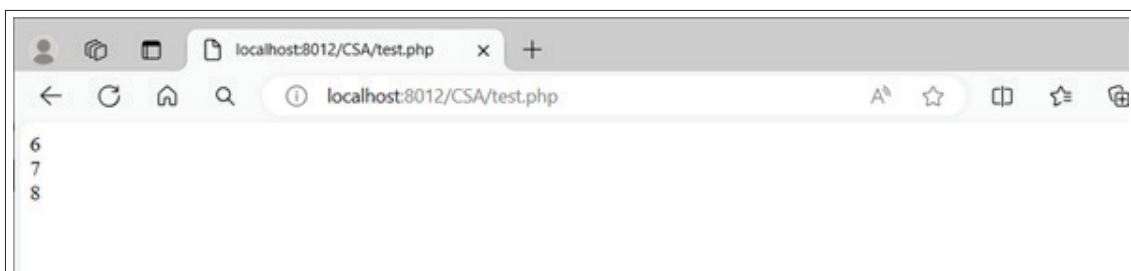


TASK 9: Function calls with the static statement

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
<?php
function keep_track()
{
    STATIC $count = 5;
    $count++;
echo $count;
    echo "<br />";
}
keep_track();
keep_track();
    keep_track();
?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



- 1 Write a function named `calculateArea` that calculates the area of a rectangle given its length and width.
- 2 Write a PHP function that takes two numbers as parameters (`$num1` and `$num2`) and print their sum. Call the function with different values and print result.
- 3 Create a function called `sayHello` that takes a parameter `$name` and defaults to "Guest" if no value is provided. The function should echo "Hello, \$name!". Call the function without providing a value for `$name` and observe the output.
- 4 Create a function named `getAgeCategory` that determines the age category based on the provided age. It should take one parameter `$age` with a default value of 25. The function should return "Child" for ages 0-12, "Teen" for ages 13-19, "Adult" for ages 20-59, and "Senior" for ages 60 and above.
- 5 Create a function called `incrementByReference` that takes an integer parameter `$num` by reference and increments it by 1 inside the function. Call the function with a variable and observe its value change after the function call.
- 6 Implement a function called `swapValues` that takes two variables `$a` and `$b` by reference and swaps their values inside the function. Call the function with two variables and print their values before and after the function call to demonstrate the swap.
- 7 Write a PHP script that defines a dynamic function named `multiply` which takes two parameters and returns their product. Then, call this dynamic function with values 5 and 3, and echo the result.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 53 : Demonstratethe Writing to the browser, Getting input from forms, Output buffering, Common math, Random numbers, File upload, File download, Environment variables

Objectives

At the end of this exercise you shall be able to

- write content to the browser using PHP
- check common math and random number functions in PHP
- upload download from server using PHP.

Requirements

Tools/Materials

- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apache web server
- PHP
- Text editor
- Web browser

Procedure

TASK 1: Writing to the browser

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
<?php
    // Example with echo
    echo "Hello, world!";

    // Example with print
    print "This is also displayed";

    // Example with printf
    $name = "Alice";
    printf("Welcome, %s!", $name);

    // Example with sprintf
    $message = sprintf("You have %d unread messages.", 3);
    echo $message;

    // Declare variable and store the string
    $output = "A computer science portal";
    echo "<script>console.log('Value of variable: " . $output . "')</script>";
?>
```

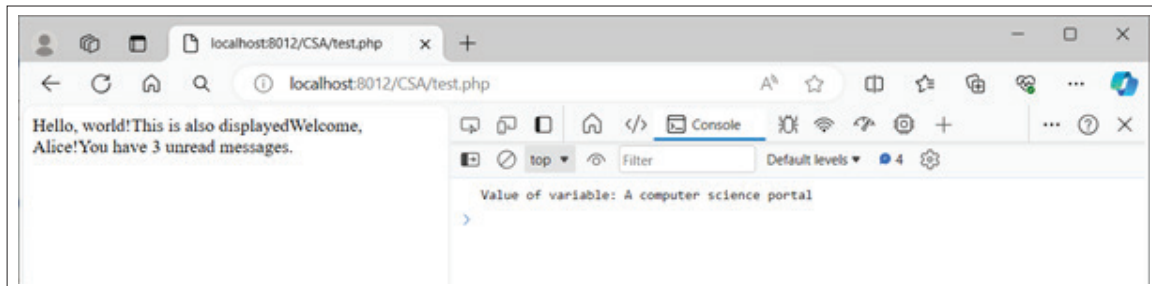


```

</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 2: Output buffering

1. Open the text editor
2. Write the following codes

```

<html>
<body>
<?php
    // Start output buffering
    ob_start();

    // Your regular PHP code
    echo "Hello, ";
    echo "World!";

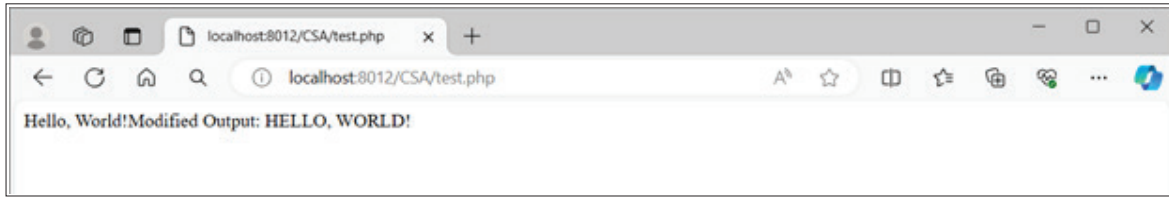
    // Capture the output
    $output = ob_get_contents();

    // End output buffering and send the content to the browser
    ob_end_flush();

    // Use the captured output
    echo "Modified Output: " . strtoupper($output);
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 3: Getting input from forms

1 Create a HTML form

- Open the text editor
- Write the following codes

```
<html >
<head>
    <title>Form Example</title>
</head>
<body>
    <form action="process.php" method="post">
    <label for="name">Name:</label>
    <input type="text" name="name" id="name" required>

    <label for="email">Email:</label>
    <input type="email" name="email" id="email" required>

    <input type="submit" value="Submit">
    </form>
</body>
</html>
```

- Save the program in C:\Apache24\htdocs in a folder as index.html

2 Create a PHP code for getting input from forms

- Open the text editor
- Write the following codes

```
<html>
<body>
    <?php

        // Check if the form is submitted
    if ($_SERVER["REQUEST_METHOD"] == "POST")
    {

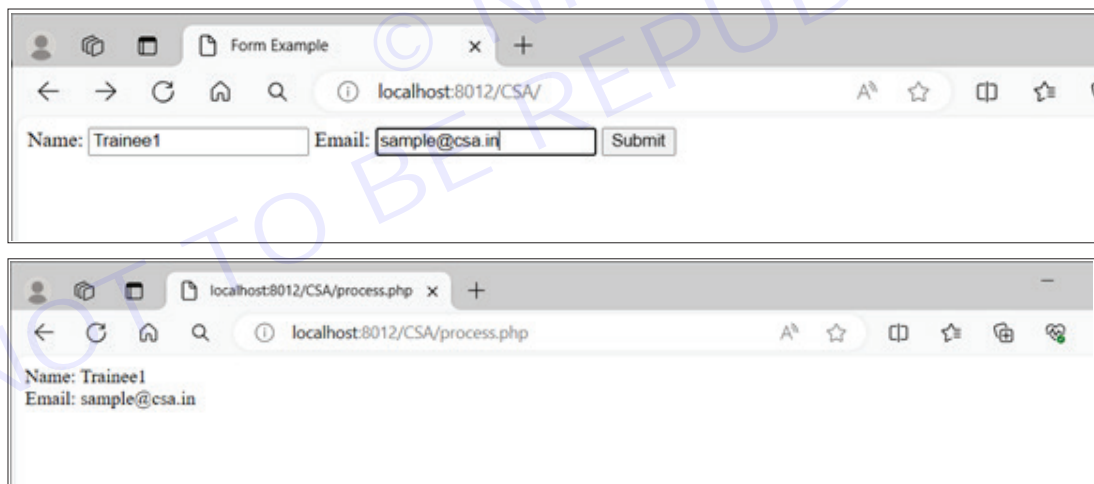
        // Retrieve form data using $_POST
        $name = $_POST["name"];
        $email = $_POST["email"];
```

```

        // Do something with the data (e.g., display it)
        echo "Name: $name<br>";
        echo "Email: $email";
    }
    else
    {
        // Redirect or display an error if someone tries to access this script directly
        header("Location: index.html");
        exit();
    }
    ?>
</body>
</html>

```

- Save the program in C:\Apache24\htdocs in a folder with .php extension
- Run the Apache services from windows services
- Open the browser and type the following address
http://localhost/foldername/
- Click the index.html file to run and verify the output



TASK 4: Display environment variables

- Open the text editor
- Write the following codes

```

<?php
echo $_SERVER['PHP_SELF'];
echo "<br>";
echo $_SERVER['SERVER_NAME'];
echo "<br>";

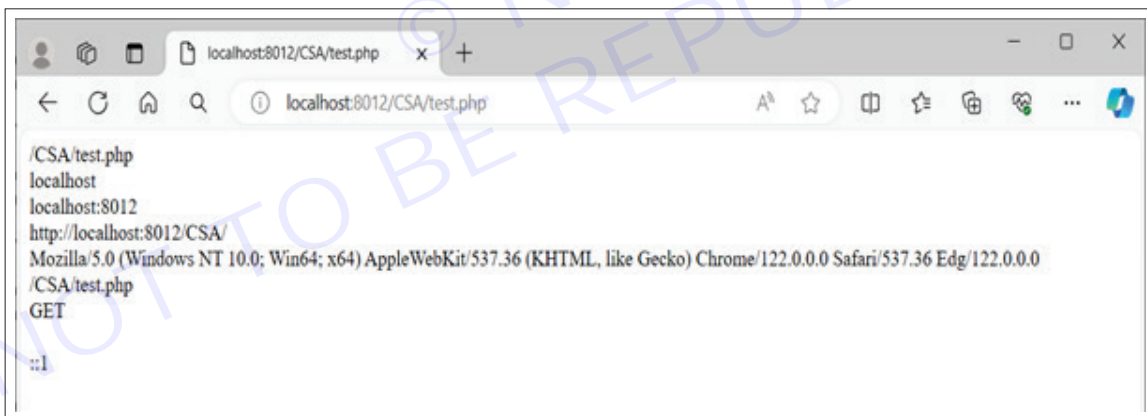
```

```

echo $_SERVER['HTTP_HOST'];
echo "<br>";
echo $_SERVER['HTTP_REFERER'];
echo "<br>";
echo $_SERVER['HTTP_USER_AGENT'];
echo "<br>";
echo $_SERVER['SCRIPT_NAME'];
echo "<br>";
echo $_SERVER['REQUEST_METHOD'];
echo "<br>";
echo $_SERVER['QUERY_STRING'];
echo "<br>";
echo $_SERVER['REMOTE_ADDR'];
    ?>

```

- Save the program in C:\Apache24\htdocs in a folder with .php extension
 - Run the Apache services from windows services
 - Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 5: File upload

- 1 Create an HTML form that allows users to select and upload files.

- Open the text editor
- Write the following codes

```

<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>File Upload</title>
</head>

```

```

<body>
<form action="upload.php" method="post" enctype="multipart/form-data">
  <label for="file">Select a file:</label>
  <input type="file" name="file" id="file" required>
  <br>
  <input type="submit" value="Upload File">
</form>
</body>
</html>

```

- Save the program in C:\Apache24\htdocs in a folder as index.html

2 Create PHP Script to Handle Upload

- Open the text editor
- Write the following codes

```

<html>
<body>
<?php
  // upload.php

  if ($_SERVER["REQUEST_METHOD"] == "POST")
  {
    // Check if the file was uploaded without errors
    if (isset($_FILES["file"]) && $_FILES["file"]["error"] == 0)
    {
      $allowedTypes = ['jpg', 'jpeg', 'png', 'gif', 'txt'];
      $maxSize = 5 * 1024 * 1024; // 5 MB

      $targetDir = "uploads/"; //create a folder in this name in your php file saved location .
      $targetFile = $targetDir . basename($_FILES["file"]["name"]);
      $fileExtension = strtolower(pathinfo($targetFile, PATHINFO_EXTENSION));

      // Check file type
      if (!in_array($fileExtension, $allowedTypes))
      {
        echo "Error: Invalid file type.";
      }
      elseif ($_FILES["file"]["size"] > $maxSize)
      {
        echo "Error: File size exceeds the limit.";
      }
      else
      {
        // Move the uploaded file to the target directory
        if (move_uploaded_file($_FILES["file"]["tmp_name"], $targetFile))
        {

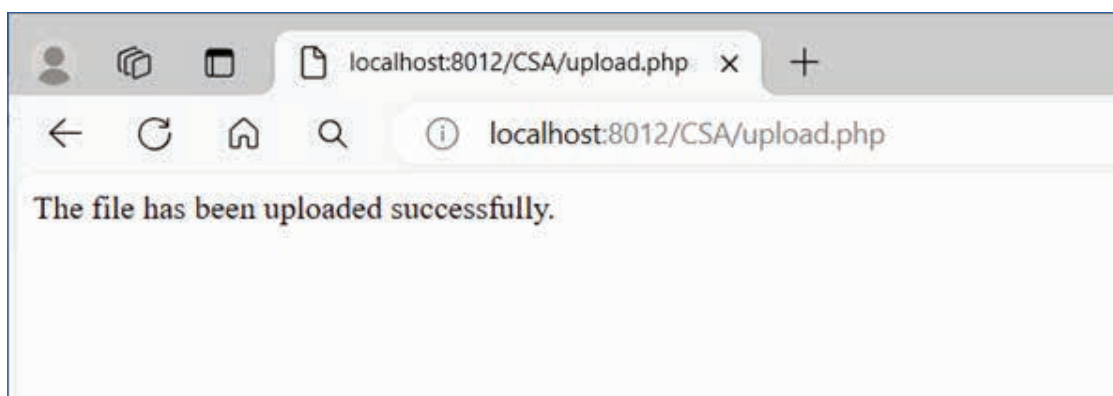
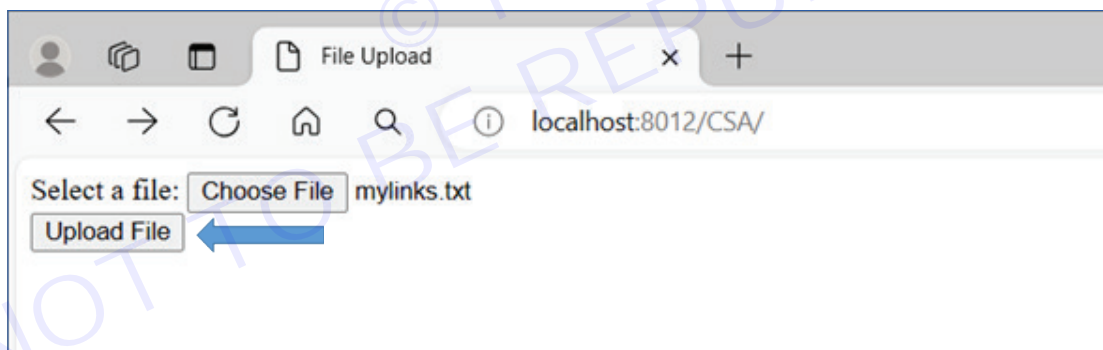
```

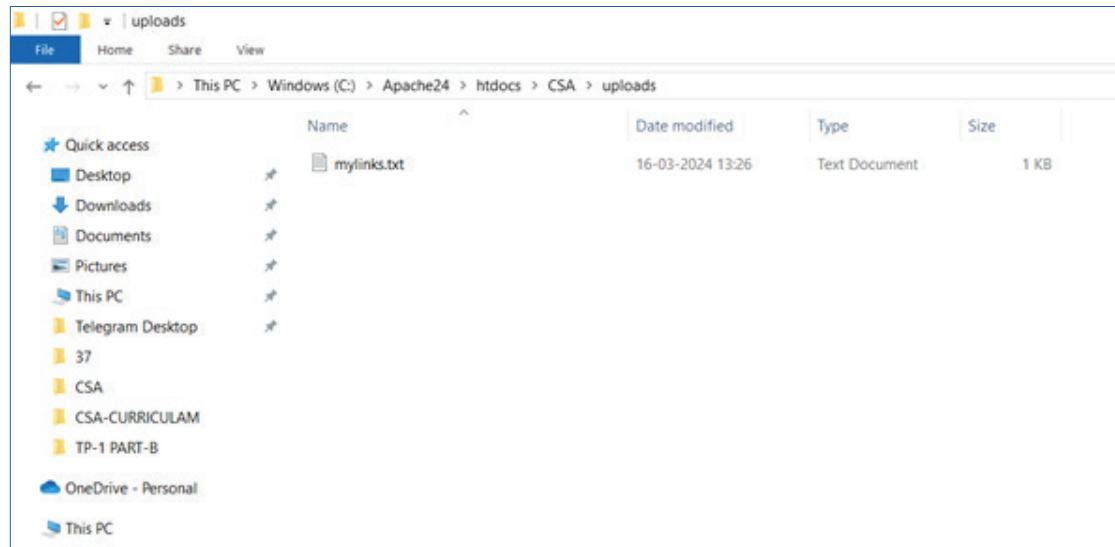
```

        echo "The file has been uploaded successfully.";
    }
    else
    {
        echo "Sorry, there was an error uploading your file.";
    }
}
}
else
{
    echo "Error: " . $_FILES["file"]["error"];
}
}
?>
</body>
</html>

```

- Save the program in C:\Apache24\htdocs in a folder with .php extension
- Run the Apache services from windows services
- Open the browser and type the following address
http://localhost/foldername/
- Click the index.html file to run and verify the output





TASK 6: File download

- Open the text editor
- Write the following codes

```
<?php
```

```
$file_path = 'path/to/your/file.pdf'; // Replace with the actual file path
```

```
$file_name = 'downloaded_file.pdf'; // Optional: Customize the filename for download
```

```
header('Content-Type: application/pdf'); // Set content type based on file extension
```

```
header('Content-Disposition: attachment; filename="' . $file_name . '"); // Force download
```

```
header('Content-Length: ' . filesize($file_path)); // Set file size for better download management
```

```
readfile($file_path); // Read and output the file contents directly
```

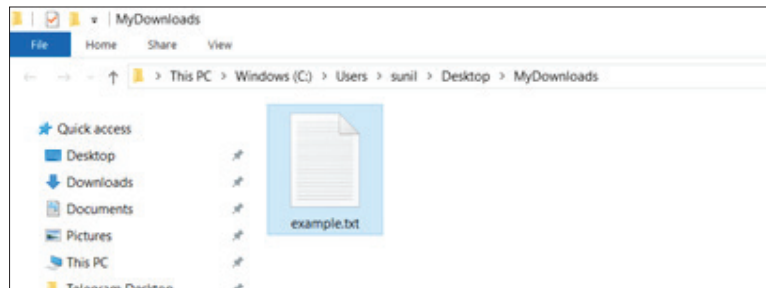
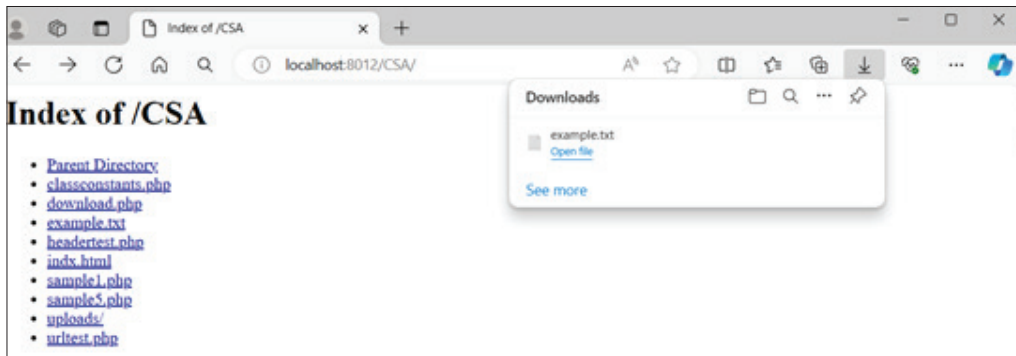
```
// Alternatively, you can use:
```

```
// echo file_get_contents($file_path);
```

```
exit(); // Prevent further output
```

```
?>
```

- Save the program in C:\Apache24\htdocs in a folder with .php extension
- Run the Apache services from windows services
- Open the browser and type the following address
http://localhost/foldername/
- Click the php file to run and verify the output



TASK 7: Common math functions

Basic Arithmetic Functions:

`abs($number)`: Returns the absolute (positive) value of a number.

`round($number, $precision = 0)`: Rounds a number to a specified number of decimal places.

`ceil($number)`: Rounds a number up to the nearest integer.

`floor($number)`: Rounds a number down to the nearest integer.

`min($number1, $number2, ...)`: Returns the smallest value from a list of numbers.

`max($number1, $number2, ...)`: Returns the largest value from a list of numbers.

`pow($base, $exponent)`: Returns the base raised to the power of the exponent.

`sqrt($number)`: Returns the square root of a number.

Trigonometric Functions:

`sin($angle)`: Returns the sine of an angle in radians.

`cos($angle)`: Returns the cosine of an angle in radians.

`tan($angle)`: Returns the tangent of an angle in radians.

`asin($number)`: Returns the arcsine of a number (in radians).

`acos($number)`: Returns the arccosine of a number (in radians).

`atan($number)`: Returns the arctangent of a number (in radians).

Logarithmic Functions:

`log($number, $base = M_E)`: Returns the logarithm of a number to a specified base (default is natural logarithm, base e).

`exp($number)`: Returns the value of e raised to the power of a number.

Number Base Conversion Functions:

`base_convert($number, $frombase, $tobase)`: Converts a number between arbitrary bases.

`bindec($binary_string)`: Converts a binary string to a decimal number.

decbin(\$number): Converts a decimal number to a binary string.

dechex(\$number): Converts a decimal number to a hexadecimal string.

hexdec(\$hex_string): Converts a hexadecimal string to a decimal number.

Other Mathematical Functions:

pi(): Returns the value of pi.

deg2rad(\$number): Converts degrees to radians.

rad2deg(\$number): Converts radians to degrees.

fmod(\$x, \$y): Returns the floating-point remainder of x divided by y.

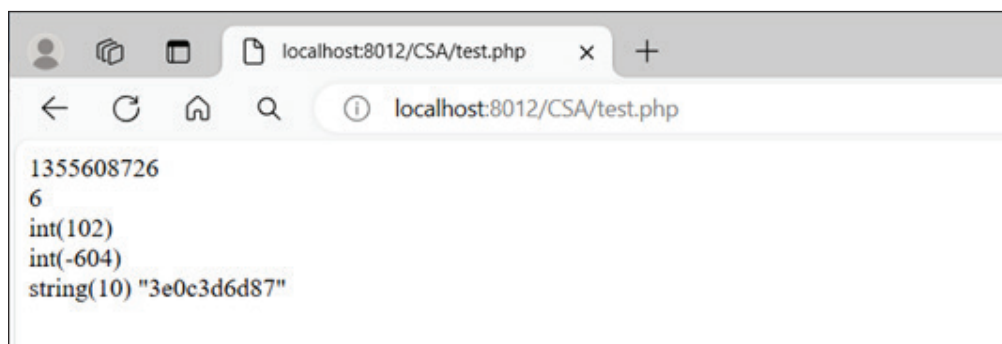
— — — — —

TASK 8: Display random numbers

- Open the text editor
- Write the following codes

```
<?php
    echo rand()." <br/>";//returns a pseudo-random integer between 0 and getrandmax().
    echo rand(5, 15). " <br/>";//returns random number between 5 and 15 (inclusive)
    var_dump(random_int(100, 999));
    echo " <br/>";
    var_dump(random_int(-1000, 0));// Get a cryptographically secure, uniformly selected
integer function for situations requiring strong randomness.
    echo " <br/>";
    $bytes = random_bytes(5);
    var_dump(bin2hex($bytes));// Get cryptographically secure random bytes
?>
```

- Save the program in C:\Apache24\htdocs in a folder with .php extension
- Run the Apache services from windows services
- Open the browser and type the following address
http://localhost/foldername/
- Click the php file to run and verify the output



- 1 Provide a code example demonstrating the use of the echo and print statements to display text on a web page.
- 2 Provide a code example demonstrating how to capture the output of a PHP script and store it in a variable for further processing.
- 3 Provide examples of how to retrieve user input from fields such as text fields, checkboxes, radio buttons, and select dropdowns.
- 4 Provide a code example demonstrating how to access form data submitted via the GET method.
- 5 Provide a code example illustrating how to create a form that allows users to upload a file to the server.
- 6 Provide a simple code example demonstrating how to force the download of a file named "example.txt".
- 7 Provide a code snippet illustrating how to set headers for downloading a PDF file named "document.pdf".
- 8 Write a PHP function that takes a floating-point number as input and returns its absolute value.
- 9 Write a PHP code to calculate the sine, cosine, and tangent of an angle in radians.
- 10 Write a PHP code snippet to generate a random integer between 1 and 100 and display it on the screen.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 54 : Demonstrate on E-mail in PHP, anatomy of a cookie, setting a cookie with PHP, deleting a cookie, creating session cookie, working with the query string, creating query string, starting a session, Working with session, variables, Destroying session, passing session IDs, Encoding and decoding session variables

Objectives

At the end of this exercise you shall be able to

- set and delete cookie using PHP
- create and destroy a session using PHP
- work with a query string using PHP.

Requirements

Tools/Materials

- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apache web server
- PHP
- Text editor
- Web browser

Procedure

TASK 1: Knowing anatomy of a cookie

Cookie Name: Choose a descriptive name that identifies the data stored.

Cookie Value: The data you want to store, like user preferences or a session ID.

Lifetime: How long the cookie should last (in seconds, hours, or days).

Path: The URL path where the cookie should be accessible. Default is current path.

Domain: The domain for which the cookie applies. Default is current domain.

Security: Whether the cookie should only be sent over secure HTTPS connections.

HttpOnly: Whether the cookie should be accessible only through HTTP, protecting it from JavaScript access.

TASK 2: setting a cookie with PHP

- 1 Open the text editor
- 2 Write the following codes

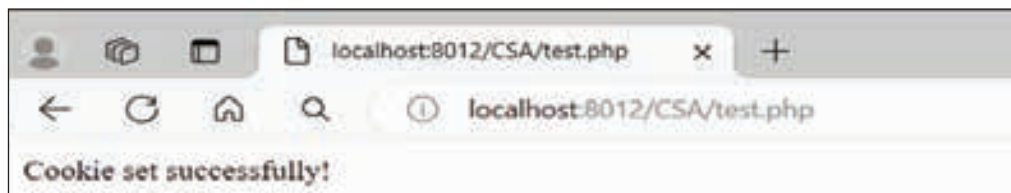
```
<html>
<body>
<?php
    $cookie_name = "username";
    $cookie_value = " John Carter";
    $expire = time()+30*24*60*60; // 30 days in seconds
    $path = "/"; // accessible on entire website
    $domain = ""; // default domain
    $secure = true; // only send over HTTPS
    $httponly = true; // not accessible through JavaScript
    setcookie($cookie_name, $cookie_value, $expire, $path, $domain, $secure,
    $httponly);
```

```

    echo "Cookie set successfully!";
    ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 3: Deleting a cookie

- 1 Deleting with unset()
- 1 Open the text editor
- 2 Write the following codes

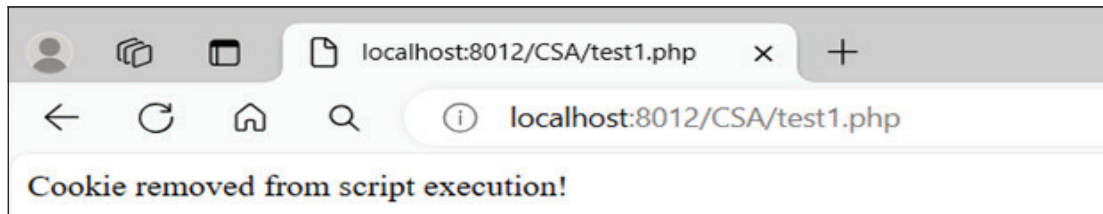
```

<html>
<body>
<?php
if (isset($_COOKIE["username"]))
{
    unset($_COOKIE["username"]);
    echo "Cookie removed from script execution!";
}
else
{
    echo "Cookie not found.";
}
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/

- 6 Click the php file to run and verify the output

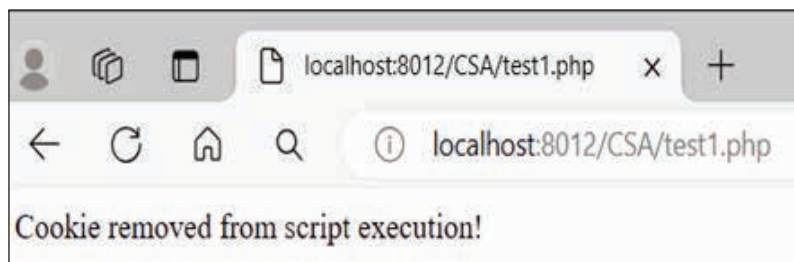


II Deleting with setcookie()

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
<?php
if (isset($_COOKIE["username"]))
{
setcookie($cookie_name, "", time() - 3600, ""); // Set expiration to past time
echo "Cookie removed from script execution!";
}
else
{
echo "Cookie not found.";
}
?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 4: Starting a session

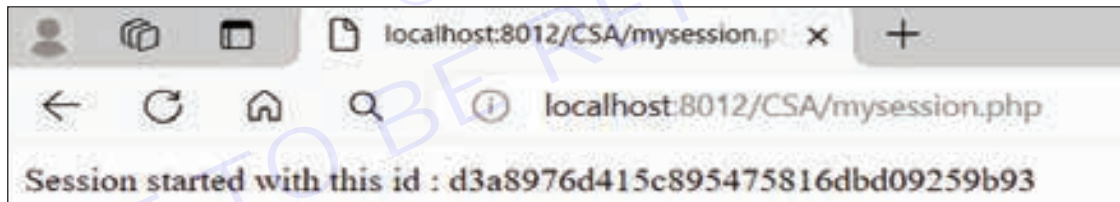
- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
// Start the session
session_start();
echo"Session started with this id : ". session_id();
// Now you can set session variables
$_SESSION['username'] = 'example_user';
$_SESSION['is_logged_in'] = true;
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

**TASK 5: Working with session variables**

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
// Start the session
session_start();
// Set session variables
$_SESSION['username'] = 'john_doe';
$_SESSION['user_email'] = 'john@example.com';
// Access session variables

```

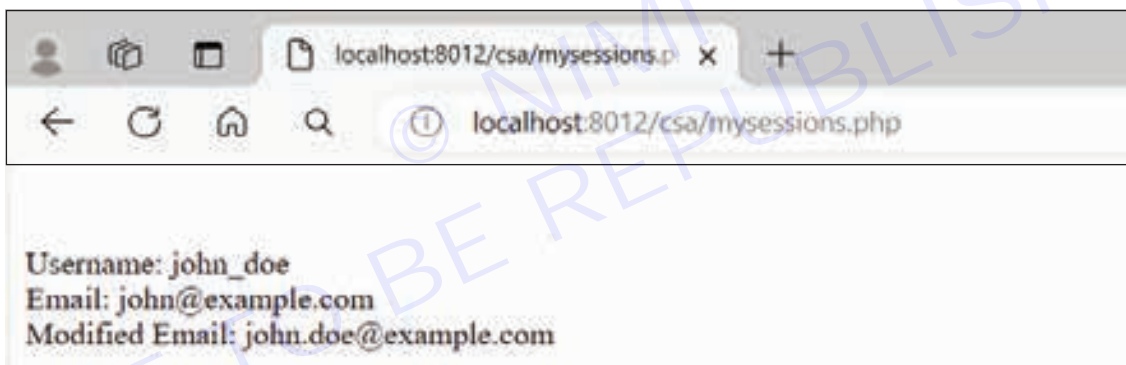
```

$username = $_SESSION['username'];
$user_email = $_SESSION['user_email'];
// Display session variables
echo "Username: $username <br>";
echo "Email: $user_email <br>";
// Modify session variable
$_SESSION['user_email'] = 'john.doe@example.com';
// Display modified session variable
echo "Modified Email: " . $_SESSION['user_email'] . "<br>";
?>

</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 6: Passing session ID

- I Pass Session ID using URLs

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
    session_start();
    $sessionId = session_id();
    echo "Session ID: $sessionId<br/>";
    echo "<a href='next_page.php?PHPSESSID=$sessionId'>Next Page</a>";
?>

```

```
</body>
```

```
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run the program
- 7 Click the "Next Page" link to pass 'Session ID' in the next page.

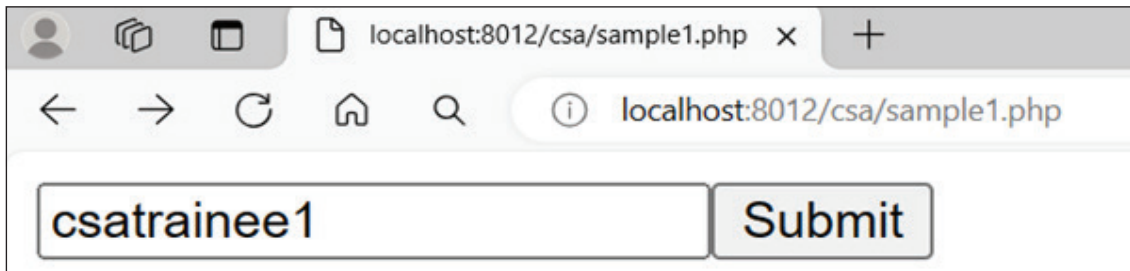


II Pass Session ID in Forms

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
<?php
    session_start();
    $sessionId = session_id();
    echo "<form action='process_form.php' method='post'>";
    echo "<input type='hidden' name='PHPSESSID' value='$sessionId'>";
    echo "<input type='text' name='username' placeholder='Enter your username'>";
    echo "<input type='submit' value='Submit'>";
    echo "</form>";
?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run the program
- 7 Click the "Submit" button to pass 'Session ID' in the next page.



III Retrieve Session ID on the Next Page

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
    session_start();
    if (isset($_GET['PHPSESSID']))
    {
        // If passed in the URL
        $sessionId = $_GET['PHPSESSID'];
    }
    elseif (isset($_POST['PHPSESSID']))
    {
        // If passed in a form
        $sessionId = $_POST['PHPSESSID'];
        echo $sessionId . "This session ID retrieved";
    }
    else
    {
        // If not explicitly passed, the session ID is already handled by PHP
        $sessionId = session_id();
    }
?>

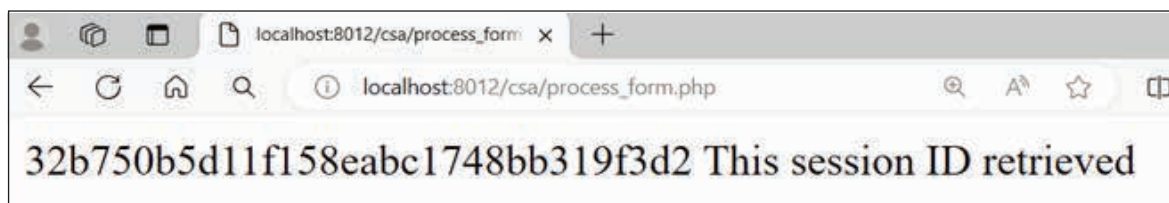
```

```

</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension



TASK 7: Setting a cookie**I Creating a session cookie**

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
session_start();
// Set the session variable
$sessionId = session_id();
echo "session started with id :".$sessionId
$_SESSION['username'] = 'JohnDoe';
// Set a cookie with the session ID
setcookie(session_name(), session_id(), time() + 60 * 60); // Expires in 1 hour
        echo "Cookie set successfully!";
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

**II Access session data on other pages**

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
session_start();
if (isset($_SESSION['username']))
{

```

```

$username = $_SESSION['username'];
echo "Welcome back, $username!";
}
else
{
    echo "Session not found!";
}
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

TASK 8: Encoding and decoding session variables

I Encoding session variables

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
// Start the session
session_start();
$username = 'JohnDoe';
//encode using base64_encode()
$encodedUsername = base64_encode($username);
$_SESSION['encoded_username'] = $encodedUsername;?>
</body>
</html>

```

3. Save the program in C:\Apache24\htdocs in a folder with .php extension
4. Run the Apache services from windows services
5. Open the browser and type the following address
http://localhost/foldername/
6. Click the php file to run and verify the output



II Decoding session variables

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
// Start the session
    session_start();

    if (isset($_SESSION['encoded_username']))
    {
        $encodedUsername = $_SESSION['encoded_username'];
        //decode using base64_decode()
        $decodedUsername = base64_decode($encodedUsername);
        echo "Decoded Username: $decodedUsername";
    }
    else
    {
        echo "Session variable 'encoded_username' not set.";
    }?>
</body>
</html>

```

3. Save the program in C:\Apache24\htdocs in a folder with .php extension
4. Run the Apache services from windows services
5. Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 9: Destroying a session

- 1 Open the text editor
- 2 Write the following codes

```

        <html>
        <body>
        <?php
// Start the session
session_start();
// Unset a session variable
unset($_SESSION['username']);
unset($_SESSION['user_email']);
// Destroy all session data
session_destroy();
echo "All session data successfully removed";
?>
    </body>
    </html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 10: Working with the query string

I Creating a query string

- 1 Open the text editor
- 2 Write the following codes

```

    <html>
    <body>
    <?php
        $data = array(
            'name' => 'John Doe',
            'age' => 30
        );
        $query_string = http_build_query($data);

```

```

$url = "http://localhost:8012/csa/querystring2.php?".$_query_string;    header("Location: $url");//
    Redirect with query string
        exit(); // Stop further script execution

?>
    </body>
    </html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder as querystring1.php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

II Processing query string

- 1 Open the text editor
- 2 Write the following codes

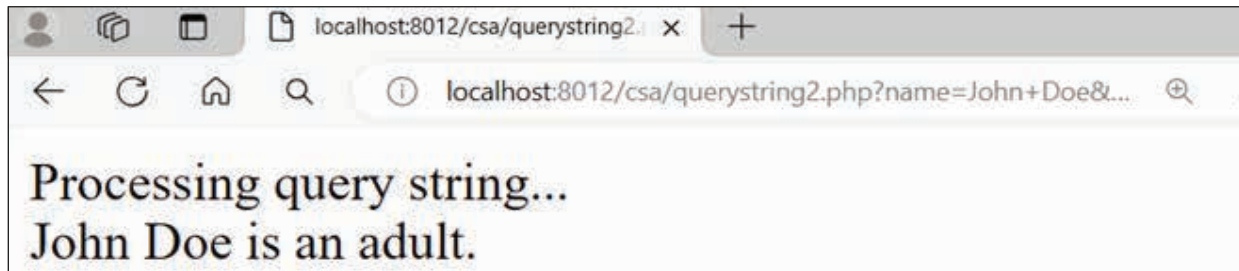
```

<html>
<body>
<?php
    echo "Processing query string...<br/> ";
    // Check if query string parameters exist
    if(isset($_GET['name']) && isset($_GET['age']))
    {
        // Retrieve and sanitize the parameters
        $name = htmlspecialchars($_GET['name']);
        $age = intval($_GET['age']); // Convert age to integer
        // Process the parameters
        if($age >= 18)
        {
            echo "$name is an adult.";
        }
    }
    else
    {
        echo "$name is a minor.";
    }
}
else
{
    // If parameters are not provided, display an error message
    echo "Please provide name and age parameters.";
}

```

```
?>
    </body>
  </html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder as querystring2.php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 11: Sending E-mail using PHP

- 1 Open the text editor
- 2 Write the following codes

Note: This code needs a functioning mail server configured in the PHP environment to send emails.

```
<html>
<body>
<?php
    $to = "recipient@example.com";//provide the actual email address
    $subject = "My Email Subject";
    $message = "This is the email body.";

    // Optional headers
    $headers = "From: sender@example.com" . "\r\n" .
        "Reply-To: reply@example.com";

    // Send the email
    if (mail($to, $subject, $message, $headers))
    {
        echo "Email sent successfully!";
    }
    else
    {
        echo "Error sending email.";
    }
}
```

```
?>  
    </body>  
    </html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
 http://localhost/foldername/
- 6 Click the php file to run and verify the output
 - 1 Provide a code example demonstrating how to set a cookie with the value that expires in 24 hours.
 - 2 Provide a code example demonstrating how to retrieve the value of a cookie named "username" set in a previous request.
 - 3 Provide a code example demonstrating how to delete a cookie named "username" that was previously set.
 - 4 Provide a code example demonstrating how to start a session and set a session variable named "myname" with the value "CSATRAINEE".
 - 5 Provide a code example demonstrating how to modify the value of the session variable "username" set in a previous request.
 - 6 Provide a code example demonstrating how to unset the session variable "username".
 - 7 Write a PHP code snippet generating a unique session ID, associating it with a session, and storing it in a cookie
 - 8 Provide examples of how to add query string parameters programmatically, either within the same page or when redirecting to another page.

EXERCISE 55 : Demonstrateon Creating and deleting a file, Reading and writing text files, working with directories in PHP, checking for existence of file, determining file size, opening a file for writing, reading, or appending, Writing Data to the file Reading characters

Objectives

At the end of this exercise you shall be able to

- create and delete files/folders in server using PHP
- read and write text files using PHP
- determine file size and checking for existence of file using PHP.

Requirements

Tools/Materials

- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apache web server
- PHP
- Text editor
- Web browser

Procedure

TASK 1: Working with directories

I Creating a Directory

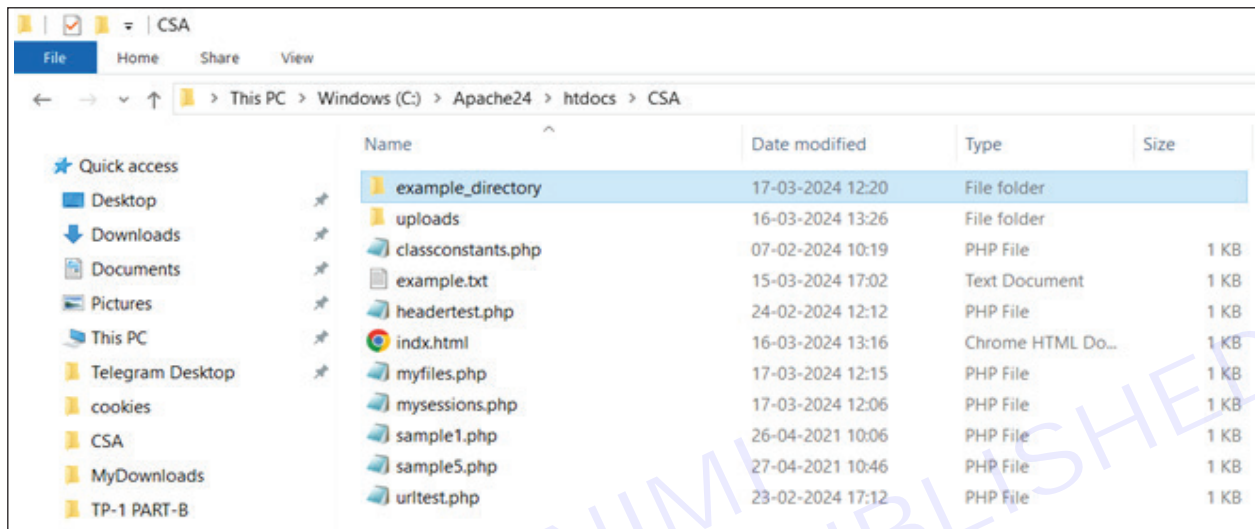
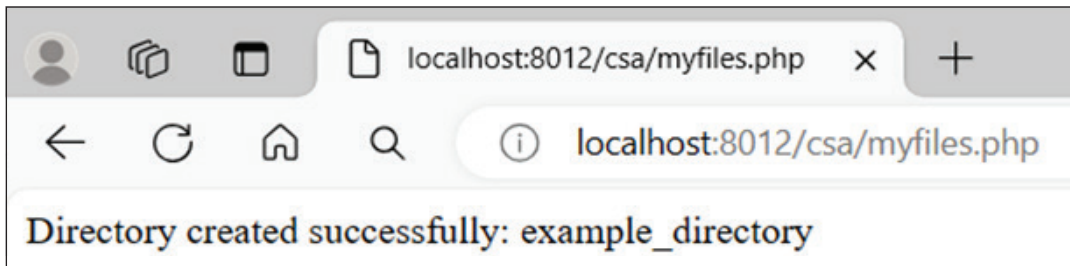
- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
<?php
```

```
$directory_path = 'example_directory';
// Check if the directory doesn't exist before attempting to create
if (!is_dir($directory_path))
{
    // Create the directory with mkdir()
    mkdir($directory_path);
    echo "Directory created successfully: $directory_path";
}
else
{
    echo "Directory already exists: $directory_path";
}
?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/

6 Click the php file to run and verify the output



II Listing Files and Directories in a Directory

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
    $directory_path = 'example_directory';
    // Check if the directory exists before listing its contents
    if (is_dir($directory_path))
    {
        // Open the directory
        $directory_handle = opendir($directory_path);
        // Read and display the contents
        echo "Contents of $directory_path:\n";
        while (($file = readdir($directory_handle)) !== false)
        {
            echo $file . "\n";
        }
        // Close the directory handle
    }

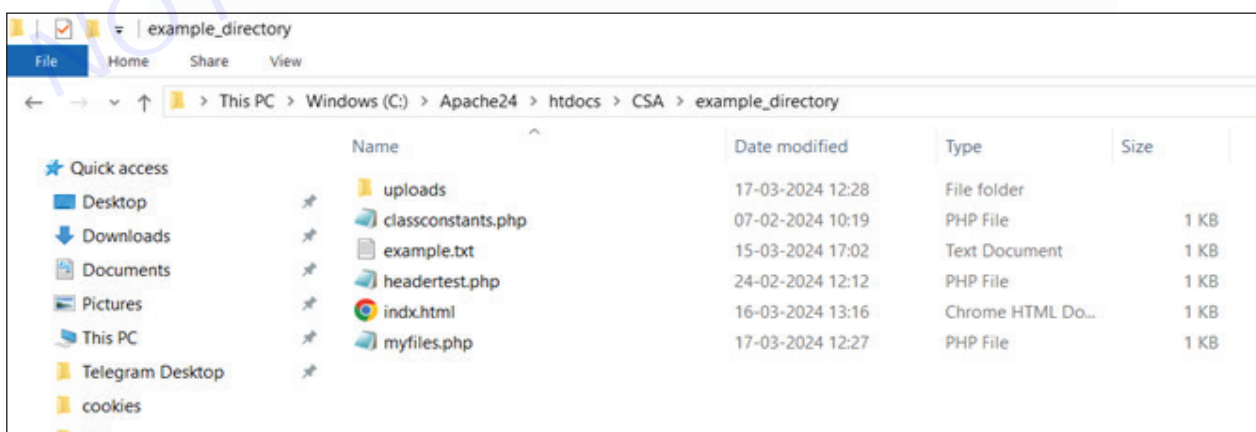
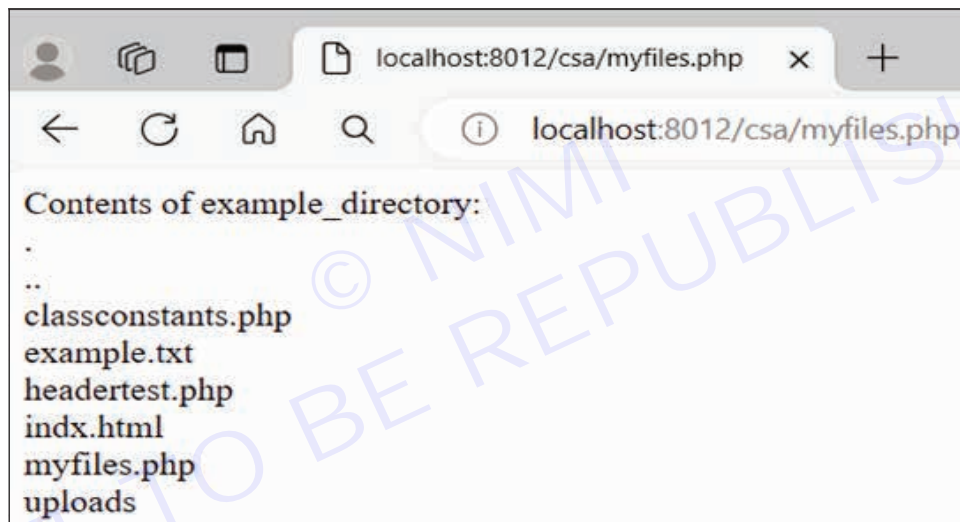
```

```

        closedir($directory_handle);
    }
    else
    {
        echo "Directory not found: $directory_path";
    }
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



III Deleting a Directory and Its Contents

- 1 Open the text editor

2 Write the following codes

```

<html>
<body>
<?php
$directory_path = 'example_directory';
// Check if the directory exists before attempting to delete
if (is_dir($directory_path))
{
    // Open the directory
    $directory_handle = opendir($directory_path);
    // Iterate over the contents and delete files and subdirectories
    while (($file = readdir($directory_handle)) !== false)
    {
        if ($file != "." && $file != "..")
        {
            // Delete files and subdirectories
            unlink($directory_path . '/' . $file);
        }
    }
    // Close the directory handle
    closedir($directory_handle);
    // Delete the directory itself
    rmdir($directory_path);
    echo "Directory and its contents deleted successfully: $directory_path";
}
else
{
    echo "Directory not found: $directory_path";
}
?>
</body>
</html>

```

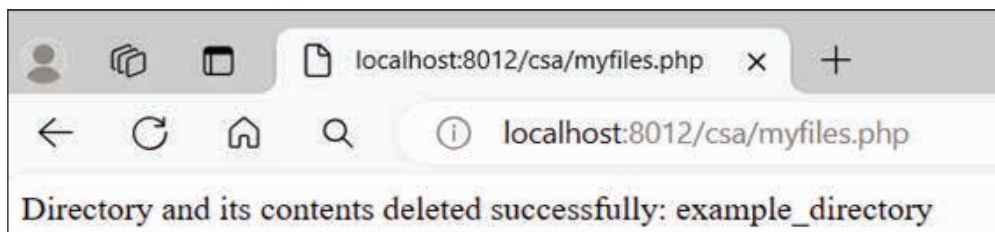
3 Save the program in C:\Apache24\htdocs in a folder with .php extension

4 Run the Apache services from windows services

5 Open the browser and type the following address

<http://localhost/foldername/>

6 Click the php file to run and verify the output



TASK 2: Creating a file

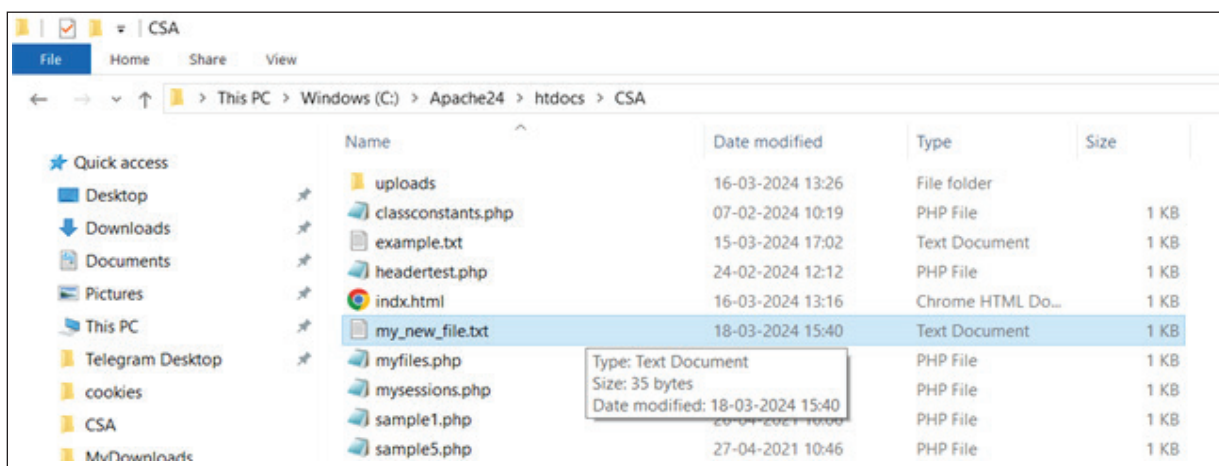
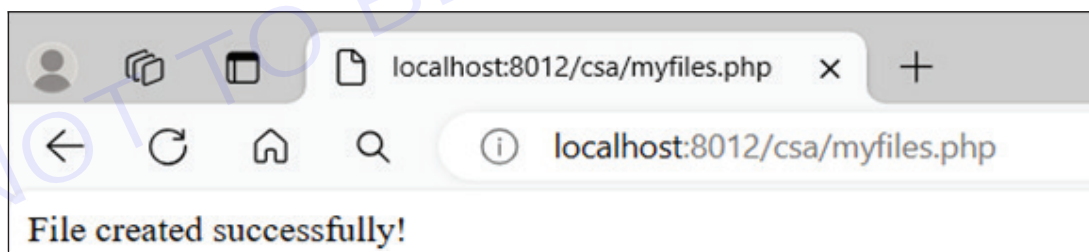
- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
    // Specify the file name and path:
    $filename = "my_new_file.txt"; // Modify with desired name and location
    // Open the file in write mode (creates the file if it doesn't exist):
    $handle = fopen($filename, "w") or die("Unable to open file!");
    // Write content to the file:
    $content = "This is the content of my new file!";
    fwrite($handle, $content);
    // Close the file:
    fclose($handle);
    // Optional: Display a success message:
    echo "File created successfully!";
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 3: Checking for existence of file

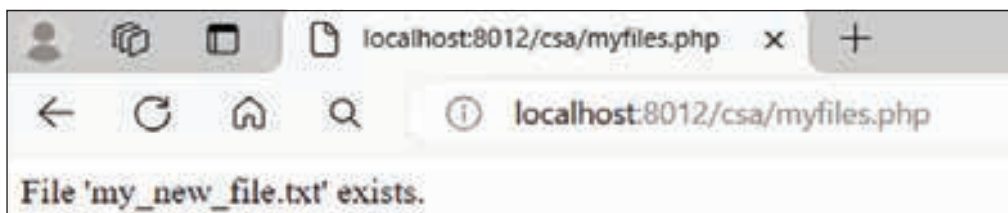
- 1 Open the text editor
- 2 Write the following codes

```

    <html>
    <body>
    <?php
    // Specify the file name
    $fileName = 'my_new_file.txt';
    // Check if the file exists
    if (file_exists($fileName))
    {
        // File exists, display a success message
        echo "File '$fileName' exists.";
    }
    else
    {
        // File does not exist, display an error message
        echo "Error: File '$fileName' does not exist.";
    }
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

**TASK 4 : Determining file size**

- 1 Open the text editor
- 2 Write the following codes


```

<html>
<body>

```

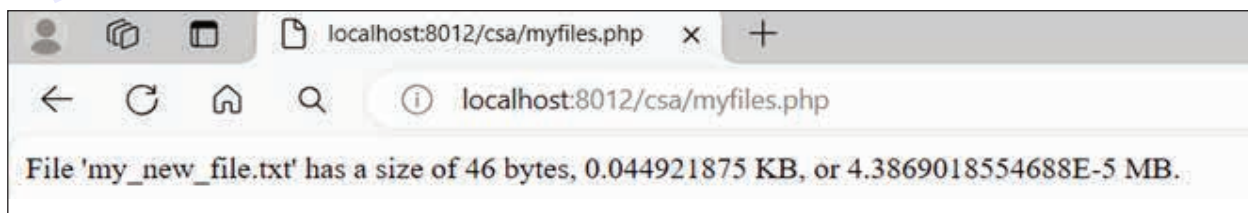
```

        // Specify the file name
        $fileName = 'my_new_file.txt';

        // Check if the file exists
        if (file_exists($fileName))
        {
            // Get the file size in bytes
            $fileSizeBytes = filesize($fileName);
            // Convert the file size to kilobytes or megabytes
            $fileSizeKB = $fileSizeBytes / 1024; // Bytes to Kilobytes
            $fileSizeMB = $fileSizeKB / 1024; // Kilobytes to Megabytes
            // Display the file size
            echo "File '$fileName' has a size of $fileSizeBytes bytes,
            $fileSizeKB KB, or $fileSizeMB MB.";
        }
        else
        {
            // Display an error message if the file does not exist
            echo "Error: File '$fileName' does not exist.";
        }
    }
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 5: Opening a File for reading

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php

```

```

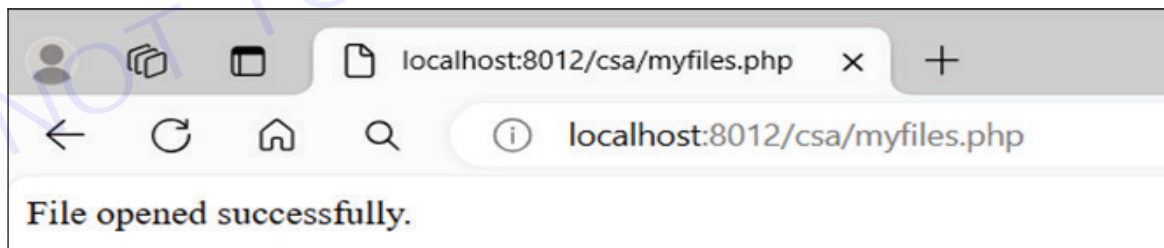
// File path
$filePath = 'my_new_file.txt';

// Open the file in read mode
$fileHandle = fopen($filePath, 'r');

// Check if file opened successfully
if ($fileHandle) {
    // File opened successfully, perform operations here
    echo "File opened successfully.";
    // Close the file handle
    fclose($fileHandle);
} else {
    // File opening failed, handle error
    echo "Failed to open file.";
}
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 6: Opening a File for writing text content

- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php

```



```

        // Specify the file name
        $fileName = 'example.txt';
        // Open the file in write mode ('w')
        $fileHandle = fopen($fileName, 'w');

        // Check if the file is opened successfully
        if ($fileHandle)
        {
            // Write content to the file
            fwrite($fileHandle, 'This is a line of text.');
```

// Close the file handle
fclose(\$fileHandle);

```

        echo "File '$fileName' has been written successfully.";
    }
    else
    {
        // Display an error message if the file cannot be opened
        echo "Error: Unable to open the file for writing.";
    }
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 7: Opening a File for appending content

- 1 Open the text editor

2 Write the following codes

```

<html>
<body>
<?php
    // Specify the file name
    $fileName = 'example.txt';
    // Open the file in append mode ('a'), create the file if it doesn't exist
    $fileHandle = fopen($fileName, 'a');
    // Check if the file is opened successfully
    if ($fileHandle)
    {
        // Append content to the file
        fwrite($fileHandle, 'This is a line of text appended to the file. ');
        // Close the file handle
        fclose($fileHandle);
        echo "File '$fileName' has been appended successfully.";
    }
    else
    {
        echo "Error: Unable to open or create the file for appending.";
    }
?>
</body>
</html>

```

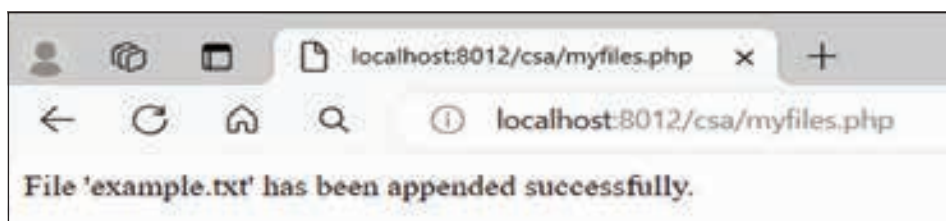
3 Save the program in C:\Apache24\htdocs in a folder with .php extension

4 Run the Apache services from windows services

5 Open the browser and type the following address

<http://localhost/foldername/>

6 Click the php file to run and verify the output



TASK 8: Writing Data to the file

1 Open the text editor

2 Write the following codes

```

<html>
<body>

```

```

<?php

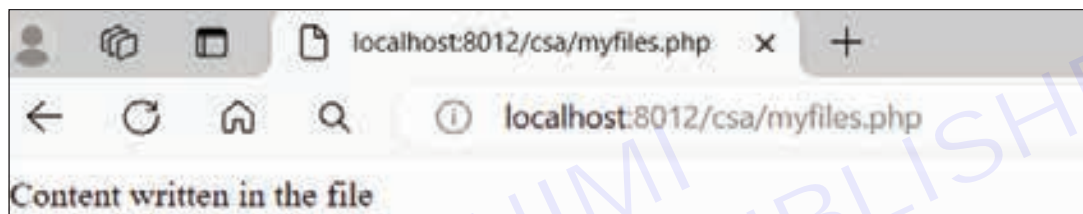
$file="info.txt";
    $content="hai you are welcome";
file_put_contents($file,$content) or die("Error not opening");
    echo "Content written in the file";

?>

</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 9: Reading characters

- I Reading characters using file_get_contents function

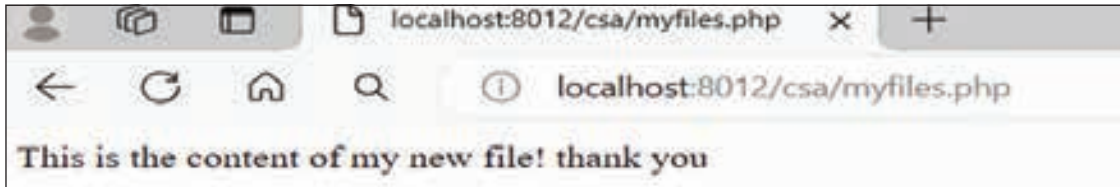
- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
    $file='my_new_file.txt';
    if(file_exists($file))
    {
        $content=file_get_contents($file) or die("Error file opening not possible ");
        echo $content;
    }
    else
    {
        echo "ERROR: File does not exist.";
    }
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
 http://localhost/foldername/
- 6 Click the php file to run and verify the output



II Reading characters using file function

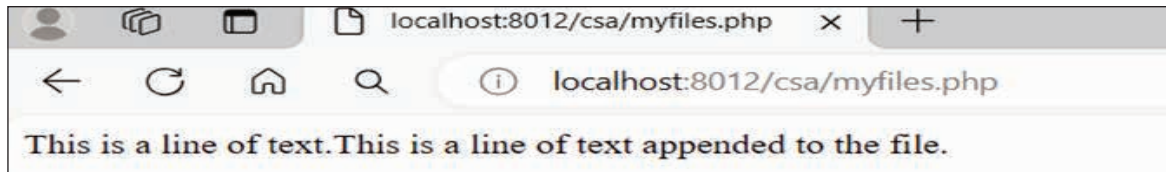
- 1 Open the text editor
2. Write the following codes

```

<html>
<body>
<?php
    $file="example.txt";
    if(file_exists($file))
    {
        $content=file($file) or die("Error file opening not possible ");
        foreach( $content as $line)
        {
            echo $line;
        }
    }
    else
    {
        echo "ERROR: File does not exist.";
    }
?>

</body>
</html>
    
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
 http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 10: Deleting a File

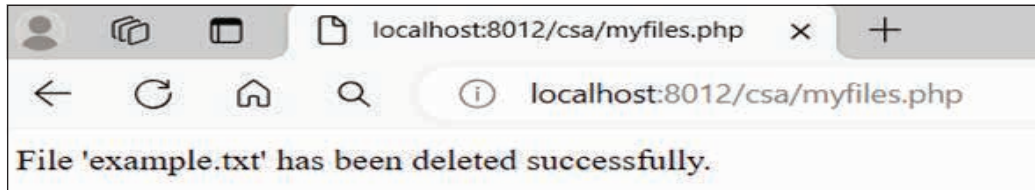
- 1 Open the text editor
- 2 Write the following codes

```

<html>
<body>
<?php
// Specify the file name
$fileName = 'example.txt';
    // Check if the file exists before attempting to delete
    if (file_exists($fileName))
    {
        // Attempt to delete the file
        if (unlink($fileName))
        {
            echo "File '$fileName' has been deleted successfully.";
        }
        else
        {
            echo "Error: Unable to delete the file.";
        }
    }
    else
    {
        echo "Error: File '$fileName' does not exist.";
    }
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



- 1 Provide a code example demonstrating how to create a directory named “uploads” within the current working directory.
- 2 Provide a code example demonstrating how to list the contents of a directory named “images”.
- 3 Provide a code example demonstrating how to delete a directory named “temp” and its contents.
- 4 Provide a code example demonstrating how to open a file named “example.txt” in read mode.
- 5 Provide a code example demonstrating how to read each line from a file handle using the fgets() function.
- 6 Provide a code example demonstrating how to use the file_get_contents()
- 7 Provide a code example demonstrating how to use the fwrite() function to write a string to a file handle.
- 8 Provide a code example demonstrating how to open a file in append mode (“a”) and write text content to it.
9. Provide a code example demonstrating how to delete a file named “example.txt” from the filesystem.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 56 : Work With Forms - Super global variables the server array A script to acquire user input, importing user input Accessing user input, Combine HTML and PHP code., using hidden feilds

Objectives

At the end of this exercise you shall be able to:

- acquire user input from a HTML form using PHP
- combine HTML and PHP code in single file
- create hidden feilds and access it using PHP.

Requirements

Tools/Materials

- computer/Laptop with latest configuration
- operating system: windows 10 or 11
- apache web server
- PHP
- Text editor
- web browser

Procedure

TASK 1: Super global variables the server array

- 1 Open the text editor
- 2 Write the following codes

```

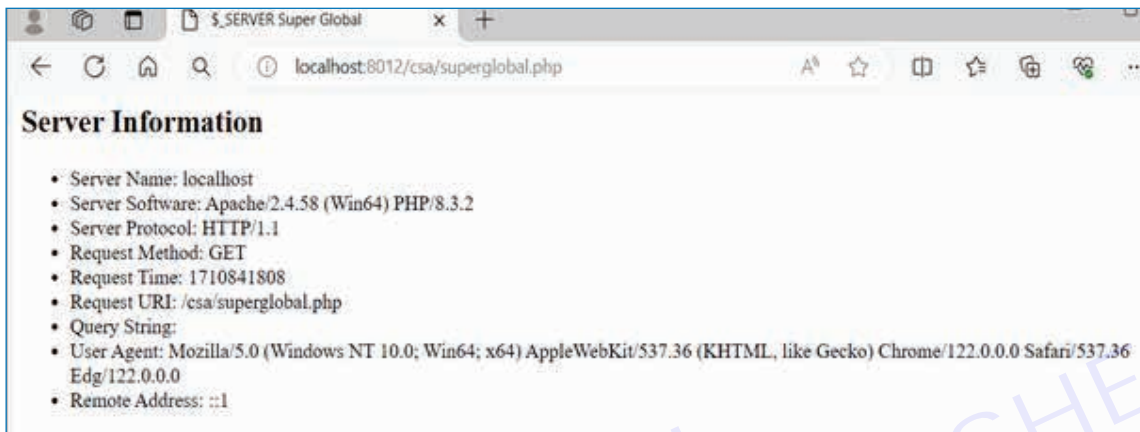
<html >
<head>
  <title>$_SERVER Super Global</title>
</head>
<body>
<h2>Server Information</h2>
<ul>
  <li>Server Name: <?php echo $_SERVER['SERVER_NAME']; ?></li>
  <li>Server Software: <?php echo $_SERVER['SERVER_SOFTWARE']; ?></li>
  <li>Server Protocol: <?php echo $_SERVER['SERVER_PROTOCOL']; ?></li>
  <li>Request Method: <?php echo $_SERVER['REQUEST_METHOD']; ?></li>
  <li>Request Time: <?php echo $_SERVER['REQUEST_TIME']; ?></li>
  <li>Request URI: <?php echo $_SERVER['REQUEST_URI']; ?></li>
  <li>Query String: <?php echo $_SERVER['QUERY_STRING']; ?></li>
  <li>User Agent: <?php echo $_SERVER['HTTP_USER_AGENT']; ?></li>
  <li>Remote Address: <?php echo $_SERVER['REMOTE_ADDR']; ?></li>
</ul>

```

```
</body>
```

```
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 2: Acquiring user input using an HTML form

I Create an HTML Form

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
<form action="process_form.php" method="POST">
Name: <input type="text" name="name"><br>
Email: <input type="email" name="email"><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .html extension

II Create a PHP script that acquires user input

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
<?php
```



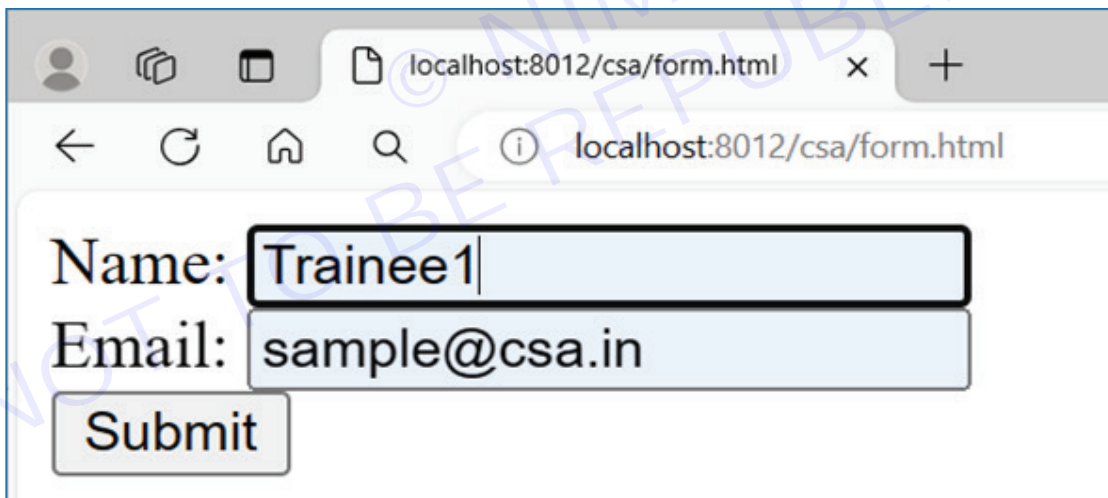
```

if ($_SERVER['REQUEST_METHOD'] == 'POST')
{
// Access form data using $_POST array
$name = $_POST['name'];
$email = $_POST['email'];
// Display a confirmation message
echo "Thank you, " . $name . " for your submission!";
}

?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the .html file to run and verify the output



TASK 3: Importing user input and accessing user input**I Create an HTML Form**

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<head>
  <title>PHP User Input</title>
</head>
<body>
<h2>Enter your name:</h2>
<form action="process.php" method="get">
  <input type="text" name="name" placeholder="Enter your name">
  <button type="submit">Submit</button>

</form>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder as index.html

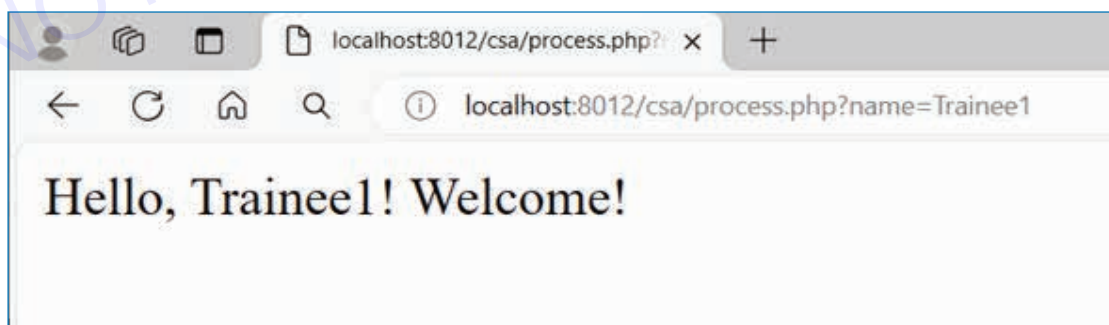
II Create a PHP script that acquires user input

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
<?php
if ($_SERVER["REQUEST_METHOD"] == "GET")
{
  // Check if the name field is set and not empty
  if (isset($_GET["name"]) && !empty($_GET["name"]))
  {
    $name = $_GET["name"];
    // Display a greeting message
    echo "Hello, $name! Welcome!";
  }
  else
  {
    // If name field is empty, display an error message
    echo "Please enter your name!";
  }
}
```

```
}  
else  
{  
    // If the request method is not POST, redirect back to the form  
    header("Location: index.html");  
    exit();  
}  
?>  
</body>  
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
<http://localhost/foldername/>
- 6 Click the index.html file to run and verify the output



TASK 4: Combine HTML and PHP code

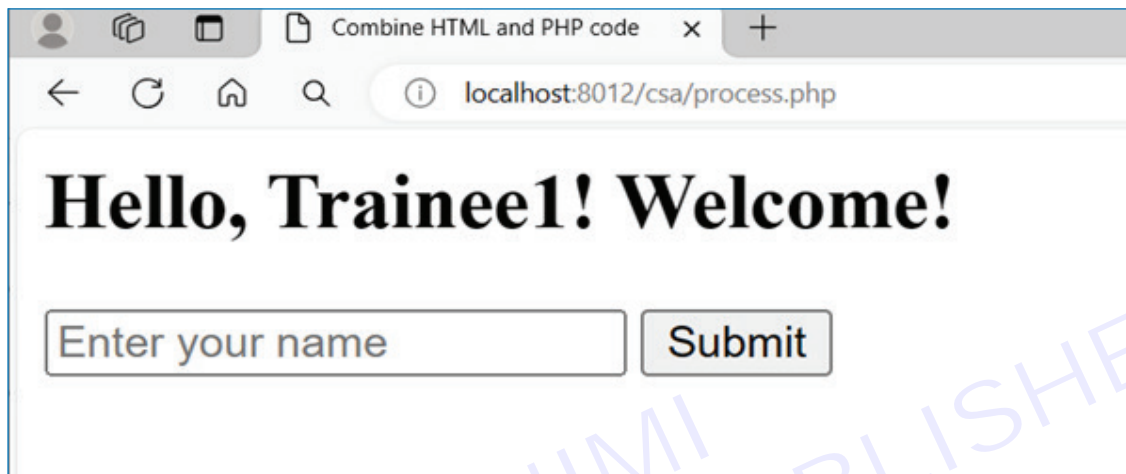
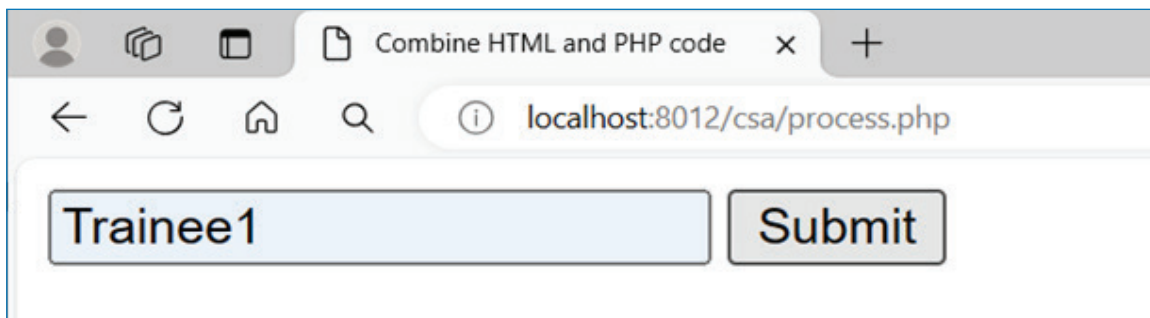
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
  <title Combine HTML and PHP code </title>
</head>
<body>
  <?php
    // Check if the form is submitted
    if ($_SERVER["REQUEST_METHOD"] == "POST")
    {
      // Check if the name field is set and not empty
      if (isset($_POST["name"]) && !empty($_POST["name"]))
      {
        $name = $_POST["name"];
        // Display a greeting message
        echo "<h2>Hello, $name! Welcome!</h2>";
      }
    }
    else
    {
      // If name field is empty, display an error message
      echo "<h2>Please enter your name!</h2>";
    }
  }
  ?>
  <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>"
    method="post">
    <input type="text" name="name" placeholder="Enter your name">
    <button type="submit">Submit</button>
  </form>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 5 : Using hidden fields

I Creating Hidden Fields

- 1 Open the text editor
- 2 Write the following codes

```
<html>
<body>
<form method="post" action="process.php">
    <input type="hidden" name="user_id" value="123">
    <input type="text" name="username">
    <button type="submit">Submit</button>
</form>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .html extension

II Accessing Hidden Field Values

- 1 Open the text editor
- 2 Write the following codes

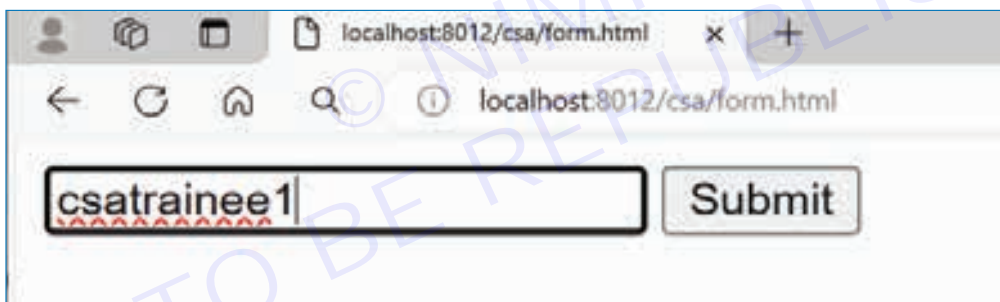
```
<html>
```

```

        <body>
        <?php
        if ($_SERVER['REQUEST_METHOD'] == 'POST')
        {
            $userId = $_POST['user_id'];
            $username = $_POST['username'];
            // Process the values
            echo "User ID: $userId, Username: $username";
        }
        ?>
    </body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the .html file to run and verify the output



- 1 Provide a code example to determine the request method (GET, POST, etc.) used by a client in PHP.
- 2 Provide examples of accessing common request headers in PHP such as \$_SERVER['HTTP_USER_AGENT'] and \$_SERVER['HTTP_REFERER'].
- 3 Provide a code example demonstrating how to create a simple HTML form with a text input field and retrieve the user's input using the \$_GET superglobal.
- 4 Provide a code example of a PHP file that contains an HTML form for acquiring user input.
- 5 Provide a code example of setting hidden field values in one page and retrieving them in another page using the \$_POST or \$_GET superglobals.

EXERCISE 57 : Demonstrate redirecting the user, File upload and scripts

Objectives

At the end of this exercise you shall be able to

- redirect user to one page to other using PHP.

Requirements

Tools/Materials

- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apache web server
- PHP
- Text editor
- Web browser

Procedure

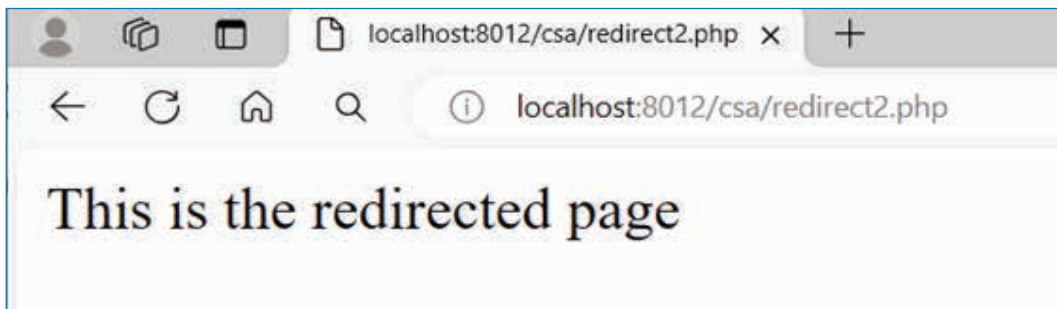
TASK 1: Redirecting a user

I Using header() function

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
  <title >Using header() function </title>
</head>
<body>
  <?php
    // Redirect to another page after 5 seconds
    header("Location:destination-page.php");
    exit; // Make sure to exit after sending the header
  ?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



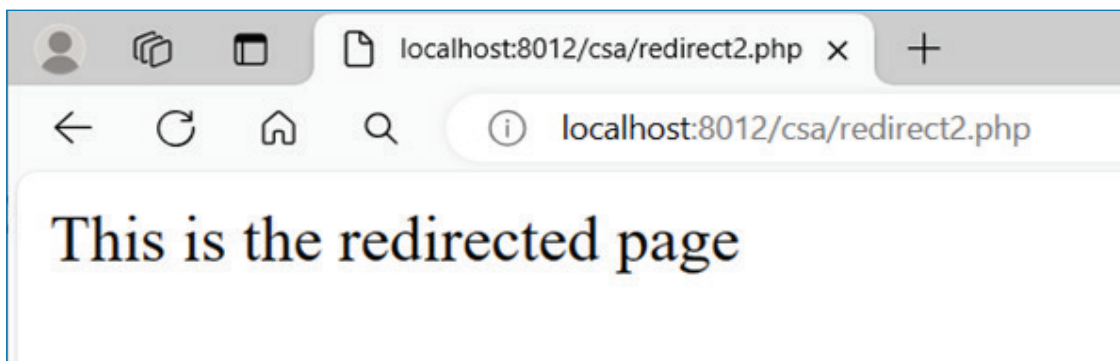
II Adding a delay before redirection

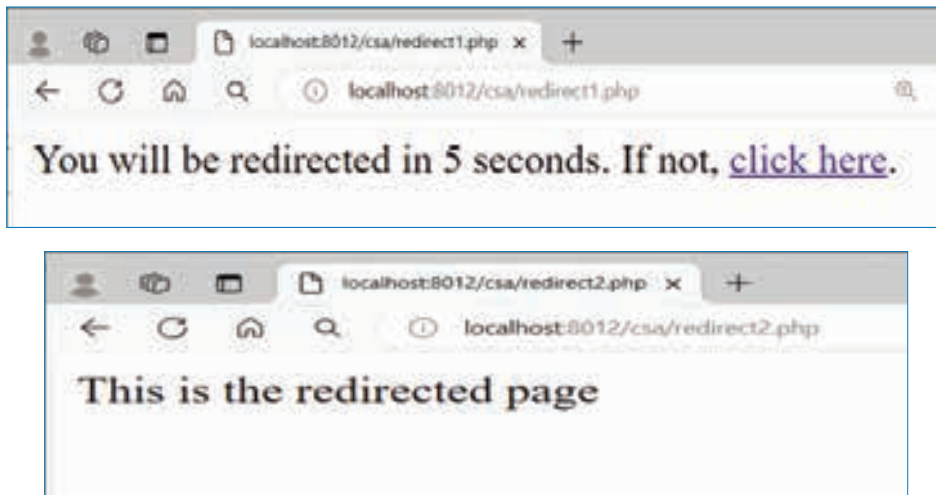
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
  <title >Adding a delay before redirection </title>
</head>
<body>
  <?php
    sleep(5);
    header("Location:destination-page.php");
    exit;
  ?>
</body>
</html>
  
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
<http://localhost/foldername/>
- 6 Click the php file to run and verify the output





- 1 Provide a code example of how to redirect a user to another URL with a delay of 5 seconds

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 58 : Demonstrate Regular Expressions - The basic regular expressions, PCRE, Matching patterns, Finding matches, Replace patterns, Modifiers, Breakup Strings

Objectives

At the end of this exercise you shall be able to:

- use the basic regular expressions in PHP
- use PCRE function in PHP.

Requirements

Tools/Materials

- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apache web server
- PHP
- Text editor
- Web browser

Procedure

TASK 1: Working with basic regular expressions

I Matching a single character

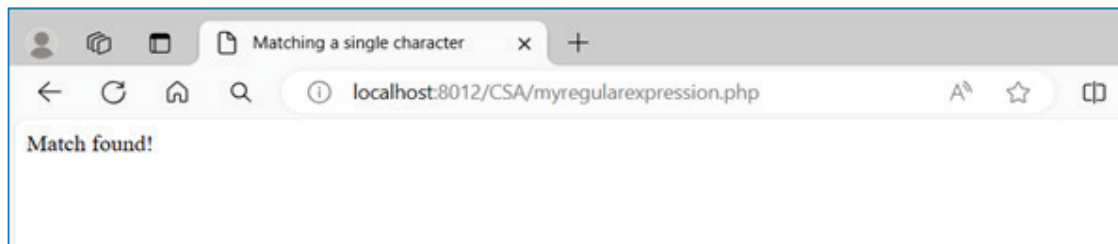
- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
<title > Matching a single character </title>
</head>
<body>
<?php
$string = "Hello, world!";
$pattern = "/w/"; // Matches any word character (alphanumeric or underscore)
if (preg_match($pattern, $string))
{
echo "Match found!"; // Output: Match found!
}
?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address

<http://localhost/foldername/>

- Click the php file to run and verify the output



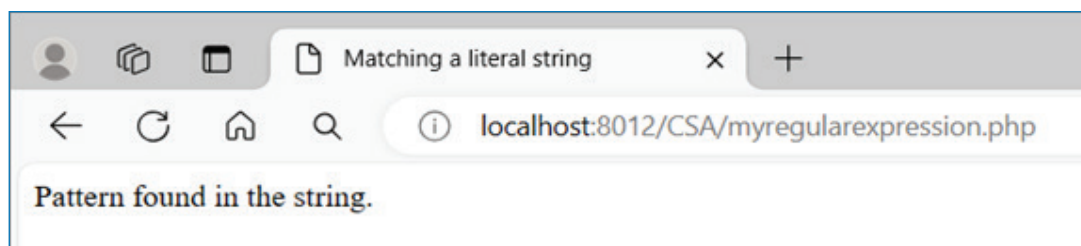
II Matching a literal string

- Open the text editor
- Write the following codes

```

<html >
<head>
  <title> Matching a literal string </title>
</head>
<body>
  <?php
    $string = "Hello, World!";
    $pattern = "/World/";
    if (preg_match($pattern, $string))
    {
        echo "Pattern found in the string.";
    }
    else
    {
        echo "Pattern not found in the string.";
    }
  ?>
</body>
</html>
  
```

- Save the program in C:\Apache24\htdocs in a folder with .php extension
- Run the Apache services from windows services
- Open the browser and type the following address
<http://localhost/foldername/>
- Click the php file to run and verify the output



III Matching any character from a set

- 1 Open the text editor
- 2 Write the following codes

```

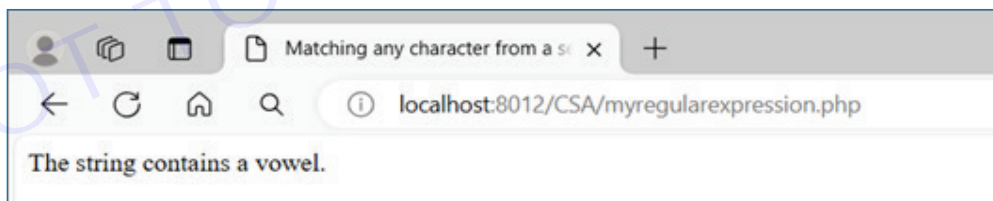
<html >
<head>
  <title> Matching any character from a set </title>
</head>
<body>
<?php
  $string = "The color is blue.";
  $pattern = "[aeiou]"; // Matches any vowel

  if (preg_match($pattern, $string))
  {
    echo "The string contains a vowel.";
  }

?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

**IV Matching any single character using the dot wildcard**

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
  <title> Matching any single character using the dot wildcard </title>
</head>
<body>
  <?php
    $string = "cat";

```

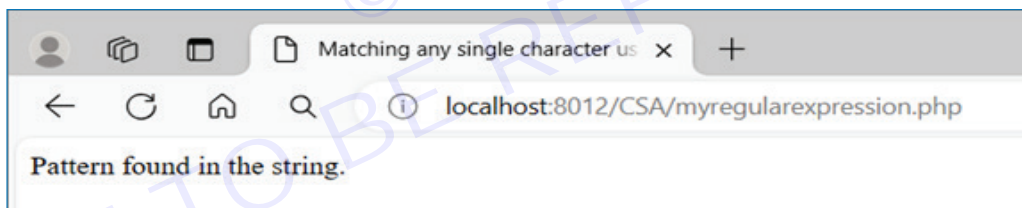
```

$pattern = "/c.t/";

if (preg_match($pattern, $string))
{
    echo "Pattern found in the string.";
}
else
{
    echo "Pattern not found in the string.";
}
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



V Matching any digit using 'd'

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Matching any digit using 'd' </title>
</head>
<body>
<?php
$string = "abc123xyz";
$pattern = "\d/";

if (preg_match($pattern, $string))
{

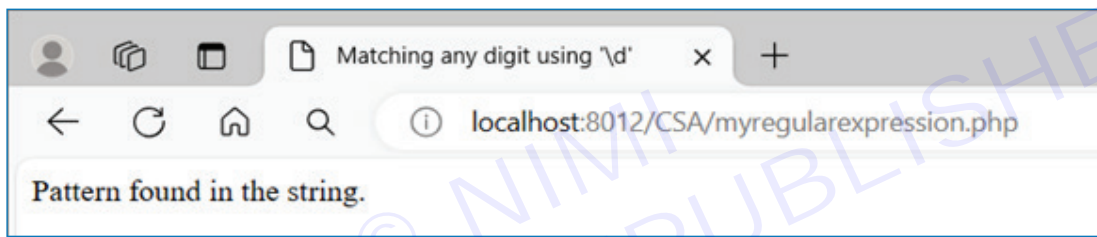
```

```

        echo "Pattern found in the string.";
    }
    else
    {
        echo "Pattern not found in the string.";
    }
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



VI Matching repeating characters using {}

- 1 Open the text editor
- 2 Write the following codes

```

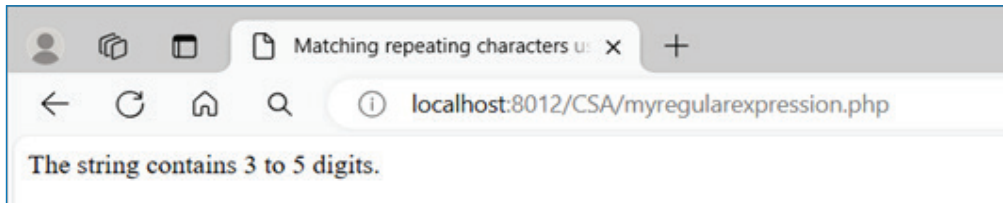
<html >
<head>
<title> Matching repeating characters using {} </title>
</head>
<body>
<?php
$string = "12345";
$pattern = "\d{3,5}"; // Matches 3 to 5 digits

if (preg_match($pattern, $string))
{
    echo "The string contains 3 to 5 digits.";
}
?>

</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



VII Matching specific number of occurrences using {}

- 1 Open the text editor
- 2 Write the following codes

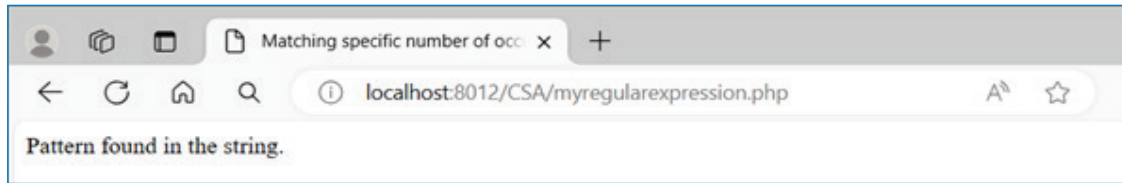
```

<html >
<head>
    <title> Matching specific number of occurrences using {} </title>
</head>
<body>
    <?php
        $string = "hellooooo";
        $pattern = "/o{5}/";

        if (preg_match($pattern, $string))
        {
            echo "Pattern found in the string.";
        }
        else
        {
            echo "Pattern not found in the string.";
        }
    ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



VIII Matching one or more occurrences using +

- 1 Open the text editor
- 2 Write the following codes

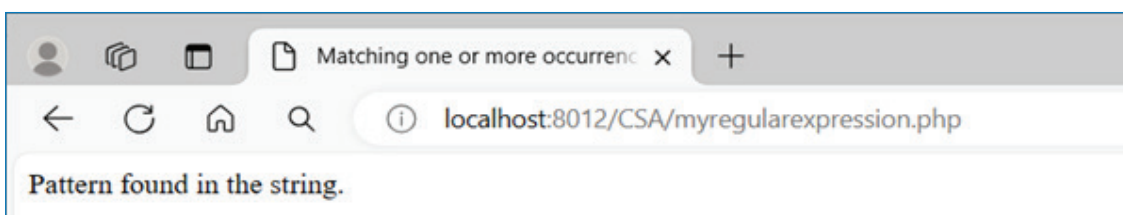
```

<html >
<head>
    <title> Matching one or more occurrences using + </title>
</head>
<body>
    <?php
        $string = "aaabbbccc";
        $pattern = "/a+/";

        if (preg_match($pattern, $string))
        {
            echo "Pattern found in the string.";
        }
        else
        {
            echo "Pattern not found in the string.";
        }
    ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



IX Matching zero or more occurrences using *

1. Open the text editor
2. Write the following codes

```

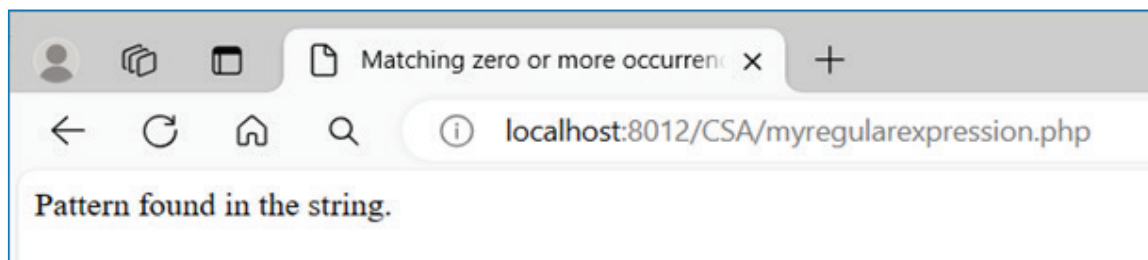
<html >
<head>
  <title> Matching zero or more occurrences using * </title>
</head>
<body>
  <?php
  $string = "abbbbbbbc";
  $pattern = "/ab*c"/;

  if (preg_match($pattern, $string))
  {
    echo "Pattern found in the string.";
  }
  else
  {
    echo "Pattern not found in the string.";
  }
?>

</body>
</html>

```

3. Save the program in C:\Apache24\htdocs in a folder with .php extension
4. Run the Apache services from windows services
5. Open the browser and type the following address
http://localhost/foldername/
6. Click the php file to run and verify the output



X Matching word boundaries

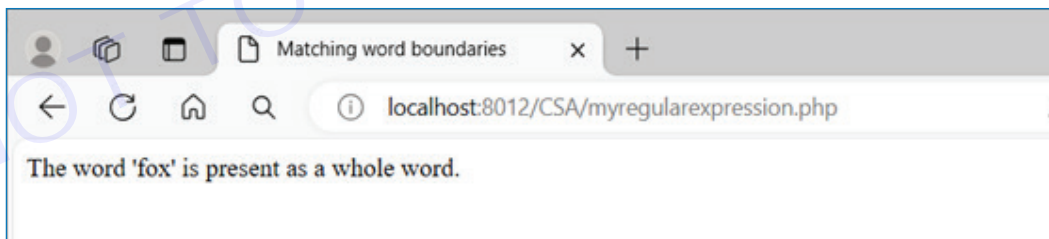
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
    <title> Matching word boundaries </title>
</head>
<body>
    <?php
        $string = "The quick brown fox jumps over the lazy dog.";
        $pattern = "\bfox\b/"; // Matches "fox" as a whole word
        if (preg_match($pattern, $string))
        {
            echo "The word 'fox' is present as a whole word.";
        }
    ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

**TASK 2 : Using modifiers****I i (case-insensitive) modifier**

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
    <title> i (case-insensitive) modifier </title>
</head>
<body>

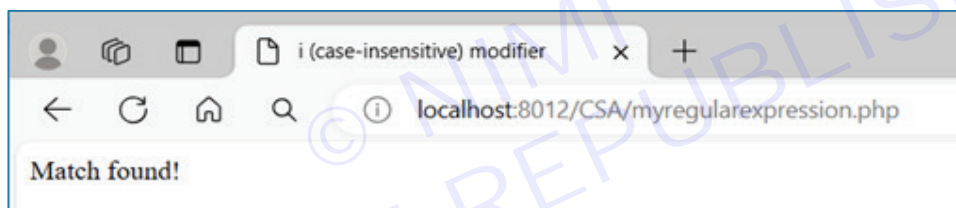
```

```

<?php
    $string = "Hello World";
if (preg_match("/hello/i", $string)) {
    echo "Match found!";
} else {
    echo "No match found!";
}
// Output: Match found!
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



II m (multiline mode) modifier

- 1 Open the text editor
- 2 Write the following codes

```

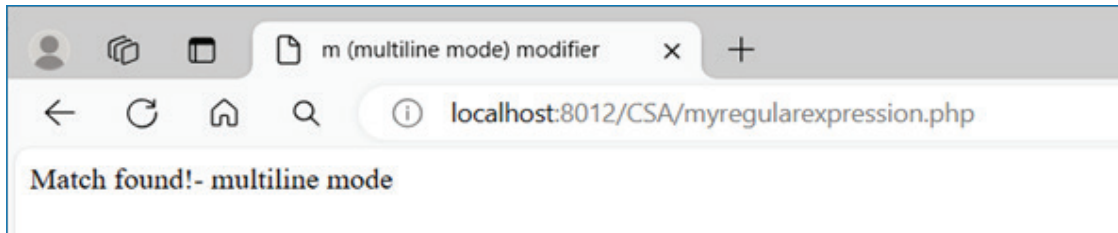
<html >
<head>
    <title> m (multiline mode) modifier </title>
</head>
<body>
    <?php
        $string = "First line\nSecond line";
if (preg_match("/^Second/m", $string)) {
    echo "Match found!- multiline mode ";
} else {
    echo "No match found!";
}
// Output: Match found!
?>

```

```
</body>
```

```
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

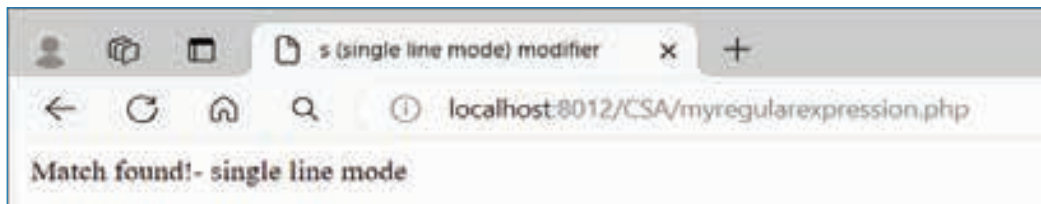


III s (single line mode) modifier

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
  <title> s (single line mode) modifier </title>
</head>
<body>
  <?php
    $string = "First line\nSecond line";
    if (preg_match("/First.*line/s", $string)) {
      echo "Match found!- single line mode ";
    } else {
      echo "No match found!";
    }
  // Output: Match found!
  ?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



IV x (extended) modifier

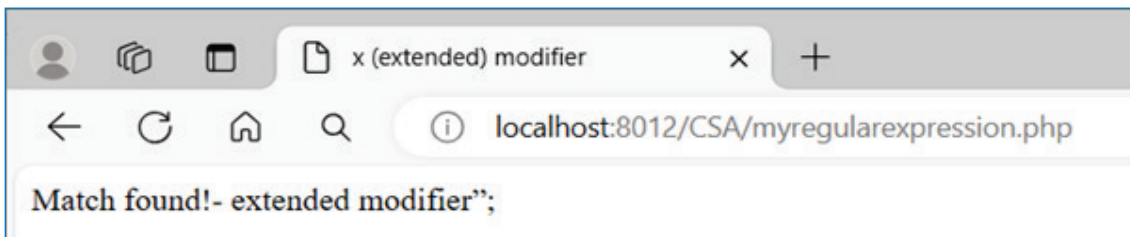
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
  <title> x (extended) modifier </title>
</head>
<body>
  <?php
    $pattern = “
        /
        ^          # Start of the string
        \d{3}      # Match exactly 3 digits
        -         # Match a hyphen
        \d{4}      # Match exactly 4 digits
        $         # End of the string
        /x
    “;

    $string = “123-4567”;
    if (preg_match($pattern, $string)) {
        echo “Match found!- extended modifier”;
    “;
    } else {
        echo “No match found!”;
    }
    // Output: Match found!
    ?>
  </body>
</html>
  
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 3 : Using PCRE Functions

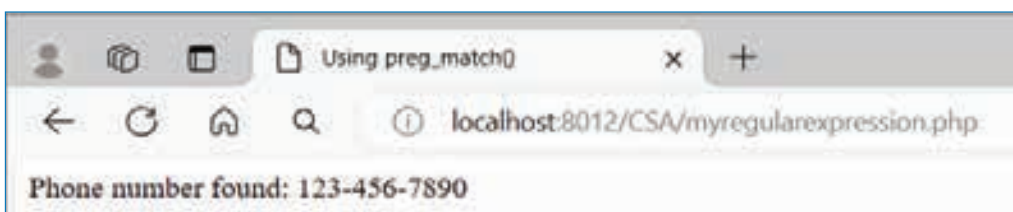
I Using preg_match() for Simple Matching

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
  <title> Using preg_match() </title>
</head>
<body>
  <?php
    $string = "My phone number is 123-456-7890.";
    $pattern = "\b\d{3}-\d{3}-\d{4}\b/";
    if (preg_match($pattern, $string, $matches))
    {
        echo "Phone number found: " . $matches[0]; // Output: 123-456-7890
    }
    else
    {
        echo "No phone number found.";
    }
  ?>
</body>
</html>
  
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

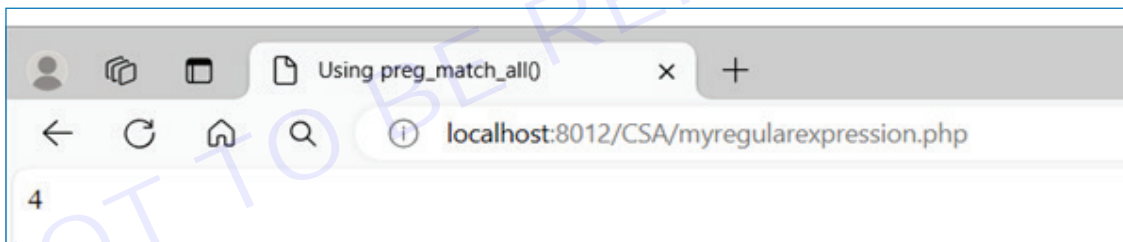


II Using preg_match_all() for Multiple Matches

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
  <title> Using preg_match_all()</title>
</head>
<body>
  <?php
    $str = "The rain in SPAIN falls mainly on the plains.";
    $pattern = "/ain/i";
    echo preg_match_all($pattern, $str);
  ?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



III Using preg_replace() for Replacement

- 1 Open the text editor
- 2 Write the following codes

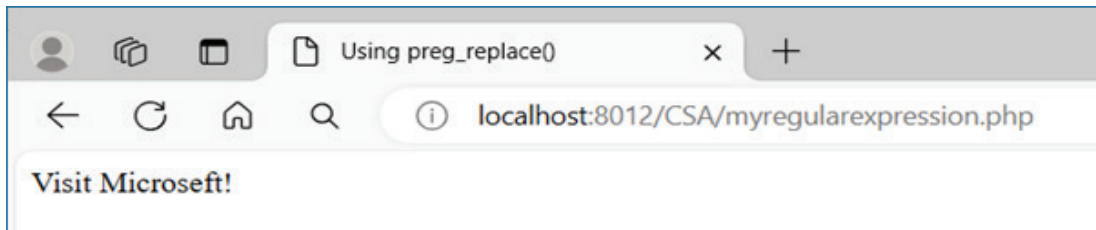
```
<html >
<head>
  <title> Using preg_replace() </title>
</head>
<body>
  <?php
    $str = "Visit Microseft!";
    $pattern = "/microsoft/i";
    echo preg_replace($pattern, "Google", $str);
```

```

    ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



IV Using preg_split() for splits a string into substrings

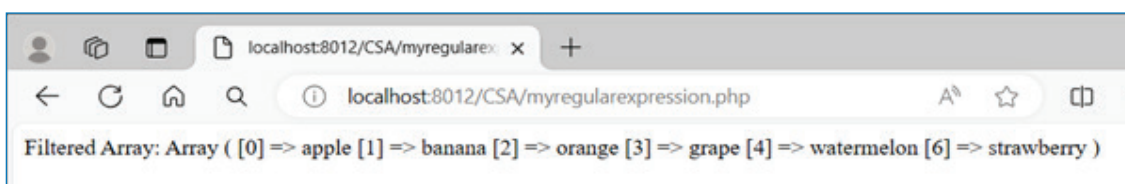
- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
<title> Using preg_split()</title>
</head>
<body>
<?php
// String to split
$string = "Hello, World! This is a test string.";
// Split the string by spaces or punctuation marks
$words = preg_split("/[\s,]+/", $string);
// Display the resulting array
print_r($words);
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



V Using preg_grep() for filtering elements of an array

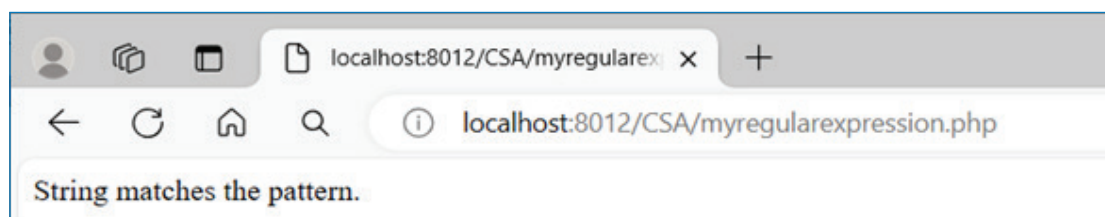
- 1 Open the text editor
- 2 Write the following codes

```

    <html >
  <head>
    <title> Using preg_grep() </title>
  </head>
  <body>
    <?php
// Sample array containing strings
$array = array(
    'apple',
    'banana',
    'orange',
    'grape',
    'watermelon',
    'kiwi',
    'strawberry'
);
// Regular expression pattern to match fruits containing 'a' followed by any character
$pattern = '/a./';
// Use preg_grep() to filter array elements based on the pattern
$filteredArray = preg_grep($pattern, $array);
// Output the filtered array
echo "Filtered Array:\n";
print_r($filteredArray);
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

**VI. Using preg_quote() for escapes special characters in a string**

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
    <title> Using preg_quote() </title>
</head>
<body>
    <?php
// String containing characters to be quoted
$string = "Escaping: .\+*?[^]{}()$=!|:-";
// Quote the string to escape regular expression characters
$quotedString = preg_quote($string);
// Regular expression pattern containing the quoted string
$pattern = '/^ . $quotedString . $/';
// Test string to match against the pattern
$testString = "Escaping: .\+*?[^]{}()$=!|:-";
// Perform regular expression matching
if (preg_match($pattern, $testString)) {
    echo "String matches the pattern.\n";
} else {
    echo "String does not match the pattern.\n";
}
?>
</body>
</html>

```

3 Save the program in C:\Apache24\htdocs in a folder with .php extension

4 Run the Apache services from windows services

5 Open the browser and type the following address
http://localhost/foldername/

6 Click the php file to run and verify the output

- 1 Provide an example of using a regular expression to match a single character in PHP
- 2 Provide an example of a regular expression pattern in PHP that matches any digits 0 to 9?
- 3 Provide an example of exclude specific characters from a set when using a regular expression in PHP?
- 4 Provide an example of a regular expression in PHP that matches exactly three occurrences of a specific pattern?
- 5 provide an example of a regular expression in PHP that matches a word boundary at the beginning of a word?
- 6 Give an example of a situation where using the (case-insensitive) modifier in PHP regular expressions?
- 7 Provide an example of a complex regular expression pattern where using the (extended) modifier in PHP would enhance readability.
- 8 Provide an example of the (extended) modifier changes the behavior of whitespace and comments within a regular expression pattern.
- 9 Provide an example of how to use preg_match_all() in PHP to extract multiple occurrences of a pattern from a string?
- 10 Provide an example of preg_split() handle capturing groups within the regular expression pattern when splitting a string in PHP
- 11 Give a sample PHP code of using regular expressions for email validation?

EXERCISE 59 : Demonstrate working with Classes And Objects - Creating an object, Object properties, Object methods, Object constructors and destructors

Objectives

At the end of this exercise you shall be able to

- create object in PHP
- use object properties in PHP
- use object constructors and destructors in PHP.

Requirements

Tools/Materials

- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apache web server
- PHP
- Text editor
- Web browser

Procedure

TASK 1: Creating an object, properties and methods

- 1 Open the text editor
- 2 Write the following codes

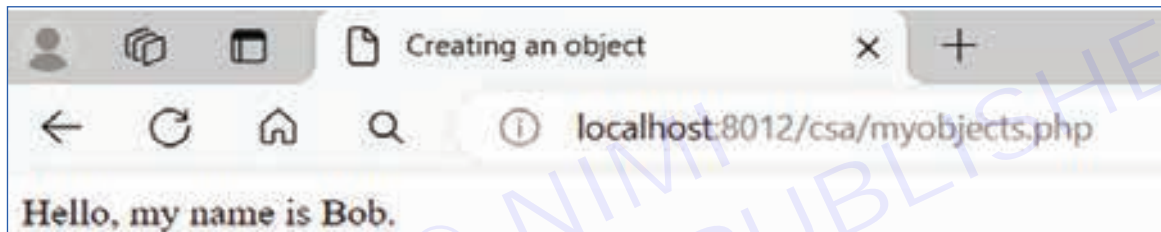
```
<html >
  <head>
    <title> Creating an object </title>
  </head>
  <body>
    <?php
      class Person
      {
          //Add Properties
          public $name;
          public $age;
          //Add Methods
          public function greet()
          {
              echo "Hello, my name is " . $this->name . ".";
          }
      }
    //Create an Object Instance
```

```

    $person1 = new Person();
    //Access and Set Properties
    $person1->name = "Bob";
    $person1->age = 25;
    //Call Methods
    $person1->greet(); // Output: Hello, my name is Bob.
    ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
<http://localhost/foldername/>
- 6 Click the php file to run and verify the output.



TASK 2: Object constructors

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
    <title> constructors</title>
</head>
<body>
    <?php

    class MyClass
    {
        // Properties
        public $name;
        public $age;
        // Constructor
        public function __construct($name, $age)
        {

```

```

        $this->name = $name;
        $this->age = $age;
        echo "Constructor called\n";
    }
    // Method to display information
    public function displayInfo()
    {
        echo "Name: " . $this->name . ", Age: " . $this->age . "\n";
    }
}

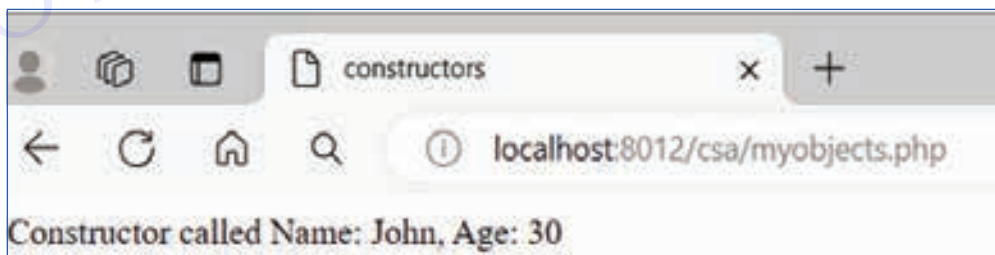
// Creating an object of MyClass
$obj = new MyClass("John", 30);

// Calling method to display information
$obj->displayInfo();

?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 3: Object destructors

- 1 Open the text editor
- 2 Write the following codes


```

<html >
<head>
<title> destructors</title>
</head>

```

```

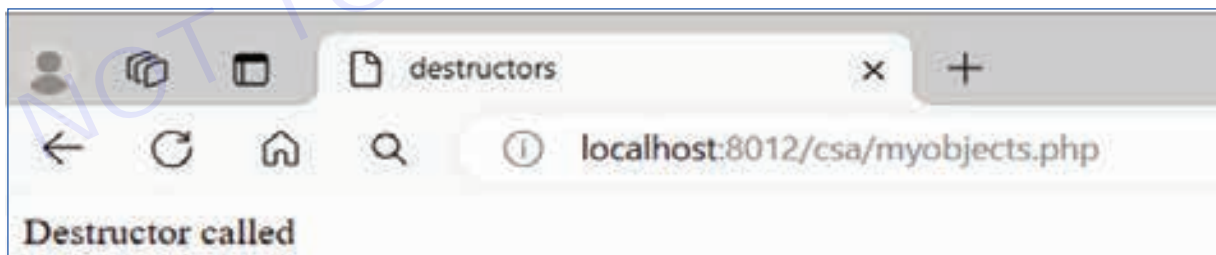
<body>
<?php
class MyClass
{
    // Destructor
    public function __destruct()
    {
        echo "Destructor called\n";
    }
}

// Creating an object of MyClass
$obj = new MyClass();

// The object will be destroyed at the end of the script
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output.



- 1 Create an object of a class named Car in PHP, which has properties such as \$make, \$model, and \$year. Assign values to these properties during object instantiation.
- 2 Create a PHP class called Book with properties \$title, \$author, and \$price. Write a constructor method to initialize these properties. Instantiate an object of the Book class with the title "PHP Programming", author "John Doe", and price \$29.99.
- 3 Create a PHP class Employee with private properties \$name and \$salary. Implement getter and setter methods for these properties. Instantiate an Employee object and set the name to "John" and salary to 50000.

EXERCISE 60 : Perform class constants, Class inheritance, Abstract classes and methods, Object serialization, checking for class and method, existence, Exceptions, Iterators

Objectives

At the end of this exercise you shall be able to

- use inheritance and abstract classes in PHP
- use object serialization in PHP
- use exceptions and Iterators in PHP.

Requirements

Tools/Materials

- Computer/Laptop with latest configuration
- Operating system: windows 10 or 11
- Apache web server
- PHP
- Text editor
- Web browser

Procedure

TASK 1: Using class constants

- 1 Open the text editor
- 2 Write the following codes

```
<html >
  <head>
    <title> class constants </title>
  </head>
  <body>
    <?php
      class MathConstants
      {
        const PI = 3.14159;
        const E = 2.71828;
        public function printConstants()
        {
          echo "The value of PI is: " . self::PI . "<br/>";
          echo "The value of E is: " . self::E . "<br/>";
        }
      }
    </?php>
  </body>
</html >
```

```

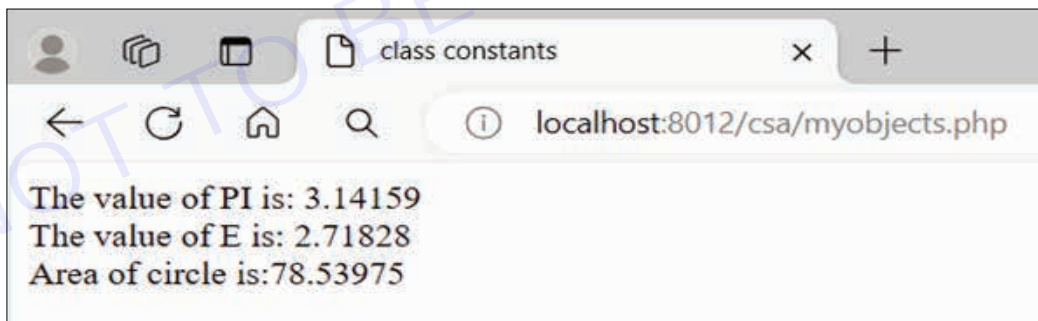
        private $radius;
        public function __construct($radius)
        {
            $this->radius = $radius;
        }
        public function getArea()
        {
            return MathConstants::PI * $this->radius * $this->radius;
        }
    }

    $con= new MathConstants();
    $con->printConstants()

    $circle = new Circle(5);
    echo "Area of circle is:". $circle->getArea(); // Output: 78.53975
    ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 2: Using class inheritance

- 1 Open the text editor
- 2 Write the following codes

```

<html >
    <head>
        <title> Class inheritance </title>
    </head>
    <body>
        <?php

```



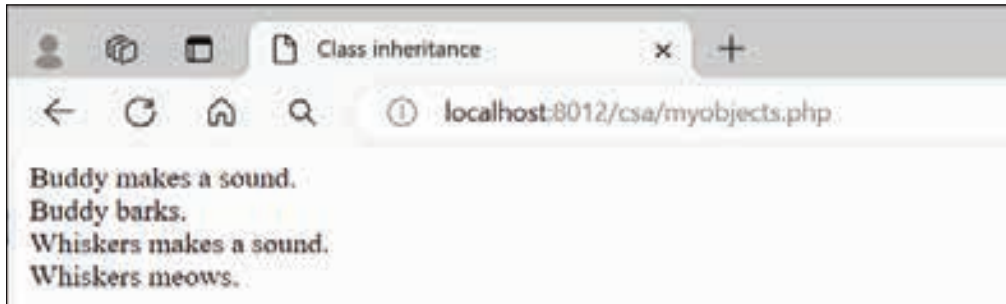
```

// Parent class
class Animal
{
    protected $name;
    public function __construct($name)
    {
        $this->name = $name;
    }
    public function speak()
    {
        echo $this->name . " makes a sound.\n";
    }
}
// Child class inheriting from Animal
class Dog extends Animal
{
    public function bark()
    {
        echo $this->name . " barks.\n";
    }
}
// Child class inheriting from Animal
class Cat extends Animal
{
    public function meow()
    {
        echo $this->name . " meows.\n";
    }
}
// Creating instances of child classes
$dog = new Dog("Buddy");
$cat = new Cat("Whiskers");
// Calling methods from parent and child classes
$dog->speak(); // Output: Buddy makes a sound.
$dog->bark(); // Output: Buddy barks.
$cat->speak(); // Output: Whiskers makes a sound.
$cat->meow(); // Output: Whiskers meows.
?>
</body>

```

```
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 3: Using abstract classes and methods

- 1 Open the text editor
- 2 Write the following codes

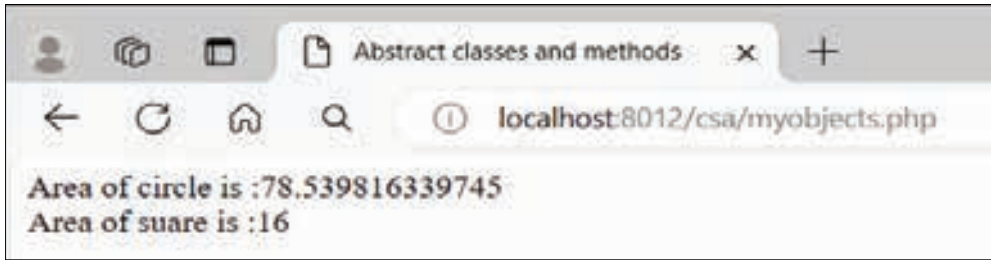
```
<html >
<head>
  <title> Abstract classes and methods </title>
</head>
<body>
<?php
// Abstract class
abstract class Shape
{
  protected $name;
  public function __construct($name)
  {
    $this->name = $name;
  }
  // Abstract method - no implementation
  abstract public function calculateArea();
}
// Concrete subclass of Shape
class Circle extends Shape
{
  private $radius;
  public function __construct($name, $radius)
```

```

{
    parent::__construct($name);
    $this->radius = $radius;
}
// Implementing abstract method
public function calculateArea()
{
    return pi() * pow($this->radius, 2);
}
}
// Concrete subclass of Shape
class Square extends Shape
{
    private $side;
    public function __construct($name, $side)
    {
        parent::__construct($name);
        $this->side = $side;
    }
    // Implementing abstract method
    public function calculateArea()
    {
        return pow($this->side, 2);
    }
}
// Creating instances of concrete subclasses
$circle = new Circle("Circle", 5);
$square = new Square("Square", 4);
// Calling methods on instances
echo"Area of circle is :".$circle->calculateArea(); // Output: 78.539816339745
echo "<br/>";
echo"Area of square is :".$square->calculateArea(); // Output: 16
    ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 4: Checking for class and method existence

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
  <title> Checking for class and method existence </title>
</head>
<body>
<?php
// Define a class
class MyClass
{
  public function myMethod()
  {
    echo "Hello from myMethod!<br/>";
  }
}
// Check if class exists
if (class_exists('MyClass'))
{
  echo "MyClass exists. <br/>";
}
else
{
  echo "MyClass does not exist. <br/>";
}
// Check if method exists in the class
if (method_exists('MyClass', 'myMethod'))
{
  echo "myMethod exists in MyClass. <br/>";
}
else

```

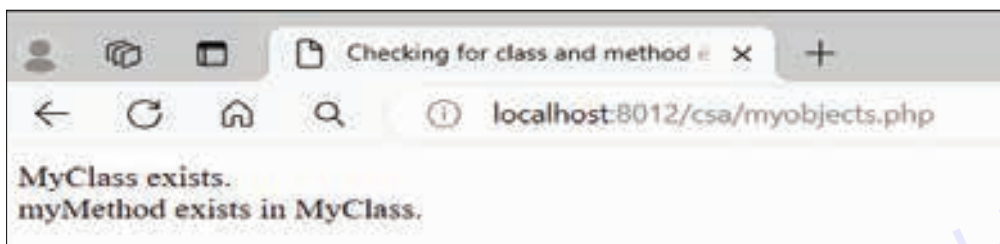
```

{
    echo "myMethod does not exist in MyClass. <br/>";
}

?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 5: Object serialization

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
    <title> Object serialization </title>
</head>
<body>
<?php
// Define a class
class MyClass
{
    public $name;
    public $age;
    public function __construct($name, $age)
    {
        $this->name = $name;
        $this->age = $age;
    }
}
// Create an object of MyClass
$obj = new MyClass("John", 30);

```

```
// Serialize the object
$serializedObj = serialize($obj);

// Output the serialized object
echo "Serialized object: $serializedObj<br/>";
// Unserialize the object
$unserializedObj = unserialize($serializedObj);
// Output the unserialized object
echo "Unserialized object:\n";
var_dump($unserializedObj);
    ?>

</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 6: Using Exceptions

I Using try, catch, and throw keywords

- 1 Open the text editor
- 2 Write the following codes

```
<html >

<head>
    <title> Exceptions </title>
</head>
<body>
<?php
    function divide($dividend, $divisor)
{
    if ($divisor === 0)
    {
        throw new MyCustomException("Cannot divide by zero!");
    }
}
```

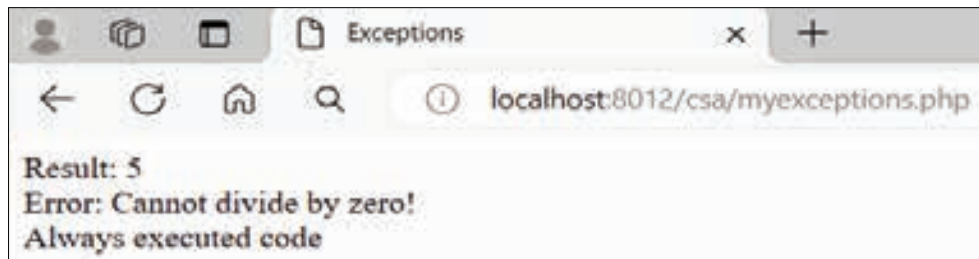
```

    }
    return $dividend / $divisor;
}
class MyCustomException extends Exception
{
    public function __construct($message, $code = 0)
    {
        parent::__construct($message, $code);
    }
}
try
{
    $result = divide(10, 2);
    echo "Result: $result<br/>";
    $result = divide(10, 0); // This will throw an exception
}
catch (MyCustomException $e)
{
    echo "Error: " . $e->getMessage();
}
catch (Exception $e)
{
    echo "General error: " . $e->getMessage();
}
finally
{
    echo "<br/>Always executed code";
}
?>

</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



II Using a custom exception class

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
  <title> custom exception </title>
</head>
<body>
<?php
// Define a custom exception class
class CustomException extends Exception
{
  public function errorMessage()
  {
    // Custom error message
    $errorMsg = 'Error on line ' . $this->getLine() . ' in '
    . $this->getFile() . '<br/> <b>' . $this->getMessage() .
    '</b>';
    return $errorMsg;
  }
}
// Function to validate an email address
function validateEmail($email)
{
  // Check if the email is valid
  if (!filter_var($email, FILTER_VALIDATE_EMAIL))
  {
    // Throw a custom exception if the email is not valid
    throw new CustomException($email);
  }
  // Return true if the email is valid
  return true;
}
// Test the validateEmail function with some email addresses
try

```

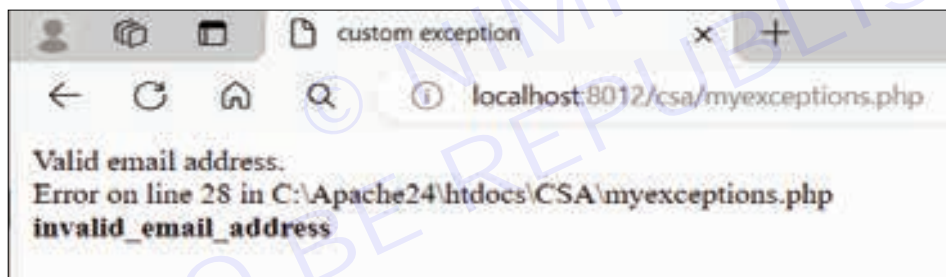


```

{
    validateEmail("john.doe@example.com");
    echo "Valid email address.<br/>";
    validateEmail("invalid_email_address");
    echo "This line will not be reached due an exception will be thrown.<br/>";
}
catch (CustomException $e)
{
    // Catch the custom exception and handle it
    echo $e->errorMessage() . "<br/>";
}
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 7: Using Iterators

I Using foreach loop

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
    <title> Iterators </title>
</head>
<body>
<?php
$numbers = [1, 2, 3, 4, 5];
foreach ($numbers as $number)
{
    echo $number . " ";
}

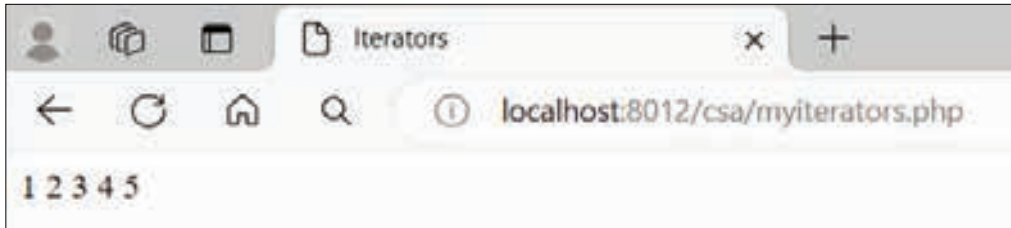
```

```
?>
```

```
</body>
```

```
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output

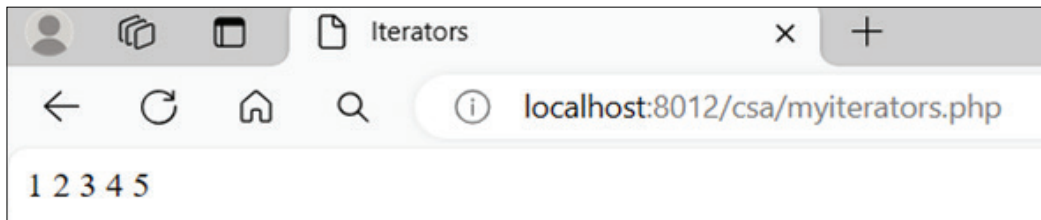


II Using ArrayIterator

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
  <title> Iterators </title>
</head>
<body>
<?php
    $numbers = [1, 2, 3, 4, 5];
    $iterator = new ArrayIterator($numbers);
    while ($iterator->valid())
    {
        echo $iterator->current() . " ";
        $iterator->next();
    }
    ?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



III Iterating over Objects

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
  <title> Iterators </title>
</head>
<body>
<?php
    class User
    {
        public $name;
        public $age;

        public function __construct($name, $age)
        {
            $this->name = $name;
            $this->age = $age;
        }
    }

    $users = [
        new User("Alice", 25),
        new User("Bob", 30),
    ];

// Using foreach loop and custom IteratorAggregate:
class UserCollection implements IteratorAggregate
{
    private $users;
    public function __construct(array $users)
    {
        $this->users = $users;
    }
    public function getIterator()
  
```

```

    {
        return new ArrayIterator($this->users);
    }
}
$userCollection = new UserCollection($users);
foreach ($userCollection as $user)
{
    echo $user->name . " (" . $user->age . ")". PHP_EOL;
}
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



- 1 Provide an example of a class constants be accessed from outside the class definition in PHP.
- 2 Create a 'Person' class with attributes like 'name' and 'age'. Create a 'Student' subclass that inherits from Person and adds a 'studentID' attribute. Instantiate a Student object and access its superclass attributes along with the subclass attribute.
- 3 Create an abstract class 'Animal' with an abstract method 'makeSound()'. Create concrete subclasses 'Cat' and 'Dog' that extend Animal and implement the makeSound() method to produce different sounds for each animal. Call the makeSound() method for a Cat and a Dog object.
- 4 Create a class 'Product' with properties like 'name', 'price', and 'quantity'. Serialize an object of this class into a string using PHP's serialization mechanism.
- 5 Create a function that reads data from a file, use PHP exceptions to handle errors such as file not found or insufficient permissions while reading the file.
- 6 Create an array of product objects representing items in a shopping cart. Use the ArrayIterator class in PHP to iterate over this array and calculate the total cost of all items in the shopping cart.

EXERCISE 61 : Connect to MySQL database from PHP and insert, delete & update data in MySQL database from webpage

Objectives

At the end of this exercise you shall be able to

- create a MySQL database and table from PHP
- insert and update data in MySQL table from PHP
- delete data in MySQL table from PHP.

Requirements

Tools/Materials

- PC/Laptop with latest configuration
- operating system: windows 10 or 11
- Text editor
- web browser
- apache web server
- PHP
- MySQL server

Procedure

TASK 1: Connect to MySQL database from PHP

I Connect to MySQL and create a database

- 1 Open the text editor
- 2 Write the following codes

```
<html >
<head>
<title> create a database </title>
</head>
<body>
<?php
$servername = "localhost"; // Change this if MySQL server is on a different host
$username = "your_username"; // Change this to your MySQL username
$password = "your_password"; // Change this to your MySQL password
```

```
// Create connection
```

```
$conn = new mysqli($servername, $username, $password);
```

```
// Check connection
```

```
if ($conn->connect_error)
```

```
{
    die("Connection failed: " . $conn->connect_error);
}
```

```
// Create database
```

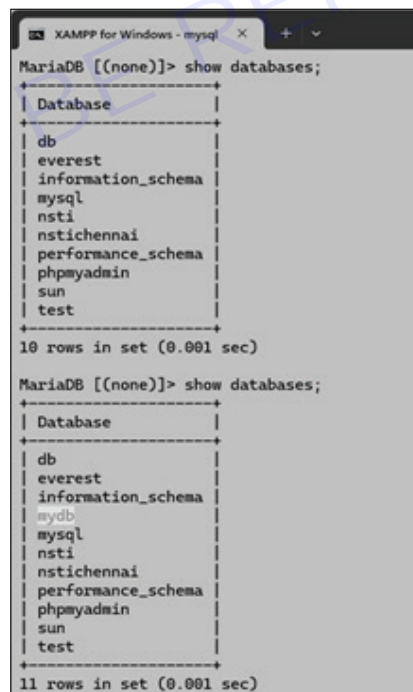
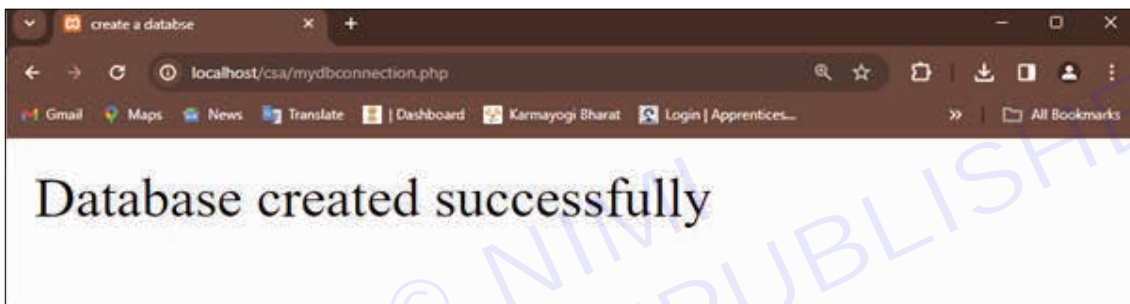
```
$sql = "CREATE DATABASE myDB";
```

```

if ($conn->query($sql) === TRUE) {
    echo "Database created successfully";
} else {
    echo "Error creating database: " . $conn->error;
}
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



- II Connect to MySQL and create a table
 - 1 Open the text editor
 - 2 Write the following codes

```

<html >
<head>
  <title> create a table </title>
</head>
<body>
<?php
  // Database connection parameters
  $servername = "localhost"; // Change this if MySQL server is on a different host
  $username = "your_username"; // Change this to your MySQL username
  $password = "your_password"; // Change this to your MySQL password
  $database = "example_db"; // Change this to your database name
  // Create connection
  $conn = new mysqli($servername, $username, $password, $database);
  // Check connection
  if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
  }
  // sql to create table
  $sql = "create table persons(
  id int, first_name varchar(30), last_name varchar(30), email varchar(70))";
  if ($conn->query($sql) === TRUE) {
    echo "Table created successfully";
  } else {
    echo "Error creating table: " . $conn->error;
  }
  $conn->close();
  ?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



```

XAMPP for Windows - mysql
MariaDB [(none)]> use mydb;
Database changed
MariaDB [mydb]> show tables;
+-----+
| Tables_in_mydb |
+-----+
| persons        |
+-----+
1 row in set (0.000 sec)

MariaDB [mydb]> desc persons;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id         | int(11)   | YES  |     | NULL    |       |
| first_name | varchar(30)| YES  |     | NULL    |       |
| last_name  | varchar(30)| YES  |     | NULL    |       |
| email      | varchar(70)| YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.035 sec)

MariaDB [mydb]>

```

TASK 2 : Connect to MySQL database from PHP and insert data

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head>
  <title> </title>
</head>
<body>
<?php
  // Database connection parameters
  $servername = "localhost"; // Change this if MySQL server is on a different host
  $username = "your_username"; // Change this to your MySQL username
  $password = "your_password"; // Change this to your MySQL password
  $database = "example_db"; // Change this to your database name
  // Create connection
  $conn = new mysqli($servername, $username, $password, $database);
  // Check connection
  if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
  }
  $sql = "INSERT INTO persons (first_name, last_name, email) VALUES ('Peter', 'Parker', 'peterparker@mail.com')";
  if ($conn->query($sql) === TRUE) {

```

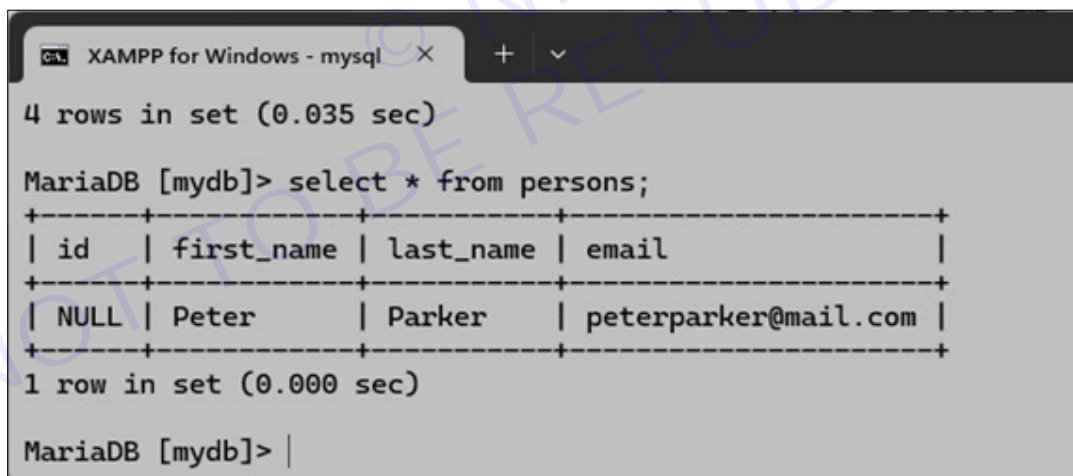


```

    echo "New record created successfully";
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}
$conn->close();
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder with .php extension
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the php file to run and verify the output



TASK 3: Connect to MySQL database from PHP and insert data using prepare statement

- i Create an HTML form in the webpage (index.html) to collect the data from the user

- 1 Open the text editor
- 2 Write the following codes


```

<html >
<head> HTML form </title>
</head>
<body>

```

```

<body>
  <h2>Insert Data into MySQL Database</h2>
  <form method="post" action="insert.php">
    <label for="username">Username:</label><br>
    <input type="text" id="username" name="username" required><br><br>
    <label for="email">Email:</label><br>
    <input type="email" id="email" name="email" required><br><br>
    <input type="submit" value="Submit">
  </form>
</body>
</html>

```

3 Save the program in C:\Apache24\htdocs in a folder as index.html

ii Create a PHP script (insert.php) to connect to the database and handle the insertion process.

1 Open the text editor

2 Write the following codes

```

<html >
<head>
  <title> </title>
</head>
<body>
<?php
// Database connection parameters
$servername = "localhost"; // Change this if the MySQL server is on a different host
$username = "your_username"; // Change this to MySQL username
$password = "your_password"; // Change this to MySQL password
$dbname = "example_db"; // Change this to database name
// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error)
{
  die("Connection failed: " . $conn->connect_error);
}

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST")
{
  // Get form data
  username = $_POST["username"];

```

```

$email = $_POST["email"];
// Prepare and execute the SQL statement to insert data into the table
$stmt = $conn->prepare("INSERT INTO users (username, email) VALUES (?,?)");
$stmt->bind_param("ss", $username, $email);

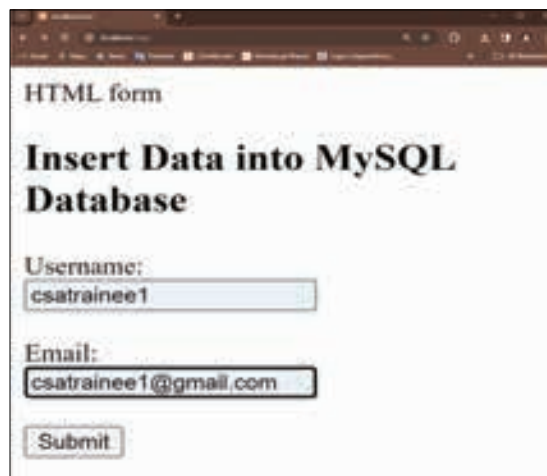
    if ($stmt->execute())
    {
        echo "New record inserted successfully";
    }
else
{
    echo "Error: " . $conn->error;
}

// Close statement
$stmt->close();
}

// Close connection
$conn->close();
?>
</body>
</html>

```

- 3 Save the program in C:\Apache24\htdocs in a folder as insert.php
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
<http://localhost/foldername/>
- 6 Click the index.html file to run and verify the output



HTML form

Insert Data into MySQL Database

Username:

Email:



 A screenshot of a MySQL command prompt window titled 'XAMPP for Windows - mysql'. The prompt shows the command 'MariaDB [mydb]> select * from users;' and its output:


```

  +-----+-----+
  | username | email |
  +-----+-----+
  | csatraineel | csatraineel@gmail.com |
  +-----+-----+
  1 row in set (0.000 sec)

  MariaDB [mydb]> |
  
```

TASK 4: Connect to MySQL database from PHP and update data

i Create an HTML form in the webpage (index.html) to collect the data from the user

- 1 Open the text editor
- 2 Write the following codes

```

<html >
<head> HTML form </title>
</head>
<body>
<h2>Update Data in MySQL Database</h2>
<form method="post" action="update.php">
    <label for="username">Username of Record to Update:</label><br>
    <input type="text" id="username" name="username" required><br><br>
    <label for="email">New Email:</label><br>
    <input type="email" id="email" name="email" required><br><br>
    <input type="submit" value="Update">
</form>
</body>
</html>
  
```

- 3 Save the program in C:\Apache24\htdocs in a folder as index.html

ii Create a PHP script (update.php) to connect to the database and handle the update process based on user input.

1 Open the text editor

2 Write the following codes

```

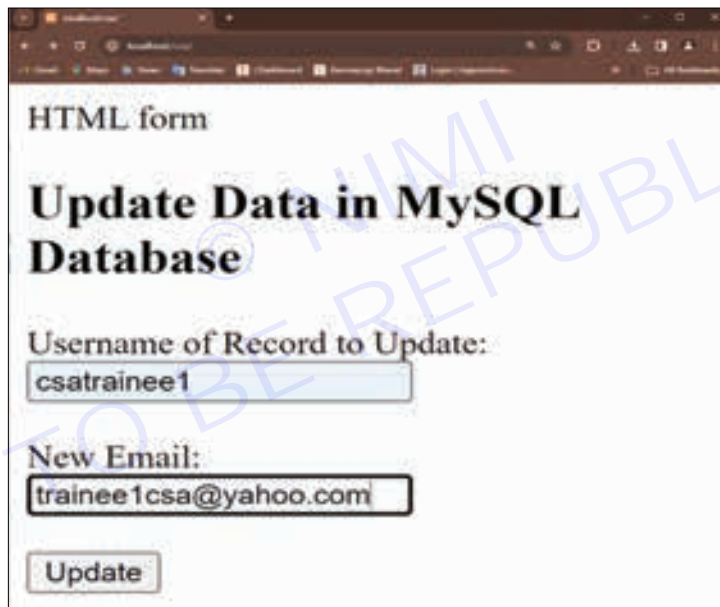
<html >
<head> HTML form </title>
</head>
<body>
    <?php
    // Database connection parameters
    $servername = "localhost"; // Change this if MySQL server is on a different host
    $username = "your_username"; // Change this to MySQL username
    $password = "your_password"; // Change this to MySQL password
    $database = "example_db"; // Change this to database name
    // Create connection
    $conn = new mysqli($servername, $username, $password, $database);
    // Check connection
    if ($conn->connect_error)
    {
        die("Connection failed: " . $conn->connect_error);
    }
    // Check if the form is submitted
    if ($_SERVER["REQUEST_METHOD"] == "POST")
    {
        // Get form data
        // $id = $_POST["id"];
        $newUsername = $_POST["username"];
        $newEmail = $_POST["email"];
        // Prepare and execute the SQL statement to update data in the table
        $stmt = $conn->prepare("UPDATE users SET email = ? WHERE username = ?");
        $stmt->bind_param("ss", $newEmail, $newUsername);
        if ($stmt->execute())
        {
            echo "Record updated successfully";
        }
        else
        {
            echo "Error: " . $conn->error;
        }
    }

```

```
// Close statement
$stmt->close();
}
```

```
// Close connection
$conn->close();
?>
</body>
</html>
```

- 3 Save the program in C:\Apache24\htdocs in a folder as update.php
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the index.html file to run and verify the output



HTML form

Update Data in MySQL Database

Username of Record to Update:

New Email:



```

XAMPP for Windows - mysql
MariaDB [mydb]> select * from users;
+-----+-----+
| username | email |
+-----+-----+
| csatraineel1 | csatraineel1@gmail.com |
| csatraineel2 | csatraineel2@gmail.com |
| csatraineel3 | csatraineel3@gmail.com |
+-----+-----+
3 rows in set (0.000 sec)

MariaDB [mydb]> select * from users;
+-----+-----+
| username | email |
+-----+-----+
| csatraineel1 | traineelcsa@yahoo.com |
| csatraineel2 | csatraineel2@gmail.com |
| csatraineel3 | csatraineel3@gmail.com |
+-----+-----+
3 rows in set (0.001 sec)

```

TASK 5: Connect to MySQL database from PHP and delete data

i Create an HTML form in the webpage (index.html) to collect the data from the user

1 Open the text editor

2 Write the following codes

```

<html >
<head> HTML form </title>
</head>
<body>
<h2>Delete Data from MySQL Database</h2>
<form method="post" action="delete.php">
    <label for="username">Username of Record to Delete:</label><br>
    <input type="text" id="username" name="username" required><br><br>
    <input type="submit" value="Delete">
</form>
</body>
</html>

```

3 Save the program in C:\Apache24\htdocs in a folder as index.html

ii Create a PHP script (delete.php) to connect to the database and handle the update process based on user input.

1 Open the text editor

2 Write the following codes

```

<html >

```

```

<head> HTML form </title>
</head>
<body>
<?php
    // Database connection parameters
    $servername = "localhost"; // Change this if MySQL server is on a different host
    $username = "root"; // Change this to your MySQL username
    $password = ""; // Change this to your MySQL password
    $database = "mydb"; // Change this to your database name

    // Create connection
    $conn = new mysqli($servername, $username, $password, $database);
    // Check connection
    if ($conn->connect_error)
    {
        die("Connection failed: " . $conn->connect_error);
    }
    // Check if the form is submitted
    if ($_SERVER["REQUEST_METHOD"] == "POST")
    {
    // Get ID from the form
    $username = $_POST["username"];
    // Prepare and execute the SQL statement to delete data from the table
    $stmt = $conn->prepare("DELETE FROM users WHERE username = ?");
    $stmt->bind_param("s", $username);
    if ($stmt->execute())
    {
        echo "Record deleted successfully";
    }
    else
    {
        echo "Error: " . $conn->error;
    }

    // Close statement
    $stmt->close();
    }
    // Close connection
    $conn->close();

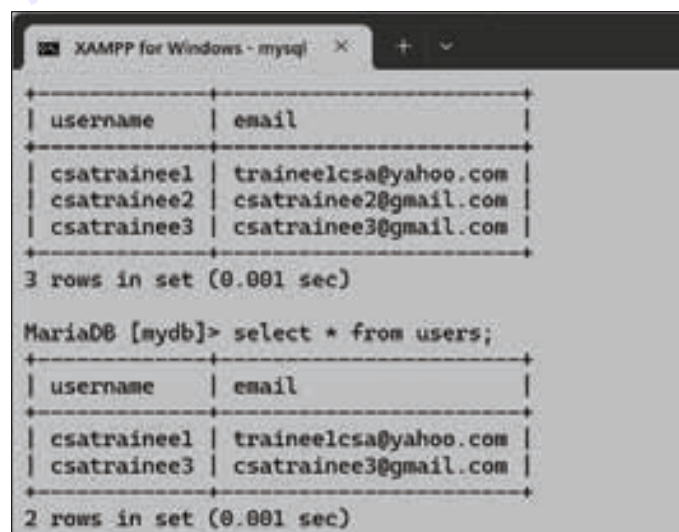
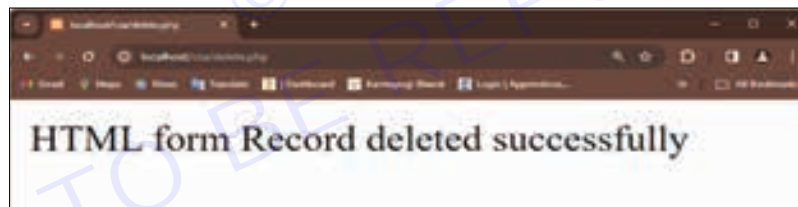
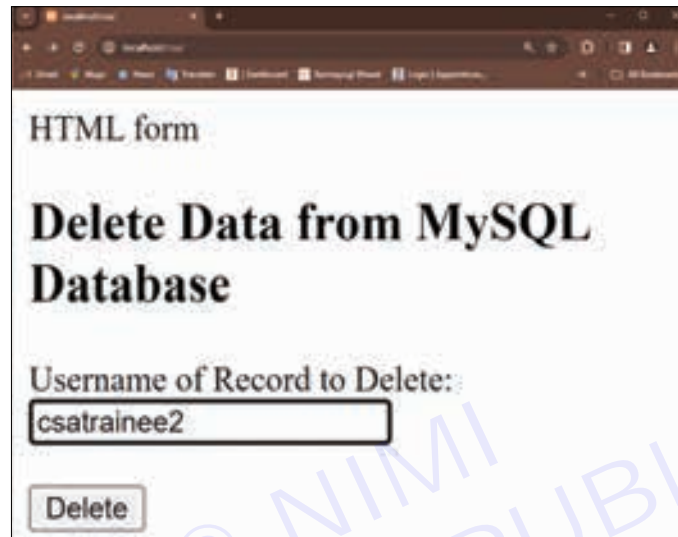
```


?>

</body>

</html>

- 3 Save the program in C:\Apache24\htdocs in a folder as delete.php
- 4 Run the Apache services from windows services
- 5 Open the browser and type the following address
http://localhost/foldername/
- 6 Click the index.html file to run and verify the output



- 1 Write the PHP code to establish a connection from a MySQL database named my_database hosted on localhost, using a username and a password.
- 2 Write the PHP code to connect to the database, execute a query to retrieve data from a table named users, and then display the results on a webpage.
- 3 Write PHP code to connect to the database, insert a new product record with specific values in to a table named products and handle any potential errors during the insertion process.
- 4 Write PHP code to connect to the database, update existing records in a MySQL database table named orders. update specific order records with new values, and ensure that the updates are applied successfully.
- 5 Write PHP code to connect to the database, delete specific customer records based on certain criteria from a MySQL database table named customers.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 62 : Create a project on PHP & MySQL for online library management

Objectives

At the end of this exercise you shall be able to

- create a project for online library management using PHP & MySQL.

Requirements

Tools/Materials

- PC/Laptop with latest configuration
- Operating system: windows 10 or 11
- Text editor
- Web browser
- Apache web server
- PHP
- MySQL server

Procedure

Project:

Develop a web-based application to manage a library's book collection, users, loans, and other functionalities.

Technologies:

Front-end: HTML, CSS, JavaScript (optional)

Back-end: PHP

Database: MySQL

System Modules:

1 User Management:

User registration and login (including different user roles, e.g., admin, librarian, member)

User profile management (update details, password change)

2 Book Management:

Add, edit, and delete book information (title, author, genre, publication date, availability)

Search books by various criteria (title, author, genre, keywords)

View detailed information about each book (summary, reviews, borrowed history)

3 Loan Management:

Borrow and return books (check user eligibility, availability, due dates)

Generate loan reports (individual, overdue, most popular books)

Set and manage loan periods and fines

4 Additional Features (optional):

Book reservation system

Online reading platform for eBooks

User reviews and ratings for books

Recommendations based on user borrowing history

Inventory management (track book condition, purchase new books)

Database Design:

Tables: Users, Books, Loans, Genres, Authors, etc.

Relationships: Users can borrow Books, Books can have multiple Loans, etc.

Implement data integrity constraints and triggers for efficient management.

Development Approach:

Implement modules iteratively, starting with core functionalities.

Use clear coding practices and modular design for maintainability.

Validate user input to prevent security vulnerabilities.

Implement user authentication and authorization for different roles.

Testing and Deployment:

Thoroughly test all functionalities manually and with automated tools.

Deploy the application on a web server accessible to users.

Evaluation and Future Improvements:

Gather user feedback and address issues during maintenance.

Implement new features based on user needs and feedback.

Consider performance optimization and scalability for large datasets.

Additional Notes:

This is a basic outline, and specific modules and features can be adjusted based on the needs and requirements.

Research existing open-source library management systems for inspiration and potential libraries to utilize.

Always keep security and data privacy in mind when developing and deploying the system.

NOT TO BE RE-PUBLISHED

◆ Module 5 : Advance Data Analysis Using Excel ◆

EXERCISE 63 : Protect sheet using password

Objectives

At the end of this exercise you shall be able to

- protect your Excel Sheet and Work Book.

Requirements

Tools/Materials

- PC/Laptop with Windows OS
- MS Excel 2013 or Higher

Procedure

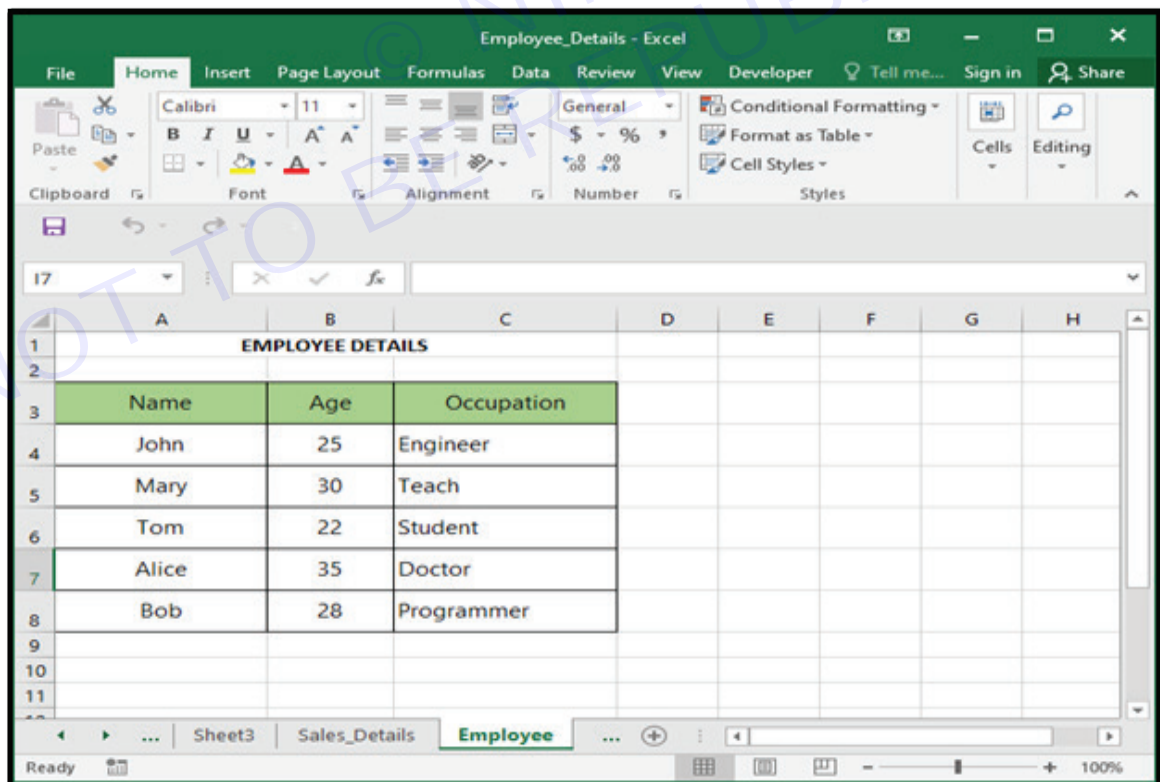
TASK 1: To protect a sheet using a password in Microsoft Excel, follow these steps

1 Open your Excel Workbook:

- Launch Microsoft Excel and open the workbook containing the sheet you want to protect.

2 Navigate to the Sheet:

- Go to the sheet that you want to protect.

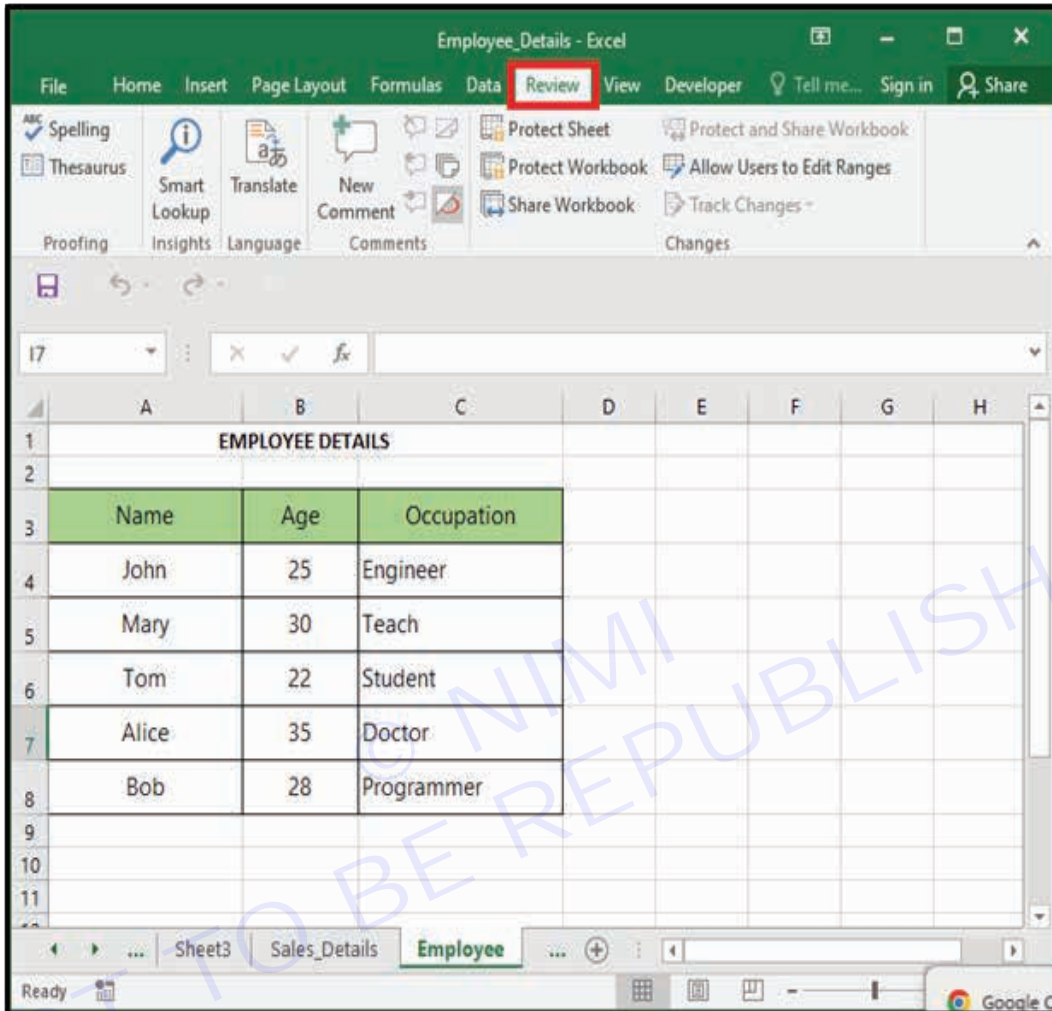


3 Select the “Review” Tab:

- Click on the “Review” tab in the Excel ribbon at the top.

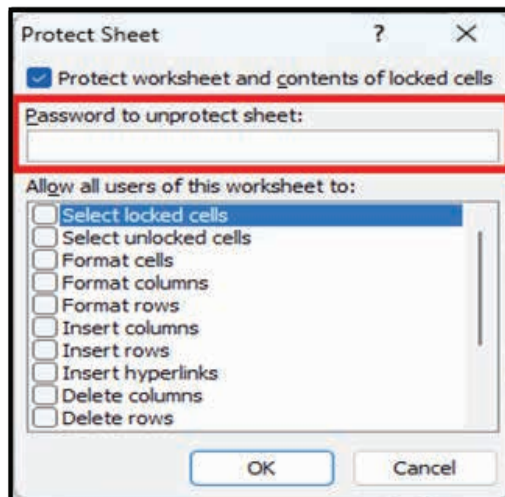
4 Click on “Protect Sheet”:

- In the “Changes” group, you will find the “Protect Sheet” option. Click on it.

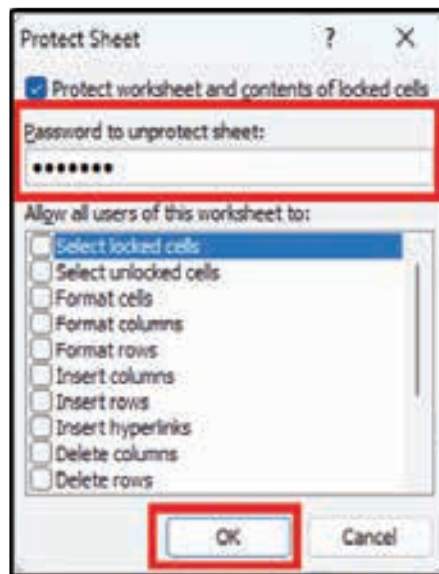


5 Set Protection Options:

- A “Protect Sheet” dialog box will appear. Here, you can set various protection options:



- Enter a password in the “Password to unprotect sheet” field. This password will be required to unprotect the sheet later.



- Choose specific options such as allowing users to select locked cells, format cells, insert rows, insert columns, etc.
- Click on “OK” when you are done.

6 Re-enter Password (Optional):



- If you set a password, you will be prompted to re-enter it to confirm.

7 Save Your Workbook:

- It's a good practice to save your workbook after protecting the sheet.

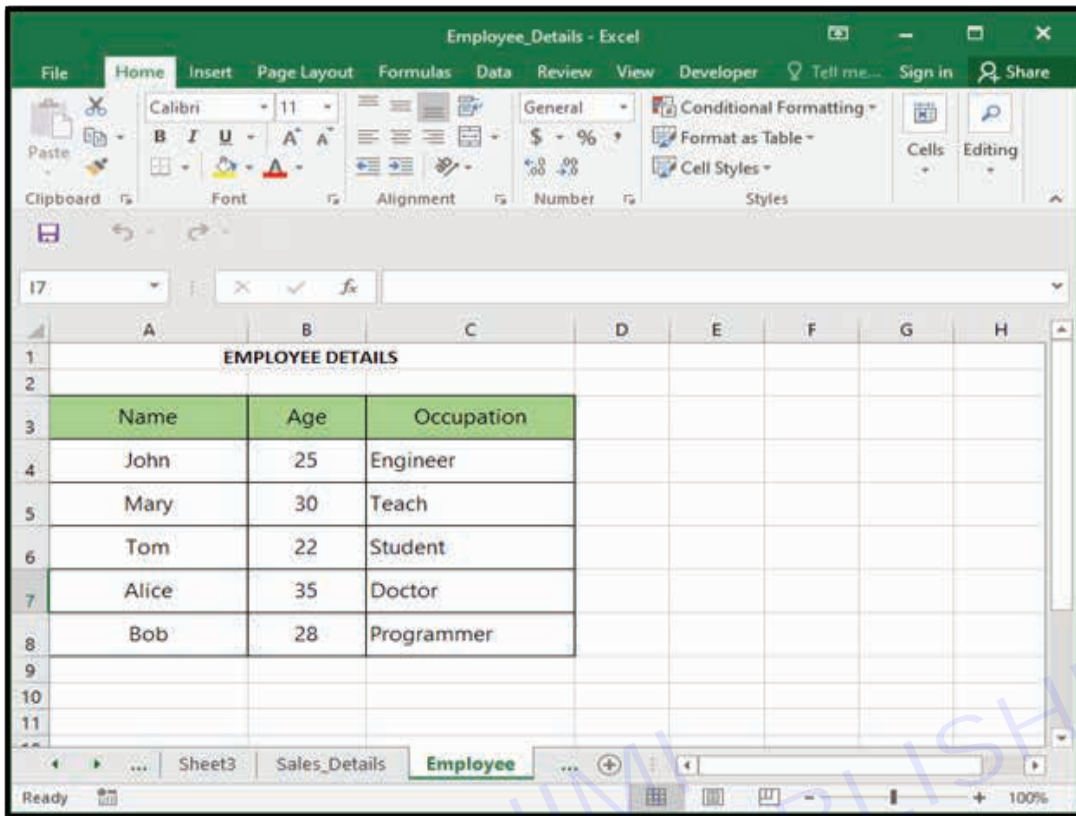
Now, the selected sheet is protected, and users will need to enter the password to make changes based on the options you've chosen.

Keep in mind that if you forget the password, there is no way to recover it. Make sure to remember the password or keep a backup of your workbook without the protection.

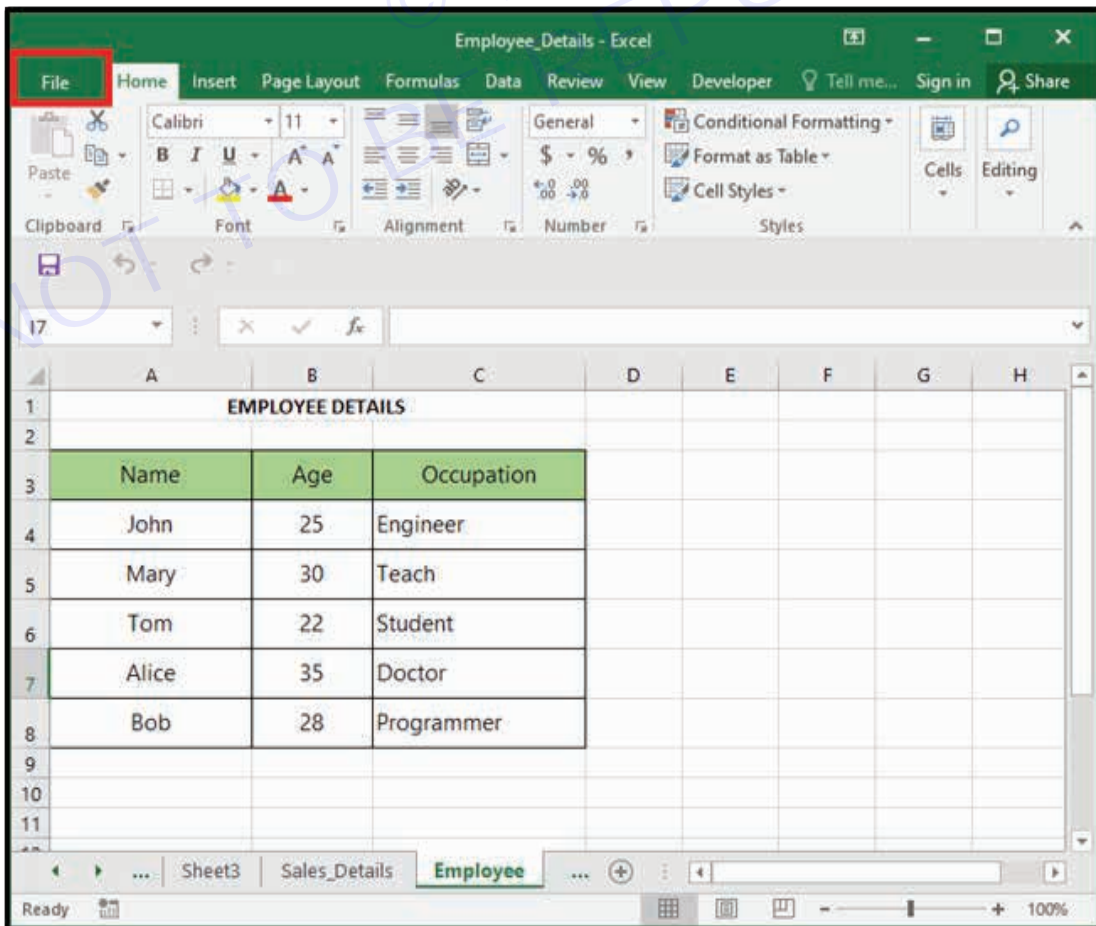
These steps apply to Microsoft Excel versions like Excel 2013, Excel 2016, Excel 2019, and Excel for Microsoft 365. The exact steps might vary slightly depending on the version you are using.

To protect a Work Book using a password in Microsoft Excel, follow these steps:

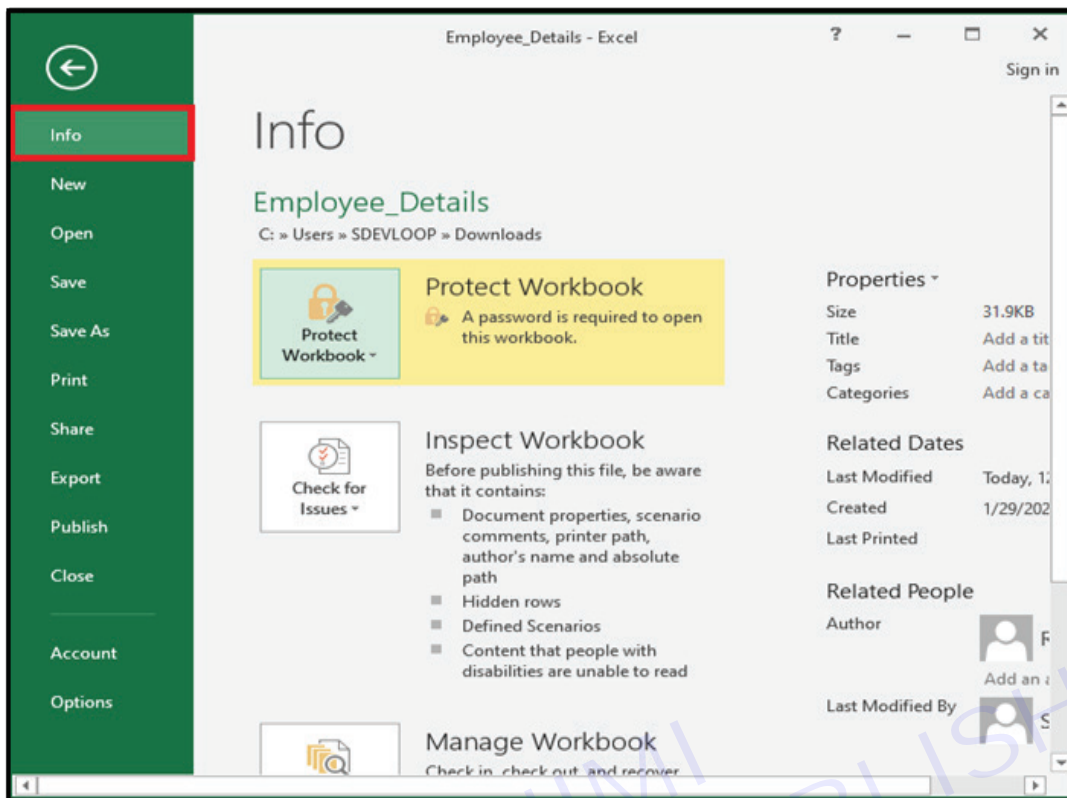
1 Open your Excel Workbook: Open the Excel workbook that you want to protect.



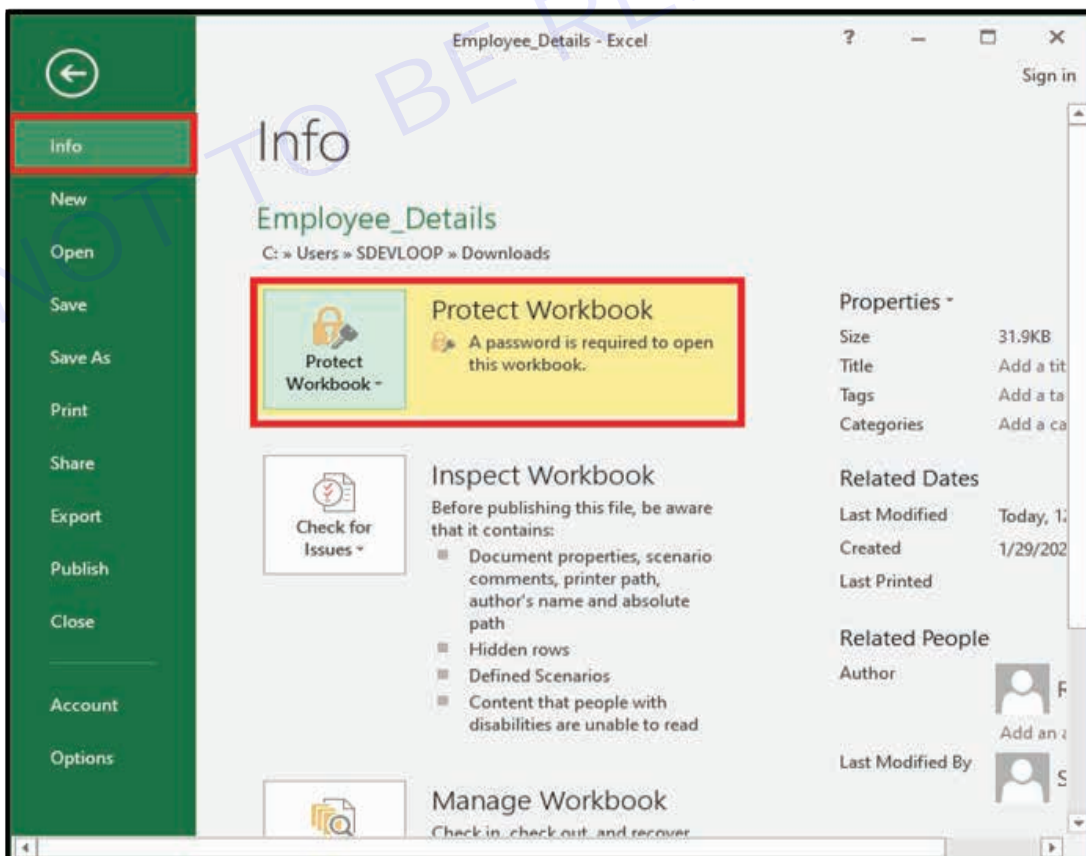
2 Click on the “File” Tab: Click on the “File” tab in the Ribbon to access the Backstage view.



3 Select “Info” from the menu: In the Backstage view, select the “Info” option from the menu on the left.

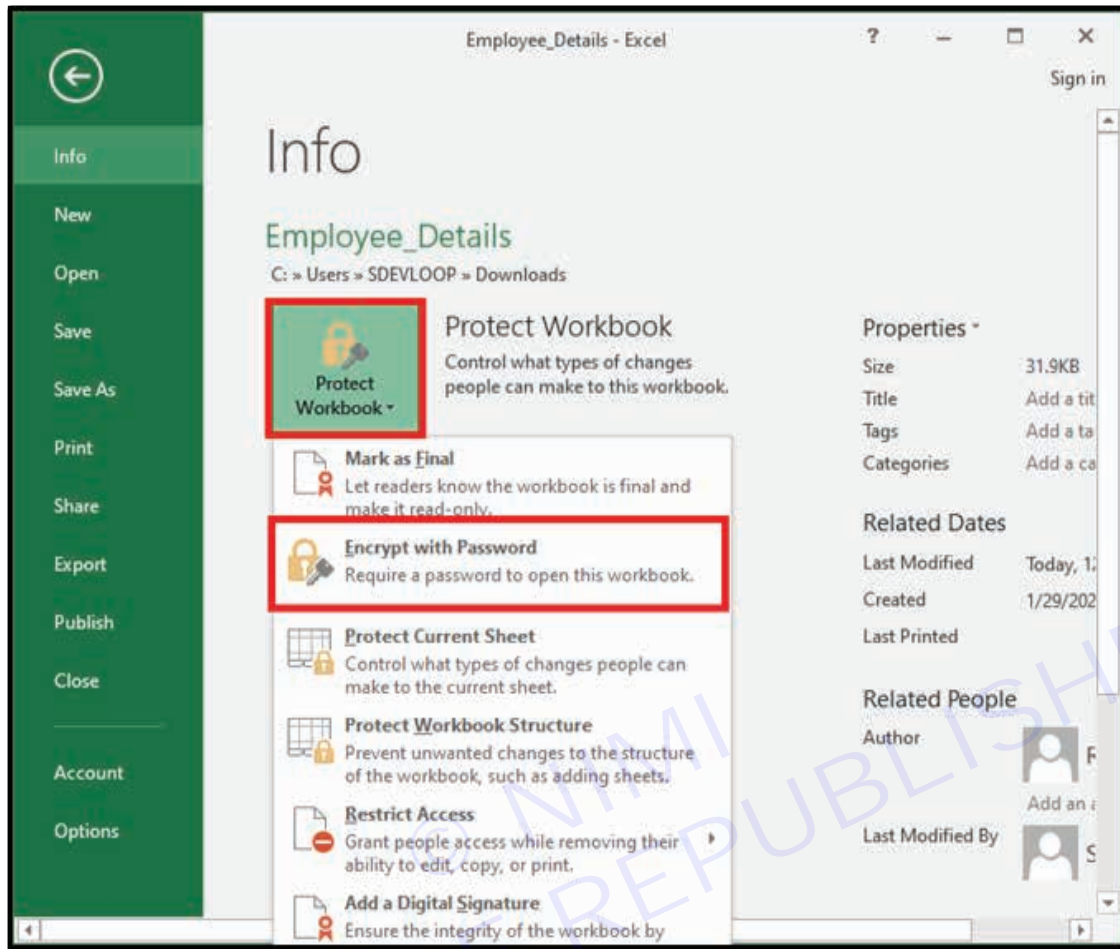


4 Click on “Protect Workbook”: Under the “Info” section, you will find an option called “Protect Workbook.” Click on it.



5 Choose a Protection Method:

- If you want to add a password to open the workbook, choose “Encrypt with Password.”



6 Enter the Password: If you selected “Encrypt with Password,” enter a password and click “OK.”

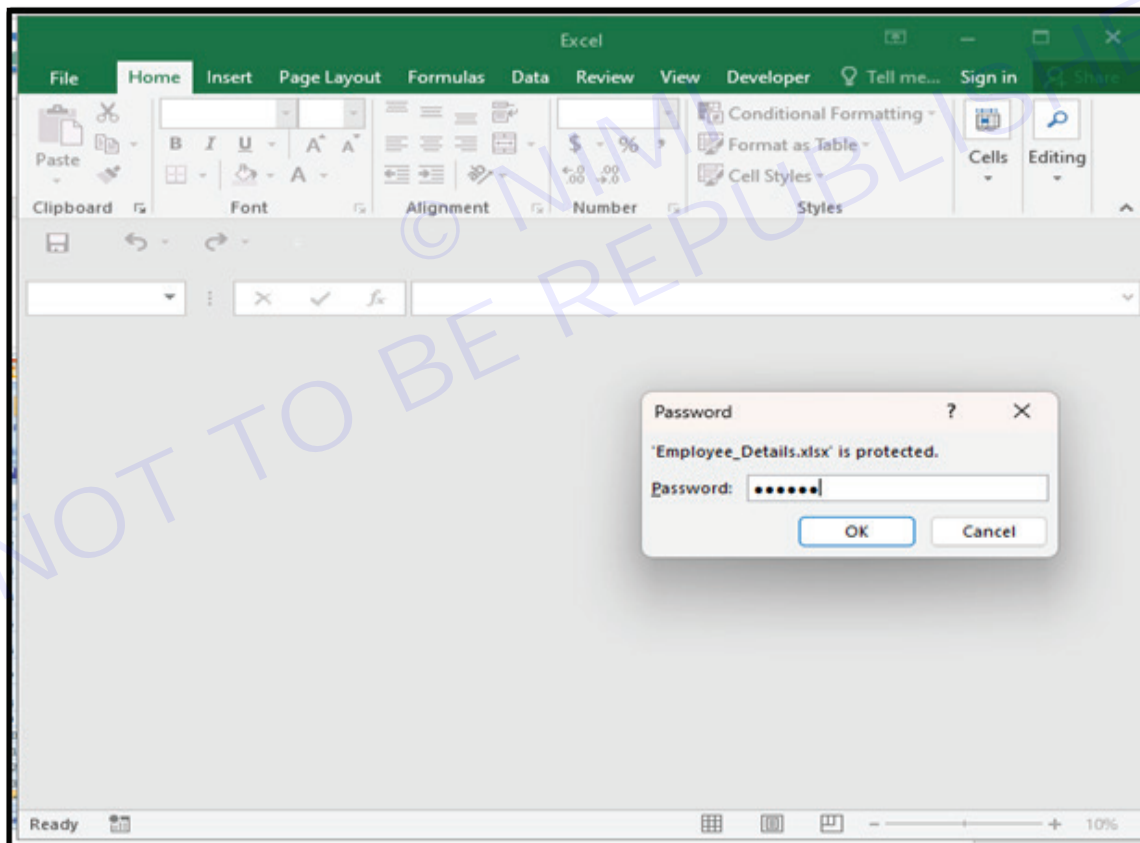


Note: Make sure to remember the password, as it will be required to unprotect the workbook.

7 **Confirm the Password:** If prompted, confirm the password by entering it again and click “OK.”



8 **Save the Workbook:** Save the workbook to apply the protection. Your workbook is now protected, and users will need the password to open or modify it.



EMPLOYEE DETAILS		
Name	Age	Occupation
John	25	Engineer
Mary	30	Teach
Tom	22	Student
Alice	35	Doctor
Bob	28	Programmer

Related Exercises:

- 1 Create a 'SalesData.xlsx' Excel file and protect the 'SalesData' sheet using the password 'Secure123.'
- 2 Create a 'Confidential.xlsx' Excel file and protect the 'Confidential' sheet using the password 'Secret789'. Modify the password from 'Secret789' to 'Classified987.' Ensure that the new password is required to unprotect the sheet.

EXERCISE 64 : Use flash fill techniques

Objectives

At the end of this exercise you shall be able to

- use flash fill techniques in your worksheet.

Requirements

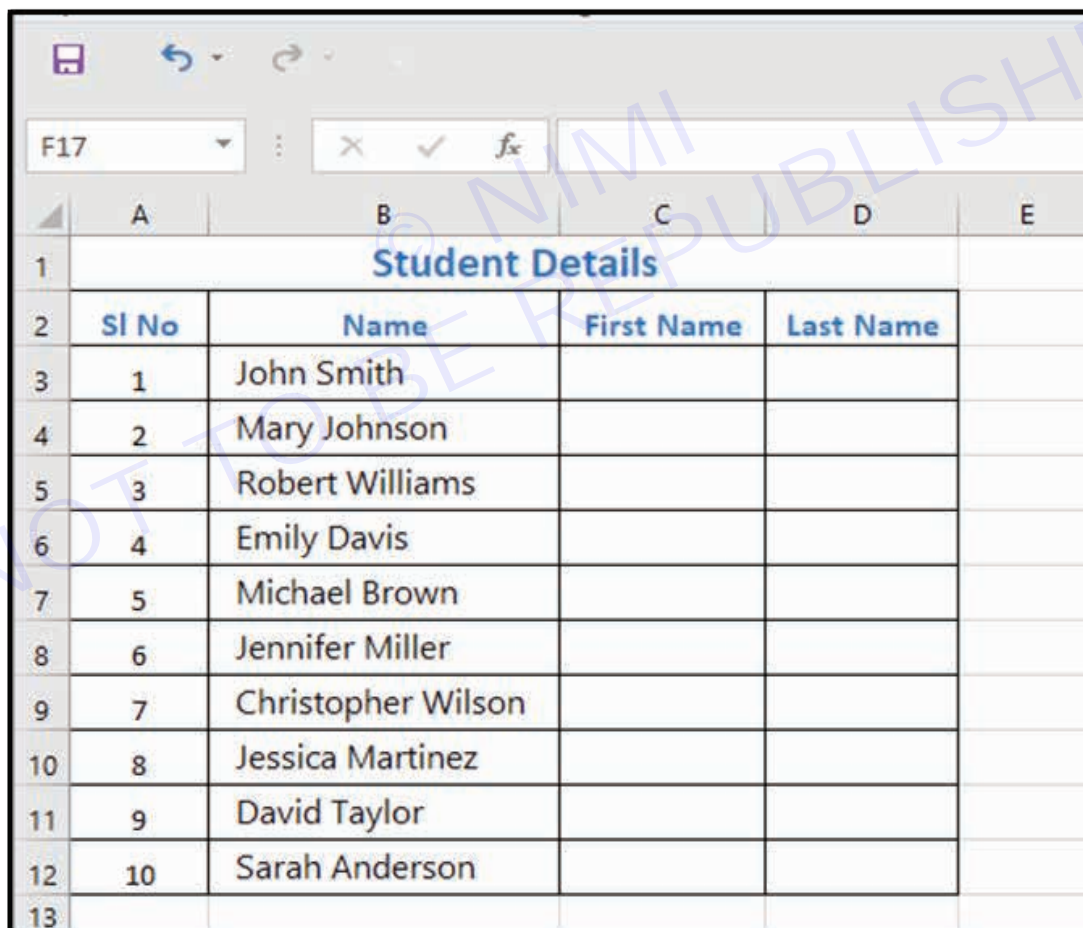
Tools/Materials

- PC/Laptop with Windows OS
- MS Excel 2013 or Higher

Procedure

TASK 1: Given a column of full names use Flash Fill to split them into separate columns for first and last names

1 Data Preparation:



Student Details			
SI No	Name	First Name	Last Name
1	John Smith		
2	Mary Johnson		
3	Robert Williams		
4	Emily Davis		
5	Michael Brown		
6	Jennifer Miller		
7	Christopher Wilson		
8	Jessica Martinez		
9	David Taylor		
10	Sarah Anderson		

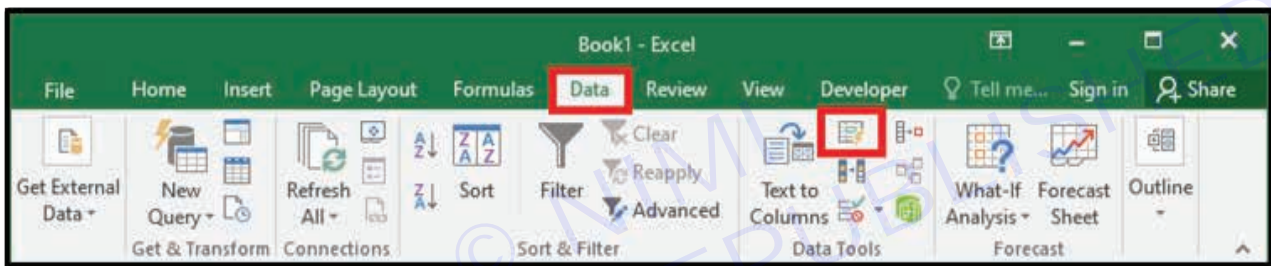
2 Activate Flash Fill:

- In the adjacent cell where you want to apply Flash Fill, start typing the pattern you want Excel to recognize.
- For example, if you want to extract first names, type the first name in the adjacent cell.

Student Details			
Sl No	Name	First Name	Last Name
1	John Smith	John	
2	Mary Johnson		
3	Robert Williams		
4	Emily Davis		
5	Michael Brown		
6	Jennifer Miller		
7	Christopher Wilson		
8	Jessica Martinez		
9	David Taylor		
10	Sarah Anderson		

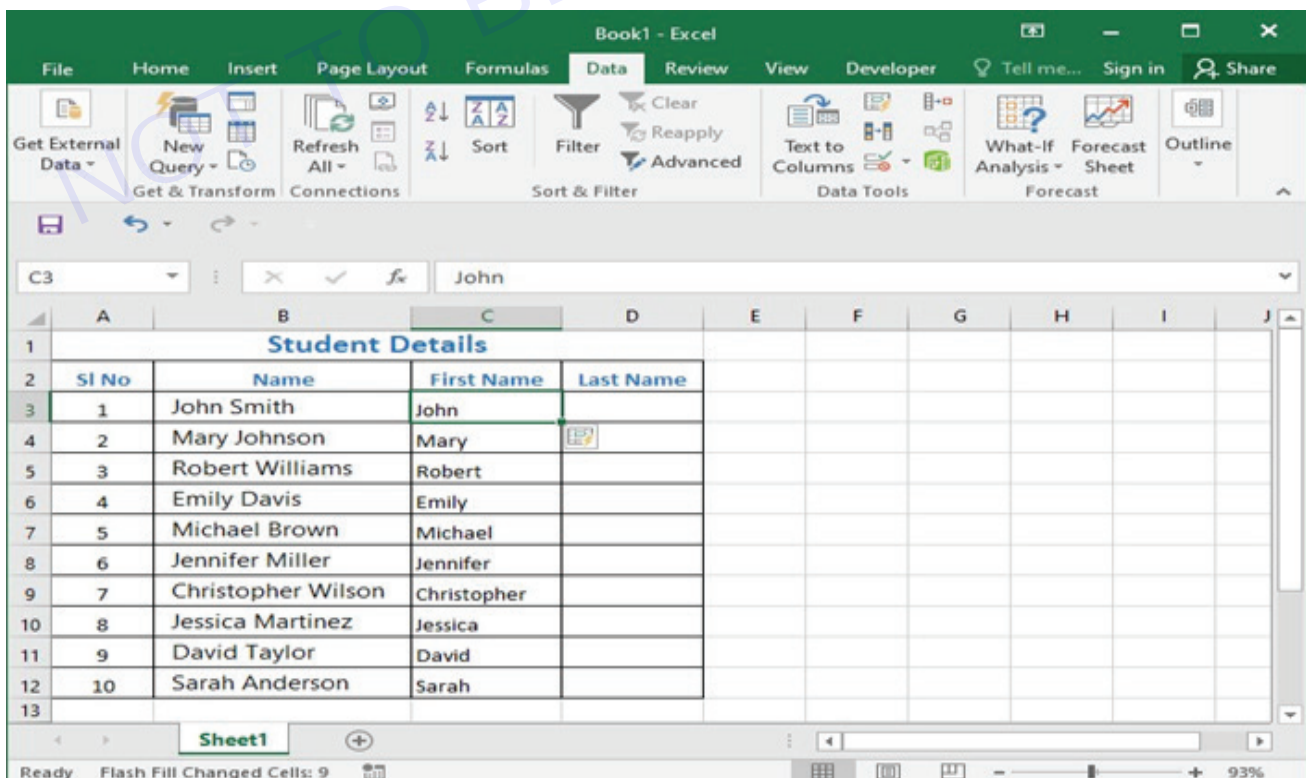
3 Execute Flash Fill:

- Press **Ctrl + E** or go to the “Data” tab on the ribbon, and in the “Data Tools” group, click on “Flash Fill.”

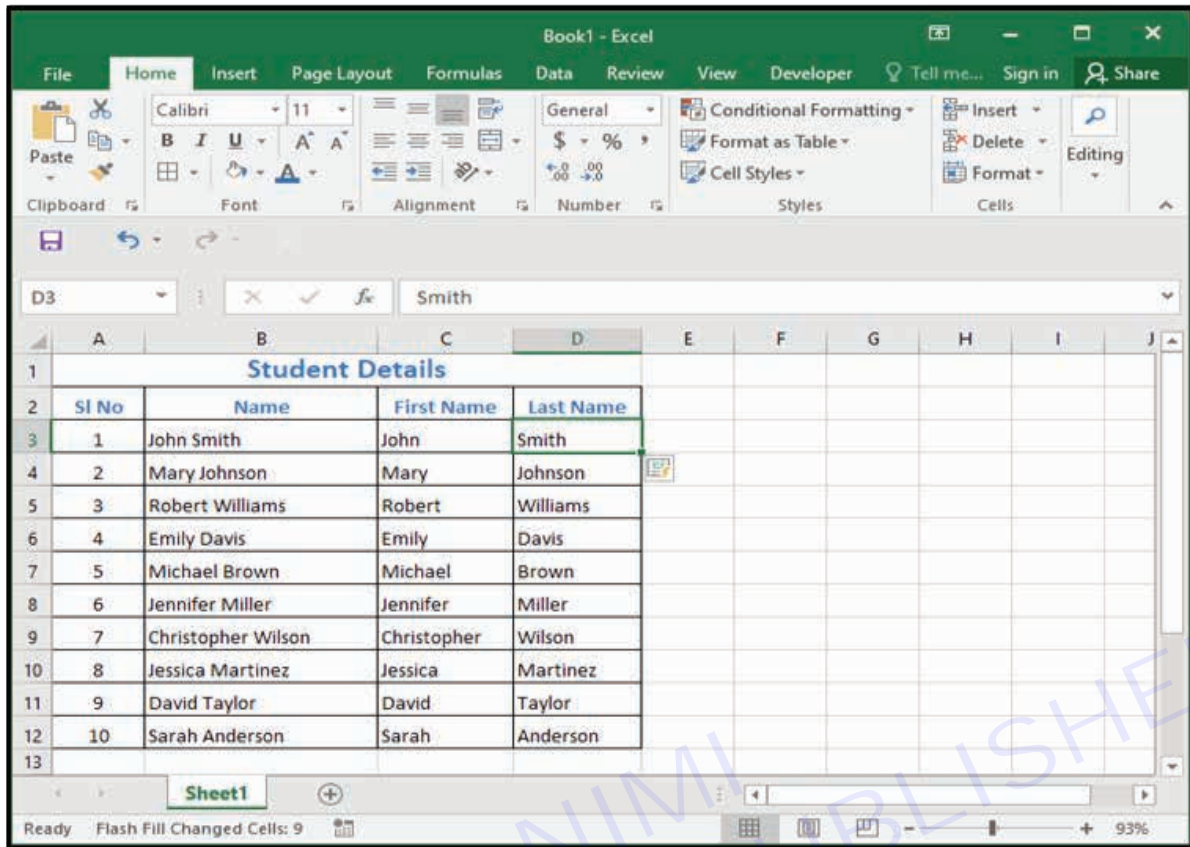


4 Review and Confirm:

- Excel will attempt to recognize the pattern and automatically fill the cells below with the expected results ie, all First Name will be displayed in the below cells.

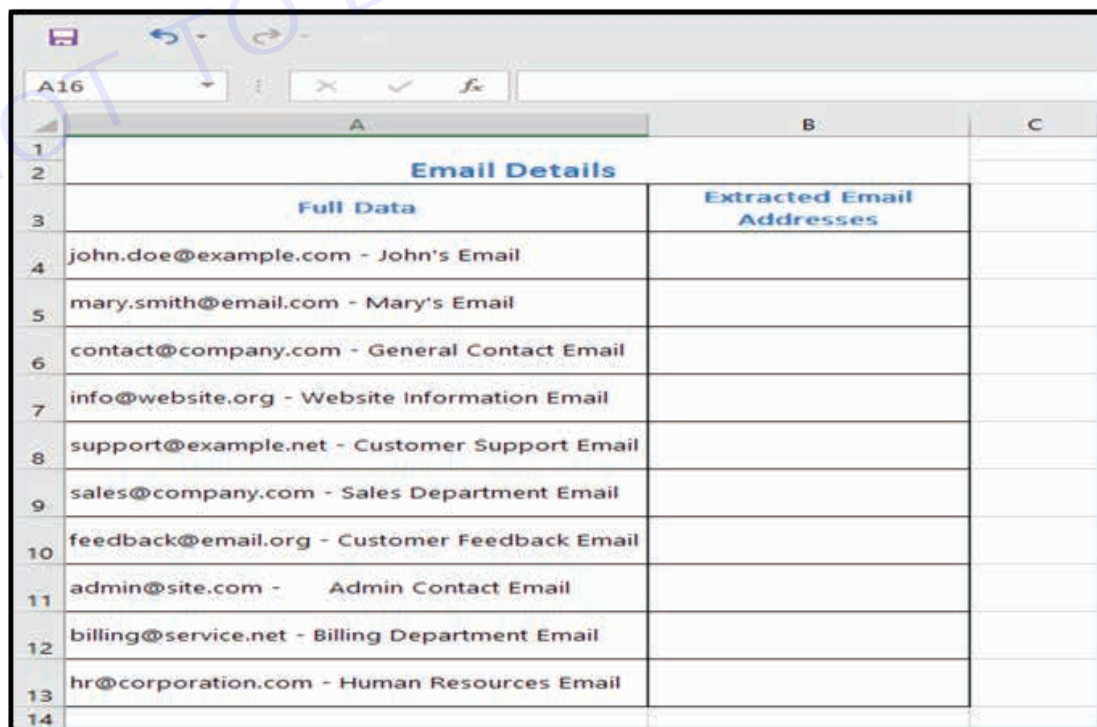


- Similar steps can be applied in the Last Name to get the result.



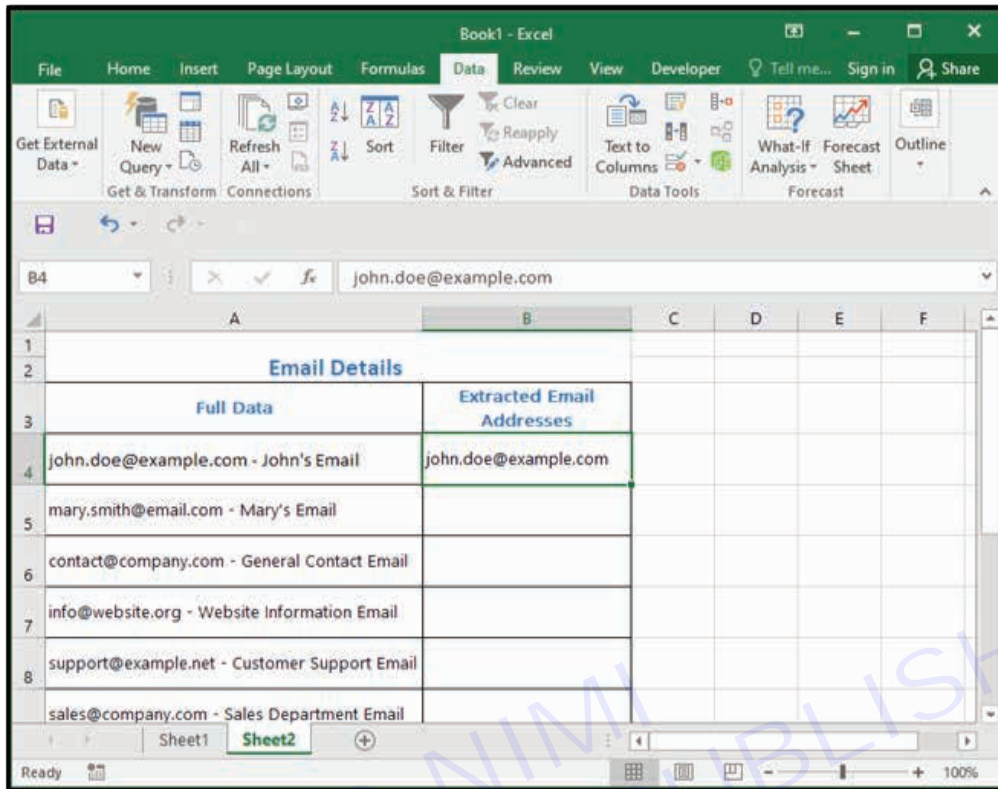
TASK 2: From a column containing full email addresses and other text, use Flash Fill to extract only the email addresses

1 Data Preparation:



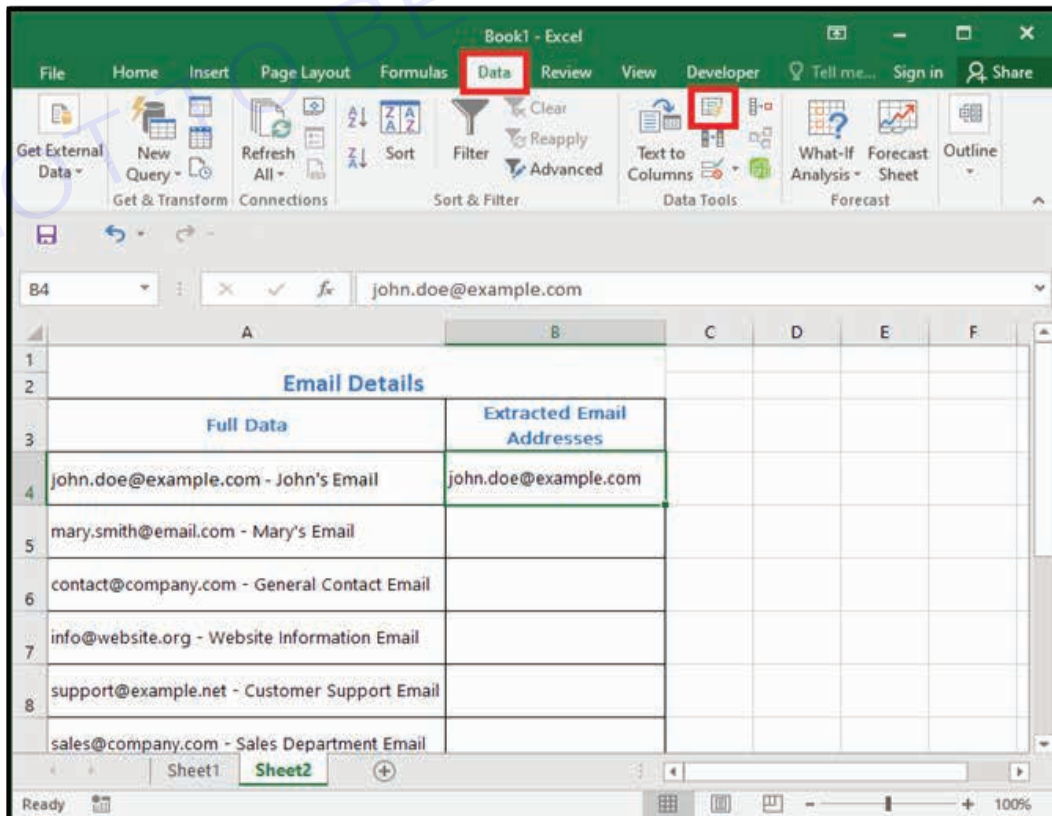
2 Activate Flash Fill:

In the adjacent cell where you want to apply Flash Fill, start typing the pattern you want Excel to recognize.



3 Execute Flash Fill:

Press **Ctrl + E** or go to the "Data" tab on the ribbon, and in the "Data Tools" group, click on "Flash Fill."



4 Review and Confirm:

Excel will attempt to recognize the pattern and automatically fill the cells below with the expected results.

	A	B	C
1	Email Details		
2	Full Data		
3	Extracted Email Addresses		
4	john.doe@example.com - John's Email	john.doe@example.com	
5	mary.smith@email.com - Mary's Email	mary.smith@email.com	
6	contact@company.com - General Contact Email	contact@company.com	
7	info@website.org - Website Information Email	info@website.org	
8	support@example.net - Customer Support Email	support@example.net	
9	sales@company.com - Sales Department Email	sales@company.com	
10	feedback@email.org - Customer Feedback Email	feedback@email.org	
11	admin@site.com - Admin Contact Email	admin@site.com	
12	billing@service.net - Billing Department Email	billing@service.net	
13	hr@corporation.com - Human Resources Email	hr@corporation.com	
14			

Related Exercises:

1 Extract the first names from the list of full names using Flash Fill?

Full Names	First Names
John Doe	
Jane Smith	
Alice Johnson	
Bob Brown	

2 Separate the email addresses into usernames and domains using Flash Fill.

Email Addresses	Username	Domain
john@example.com		
jane@example.com		
bob@example.com		
alice@example.com		

Hint: Separate the first email address into the "Username" and "Domain" columns (e.g., "john" in the Username column and "example.com" in the Domain column).

EXERCISE 65 : Perform Goal Seek, Solver & Scenarios on Data

Objectives

At the end of this exercise you shall be able to

- perform Goal seek on data sheet to achieve your target
- use excel solver for data optimization & complex problem solving
- use scenario for various data sheet analysis.

Requirements

Tools/Materials

- PC/Laptop with Windows OS
- MS Excel 2013 or Higher

Procedure

a Goal Seek :

Goal Seek in Excel is a feature that allows you to find the value of a particular cell that achieves a desired result in another cell by adjusting the value of one input cell. It's often used in financial modeling, engineering, and other fields where you need to determine the input necessary to achieve a specific outcome.

TASK 1: Imagine you are managing the financial data for a company. The NET SALES are determined by the formula $\text{NET SALES} = \text{GROSS SALES} * (\text{PROFIT}/100)$, and the ANNUAL PROFIT is the sum of the net sales from four quarters. Your goal is to set a target ANNUAL PROFIT (23, 00,000) and use the Goal Seek function to find the required PROFIT percentage for each quarter to achieve this target

1 Create an Excel table with the following columns:

Sales Details			
Quarter	Gross Sales	Profit	Net Sale
Q1	₹ 2,850,000	20	
Q2	₹ 3,155,000	20	
Q3	₹ 2,940,318	20	
Q4	₹ 0	20	
		Annual Profit	
		Net Profit	₹ 2,300,000

2 Formulas

- In cell C3 (Profit (%) for Q1), enter the initial profit percentage (e.g., 20%).
- In cell D9(Net Profit), enter the target annual profit value (e.g., ₹ 2,300,000).

c In cell D3 (Net Sales for Q1), enter the formula and press Enter Key.

Sales Details			
Quarter	Gross Sales	Profit	Net Sales
Q1	₹ 2,850,000	20	=[@[Gross Sales]]*([@Profit]/100)
Q2	₹ 3,155,000	20	
Q3	₹ 2,940,318	20	
Q4	₹ 0	20	
Annual Profit			
Net Profit			₹ 2,300,000

Drag it in the below cells to get the Net Sales Value in Q2, Q3 & Q4 as shown below:

Sales Details			
Quarter	Gross Sales	Profit	Net Sales
Q1	₹ 2,850,000	20	₹ 570,000
Q2	₹ 3,155,000	20	₹ 631,000
Q3	₹ 2,940,318	20	₹ 588,064
Q4	₹ 0	20	₹ 0
Annual Profit			
Net Profit			₹ 2,300,000

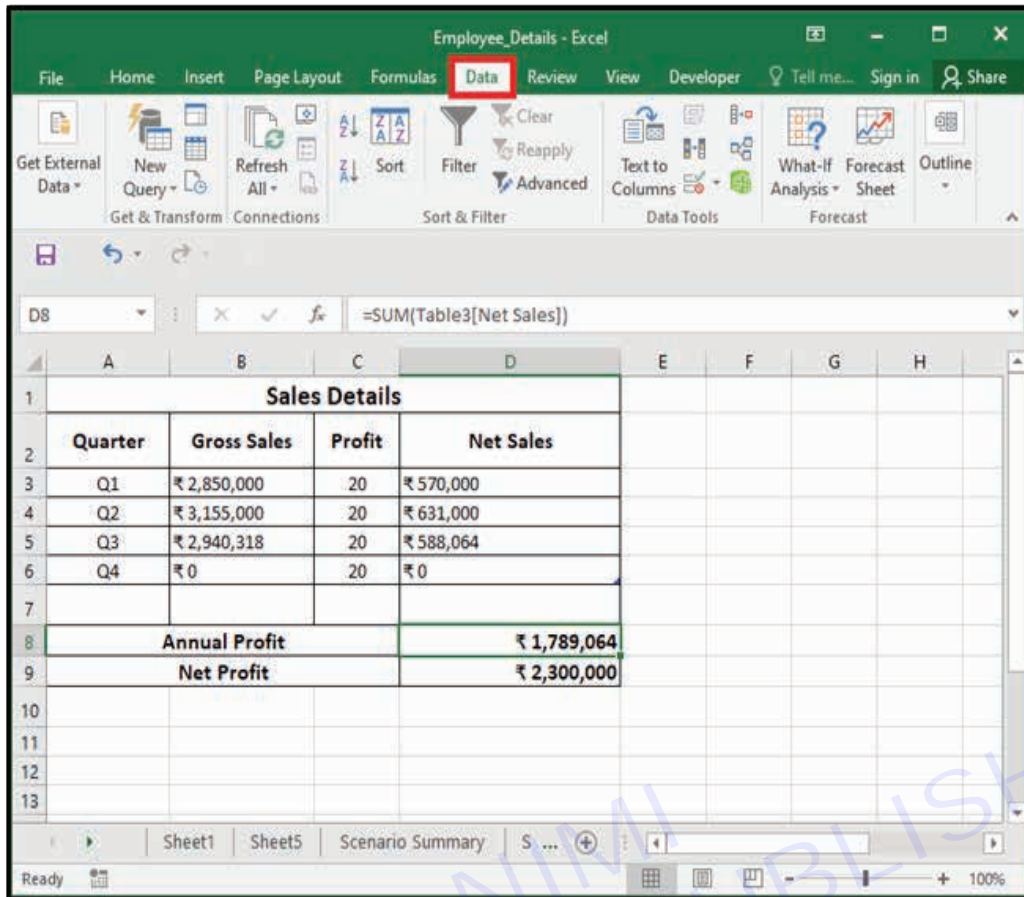
3 In cell D8 (SUM of NET SALES), enter the formula to sum the NET SALES from all four quarters:

Sales Details				
Quarter	Gross Sales	Profit	Net Sales	
Q1	₹ 2,850,000	20	₹ 570,000	
Q2	₹ 3,155,000	20	₹ 631,000	
Q3	₹ 2,940,318	20	₹ 588,064	
Q4	₹ 0	20	₹ 0	
Annual Profit			₹ 1,789,064	
Net Profit			₹ 2,300,000	

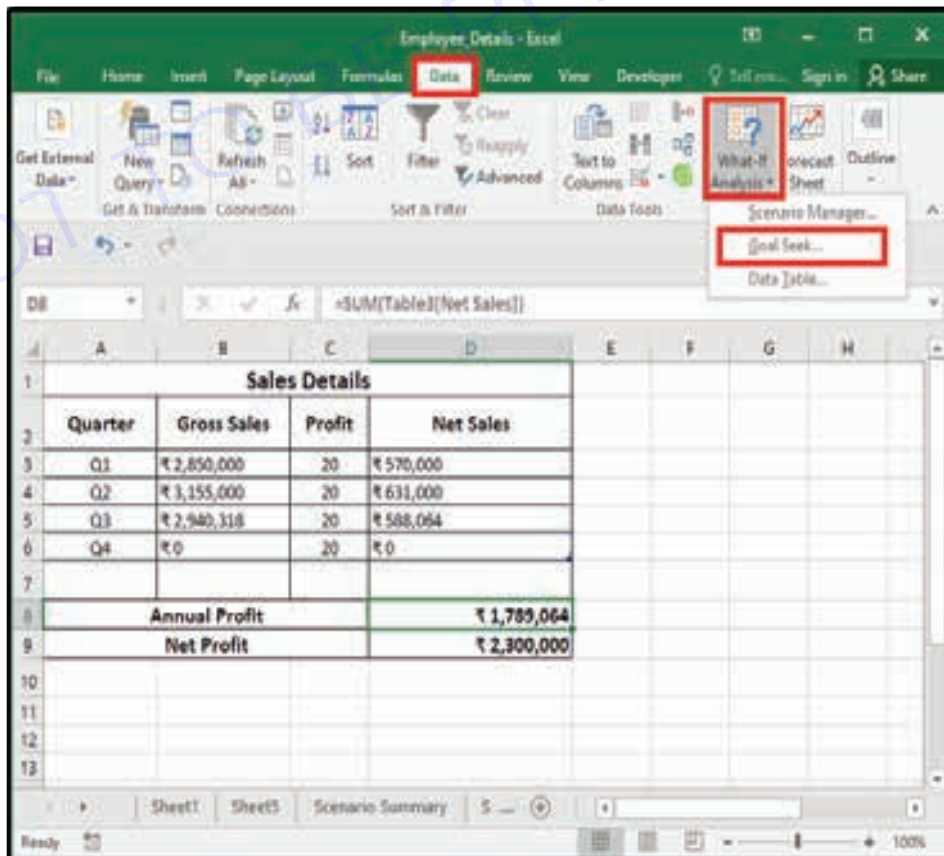
Sales Details				
Quarter	Gross Sales	Profit	Net Sales	
Q1	₹ 2,850,000	20	₹ 570,000	
Q2	₹ 3,155,000	20	₹ 631,000	
Q3	₹ 2,940,318	20	₹ 588,064	
Q4	₹ 0	20	₹ 0	
Annual Profit			₹ 1,789,064	
Net Profit			₹ 2,300,000	

4 Use Goal Seek

- a Select cell D8 (SUM of Net Sales).
- b Go to the "Data" tab in the Excel ribbon.



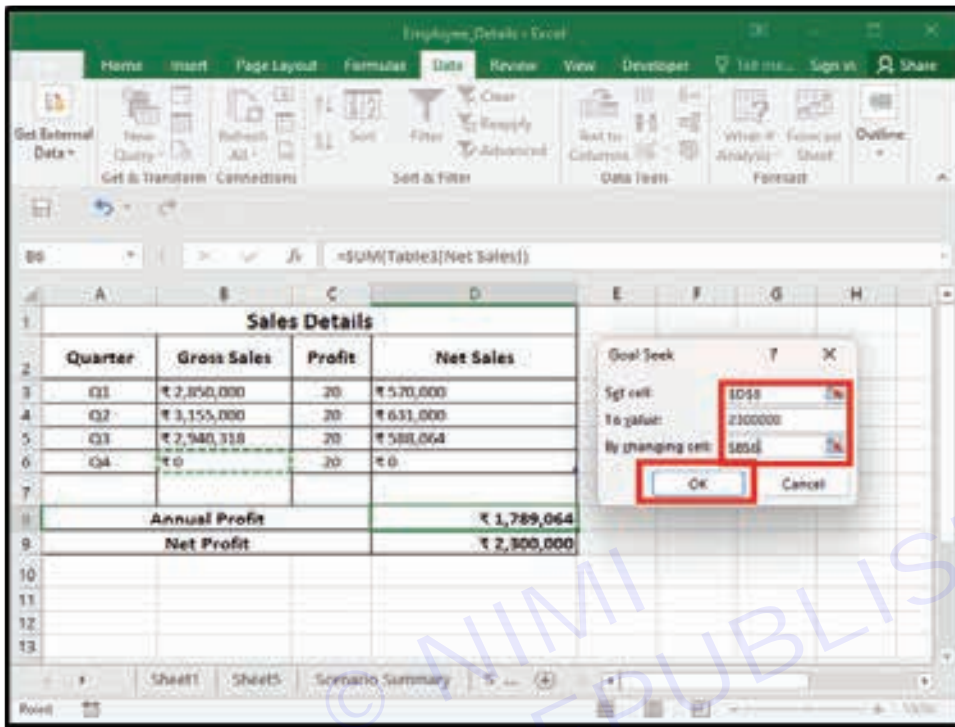
c Under “Data Tools,” find and click on “What-If Analysis,” then select “Goal Seek.”



4 Set Goal Seek Parameters

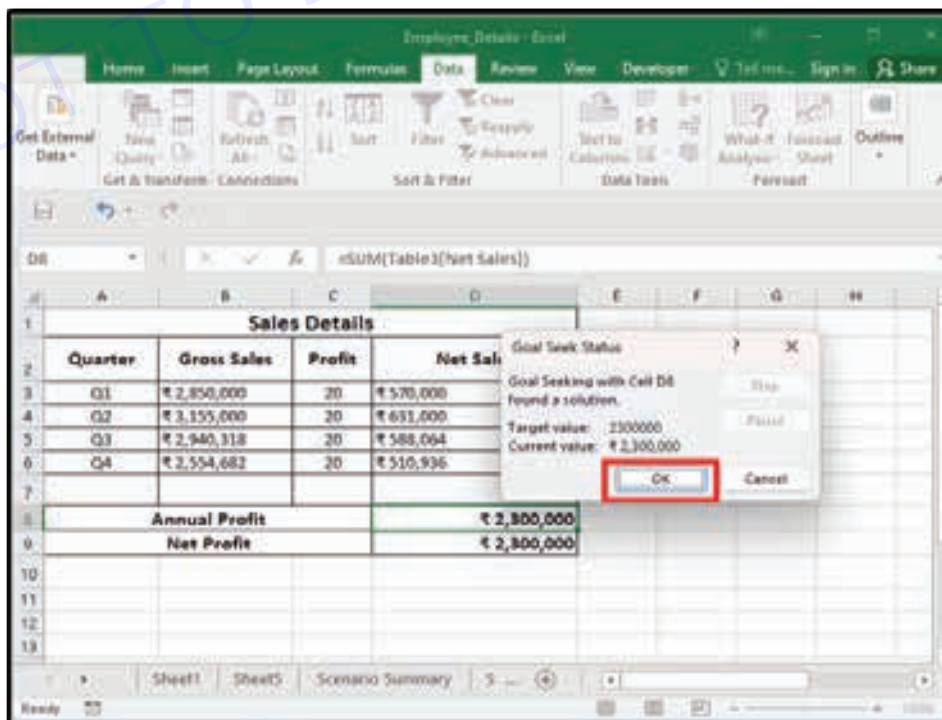
In the Goal Seek dialog box:

- Set “Set cell” to the cell containing the SUM of Net Sales (D8).
- Set “To value” to the desired Target PROFIT (23,00,000).
- Set “By changing cell” to the cell containing the initial PROFIT percentage for Q4 (B6).



5 Run Goal Seek

Click “OK” in the Goal Seek dialog box. Excel will perform calculations to find the required PROFIT percentage for each quarter to achieve the target Annual Profit.



6 Review the Result

Excel will automatically adjust the PROFIT percentages for each quarter to achieve the target Annual Profit of 2,300,000.

TASK 2: You want to use Goal Seek to determine the number of units that need to be produced to achieve a target production cost of 50,0000, assuming fixed costs of 20,0000 and a variable cost per unit of 150

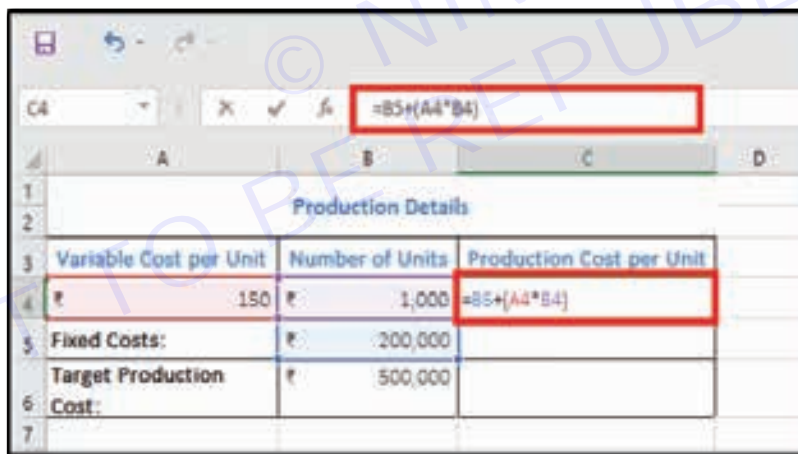
1 Set Up the Initial Table

Create an Excel table with the following columns:

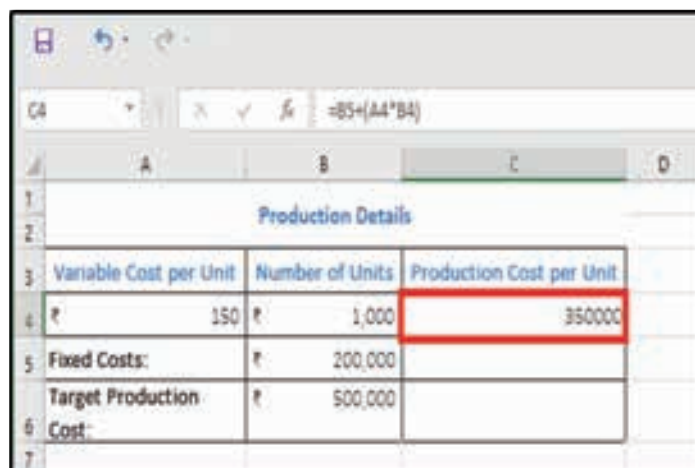
	A	B	C
1	Production Details		
2			
3	Variable Cost per Unit	Number of Units	Production Cost per Unit
4	₹ 150	₹ 1,000	
5	Fixed Costs:	₹ 200,000	
6	Target Production Cost:	₹ 500,000	

2 Formulas

- 1 In cell B4 (Number of Units), enter the initial number of units (e.g., 1,000).
- 2 In cell C4(Production Cost per Unit), enter the formula:

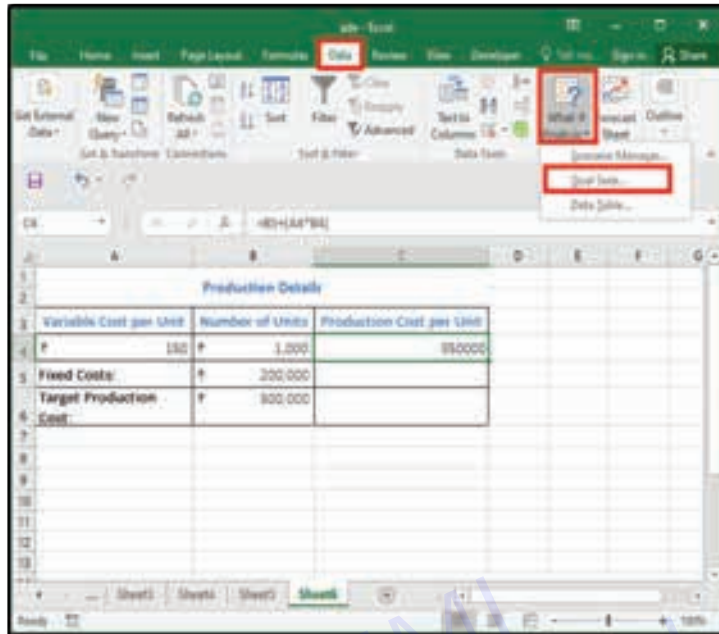


- 3 Press Enter



3 Use Goal Seek

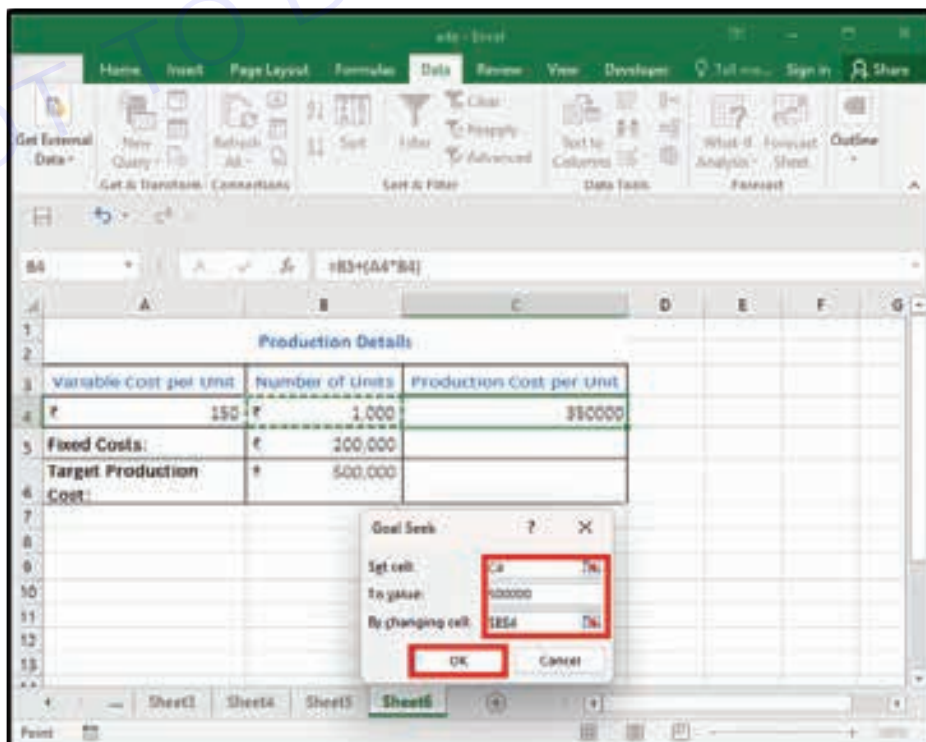
- 1 Select cell C4 (Production Cost per Unit).
- 2 Go to the “Data” tab in the Excel ribbon.
- 3 Under “Data Tools,” find and click on “What-If Analysis,” then select “Goal Seek.”



4 Set Goal Seek Parameters

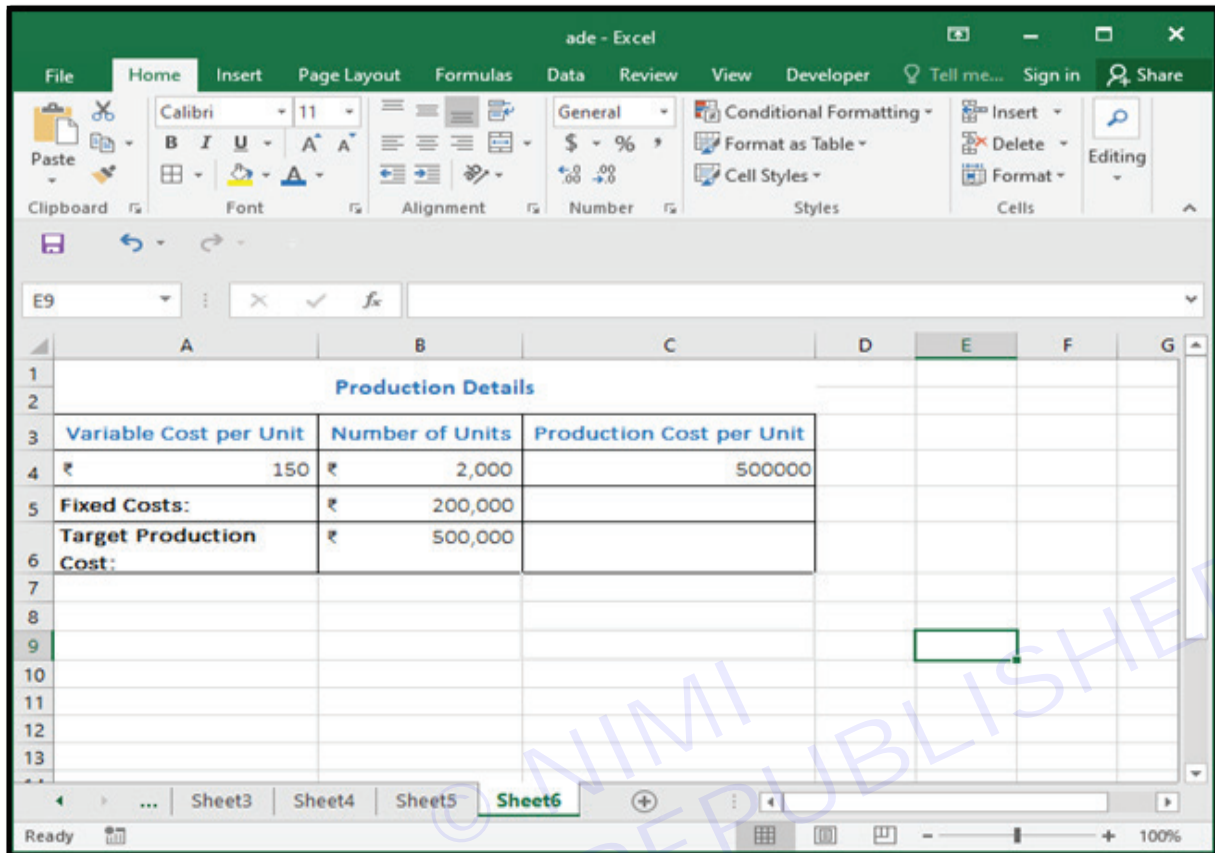
In the Goal Seek dialog box:

- Set “Set cell” to the cell containing the Production Cost per Unit (C4).
- Set “To value” to the desired Target Production Cost (Rs. 500,000).
- Set “By changing cell” to the cell containing the Number of Units (B4).



5 Run Goal Seek

Click “OK” in the Goal Seek dialog box. Excel will perform calculations to find the required Number of Units to achieve the target Production Cost.



Related Exercises:

- You are managing a project budget, and the total project cost (Total Cost) is determined by the formula:

$$\text{Total Cost} = \text{Fixed Costs} + (\text{Variable Cost per Unit} \times \text{Number of Units})$$
 You want to use Goal Seek to determine the required number of units if the fixed costs are \$10,000, and the target total cost is \$25,000.
- In a manufacturing process, the production time per unit (Production Time) is determined by the formula:

$$\text{Production Time} = \text{Fixed Time} + (\text{Variable Time per Unit} \times \text{Number of Units})$$
 You want to use Goal Seek to determine the required number of units if the fixed time is 5 hours, and the target production time is 20 hours.

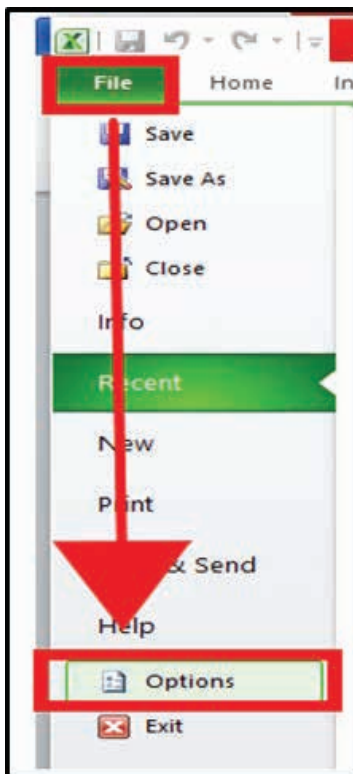
b Excel Solver:

Excel Solver is a powerful tool used for optimization and solving complex problems by finding the best solution based on a set of constraints. It is commonly used in operations research, engineering, finance, and other fields where optimization is required.

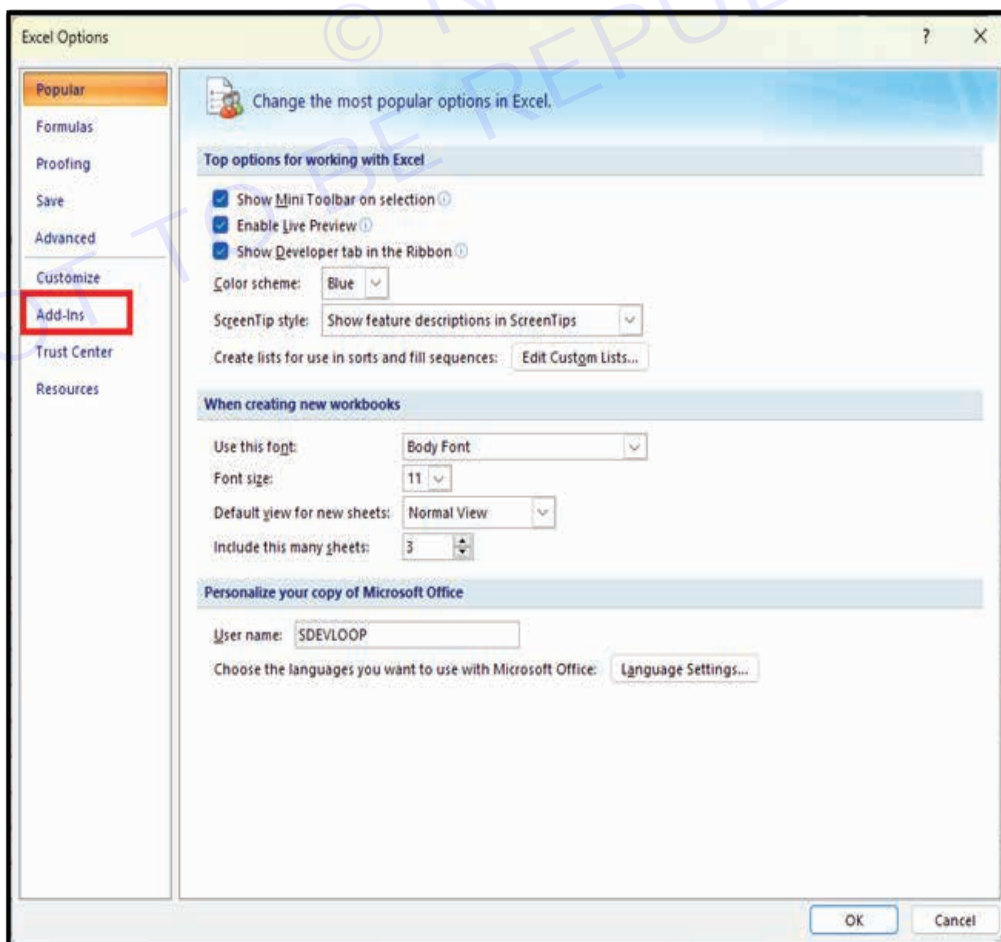
Add Solver to your Excel worksheet

The Solver add-in is added with all versions of Microsoft Excel though it is not enabled by default. You need to manually add Solver to your Excel worksheet. To incorporate a Solver into your Excel worksheet, follow the below-given steps.

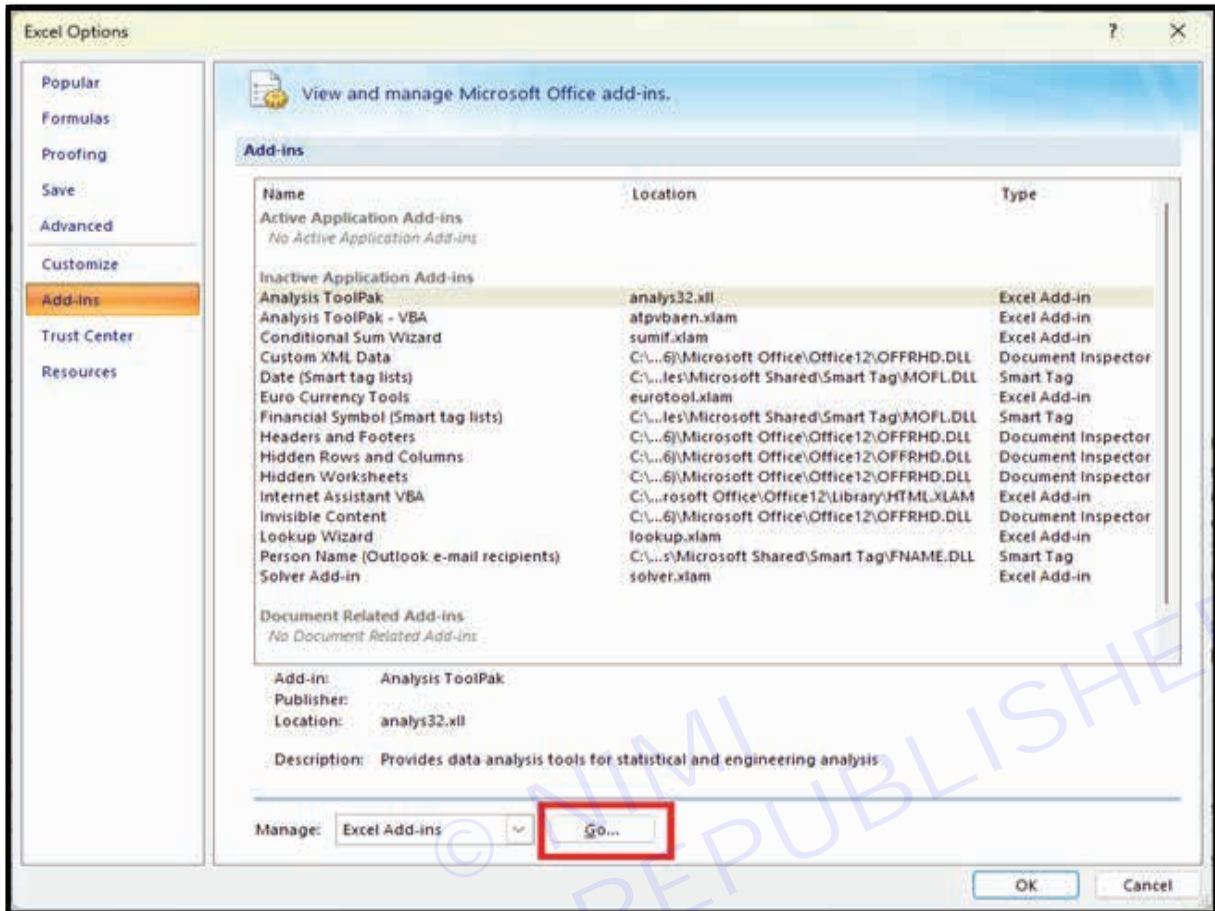
1 Open your Excel worksheet, click on **Files-> Options**.



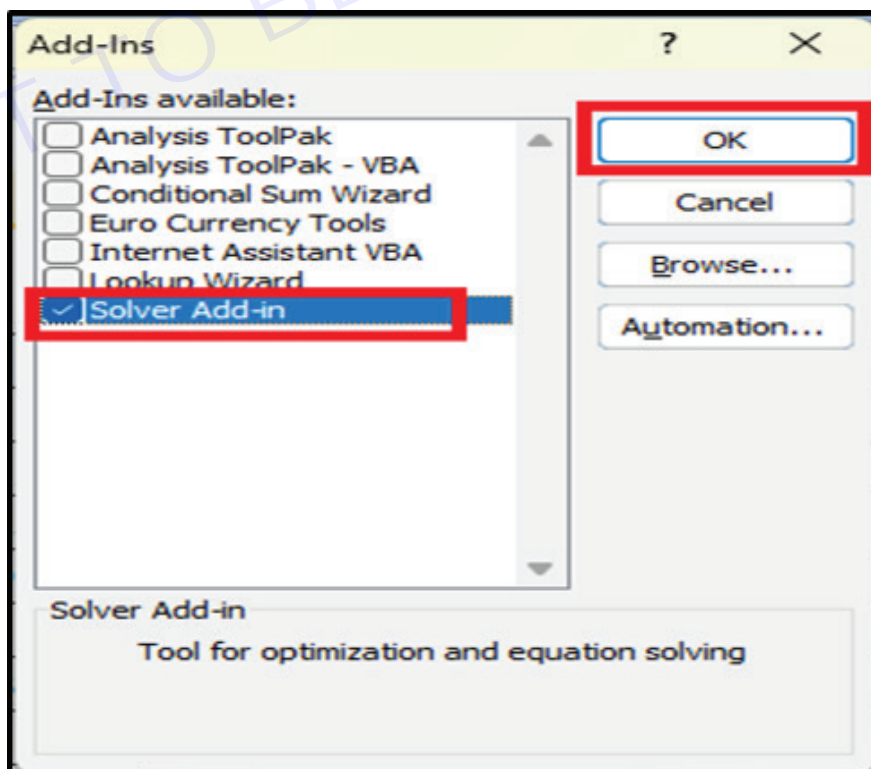
2 The MS Excel options dialog box will appear. From the left side of the pane, click on the Add-Ins options.



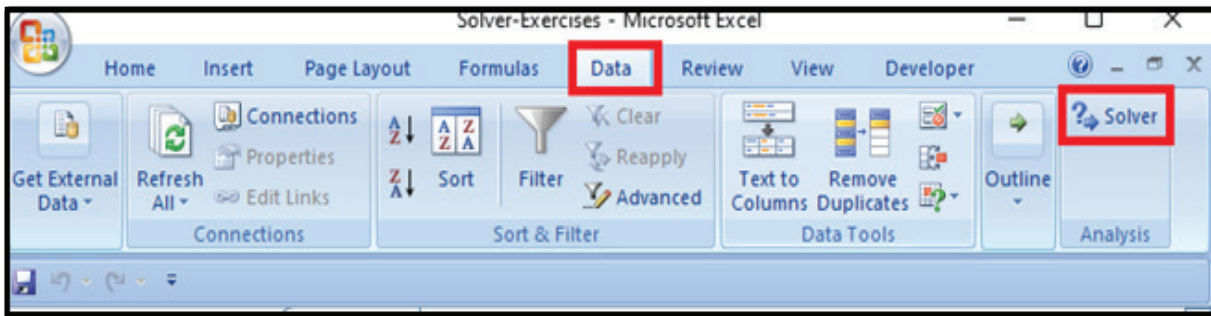
- 3 The Add-Ins screen will be displayed in the center. At the bottom of the screen where the **Manage** box is located, make sure that the **add-ins** option is selected in its field. Click on **Go**.



- 4 The **Add-Ins** dialog box will appear. Check the Solver 'Add-ins' box and click on the **OK** button.



- 5 That's it; it will add the Solver tool to your Excel Worksheet.
- 6 The Solver add-ins programming tool is located on the **Data tab, in the Analysis group.**



TASK 1: Application of Excel Solver to Get Maximize Profit of Products

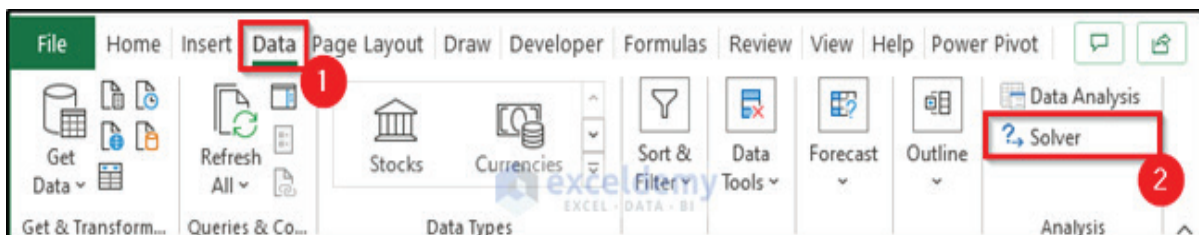
This company has some constraints that must be met to produce products:

- The combined production capacity is 300 units per day.
- The company needs 50 units of Software to fill an existing order.
- The company needs 40 units of Computer to fill an expected order.
- The company needs 40 units of Networking Kit to fill an existing order.

Products	Units	Profit/Unit	Profit
Software	25	₹ 6,000	₹ 150,000
Computer	25	₹ 50,000	₹ 1,250,000
Networkng Kit	25	₹ 3,000	₹ 75,000
Total	75		₹ 1,475,000

Step 1: First, go to the Data tab on your ribbon.

- Then select Solver from the Analysis group.



Step 2:

- Select cell E8 as the objective cell / Set Target Cell of the Solver Parameter box.
- Besides the To: options select Max as we are trying the maximize the value of the cell.

Products	Units	Profit/Unit	Profit
Software	25	₹ 600	₹ 15,000
Computer	25	₹ 5,000	₹ 125,000
Networking Kit	25	₹ 300	₹ 7,500
Total	75		₹ 147,500

Step 3:

- In the By Changing Variable Cells, select the cell values we are mainly focusing on changing. Here, they belong to the range C5:C7.

Products	Units	Profit/Unit	Profit
Software	25	₹ 600	₹ 15,000
Computer	25	₹ 5,000	₹ 125,000
Networking Kit	25	₹ 300	₹ 7,500
Total	75		₹ 147,500

Step 4:

- Now add the constraints by clicking on the Add button on the right of the box.

Products	Units	Profit/Unit	Profit
Software	25	₹ 600	₹ 15,000
Computer	25	₹ 5,000	₹ 125,000
Networking Kit	25	₹ 300	₹ 7,500
Total	75		₹ 147,500

- The company needs 50 units of Software to fill an existing order.

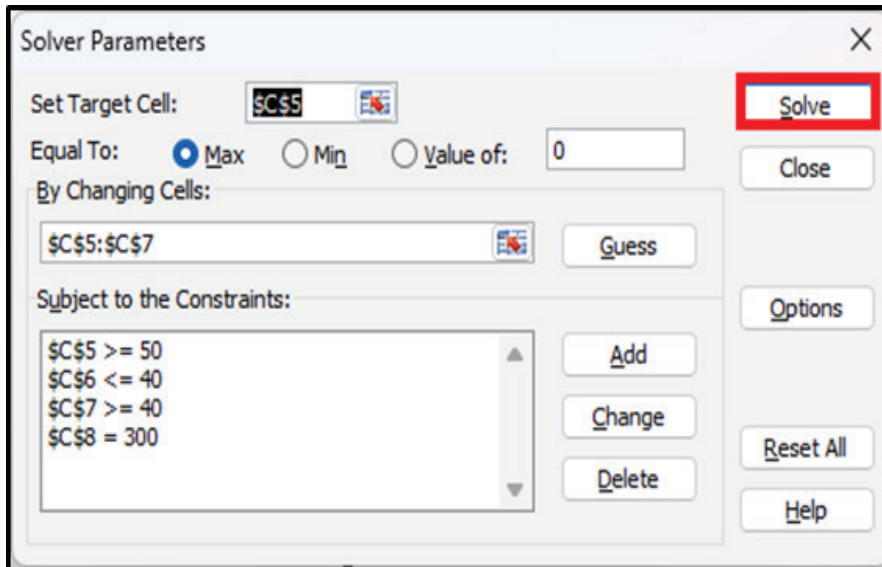
- The company needs 40 units of Computer to fill an expected order.

- The company needs 40 units of Networking Kit to fill an existing order.

- The combined production capacity is 300 units per day.

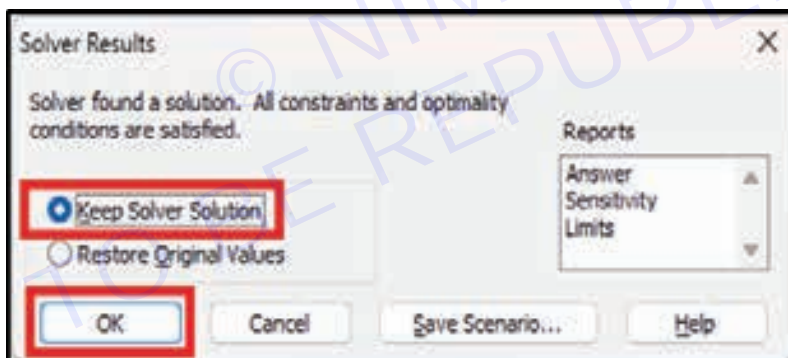
Step 5:

- Once you are done with all the steps above, click on Solve at the bottom of the box.



Step 6:

- After that, the Solver Results box will appear.
- Now select the options and reports you want to prefer in this box. For the demonstration, we are choosing to enable the Keep Solver Solution option only.



- Click on OK button, then you will get the output like this:

	A	B	C	D	E	F
1						
2		Maximize Profit of Products				
3						
4		Products	Units	Profit/Unit	Profit	
5		Software	220	₹ 600	₹ 132,000	
6		Computer	40	₹ 5,000	₹ 200,000	
7		Networking Kit	40	₹ 300	₹ 12,000	
8		Total	300		₹ 344,000	
9						

Related Exercise:

Question1 : Employee Scheduling

You are a manager responsible for scheduling employees for a customer service department. You want to minimize the total number of hours scheduled while ensuring adequate coverage during peak hours.

Shift	Monday	Tuesday	Wednesday	Thursday	Friday
9am - 5pm	4	3	4	5	4
1pm - 9pm	5	4	3	4	5
5pm - 1am	3	4	3	4	3

Question: How should you schedule employees for each shift to minimize total hours scheduled while meeting staffing requirements for each day?

Using Solver:

- 1 For each example, set up the objective function to maximize or minimize.
- 2 Define the decision variables and constraints based on the problem.
- 3 Access the Solver tool in Excel, specify the objective cell, decision variables, and constraints.
- 4 Run Solver to find the optimal solution.

c Scenarios on data

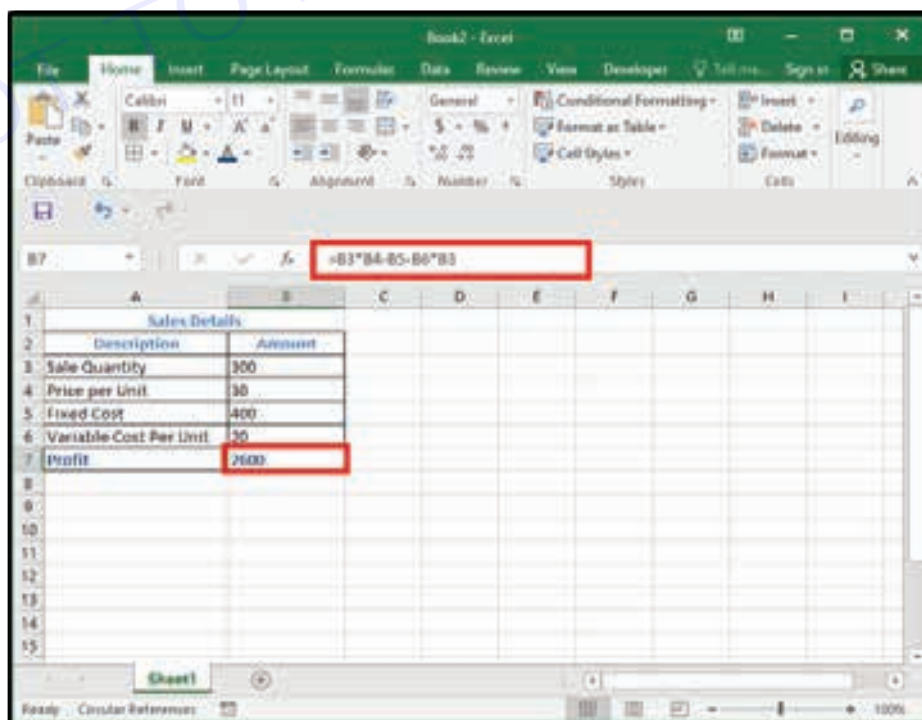
Excel scenarios allow you to create and save different sets of values that can be quickly substituted into your worksheet

TASK 1:

1 Create Dataset with Proper Parameters

Suppose, that we are going to sell a book and would like to know how the Sale Units, Price per Unit, and Variable Cost per Unit can affect the final profits. The profit is dependent on Sale Units (Cell B2), Price per Unit (Cell B3), and the Variable Cost per Unit (Cell B5). Therefore, type the below formula in cell B7.

= B3*B4-B5-B6*B3

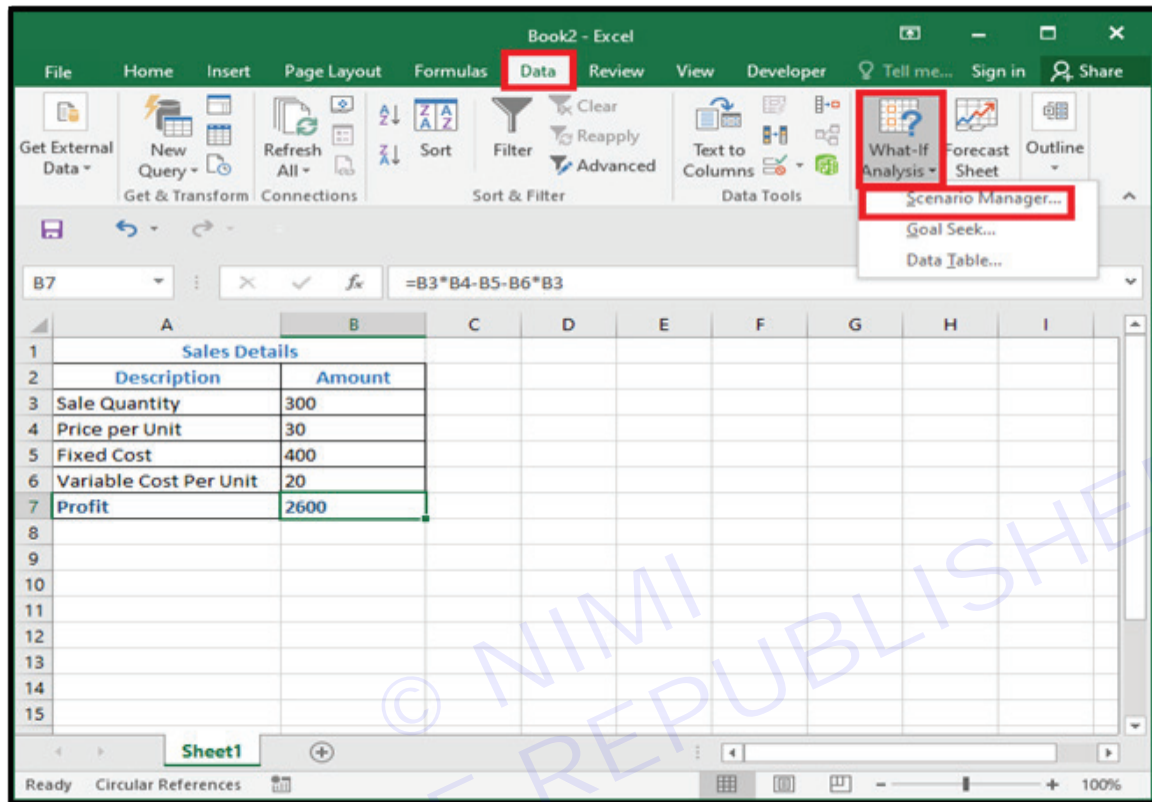


2 Make Scenario Manager

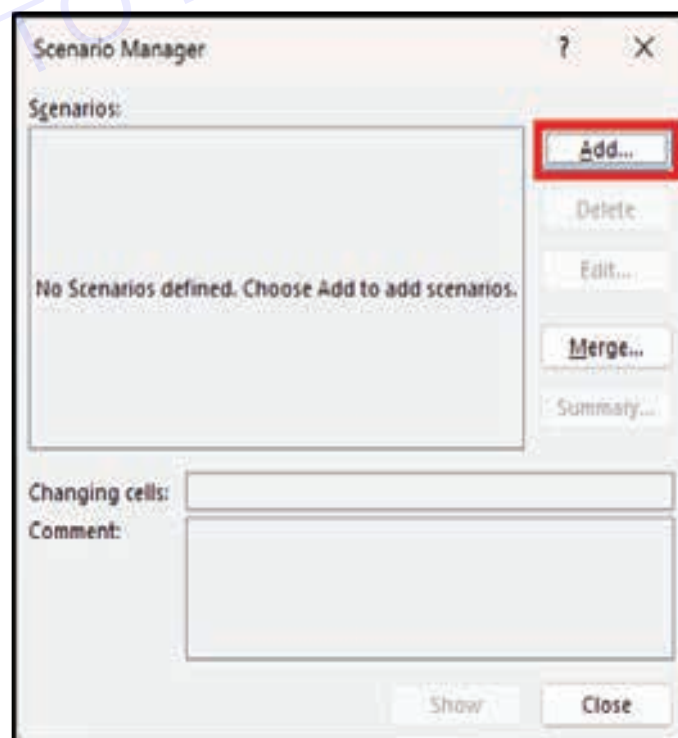
Now if you want to know how to use scenario manager in Excel, let's set up a Scenario Manager. To do that, follow the instructions below:

- First of all, from your Data tab, go to,

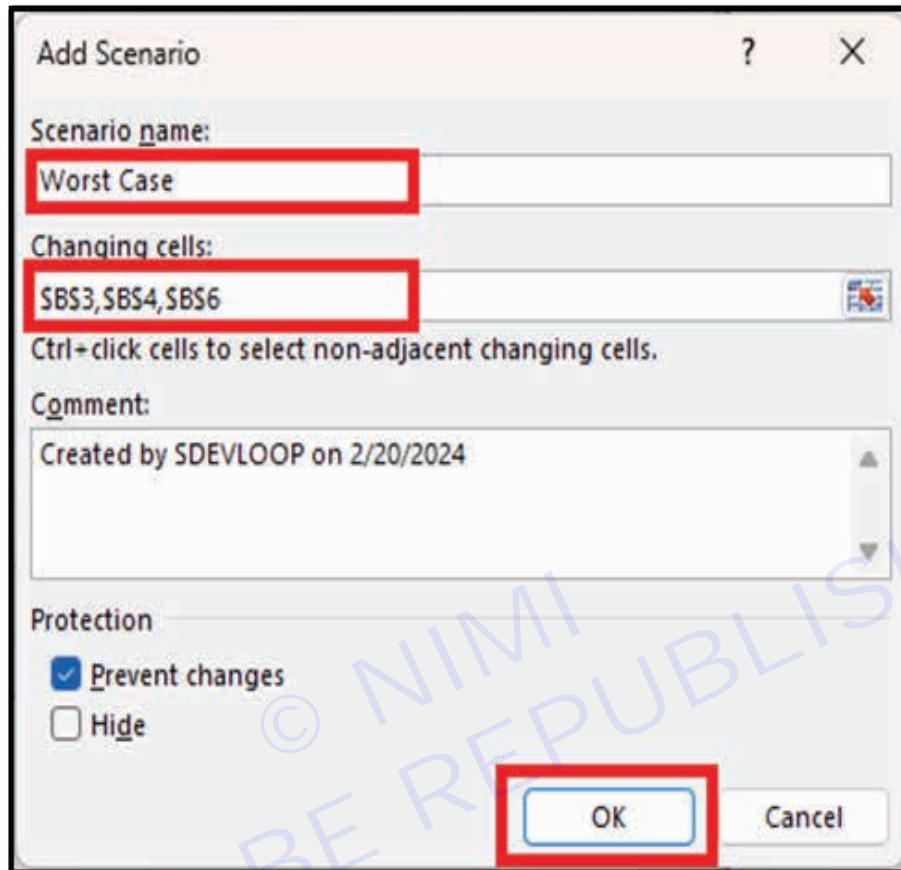
Data → Forecast → What-If Analysis → Scenario Manager



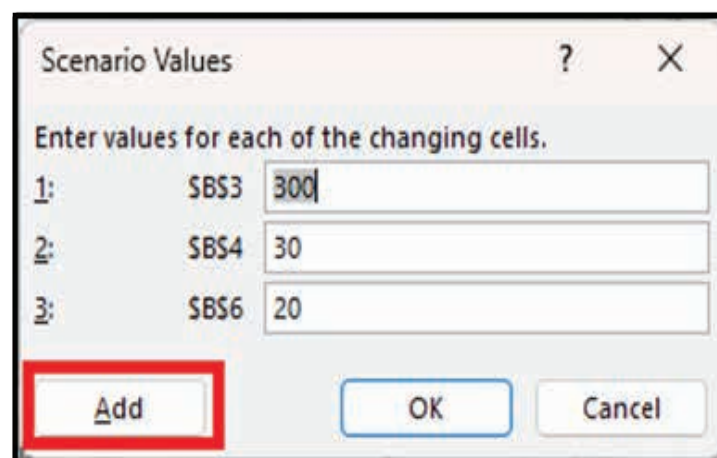
- As a result, a Scenario Manager dialogue box will appear in front of you. From the dialog box, click on Add option.



- In the prompted Add Scenario dialogue box, fill in the required details. Enter a name (Worst Case) for the Scenario name Add any comment that you wish to into the Comment box. Or you can also leave it blank. As for the Changing cells, fill in all the reference cells (B3,B4,B6 in this case) that contain the input values. Please note that the references must be separated by commas. Or, just press the CTRL key on your keyboard and select all the cells, one by one, that contain the input values. At last, press the OK option.



- Hence, the Scenario Values dialogue box pops up. Fill in the Scenario Values dialog box with the input values that define the worst case, and press the Add option to add another scenario. Click on OK, and the Worst Case scenario will be successfully created.



- Since we'd like to create another scenario, we click on Add. After clicking on Add, another Add Scenario dialogue box will appear. Use the same approach that we applied when creating the Worst Case scenario to build the Best Case scenario. Please note that Excel has set changing cells for the worst-case scenario as the default changing cells for the best-case scenario. The details are given in the following screenshot.

Scenario Values

Enter values for each of the changing cells.

1: SBS3 5000

2: SBS4 30

3: SBS6 20

Add OK Cancel

- With the same approach, create the Most Likely Case Here the below screenshot presents the details.

Scenario Values

Enter values for each of the changing cells.

1: SBS3 3000

2: SBS4 30

3: SBS6 15

OK Cancel

- You can also use the same above approach to creating other scenarios if you have other combinations of input values. In this example, we assume that there are only 3 scenarios available and thus we click on the OK button in the Scenario Values dialogue box. Now, you can see that three scenarios have been successfully created and they are listed in sequence. Click on Close, and the Scenario Manager dialogue box will be closed.

Scenario Manager

Scenarios:

Worst Case

Best Case

Most Likely Case

Add...

Delete

Edit...

Merge...

Summary...

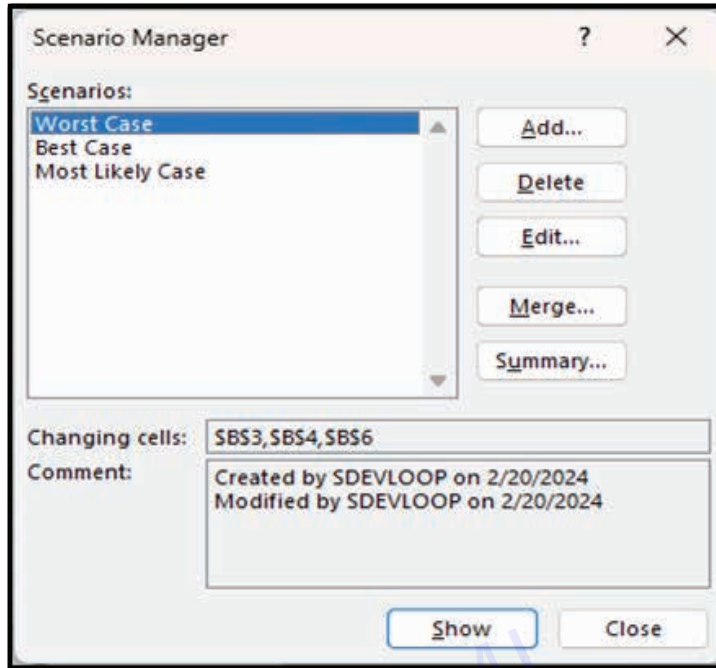
Changing cells: SBS3,SBS4,SBS6

Comment: Created by SDEVLOOP on 2/20/2024
Modified by SDEVLOOP on 2/20/2024

Show Close

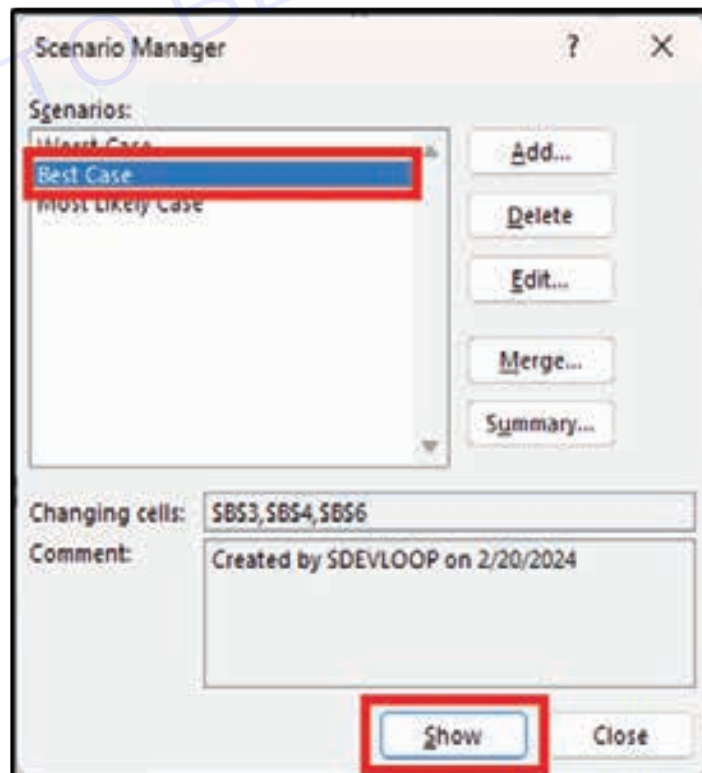
3 View Different Scenarios

So far, you have saved all of those 3 scenarios in your workbook. If you go to the Data tab and click on What-If Analysis in Forecast Group, then choose Scenario Manager in the drop-down, you will see the same Scenario Manager dialogue box as that is shown in the below screenshot.



The Scenario Manager dialogue box is no longer blank. Now you can view the result from each of the scenarios by simply double-clicking on any scenario.

To view a specific scenario and its corresponding outputs, you can click on that scenario and then click on the Show button at the bottom. For example, if we double-click on Best Case, the input values in the Excel worksheet will change into what has been filled for Best Case, and the output value will be calculated automatically based on the formula in cell B7. The dataset will change automatically.

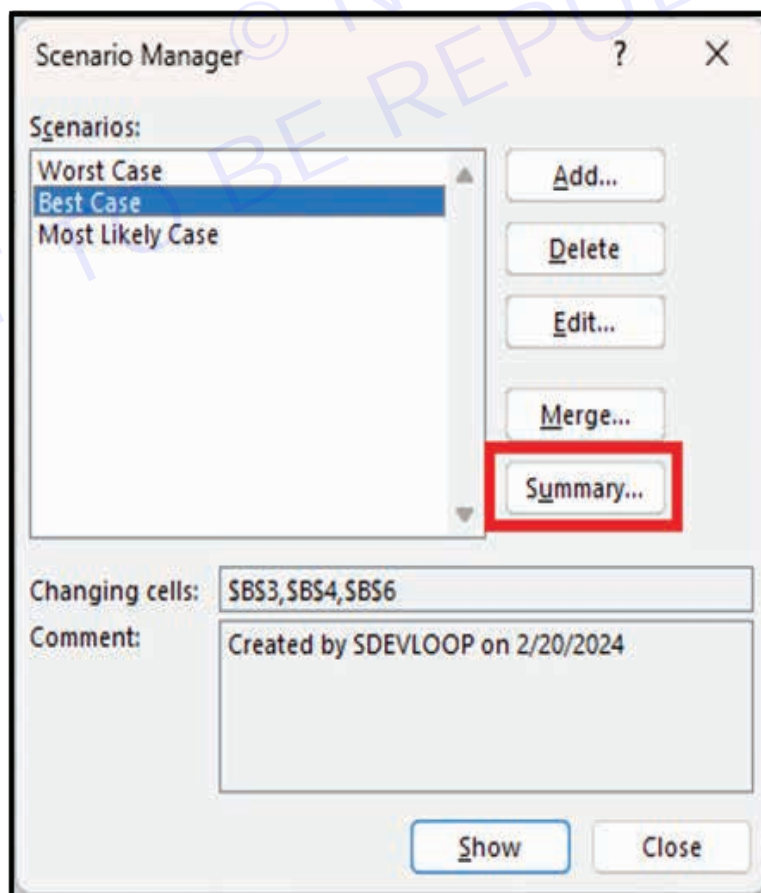


Sales Details	
Description	Amount
Sale Quantity	5000
Price per Unit	30
Fixed Cost	400
Variable Cost Per Unit	20
Profit	49600

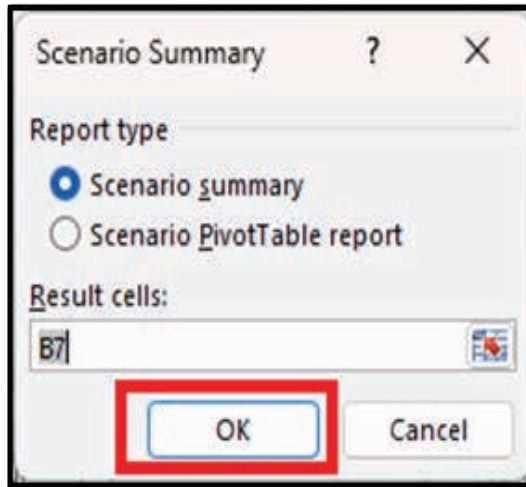
4 Create Scenario Summary Report in Excel

Excel can create a summary report based on the saved scenarios. Now let's see how to make a summary report. To do that, follow the instructions below:

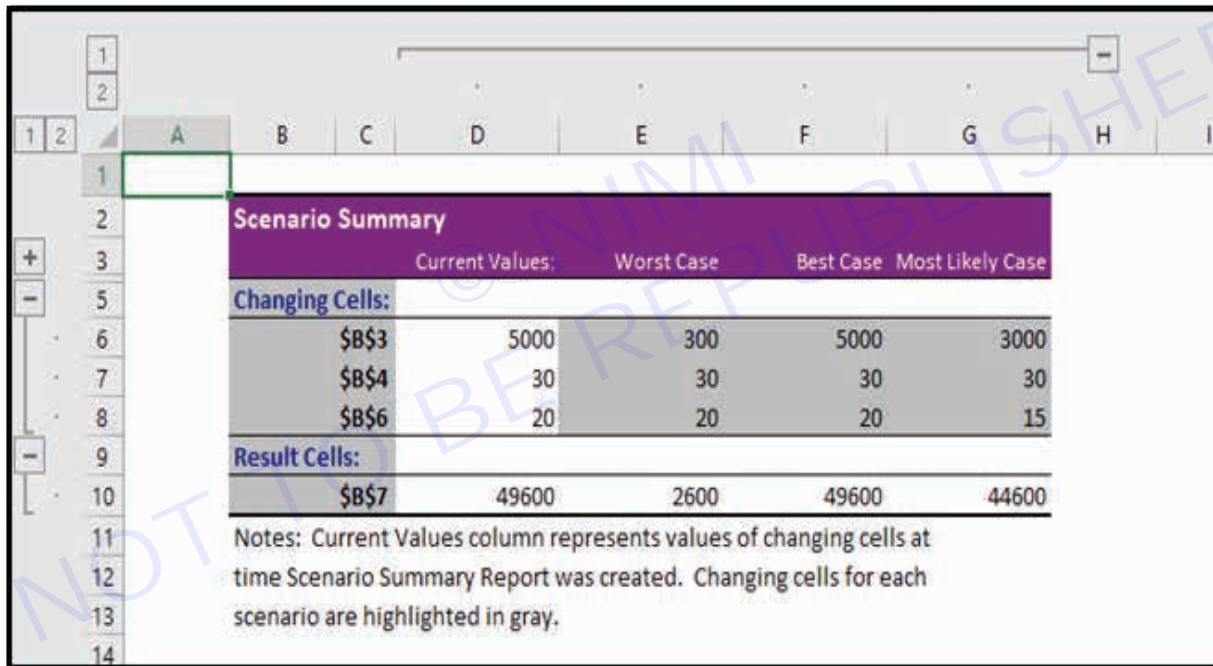
- First of all, from your Data tab, go to,
Data → **Forecast** → **What-If Analysis** → **Scenario Manager**
- After that, a Scenario Manager dialogue box will appear in front of you. From the Scenario Manager dialog box, click on the Summary option.



- After clicking on Summary, a Scenario Summary dialog box appears for you to put Result cells (B7 in this case) and choose between Scenario summary. At last, press the OK option.



As a result, you will be able to create the scenario summary report



Related Exercises:

Question 1: Sales Projection Scenarios

Suppose you have a sales projection table like this:

Month	Scenario A	Scenario B	Scenario C
January	1000	1200	900
February	1100	1300	950
March	1050	1250	920

Question: How can you use Excel scenarios to compare different sales projections for the first quarter?

Steps to do :

- 1 Fill in the actual budget allocations in the “Actual” column.
- 2 Go to the “Data” tab, click on “What-If Analysis,” and select “Scenario Manager.”
- 3 Click on “Add” to create new scenarios, naming them as needed (e.g., Scenario 1, Scenario 2).
- 4 Input different budget allocations for each scenario.
- 5 By switching between scenarios, you can compare the impact of different budget allocations on departmental spending.

Question 2: Project Timeline Scenarios

Imagine a project timeline table like this:

Task	Duration (Days)	Scenario 1	Scenario 2
Research	10		
Development	20		
Testing	15		
Deployment	5		

Question: How can you use scenarios to analyze different project completion timelines?

Steps to do:

- 1 Enter the duration of each task in the “Duration (Days)” column.
- 2 Go to the “Data” tab, click on “What-If Analysis,” and select “Scenario Manager.”
- 3 Create new scenarios (e.g., Scenario 1, Scenario 2) and input different durations for each task.
- 4 By comparing scenarios, you can assess how adjustments in task durations impact the overall project timeline.

EXERCISE 66 : Use different types of cell references

Objectives

At the end of this exercise you shall be able to

- use different types of cell references in your tasks.

Requirements

Tools/Materials

- PC/Laptop with Windows OS
- MS Excel 2013 or Higher

Procedure

In Excel, there are three main types of cell references: relative, absolute, and mixed. Using different types of cell references can be helpful when creating formulas that you plan to copy or fill to other cells.

TASK 1: Different types of Cell References

Method 1: Relative Cell Reference

A relative cell reference adjusts when you copy the formula to other cells.

In a scenario where the objective is to establish a formula P (Price) equals INR (Indian Rupees), the suggestion is to create a single formula in cell E3 and then copy it to other rows to avoid the need for a new formula for each row. The emphasis is on using relative references to ensure the formula calculates the total for each item correctly.

	A	B	C	D	E
1	INTEREST CALCULATION				
2	SL NO	NAME	SALARY	NO. OF YEAR	INTEREST RATE(12%)
3	1	ANU.R	4700	5	
4	2	ARUNIMA.R	3500	4	
5	3	ARYA.A.R	1100	8	
6	4	ASWATHY.K	2500	3	
7	5	BISMINA.S	4000	2	
8	6	ISHA.S.S	4500	7	
9	7	KARTHIKA. B.B	1500	6	
10	8	RESHMA.R	5500	3	
11	9	RIYA.L	3800	4	
12	10	SREEDHA.A	6000	9	
13					
14					

Step 1: Select the cell that will contain the formula. In our example, we'll select cell E3.

INTEREST CALCULATION						
SI No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5		12%	
2	ARUNIMA.R	3500	4			
3	ARYA.A.R	1100	8			
4	ASWATHY.K	2500	3			
5	BISMINA.S	4000	2			
6	ISHA.S.S	4500	7			
7	KARTHIKA. B.B	1500	6			
8	RESHMA.R	5500	3			
9	RIYA.L	3800	4			
10	SREEDHA.A	6000	9			

Step 2: Enter the formula to calculate the desired value. In our example, we'll type $=(C3*D3)*(12/100)$. And Press Enter on your keyboard. The formula will be calculated, and the result will be displayed in the cell.

INTEREST CALCULATION						
SI No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5	$=[@[Principal Amount]]*[@[No of Years]]*12/100$		
2	ARUNIMA.R	3500	4			
3	ARYA.A.R	1100	8			
4	ASWATHY.K	2500	3			
5	BISMINA.S	4000	2			
6	ISHA.S.S	4500	7			
7	KARTHIKA. B.B	1500	6			
8	RESHMA.R	5500	3			
9	RIYA.L	3800	4			
10	SREEDHA.A	6000	9			

Step 3: Select the cell E3. The fill handle will appear in the bottom-right corner of the cell.

INTEREST CALCULATION						
SI No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5	2820	12%	
2	ARUNIMA.R	3500	4			
3	ARYA.A.R	1100	8			
4	ASWATHY.K	2500	3			
5	BISMINA.S	4000	2			
6	ISHA.S.S	4500	7			
7	KARTHIKA. B.B	1500	6			
8	RESHMA.R	5500	3			
9	RIYA.L	3800	4			
10	SREEDHA.A	6000	9			

Step 4: Click and drag the fill handle over the cells. Select cells E3:E12, Release the mouse. The formula will be copied to the selected cells with relative references, displaying the result in each cell.

INTEREST CALCULATION						
SI No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5	2820	12%	
2	ARUNIMA.R	3500	4	1680		
3	ARYA.A.R	1100	8	1056		
4	ASWATHY.K	2500	3	900		
5	BISMINA.S	4000	2	960		
6	ISHA.S.S	4500	7	3780		
7	KARTHIKA. B.B	1500	6	1080		
8	RESHMA.R	5500	3	1980		
9	RIYA.L	3800	4	1824		
10	SREEDHA.A	6000	9	6480		

Note: Double-click the filled cells to check formulas for accuracy. The relative cell references should be different for each cell, depending on their rows.

Method 2: Absolute references

Excel Absolute Reference refers to a 'locked' reference so that the address of its corresponding row and column does not change when copied.

- \$A\$2 –Denote the column and the row do not changed when copied

In the example below, we're going to use cell E3 (which contains the Interest rate at 12%) to calculate the interest for each item.

INTEREST CALCULATION						
SI No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5		12%	
2	ARUNIMA.R	3500	4			
3	ARYA.A.R	1100	8			
4	ASWATHY.K	2500	3			
5	BISMINA.S	4000	2			
6	ISHA.S.S	4500	7			
7	KARTHIKA. B.B	1500	6			
8	RESHMA.R	5500	3			
9	RIYA.L	3800	4			
10	SREEDHA.A	6000	9			

Step 1: Input the formula in the selected cell (E3) to compute the desired value. In this instance, the formula will be $= (C3 * D3) * (\$F\$3)$, with $\$F\3 being specified as an absolute reference.

INTEREST CALCULATION						
SI No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5	$=[@[Principal Amount]]*[@[No of Years]]*\$F\$3$	12%	
2	ARUNIMA.R	3500	4			
3	ARYA.A.R	1100	8			
4	ASWATHY.K	2500	3			
5	BISMINA.S	4000	2			
6	ISHA.S.S	4500	7			
7	KARTHIKA. B.B	1500	6			
8	RESHMA.R	5500	3			
9	RIYA.L	3800	4			
10	SREEDHA.A	6000	9			

Step 2: Press Enter on your keyboard. The formula will calculate, and the result will display in the cell.

INTEREST CALCULATION						
Sl No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5	2820	12%	
2	ARUNIMA.R	3500	4			
3	ARYA.A.R	1100	8			
4	ASWATHY.K	2500	3			
5	BISMINA.S	4000	2			
6	ISHA.S.S	4500	7			
7	KARTHIKA. B.B	1500	6			
8	RESHMA.R	5500	3			
9	RIYA.L	3800	4			
10	SREEDHA.A	6000	9			

Step 3: Click and drag the fill handle across the cells where you intend to replicate the formula. Once you release the mouse, the formula will be copied to the designated cells, retaining the absolute reference, and the values will be automatically calculated in each corresponding cell.

INTEREST CALCULATION						
Sl No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5	2820	12%	
2	ARUNIMA.R	3500	4	1680		
3	ARYA.A.R	1100	8	1056		
4	ASWATHY.K	2500	3	900		
5	BISMINA.S	4000	2	960		
6	ISHA.S.S	4500	7	3780		
7	KARTHIKA. B.B	1500	6	1080		
8	RESHMA.R	5500	3	1980		
9	RIYA.L	3800	4	1824		
10	SREEDHA.A	6000	9	6480		

Double-click the filled cells to check their formulas for accuracy. The absolute reference should be the same for each cell, while the other references are relative to the cell's row.

Method 3: Mixed references

A Mixed cell reference is a mixture of both relative and absolute cell reference. In mixed cell reference, dollar signs are attached to either the row letter or the column number.

- \$A2 – Denote the column does not changed when copied
- A\$2 – Denote the row does not changed when copied.

INTEREST CALCULATION						
SI No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5		12%	
2	ARUNIMA.R	3500	4			
3	ARYA.A.R	1100	8			
4	ASWATHY.K	2500	3			
5	BISMINA.S	4000	2			
6	ISHA.S.S	4500	7			
7	KARTHIKA. B.B	1500	6			
8	RESHMA.R	5500	3			
9	RIYA.L	3800	4			
10	SREEDHA.A	6000	9			

Step 1: Input the formula in the selected cell (E3) to compute the desired value. In this instance, the formula will be = (\$C3*\$D3)/(\$F\$3), with \$F\$3 being specified as an absolute reference and \$C3 and \$D3 is the mixed reference.

INTEREST CALCULATION						
SI No	Name	Principal Amount	No of Years	Interest	Rate of Interest	
1	ANU.R	4700	5	=(\$C3*\$D3)/(\$F\$3)	12%	
2	ARUNIMA.R	3500	4			
3	ARYA.A.R	1100	8			
4	ASWATHY.K	2500	3			
5	BISMINA.S	4000	2			
6	ISHA.S.S	4500	7			
7	KARTHIKA. B.B	1500	6			
8	RESHMA.R	5500	3			
9	RIYA.L	3800	4			
10	SREEDHA.A	6000	9			

Step 2: Press Enter on your keyboard. The formula will calculate, and the result will display in the cell.

INTEREST CALCULATION						
SL NO	NAME	Principal Amount	NO. OF YEAR	INTEREST RATE		
1	ANU.R	4700	5	2820	Interest	12%
2	ARUNIMA.R	3500	4			
3	ARYA.A.R	1100	8			
4	ASWATHY.K	2500	3			
5	BISMINA.S	4000	2			
6	ISHA.S.S	4500	7			
7	KARTHIKA. B.B	1500	6			
8	RESHMA.R	5500	3			
9	RIYA.L	3800	4			
10	SREEDHA.A	6000	9			

Step 3: Click and drag the fill handle across the cells where you intend to replicate the formula. Once you release the mouse, the formula will be copied to the designated cells, retaining the absolute reference, and the values will be automatically calculated in each corresponding cell

INTEREST CALCULATION						
Sl No	Name	Principal Amount	No of Years	Interest		Rate of Interest
1	ANU.R	4700	5	2820		12%
2	ARUNIMA.R	3500	4	1680		
3	ARYA.A.R	1100	8	1056		
4	ASWATHY.K	2500	3	900		
5	BISMINA.S	4000	2	960		
6	ISHA.S.S	4500	7	3780		
7	KARTHIKA. B.B	1500	6	1080		
8	RESHMA.R	5500	3	1980		
9	RIYA.L	3800	4	1824		
10	SREEDHA.A	6000	9	6480		

Related Exercises:**Question 1: Sales Commission Calculation**

Suppose you work as a sales representative and earn a commission based on the total sales amount. Your commission rate is 5% of total sales. You have a table with sales data for different months.

Month	Sales Amount (\$)	Commission Earned (\$)
January	\$5,000	
February	\$7,500	
March	\$6,200	
April	\$8,000	
May	\$9,500	

Calculate the commission earned for each month using a formula with absolute references?

Question 2: Product Inventory Management

Suppose you run a small retail store and want to track the inventory levels of different products. You have a table with the product names and their current inventory counts:

Product	Inventory Count
Product A	100
Product B	50
Product C	75
Product D	120
Product E	80

Calculate the percentage of inventory remaining for each product using a formula with absolute references?

Solution: You can calculate the percentage of inventory remaining as (Inventory Count / Total Inventory) * 100%.

Question 3: Monthly Expense Analysis

Suppose you are analyzing your monthly expenses and want to calculate the percentage of each expense category relative to the total expenses. You have a table with the expense categories and their amounts:

Category	Amount (\$)
Groceries	\$300
Dining Out	\$200
Utilities	\$150
Transportation	\$100
Entertainment	\$150

Calculate the percentage of each expense category relative to the total expenses using a formula with relative references?

Solution: You can calculate the percentage of each expense category relative to the total expenses by dividing the amount of each category by the total expenses, multiplied by 100. Here's the formula you can use in the "Percentage of Total Expenses (%)" column:

$$= (B2 / \text{SUM}(B\$2:B\$6)) * 100$$

EXERCISE 67 : Use R1C1 notation

Objectives

At the end of this exercise you shall be able to

- use R1C1 notation in your worksheet.

Requirements

Tools/Materials

- PC/Laptop with Windows OS
- MS Excel 2013 or Higher

Procedure

R1C1 notation is an alternative way of referencing cells in Microsoft Excel, as opposed to the more common A1 notation. In R1C1 notation:

- R stands for Row.
- C stands for Column.
- The numbers following R and C represent the row and column numbers, respectively.

The basic format of an R1C1 reference is **R[row]C[column]**.

For example:

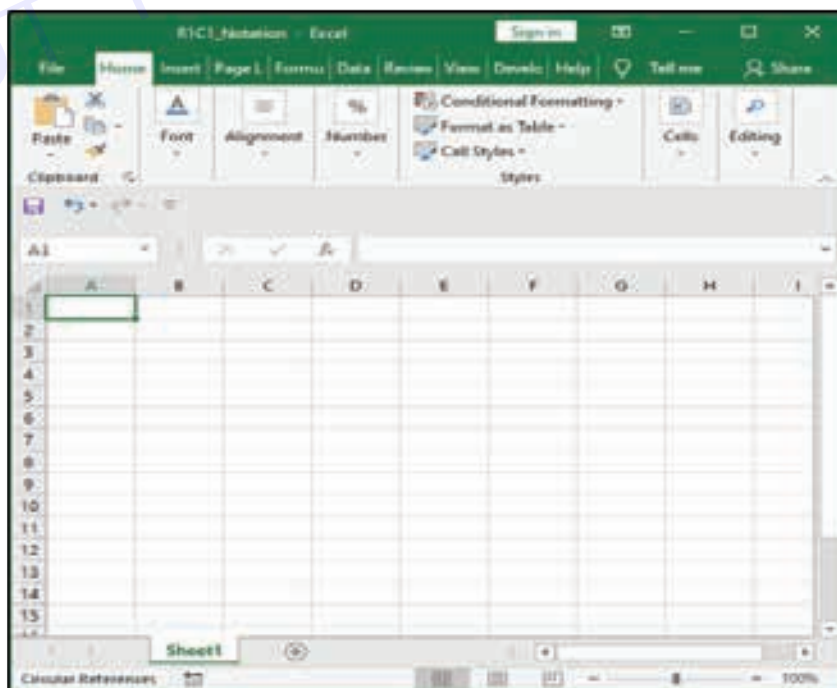
- R1C1 refers to the cell in the first row and first column.
- R2C3 refers to the cell in the second row and third column.

In addition to explicit cell references, you can use relative references in R1C1 notation. For instance:

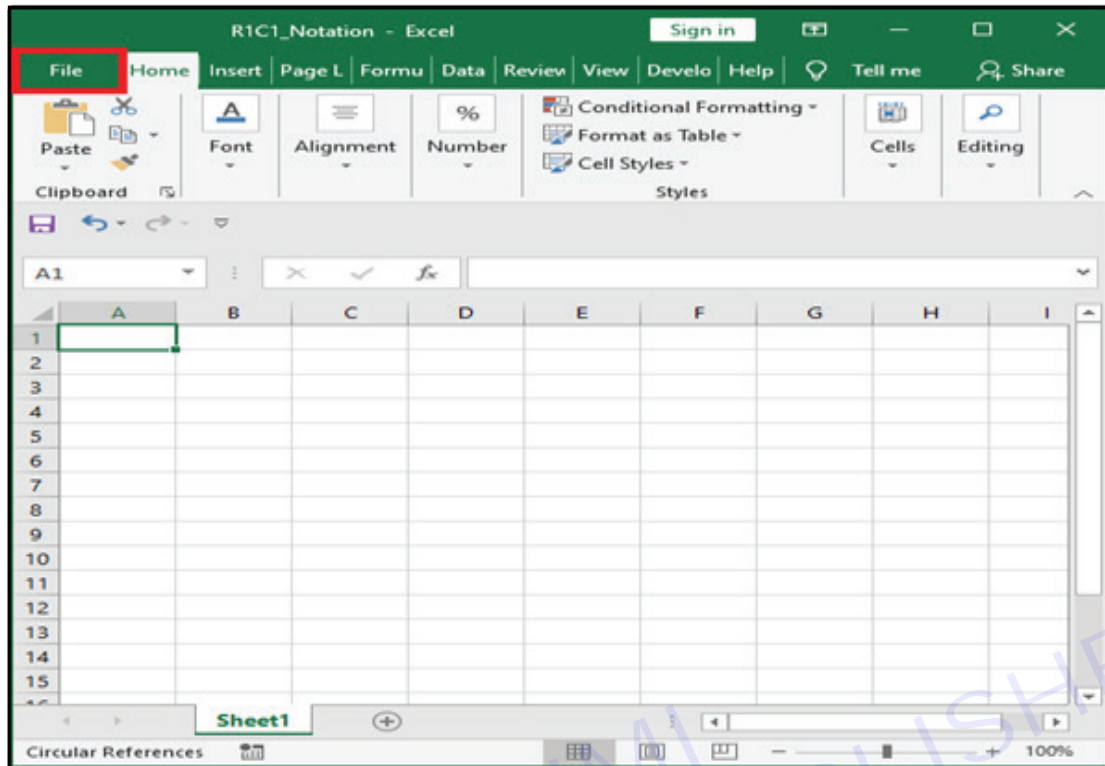
- RC[-1] refers to the cell in the same row and one column to the left of the active cell/current cell.
- R[2]C refers to the cell two rows below and in the same column.

How to Switch to R1C1 Notation in Excel:

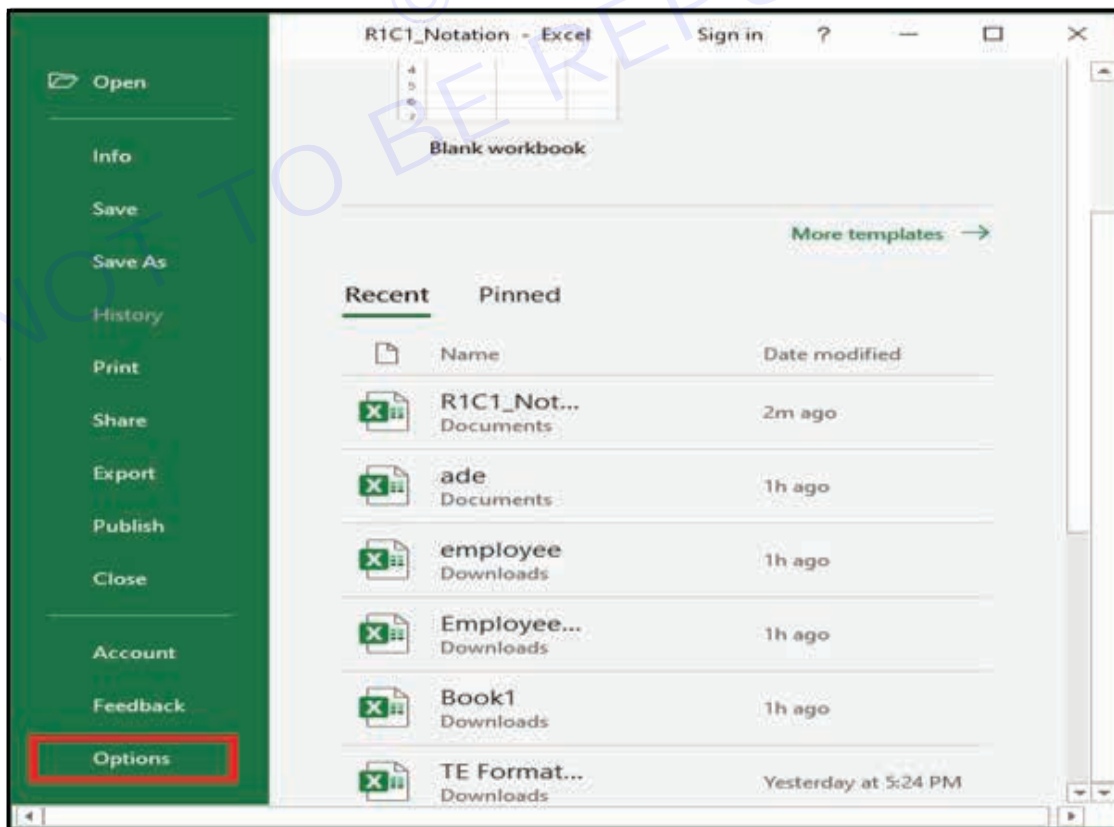
- 1 Open Excel.



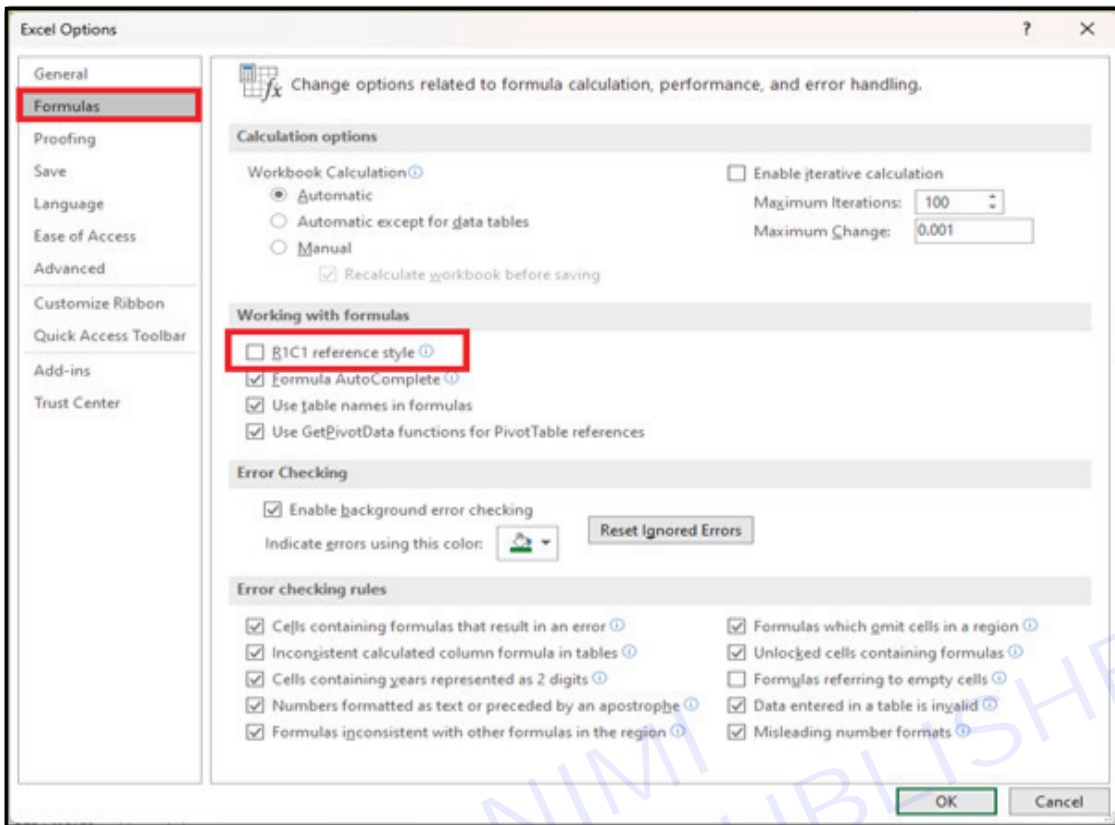
2 Go to the "File" tab. Now you can see the Excel in A1 reference style.



3 Select "Options." from File menu



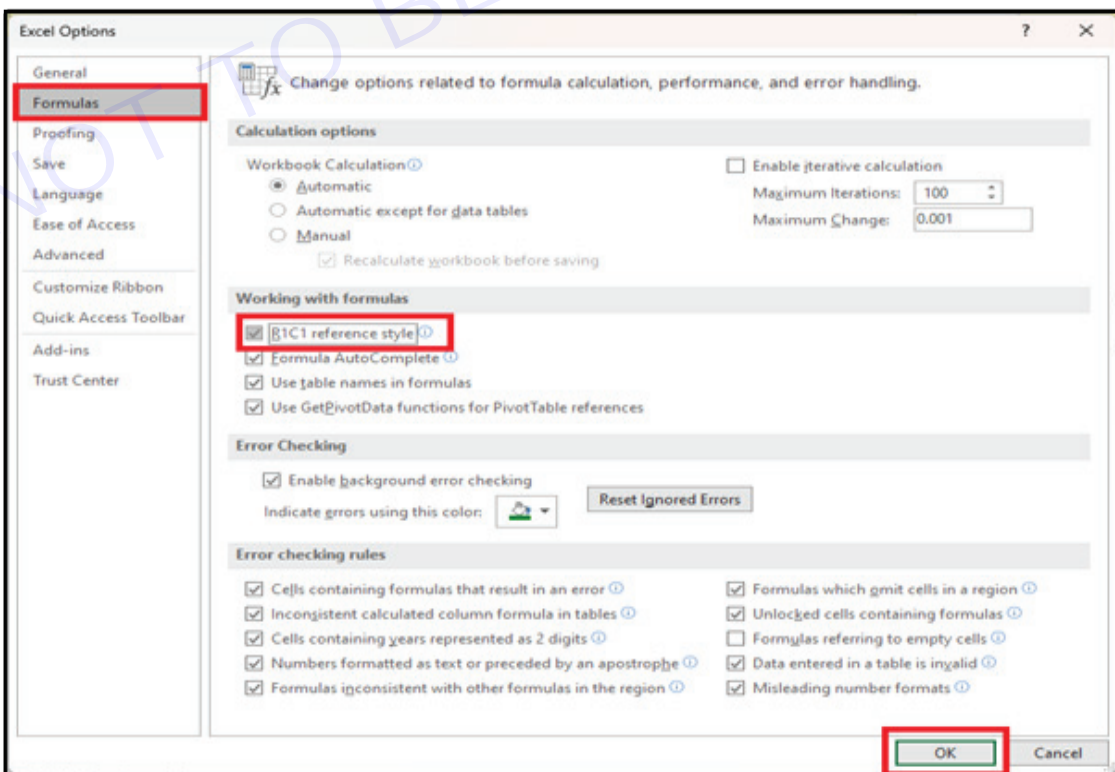
4 In the Excel Options dialog box, go to the “Formulas” category.



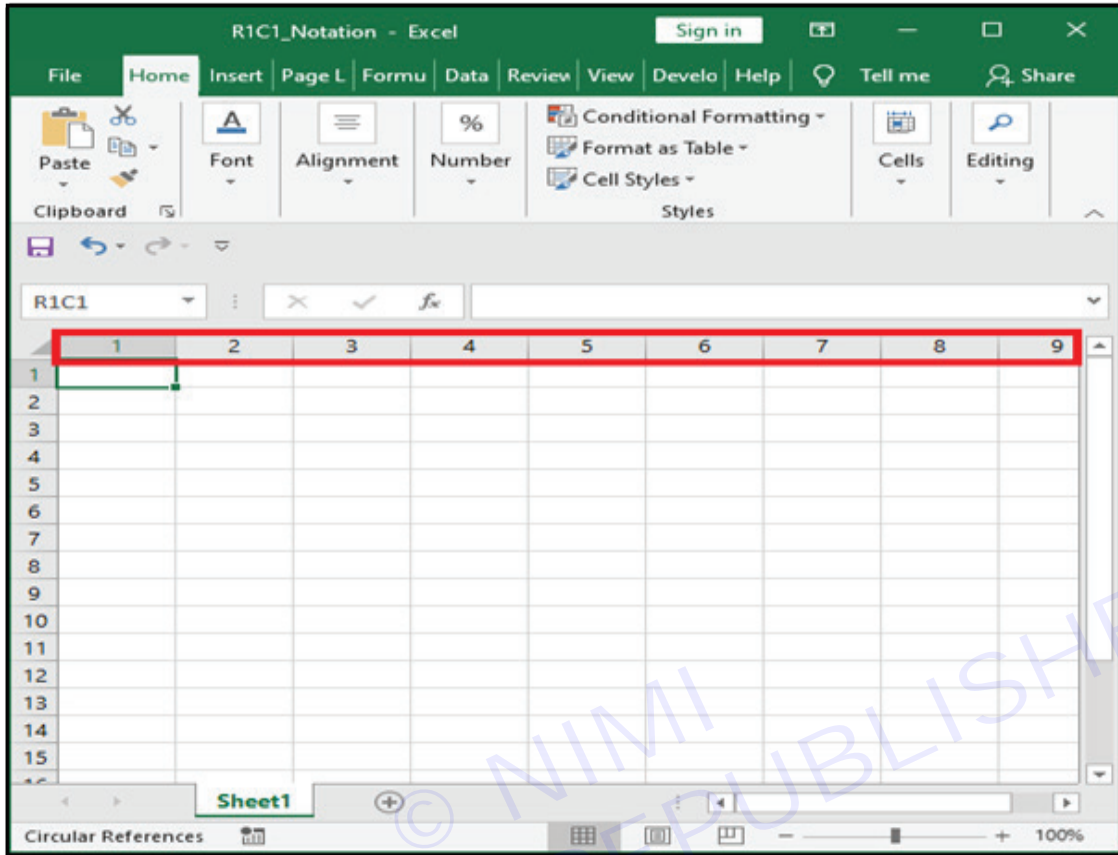
5 Look for the “Working with formulas” section.

6 Check the box for “R1C1 reference style.”

7 Click “OK” to apply the changes.



Once switched, you'll notice that the column headers change from letters (A, B, C, etc.) to numbers (1, 2, 3, etc.), indicating the R1C1 notation mode.



TASK 1: Sum of Two numbers using R1C1

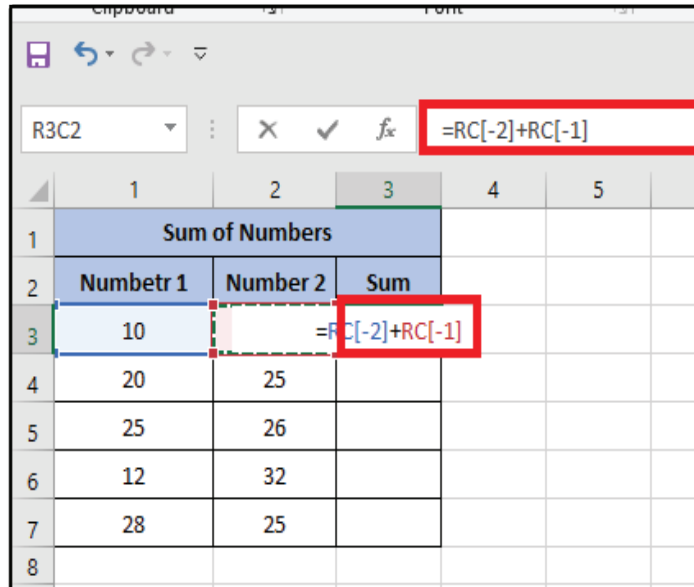
Step 1: Switch to R1C1 Notation

Follow the steps from the previous answer to switch to R1C1 notation.

	1	2	3	4
1	Sum of Numbers			
2	Number 1	Number 2	Sum	
3	10	54		
4	20	25		
5	25	26		
6	12	32		
7	28	25		
8				

Step 2: Write the Formula

In cell R3C3, enter the following formula using R1C1 notation:

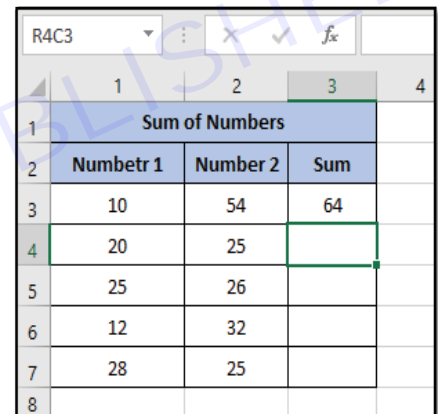


This formula instructs Excel to sum the value in the cell two columns to the left (column A ie, C[-2]) from the current cell(R3C3) and the value in the cell one column to the left (column B ie, C[-1]).

$$R3C3= RC[-2]+RC[-1]$$

Step 3: Press Enter

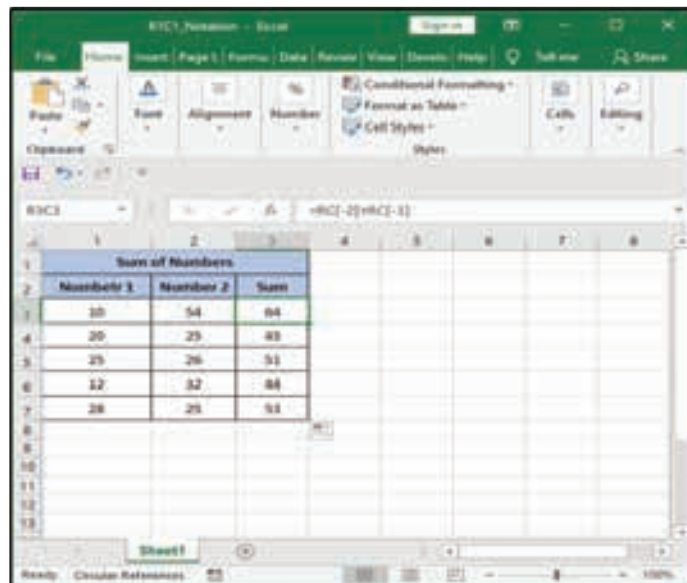
Press Enter to execute the formula. Excel will calculate the sum and display the result in cell R3C3.



Step 4:

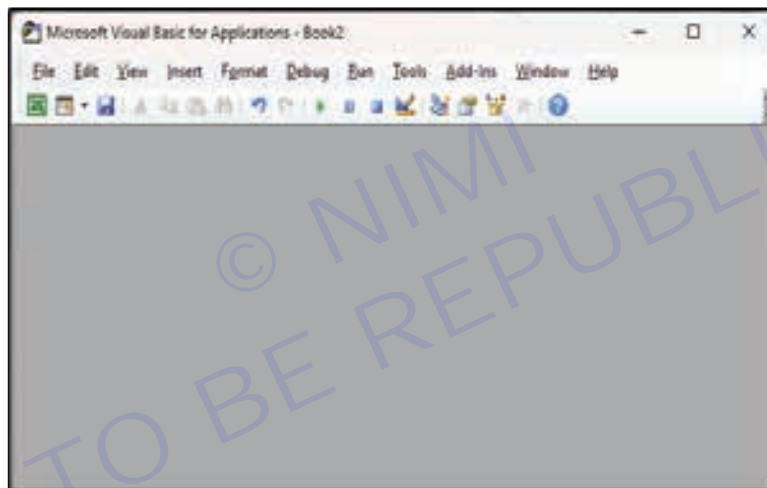
Drag down from R3C3:R7C3 to obtain the result.

Result:

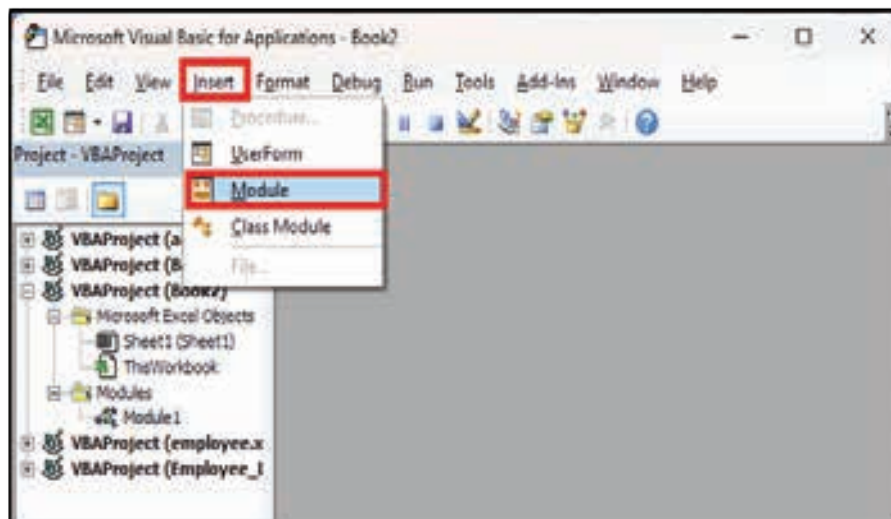


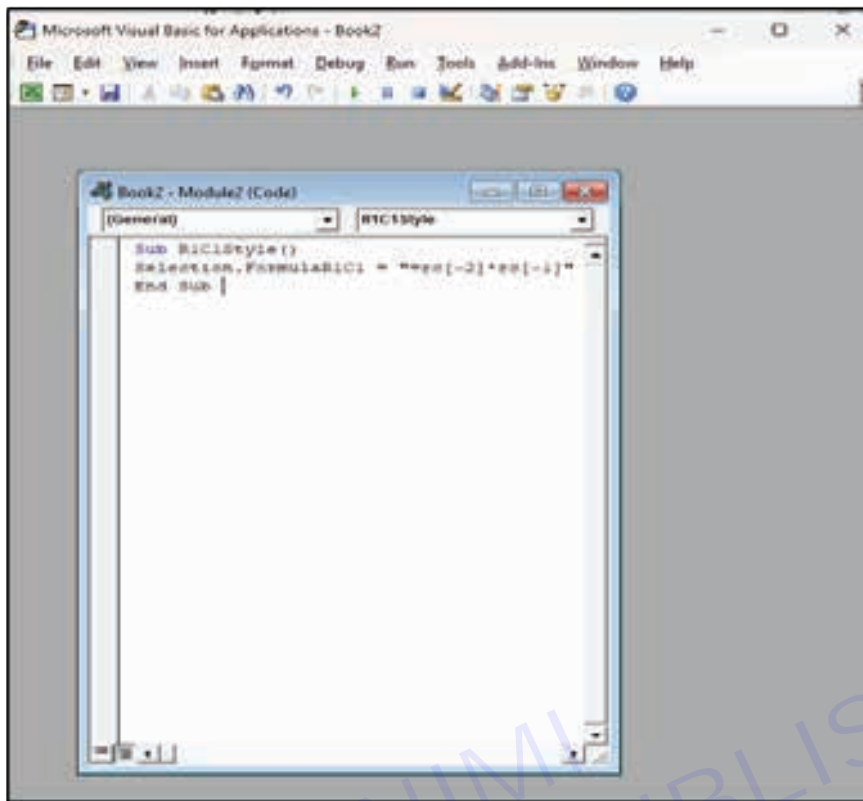
TASK 2 : Using R1C1 Reference in VBA**Step 1:** Open the workbook in Excel.

	1	2	3	4	5
1	Sales Details				
2	Product Name	Quantity	Price	Amount	
3	Key Board	14	700		
4	Mouse	25	500		
5	Pendrive	33	1000		
6	Hard Disk	34	5500		
7					

Step 2: Access VBA Editor: Press Alt + F11 to open the Visual Basic for Applications (VBA) editor.**Step 3:** Insert a Module

- In the VBA editor, ensure that your workbook is selected in the left Project Explorer window.
- If there isn't an existing module, right-click on your workbook name, choose Insert, and then click on Module.



Step 4: Write the Code:**Explanation:**

- 1 Sub R1C1Style(): This line defines the beginning of a subroutine named R1C1Style. Sub indicates the start of a subroutine, followed by the name of the subroutine (R1C1Style), and parentheses to denote that it doesn't take any arguments.
- 2 Selection.FormulaR1C1 = "=rc[-2]*rc[-1]": This line sets the formula of the selected cell or range using R1C1 reference style. Here's the breakdown:
 - Selection: Refers to the currently selected cell or range in Excel.
 - .FormulaR1C1: Indicates that the formula will be assigned using R1C1 reference style.
 - =: Assignment operator used to assign the following formula to the selected cell(s).
 - "=rc[-2]*rc[-1]": The formula being assigned. In R1C1 notation:
 - rc[-2]: Refers to the cell in the same row (relative reference) and two columns to the left.
 - *: Represents the multiplication operator.
 - rc[-1]: Refers to the cell in the same row (relative reference) and one column to the left.
- 3 End Sub: Marks the end of the subroutine R1C1Style. This line tells VBA that the subroutine has finished and returns control to the main program.

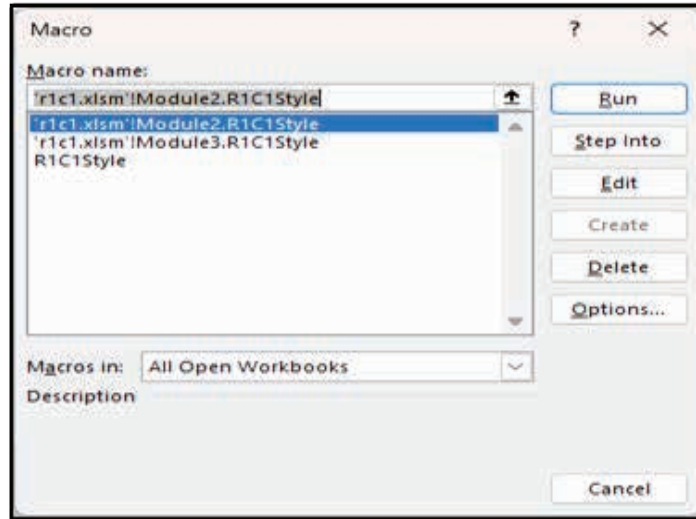
In summary, this subroutine sets the formula of the selected cell(s) to multiply the value in the cell two columns to the left by the value in the cell one column to the left, using R1C1 reference style.

Step 5: Close VBA Editor:

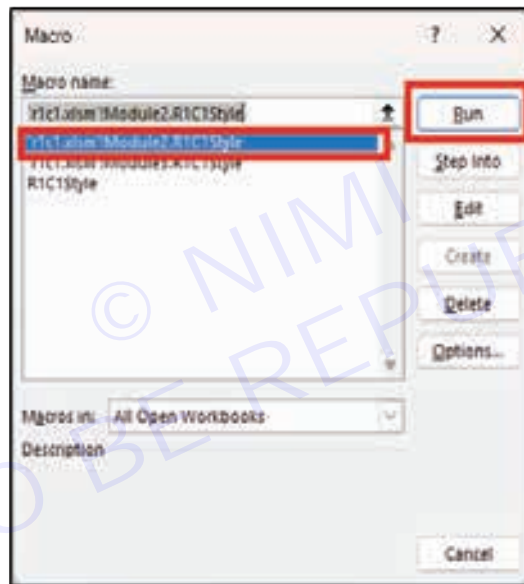
- Close the VBA editor by clicking the close button or pressing Alt + Q.

Step 6: Run the Macro:

- Press Alt + F8 to open the "Macro" dialog.
- Choose the macro named " 'r1c1.xlsm!Module2.R1C1Style' " from the list.



- Click "Run."



Result:

	1	2	3	4	5
1	Sales Details				
2	Product Name	Quantity	Price	Amount	
3	Key Board	14	700	9800	
4	Mouse	25	500	12500	
5	Pendrive	33	1000	33000	
6	Hard Disk	34	5500	187000	
7					
8					

Related Exercises:

- 1 Create Mark Sheet of Students Using R1C1 Reference in VBA and Calculate the following:
 - Calculate Total and Average Marks
 - Calculate Average

Student Name	Python	JAVA	PHP	Total	Average
John	80	75	85		
Mary	85	90	80		
Alice	70	65	75		
Bob	75	80	70		

- 2 Create an EMI Calculator Using R1C1 Reference in VBA

EMI Calculator	
Amount	
Rate of Interest	
Year	
EMI	

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 68 : Use array formula

Objectives

At the end of this exercise you shall be able to

- use various array formula in your tasks.

Requirements

Tools/Materials

- PC/Laptop with Windows OS
- MS Excel 2013 or Higher

Procedure

TASK 1: Calculating Total Marks of Students

In our first task we'll calculate the total marks for each student using an array formula in Excel. It's simple and easy. Just follow along.

Name	Maths	Science	Total Marks
Arun	95	76	
Akhila	54	46	
Sona Sajeew	69	84	
Aliya	54	73	
Neelima	89	51	
Anna	52	50	
Rahul	74	72	
Varun Prakash	62	49	
Elizabeth	59	72	
Praveen	85	68	

Step 1: In Office 365, Select cell E5 and enter the following formula.

=C5:C14+D5:D14

Name	Maths	Science	Total Marks
Arun	95	76	=C5:C14+D5:D14
Akhila	54	46	
Sona Sajeew	69	84	
Aliya	54	73	
Neelima	89	51	
Anna	52	50	
Rahul	74	72	
Varun Prakash	62	49	
Elizabeth	59	72	
Praveen	85	68	

- After that, press ENTER.

Name	Maths	Science	Total Marks
Arun	95	76	171
Akhila	54	46	100
Sona Sajeer	69	84	153
Aliya	54	73	127
Neelima	89	51	140
Anna	52	50	102
Rahul	74	72	146
Varun Prakash	62	49	111
Elizabeth	59	72	131
Praveen	85	68	153

Note: Check the image carefully until you see a light blue line around the cells in Column E. This is because they are in an array. We didn't use the Fill Handle or anything like that to fill these cells.

Here, we're using Microsoft Excel 365 version. So, we can run the array formula by pressing ENTER. But, in the older version do the following steps.

- So, first select E5:E14 and type the above formula in cell E5.
- After that, press CTRL + SHIFT + ENTER at a time.

Name	Maths	Science	Total Marks
Arun	95	76	171
Akhila	54	46	100
Sona Sajeer	69	84	153
Aliya	54	73	127
Neelima	89	51	140
Anna	52	50	102
Rahul	74	72	146
Varun Prakash	62	49	111
Elizabeth	59	72	131
Praveen	85	68	153

Note: After pressing these buttons, a pair of curly brackets will automatically be applied within two sides of the formula. You don't have to write them manually.

Also, you can't make any changes inside the cell of an array. Excel will not allow you to do this. If you ever try to do this kind of thing, the following phenomena will happen.

- Here, go to cell E8 and change the cell element to 5.
- After that, press ENTER.

Immediately, Excel will show a warning box on the display like the image below.

Name	Maths	Science	Total Marks
Arun	95	76	171
Akhila	54	46	100
Sona Sajeer	69	84	153
Aliya	54	73	5
Neelima	89	51	140
Anna	52	50	102
Rahul	74	72	146
Varun Prakash	62	49	111
Elizabeth	59	72	131
Praveen	85	68	153

TASK 2 : Determining Highest Marks Obtained

Now, we want to discover the highest marks obtained by the students on any of the two subjects. So, let's begin.

- Firstly, we've made an output range in **B16:C16**.
- Secondly, select cell C16 and write down the formula below.

=MAX(C5:D14)

Here, the MAX function returns the maximum value among the numbers in this range.

- Lastly, press ENTER on the keyboard.

Name	Maths	Science
Arun	95	76
Akhila	54	46
Sona Sajeer	69	84
Aliya	54	73
Neelima	89	51
Anna	52	50
Rahul	74	72
Varun Prakash	62	49
Elizabeth	59	72
Praveen	85	68

Highest Marks	95
---------------	----

TASK 3 : Excel Array Formula with Multiple Criteria

In this task, will show an array formula that will return a two-dimensional array with multiple criteria. Here, will be got the Name, Section, Subject, and Mark in columns B, C, D, and E consecutively. Also, will got Section A and Maths as the Subject in cells in the B16:E17 range

Now, we'll filter out the above array in the B5:E14 range with the criteria in the B16:E17 range. We can clearly see there are two criteria. One is Section, another is Subject. So, obviously, there are multiple criteria for filtering. So, without further delay, let's dive in!

- At first, copy the headings in the B4:E4 range and paste them into the B19:E19 range.

Name	Section	Subject	Mark
Arun	A	Maths	76
Akhila	C	English	46
Sona Sajeer	A	Science	84
Aliya	B	History	73
Neelima	C	Maths	51
Ansa	A	English	50
Rahul	B	English	72
Varun Prakash	A	Maths	49
Elizabeth	C	Science	72
Praveen	A	Maths	68

Section	Subject
A	Maths

- Then, go to cell B20 and paste the following formula into that cell.

`=FILTER(B5:E14,(C5:C14=B17)*(D5:D14=D17),"`

Here, **B5:E14** represents the range of the array. **C5:C14** serves as the Section column. B17 performs as the section to be filtered. **D5:D14** means the Subject column. Lastly, D17 acts as the subject to be filtered.

The FILTER function has three arguments. Here, B5:E14 is the array argument. And **(C5:C14=B17)*(D5:D14=D17)** is the include argument. Basically, this works like a Boolean array; it carries the condition or criteria for filtering. In this case, we've two criteria combined by a (*) sign.

- After that, tap **ENTER**.

Name	Section	Subject	Mark
Arun	A	Maths	76
Varun Prakash	A	Maths	49

Note: Filter function available in Excel 2021 and Above Versions.

TASK 4 : Computing Total Cost based on Quantity

This example is actually fun to learn. You can apply this theory to other problems as well. Here, in Column B, we have the Quantity of Unit. And, in Column C, there is their corresponding Unit Price. In essence, we want to know how much this will cost if we order a certain quantity of units.

	A	B	C	D
1				
2		Computing Total Cost based on Quantity		
3				
4		Quantity of Unit	Unit Price	
5		1 to 10	₹ 20	
6		11 to 20	₹ 19	
7		21 to 50	₹ 16	
8		51 to 100	₹ 15	
9		101 to 200	₹ 13	
10				
11		Quantity	80	
12		Total Cost		
13				

From the picture above, we can see that the unit price is decreasing gradually with the increase in order amount. So, it will cost us less if we order in bulk. Let's see it in action.

- Initially, go to cell C12 and put the following formula into the cell.

$$=C11*IF(C11>=101,C9, IF(C11>=51, C8, IF(C11>=21, C7, IF(C11>=11, C6, IF(C11>=1, C5, ""))))))$$

Formula Breakdown

- IF(C11>=1, C5, " ")** → The IF function checks whether a condition is met, and returns one value if TRUE, and another one if FALSE. Here, **C11>=1** is the logical_test argument which compares the value of the C11 cell with 1. If this value is greater than or equal to 1 then the function returns the value of cell C5 (**value_if_true** argument) otherwise it returns blank(**value_if_false** argument).

Output → 20

- IF(C11>=11, C6, IF(C11>=1, C5, ""))** → this becomes **IF(C11>=11, C6, 20)**.

Output → 19

- IF(C11>=21, C7, IF(C11>=11, C6, IF(C11>=1, C5, "")))** → this becomes **IF(C11>=21, C7, 19)**.

Output → 16

- IF(C11>=51, C8, IF(C11>=21, C7, IF(C11>=11, C6, IF(C11>=1, C5, ""))))** → this becomes **IF(C11>=51, C8, 16)**.

Output → 15

- IF(C11>=101,C9, IF(C11>=51, C8, IF(C11>=21, C7, IF(C11>=11, C6, IF(C11>=1, C5, ""))))))** → this becomes **IF(C11>=101,C9, 15)**.

Output → 13

- C11*IF(C11>=101,C9, IF(C11>=51, C8, IF(C11>=21, C7, IF(C11>=11, C6, IF(C11>=1, C5, ""))))))** → this becomes **C11*15**.

Output → 80*15 → 1200

- Following this, press the ENTER key.

Quantity of Unit	Unit Price
1 to 10	₹ 20
11 to 20	₹ 19
21 to 50	₹ 16
51 to 100	₹ 15
101 to 200	₹ 13

Quantity	80
Total Cost	1200

TASK 5 : Finding out Average of Positive Numbers

In this worksheet, you see there are some numbers in Column B of the worksheet. Some numbers are positive, and some numbers are negative. We are going to find out the average of the positive numbers in this range.

Numbers	Average
15	
-5	
14	
-8	
-98	
45	
78	
65	
54	
-25	

You can do it in two ways. The first way is: separate the positive numbers manually, create a new range, and at the end use the AVERAGE function to find out the average of the positive numbers. Another way is: we can use an array formula. It will save us time and effort. So, without further delay, let's see how we can do it.

Steps:

- Primarily, go to cell C5 and enter the following formula.

=AVERAGE(IF(B5:B14>0,B5:B14,FALSE))

Formula Breakdown

Observe closely the arguments of the IF function. You know how the IF function works. If the first argument of the function is TRUE, then the second argument is returned by the IF function. If the first argument is FALSE, then the third argument is returned by the IF function. Arguments are separated by commas.

In our case, the third argument is FALSE, so if the first argument is FALSE, then the IF function will return the FALSE statement.

OKAY! The main focus is now on the first argument and the second argument. You see the first argument is a range, and the second argument is also a range. And the whole formula is an array formula.

- **IF(B5:B14>0,B5:B14,FALSE)** → here Excel will create an array internally with the positive numbers and False statements.
- **Output** → {15, FALSE, 14, FALSE, FALSE, 45, 78, 65, 54, FALSE}

AVERAGE(IF(B5:B14>0,B5:B14,FALSE)) → this becomes **AVERAGE({15, FALSE, 14, FALSE, FALSE, 45, 78, 65, 54, FALSE})**.

- **Output** → 45.167

The AVERAGE function finds out the average of the values in the array, except the FALSE values, the False values are not numbers, so the AVERAGE function neglects the FALSE values.

- As usual, press ENTER.

Finding out Average of Positive Numbers	
Numbers	Average
15	45.167
-5	
14	
-8	
-98	
45	
78	
65	
54	
-25	

Related Exercises:

- You are managing a bookstore and want to calculate the total revenue generated by each genre of books over a specific period. Create an Excel spreadsheet with the following details:
 - Column A: Book titles.
 - Column B: Genre of each book.
 - Column C: Unit price for each book.
 - Column D: Number of units sold for each book.
 - Row 1: Use an array formula to calculate the total revenue generated by each genre by summing the revenue from each book sold in that genre. Ensure that the array formula adjusts correctly for each genre as you copy it across.
- You are organizing a charity event and want to track the donations received from different sources. Create an Excel spreadsheet with the following details:
 - Column A: Donation amounts.
 - Column B: Donation sources (e.g., Individuals, Corporations, Organizations, etc.).
 - Row 1: Use an array formula to calculate the total donations received from each source by summing the donations from each source. Ensure that the array formula adjusts correctly for each donation source as you copy it across.

- You are analyzing the performance of different marketing campaigns and want to calculate the conversion rate for each campaign based on the number of leads generated and the number of conversions. Create an Excel spreadsheet with the following details:
 - Column A: Campaign names.
 - Column B: Number of leads generated by each campaign.
 - Column C: Number of conversions from leads for each campaign.
 - Row 1: Use an array formula to calculate the conversion rate for each campaign by dividing the number of conversions by the number of leads generated. Ensure that the array formula adjusts correctly for each campaign as you copy it across.
- You are managing a restaurant and want to analyze the popularity of different menu items. Create an Excel spreadsheet with the following details:
 - Column A: Menu item names.
 - Columns B through G: Weekly sales figures for each menu item for the past six weeks (varying values).
 - Row 1: Use an array formula to calculate the total sales for each menu item by summing the sales figures across all weeks. Ensure that the array formula adjusts correctly for each menu item as you copy it across.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 69 : Audit excel formula

Objectives

At the end of this exercise you shall be able to

- use various Audit Excel formulas in your task for trouble shooting errors in large data sheets
- check the accuracy of financial reports quickly.

Requirements

Tools/Materials

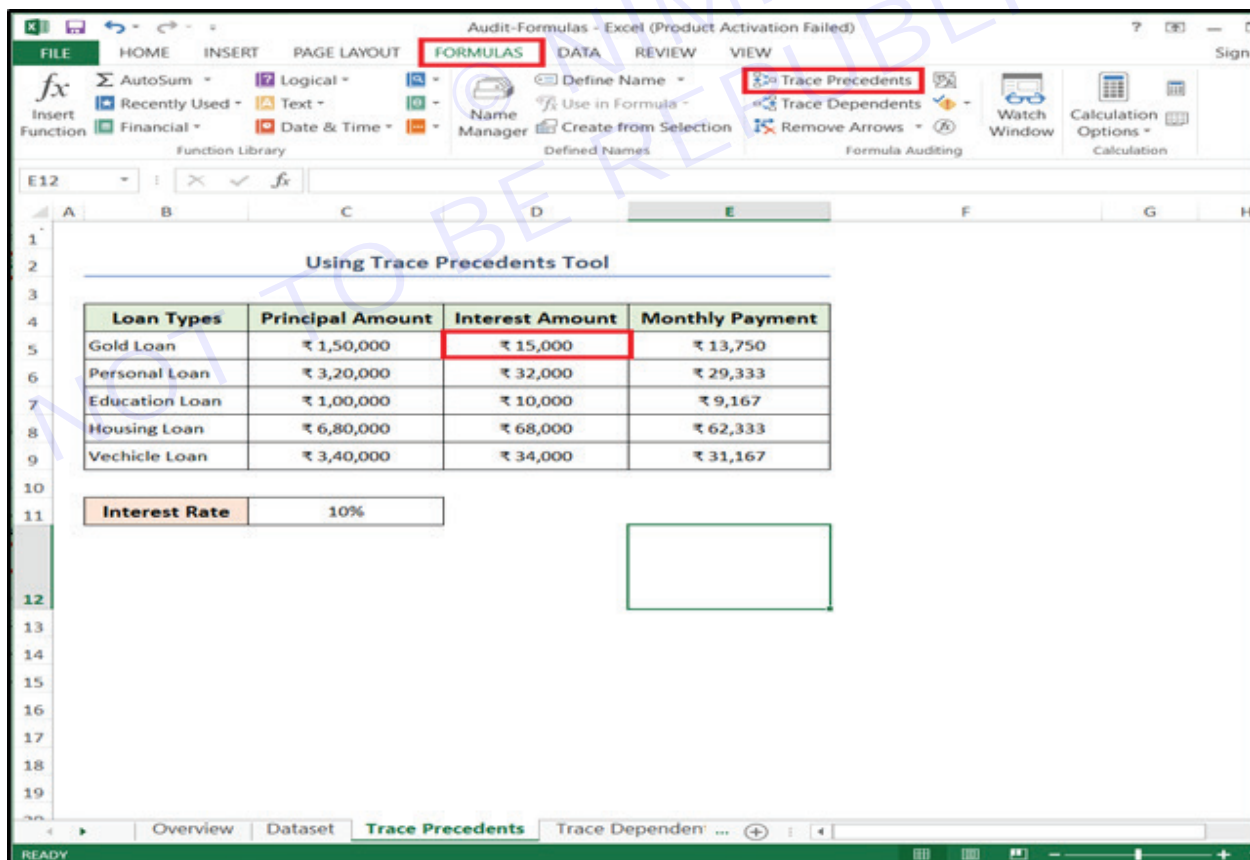
- PC/Laptop with Windows OS
- MS Excel 2013 or Higher

Procedure

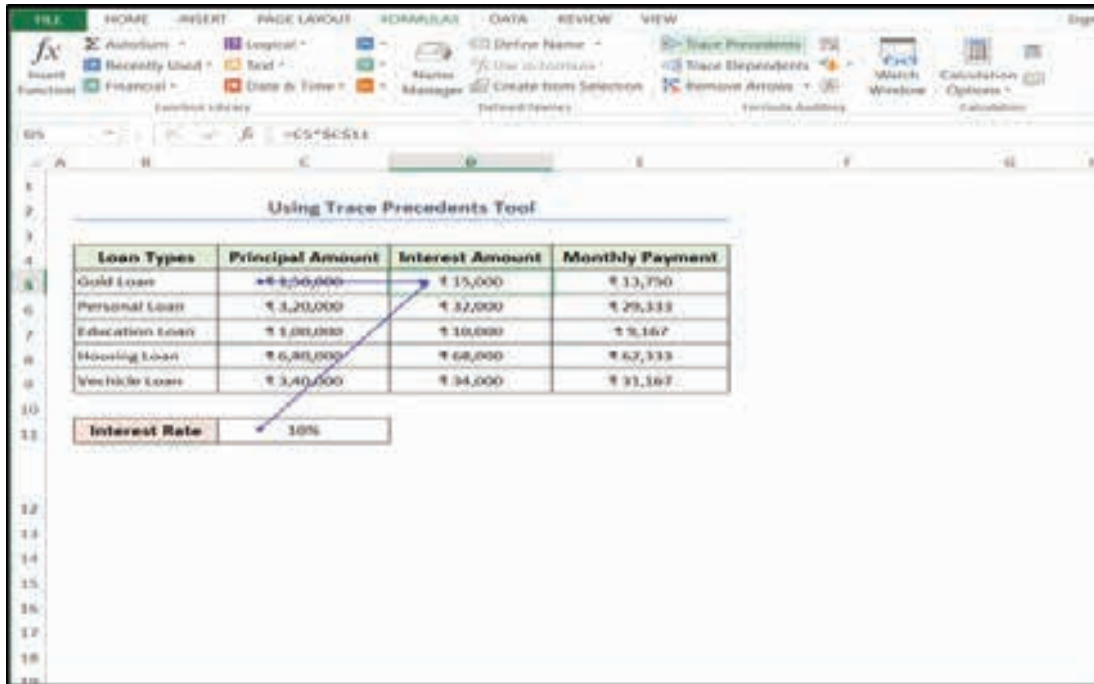
TASK 1 : Trace Precedents

Trace Precedents displays tracer arrows from the cells showing the direction of information flow. You see a blue box around the cells when this method is active. However, one can press this button multiple times to catch additional levels.

- Choose a cell (D5), and visit the Trace Precedents option from the Formulas tab.



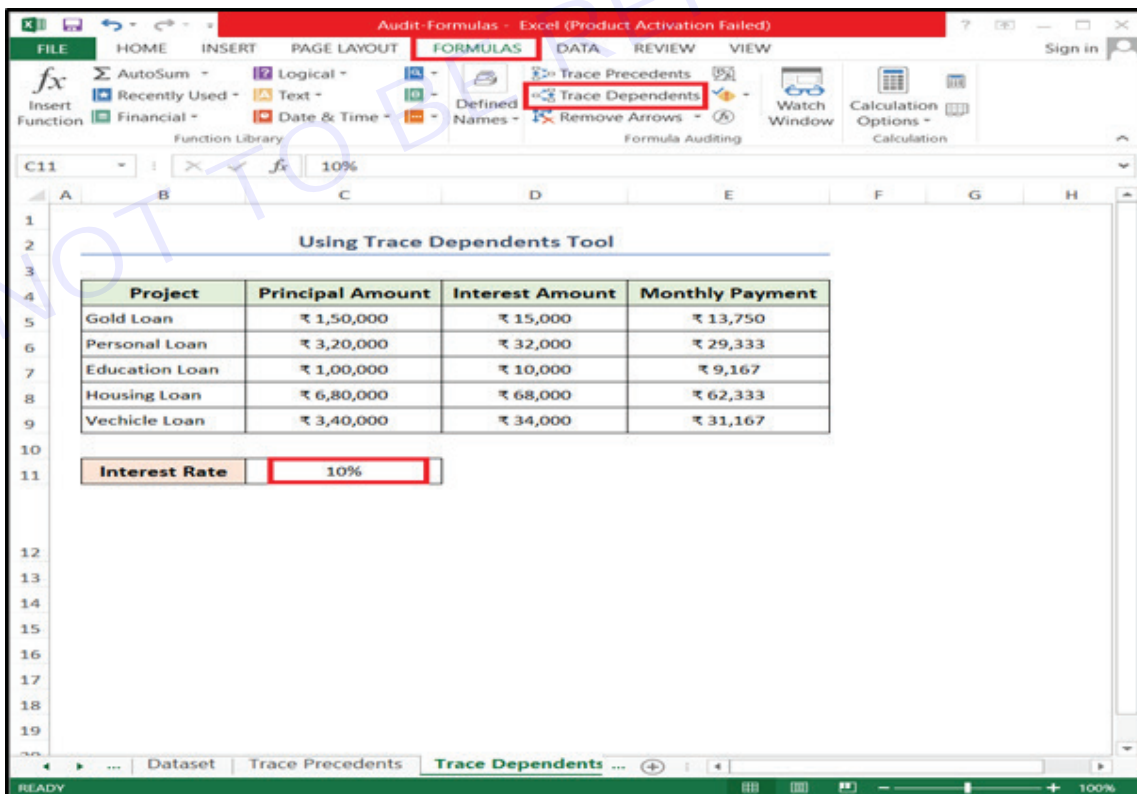
- As a result, we will see two arrows from cell (C5) and cell (C11) indicating towards cell (D5) as the interest amount is calculated using the principal amount and interest rate.



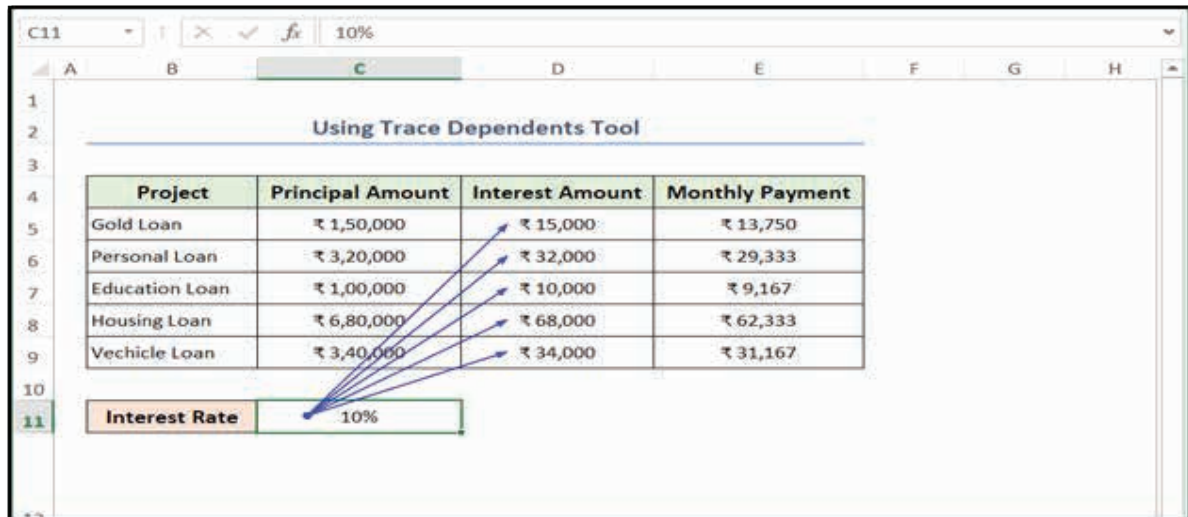
TASK 2: Trace Dependents

In order to visually highlight the cells that depend on the value of a selected cell, you can try the Trace Dependents feature in Excel. This is a powerful tool for understanding the relationship between cells. Here, let's see how the interest rate is dependent on the cells.

- Simply, choose a cell (C11), and then visit the Trace Dependents feature from the Formulas tab.



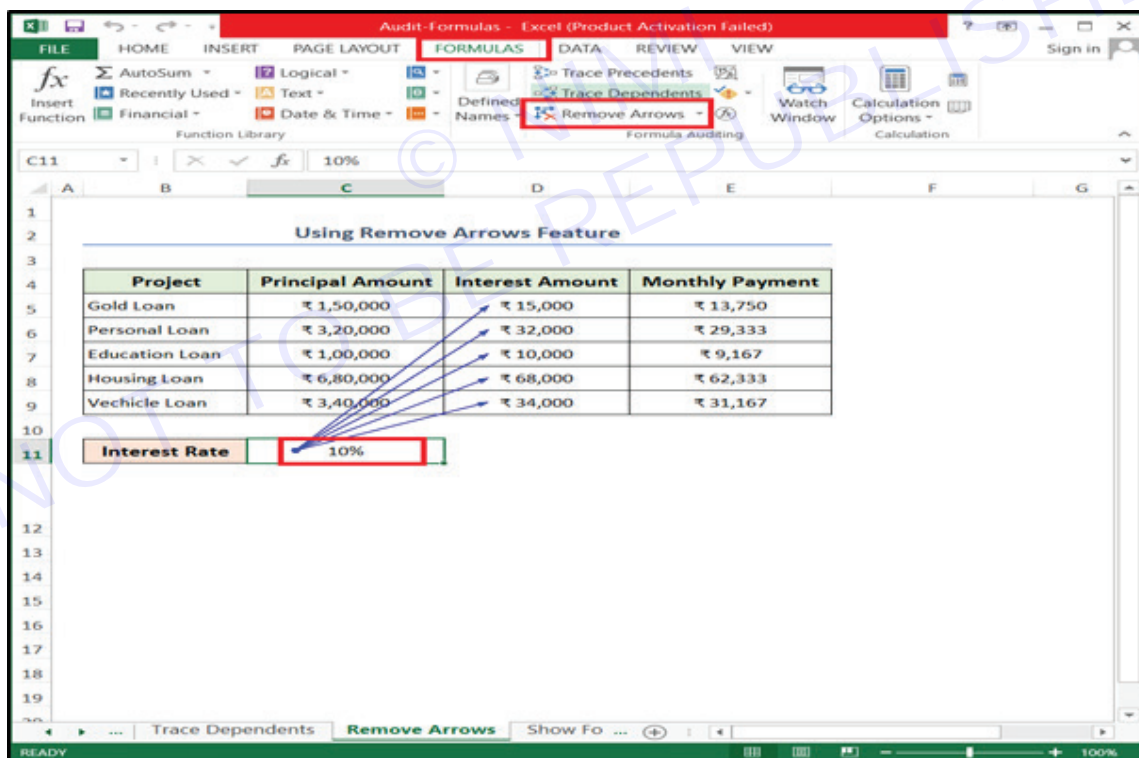
- Finally, you will see the arrows from the cell (C11) to other cells indicating the cells that are dependent on the value of the selected cell.



TASK 3: Remove Arrows

After inserting arrows using the above features, you can also delete them by utilizing the Remove Arrows option.

- Start with, selecting the cell (C11) and clicking the Remove Arrows option from the Formulas tab

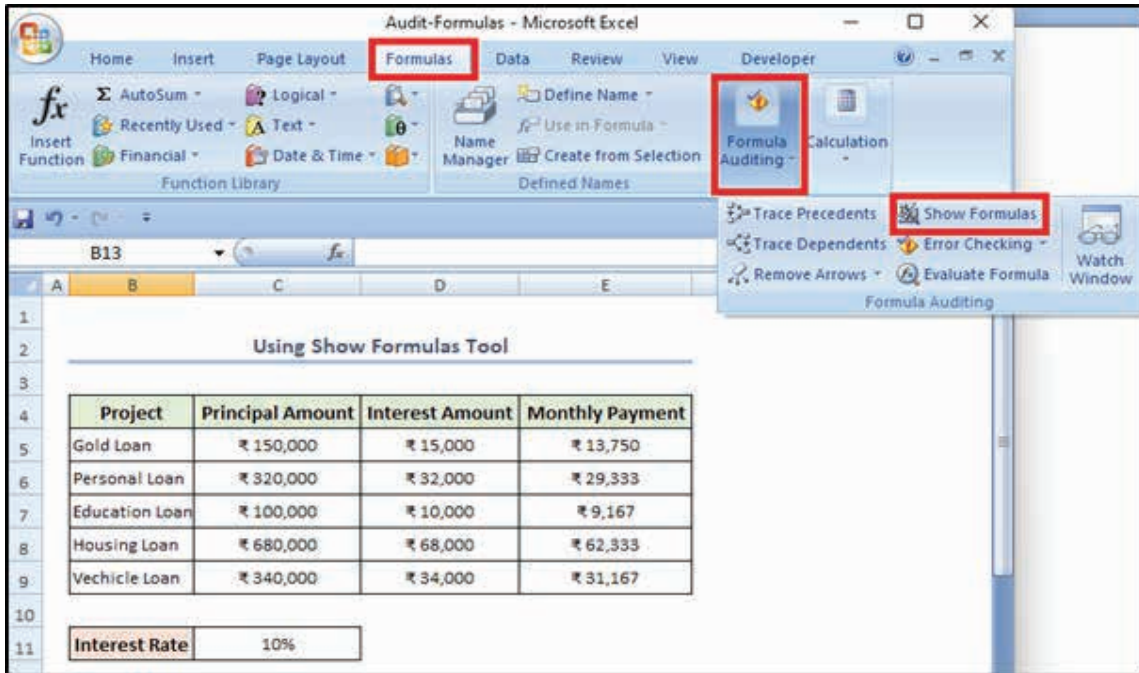


- Within a blink of an eye, the arrows will be removed.

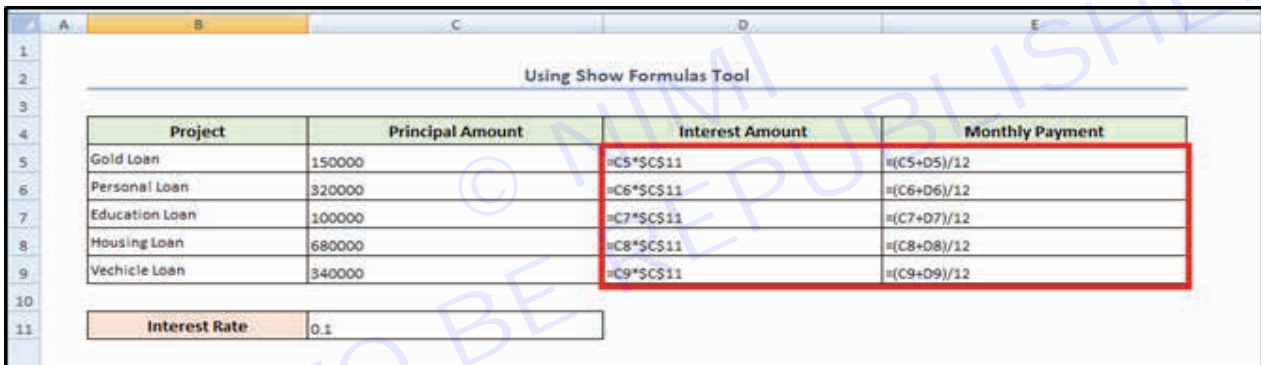
TASK 4: Show Formulas

Show Formulas in Excel is a helpful tool that allows you to view the actual formulas within cells instead of their calculated results. This feature offers transparency into complex calculations, aiding in formula debugging and verification.

- While the worksheet is open, visit the Formulas tab and press the Show Formulas option.



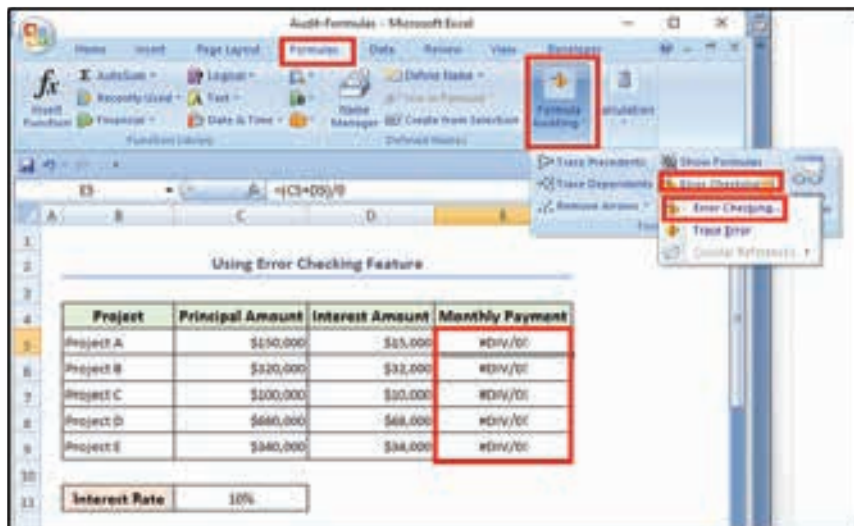
- Immediately, all the cells with formulas will represent the formulas inside the cells.



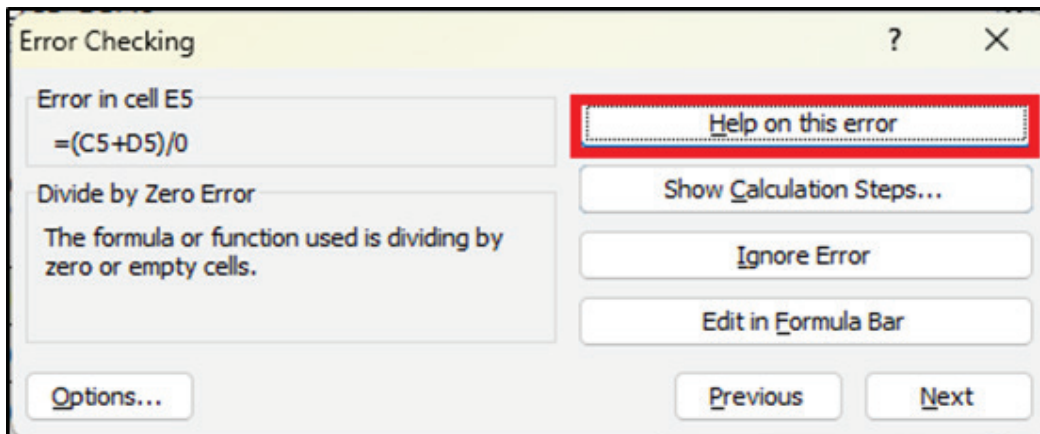
TASK 5: Error Checking (Includes Error Checking, Trace Error, Circular References)

Sometimes while applying formulas, you will get errors like #DIV/0!, #VALUE!, #NAME? Errors. To check why it's happening you can visit Error Checking option from the Formula tab.

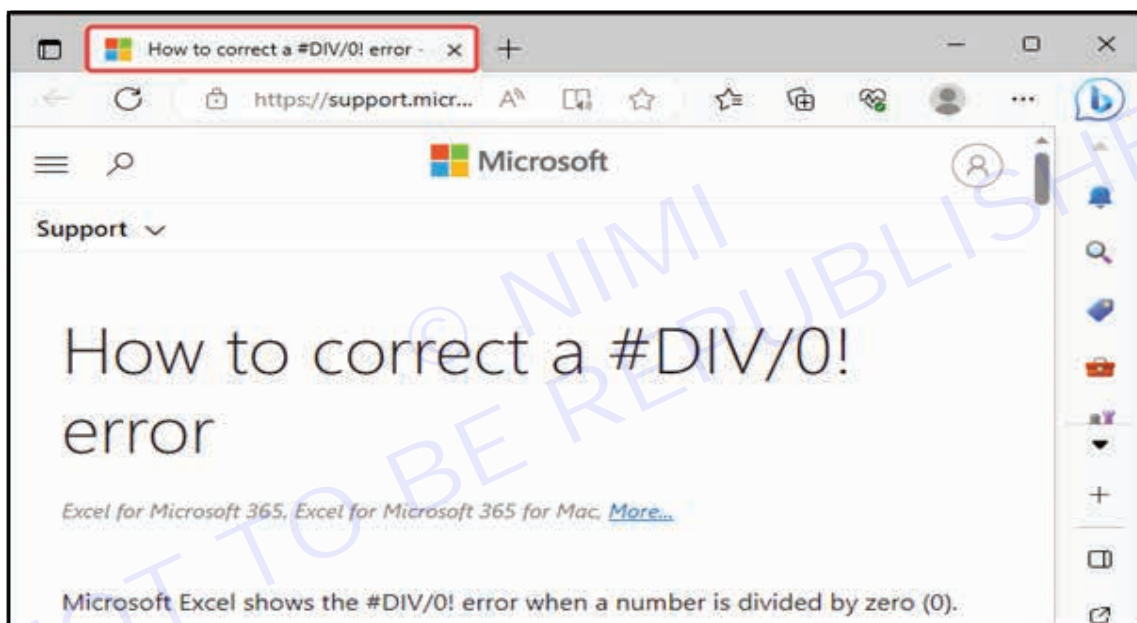
- Select a cell (E5), and hit the Error Checking option from the Formulas tab.



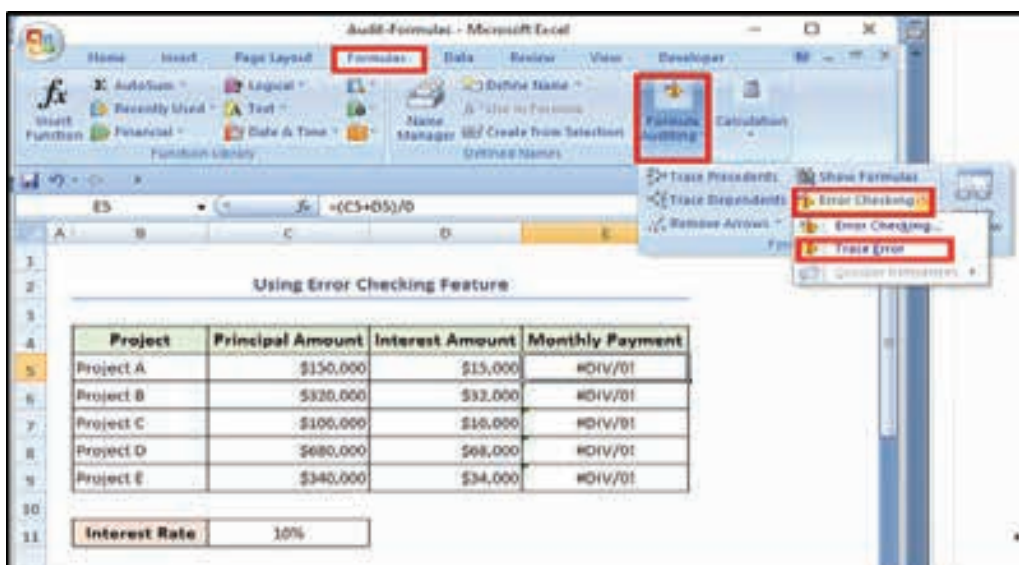
Now, you can click the Help on this Error option to check in detail about the error



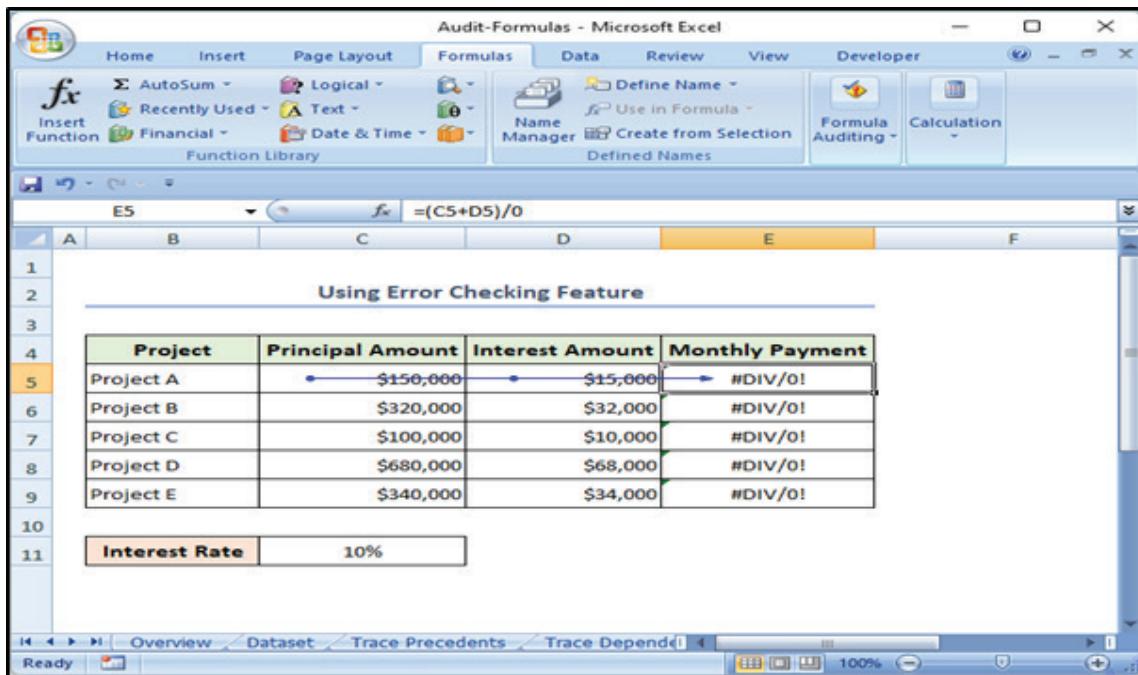
- As a result, a new window will open in your browser providing you details about the error and its solution.



- In order to trace from which cells these are happening, click the Trace Error option from the Formulas tab.



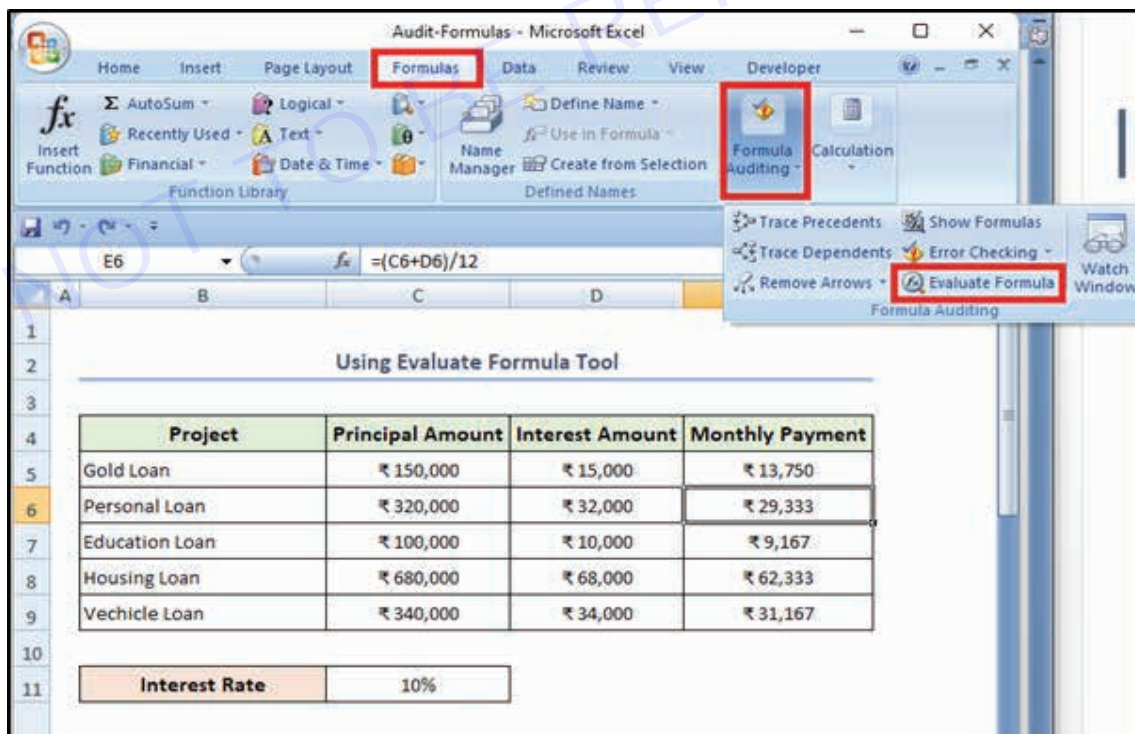
- Immediately, an arrow will appear to indicate the error and its corresponding cells.



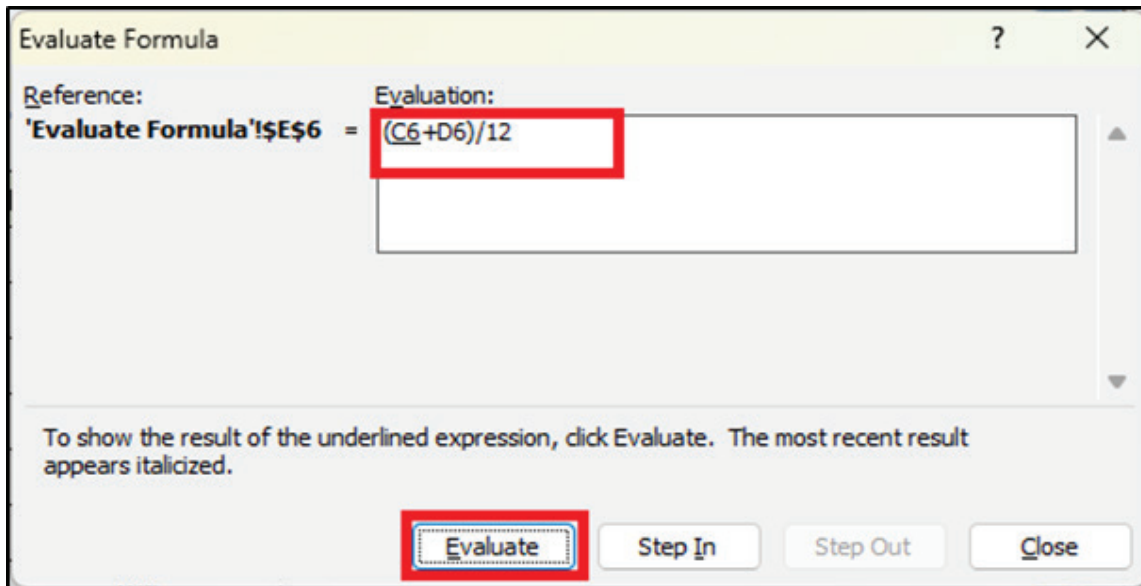
TASK 6: Evaluate Formula

When you are dealing with complex formulas and you are having trouble understanding the formulas, at that time you can visit the Evaluate Formula option to have a better understanding.

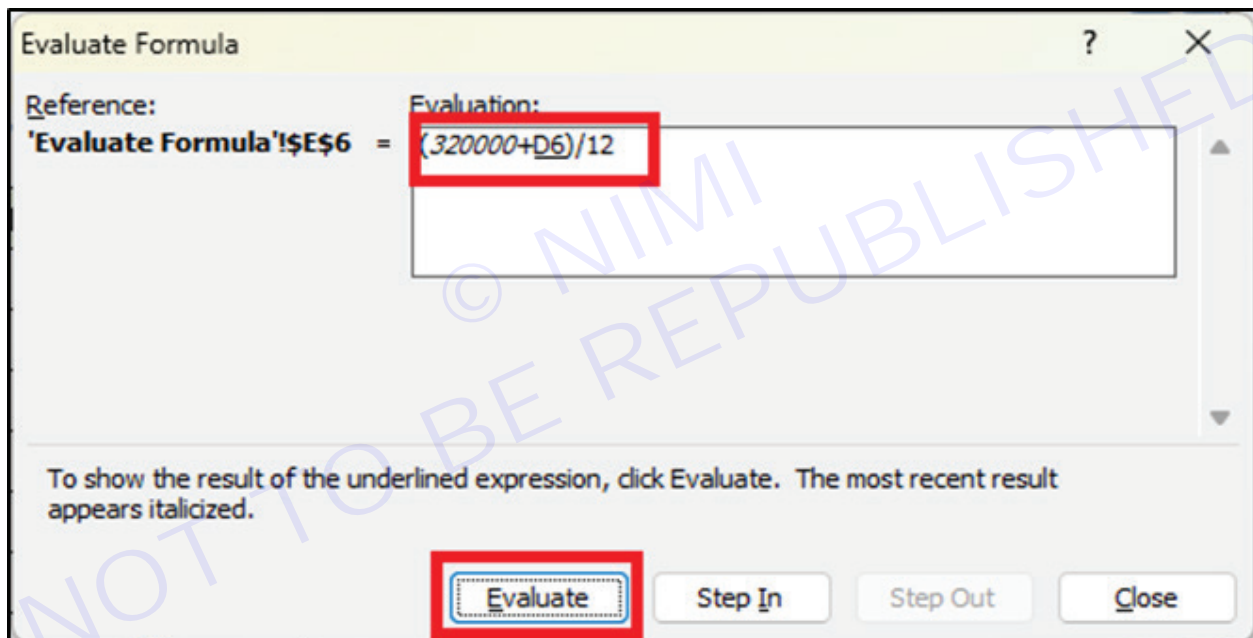
- Simply, choose a cell (E5) consisting of the formula and hit the Evaluate Formula option.



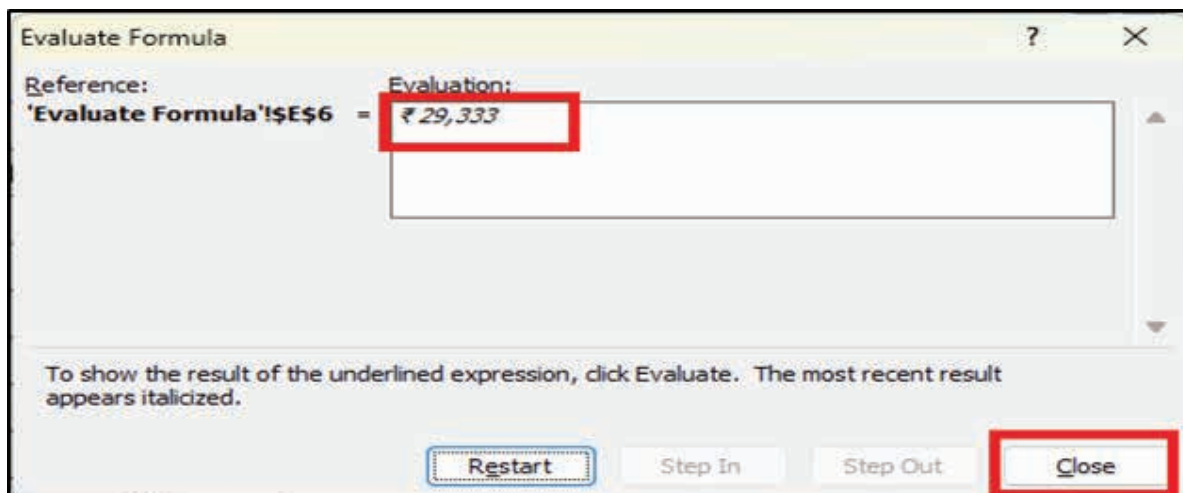
- Immediately, a window will open evaluating the formula.
- From there to evaluate more deeply click the Evaluate option.



- Another update will come describing the formula. Click Evaluate again to get the result.



- After completing the evaluation, you will get the cell value in the window.

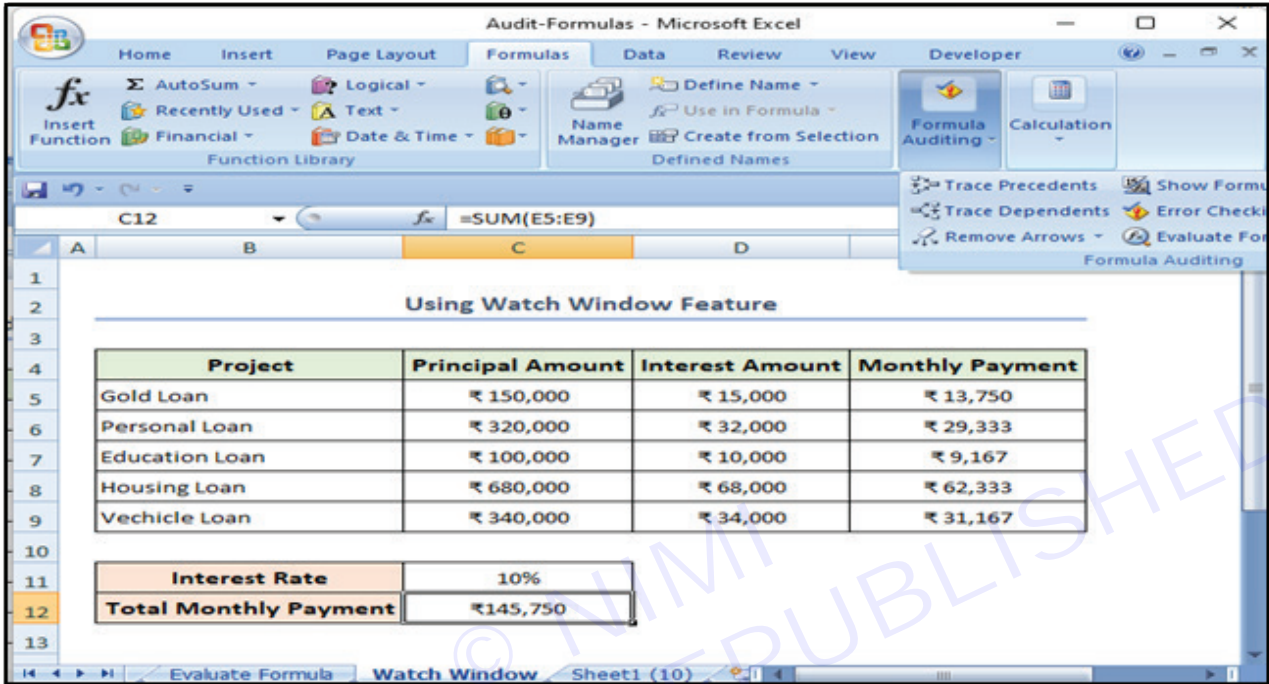


TASK 7: Watch Window

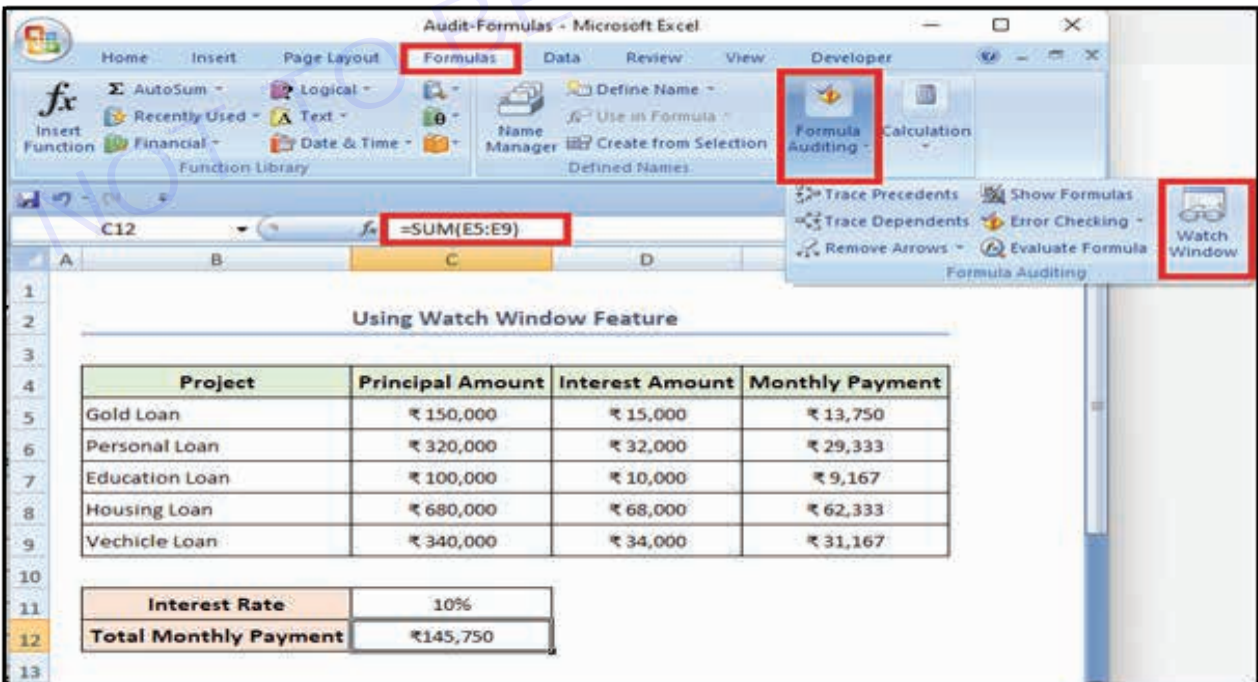
While working with a large dataset sometimes you might need to look over some cell values immediately and all time in a specific space. For that, you can add a watch window at the top of the spreadsheet. Here, we have calculated the total monthly payment amount using the SUM function in Excel. Now we will add a watch window for this specific cell.

- Choose a cell (C12), apply the below formula, and hit ENTER.

=SUM(E5:E9)



- Now choosing the cell (C12), hit the Watch Window feature from the Formulas tab.



- Within a glimpse, a window will open at the top of the spreadsheet showing cell value and formula. This watch window is really a helpful tool for making a summary of your dataset. And if you scroll or jump to another sheet the watch window will always be visible on that place.

The screenshot displays the 'Watch Window' in Microsoft Excel. The window lists a watch for cell C12, which contains the formula `=SUM(E5:E9)` and a value of ₹145,750. Below the window, the worksheet is visible, showing a table titled 'Using Watch Window Feature'. The table lists various loan projects with their principal amounts, interest amounts, and monthly payments. A summary row at the bottom shows the 'Total Monthly Payment' as ₹145,750, which corresponds to the value in the Watch Window.

Project	Principal Amount	Interest Amount	Monthly Payment
Gold Loan	₹ 150,000	₹ 15,000	₹ 13,750
Personal Loan	₹ 320,000	₹ 32,000	₹ 29,333
Education Loan	₹ 100,000	₹ 10,000	₹ 9,167
Housing Loan	₹ 680,000	₹ 68,000	₹ 62,333
Vehicle Loan	₹ 340,000	₹ 34,000	₹ 31,167
Interest Rate		10%	
Total Monthly Payment		₹145,750	

Related Exercises:

- You are reviewing a complex financial model and want to trace precedents to understand how a particular calculation is derived. Perform the following tasks:
 - Identify a cell containing a formula that you want to audit.
 - Use the “Trace Precedents” feature in Excel to visually trace the cells that contribute to the formula’s calculation.
 - Document the path of precedents to understand the formula’s logic.
- You are troubleshooting errors in a large dataset and suspect that circular references may be causing incorrect calculations. Perform the following tasks:
 - Use the “Error Checking” feature in Excel to check for circular references within the worksheet.
 - If circular references are found, analyze the affected cells to identify and resolve the circular dependency.
- You are verifying the accuracy of a financial report and want to identify any external references to cells in other workbooks. Perform the following tasks:
 - Use the “Workbook Audit” feature in Excel to audit external references to cells in other workbooks.
 - Review the list of external references and ensure that they are accurately referencing the intended data sources.
- You are collaborating on a spreadsheet with multiple colleagues and want to track changes made by each user. Perform the following tasks:
 - Enable the “Track Changes” feature in Excel to track changes made to the workbook by different users.
 - Review the change history to identify who made specific changes and when they were made.

EXERCISE 70 : Create and modify simple macros

Objectives

At the end of this exercise you shall be able to

- create simple macros
- modify macros.

Requirements

Tools/Materials

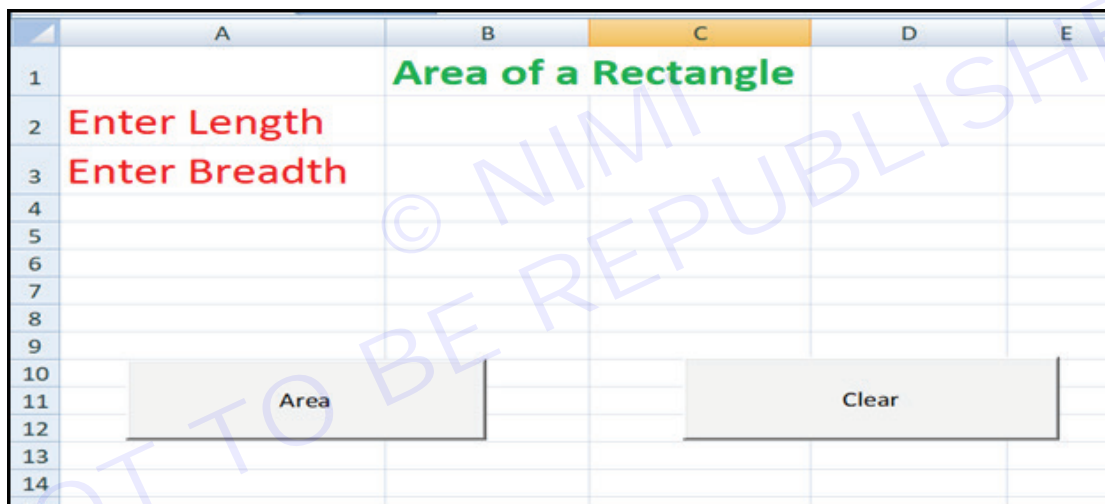
- PC/Laptop with Windows OS
- MS Excel 2013 or Higher

Procedure

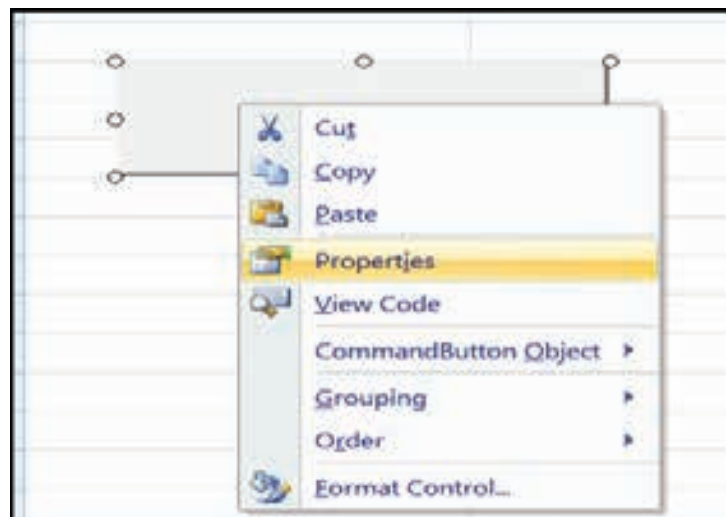
TASK 1: Area of a rectangle

Step 1 : Open MS Excel and press Alt+F11 to activate Visual Basic Editor

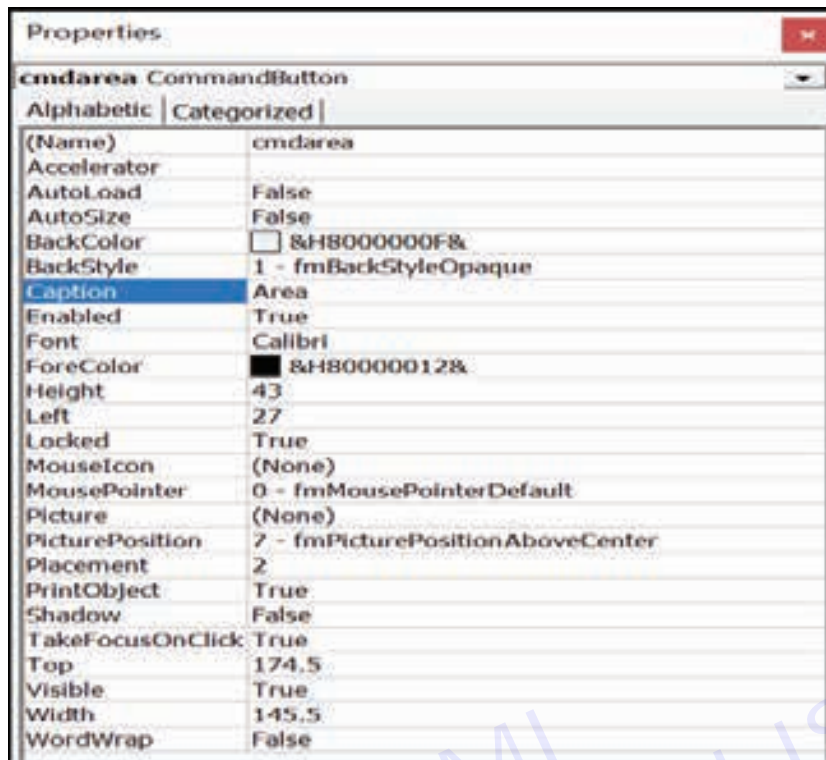
Step 2: Develop the Interface i.e., Place the required Controls from Developer tab in the Excel Sheet and Set the Properties .



Step 2.1: To set the property Right Click on each controls (CommandButton1& CommandButton2) and select 'properties' from the drop down menu.



Step 2.2: Set the required properties as follows:



For Example:

Sl. No	Control	Property	Value
1	CommndButton1	Name	cmdarea
		Caption	Area
2	CommandButton2	Name	cmdclear
		Caption	Clear

Step 3: Double Click on each Control and write down the Code .

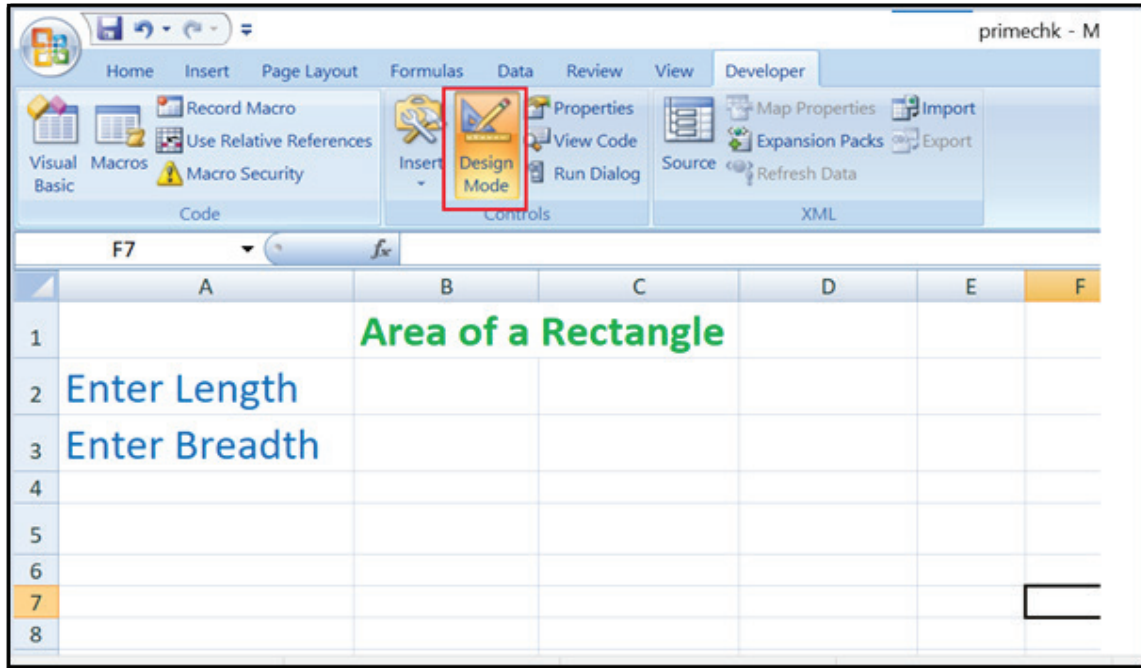
```
Private Sub cmdarea_Click()
Dim length As Integer, breadth As Integer
length = Range("B2")
breadth = Range("B3")
Range("A5") = "Area is"
Range("B5") = length * breadth
End Sub

Private Sub cmdclear_Click()
Range("A5,B2:B5") = " "
End Sub
```

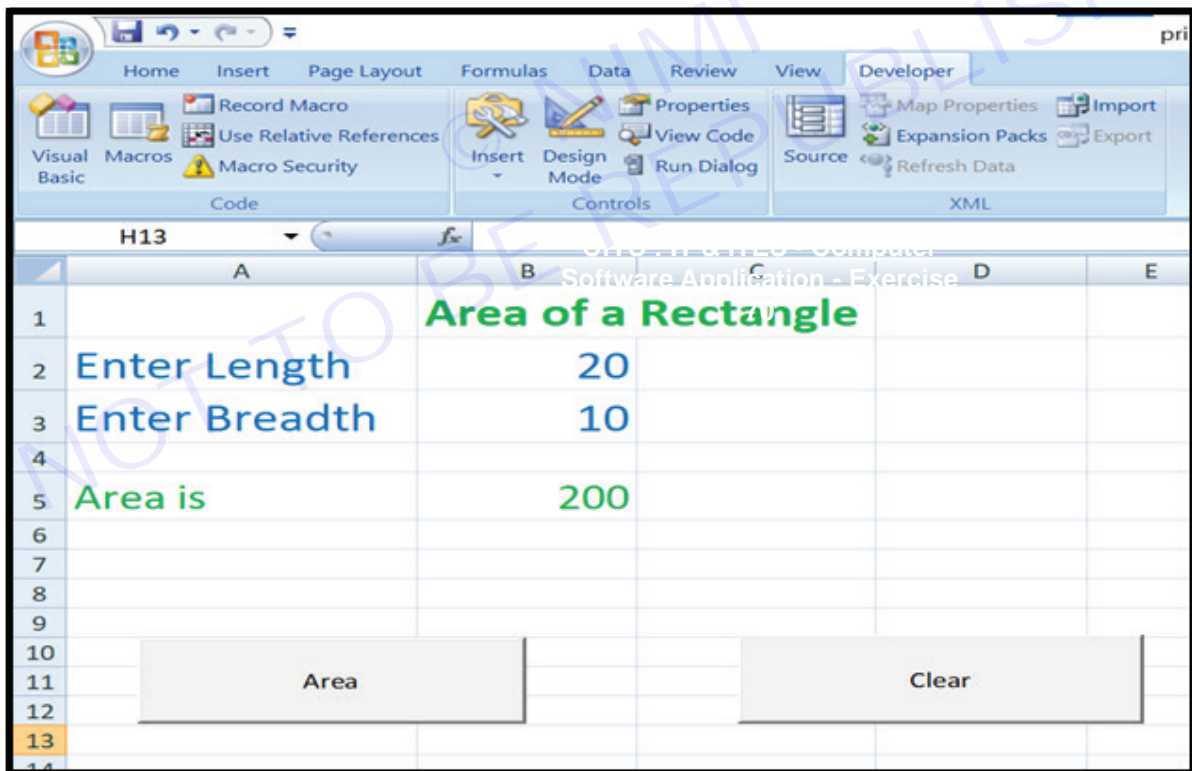
Step 4 : Then save the workbook type as a 'Excel Macro- Enabled Workbook'

Step 5: Run the Macro

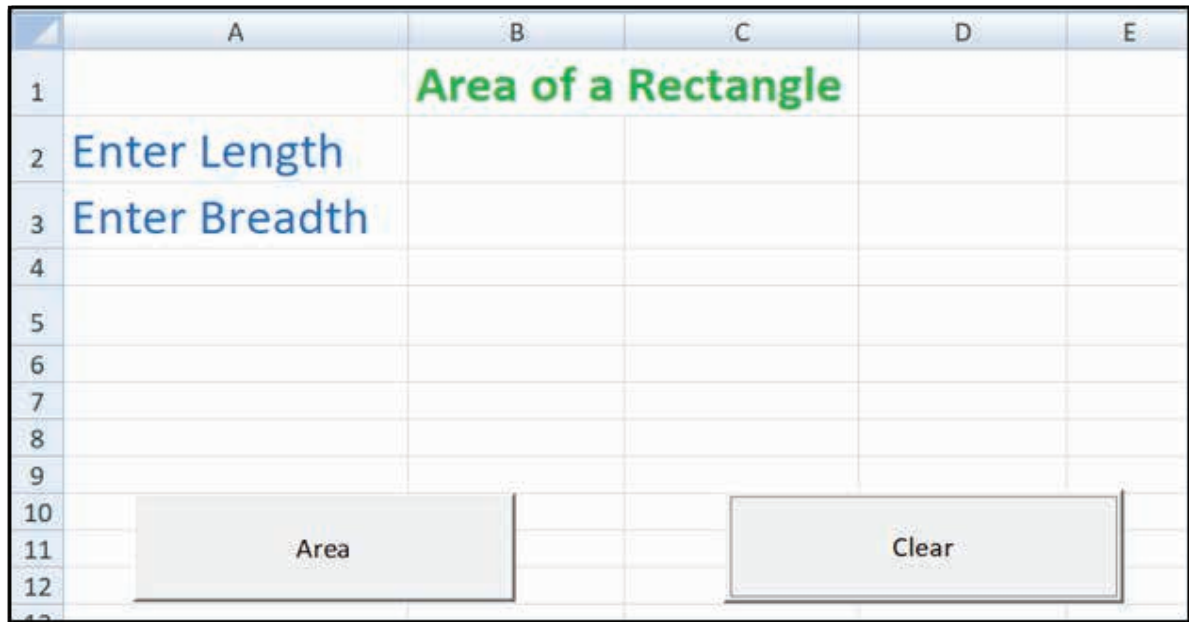
Step 5.1: Click on the Design mode icon to turn of the Design mode .



Step 5.2: Enter the values of Length in Cell B2 and Breadth in Cell B3 AND Click on the Area Button to view the Area in the Cell B5.



Step 5.3: Clear Button is used to clear the input data and result .

**Explanation :**

This VBA macro is designed to calculate the area of a rectangle based on the length and breadth entered into specific cells in an Excel worksheet. Let's break down the code:

1 Variable Declarations:

- Two integer variables, length and breadth, are declared to store the dimensions of the rectangle.

2 Assigning Values:

- The length variable is assigned the value of the cell B2, which presumably holds the length of the rectangle.
- The breadth variable is assigned the value of the cell B3, which presumably holds the breadth of the rectangle.

3 Calculating Area:

- The area of the rectangle is calculated by multiplying the length and breadth variables.
- The result is then displayed in cell B5 of the Excel worksheet.

4 Displaying Result:

- The message "Area is" is displayed in cell A5, followed by the calculated area in cell B5.

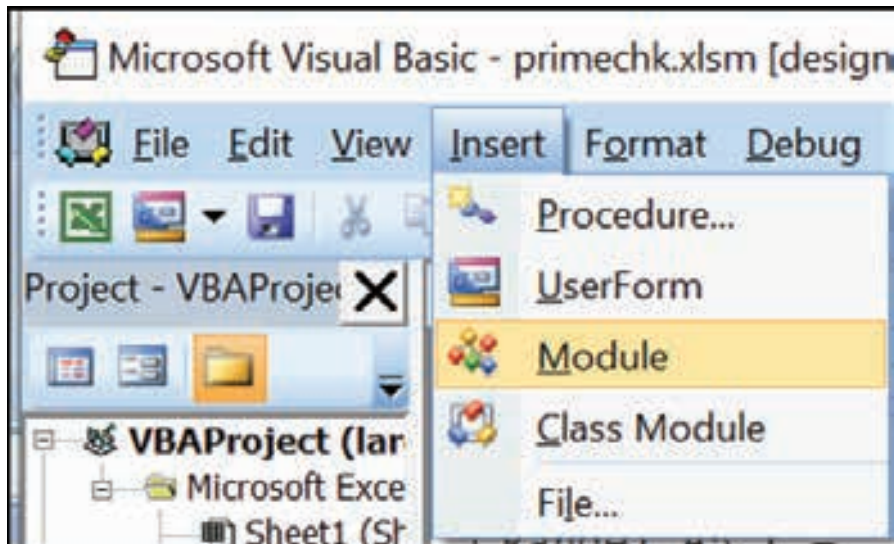
Explanation Summary:

- The macro retrieves the length and breadth of a rectangle from specific cells in the Excel worksheet.
- It calculates the area of the rectangle using the formula: $\text{Area} = \text{length} \times \text{breadth}$.
- The calculated area is displayed in another cell of the worksheet, along with a descriptive label.

This macro simplifies the process of calculating the area of a rectangle in Excel, allowing users to quickly obtain results based on the provided dimensions.

TASK 2: Create and modify a simple macro in Excel to calculate Body Mass Index (BMI) based on user input of weight in kilograms and height in meters

Step 1: Open Ms Excel and press Alt+F11 to activate VBE(Visual Basic Editor) and go to Insert Module as follows:



Step 2: Type the following Sub Procedure and save it in a macro enabled work book

```
Sub Calculate_BMI()
```

```
    Dim weight As Double
```

```
    Dim height As Double
```

```
    Dim bmi As Double
```

```
    Dim result As String
```

```
    ' Prompt user to enter weight in kilograms
```

```
    weight = InputBox("Enter your weight in kilograms:")
```

```
    ' Prompt user to enter height in meters
```

```
    height = InputBox("Enter your height in meters:")
```

```
    ' Check if weight and height are greater than 0
```

```
    If weight > 0 And height > 0 Then
```

```
        ' Calculate BMI using the formula: weight / (height * height)
```

```
        bmi = weight / (height * height)
```

```
    ' Determine the BMI category
```

```
        If bmi < 18.5 Then
```

```
            result = "Underweight"
```

```
        ElseIf bmi >= 18.5 And bmi < 25 Then
```

```
            result = "Normal weight"
```

```
        ElseIf bmi >= 25 And bmi < 30 Then
```

```
            result = "Overweight"
```

```
        Else
```

```
            result = "Obese"
```

```
        End If
```

```
    ' Display the calculated BMI and category
```

```
    MsgBox "Your BMI is: " & Round(bmi, 2) & vbCrLf & "BMI Category: " & result
```

Else

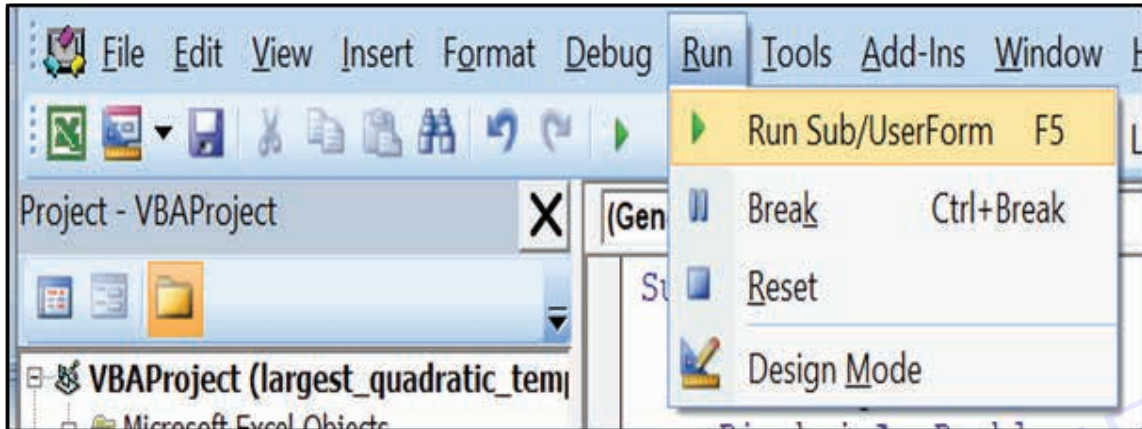
' Display error message if weight or height is not valid

MsgBox "Invalid input. Please enter valid weight and height."

End If

End Sub

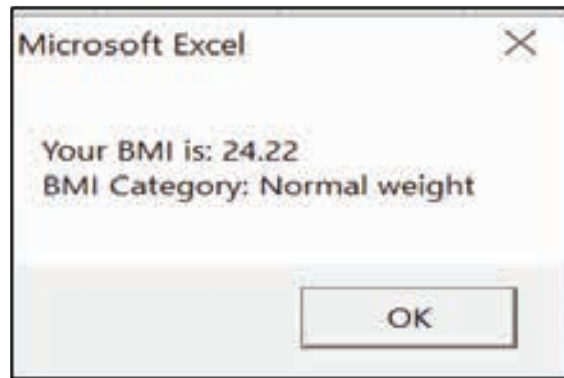
Step 3 : Run the Macro and view the result.



Output:

 A screenshot of a Microsoft Excel dialog box titled 'Microsoft Excel'. The text inside reads 'Enter your weight in kilograms:'. There are 'OK' and 'Cancel' buttons on the right. A text input field at the bottom contains the number '70'. A large watermark 'NOT TO BE REPUBLISHED' is overlaid on the image.

 A screenshot of a Microsoft Excel dialog box titled 'Microsoft Excel'. The text inside reads 'Enter your height in meters:'. There are 'OK' and 'Cancel' buttons on the right. A text input field at the bottom contains the number '1.70'. A large watermark 'NOT TO BE REPUBLISHED' is overlaid on the image.



Explanation:

This VBA subroutine named Calculate_BMI calculates the Body Mass Index (BMI) based on the weight and height entered by the user and categorizes the BMI into different weight categories such as underweight, normal weight, overweight, and obese. Here's an explanation of the code:

1 Variable Declarations:

- weight, height, bmi: These variables of type Double are used to store the weight, height, and calculated BMI, respectively.
- result: This variable of type String holds the category of BMI.

2 Prompt for Weight and Height:

- The user is prompted to enter their weight in kilograms and height in meters using the InputBox function.

3 Check for Valid Input:

- The code checks if the entered weight and height are greater than 0.

4 Calculate BMI:

- If the weight and height are valid, the BMI is calculated using the formula $\text{weight} / (\text{height} * \text{height})$.

5 Determine BMI Category:

- The BMI is categorized based on standard BMI ranges:
- BMI less than 18.5 is considered "Underweight".
- BMI between 18.5 and 24.9 is considered "Normal weight".
- BMI between 25 and 29.9 is considered "Overweight".
- BMI 30 or greater is considered "Obese".

6 Display Results:

- The calculated BMI, rounded to two decimal places using the Round function, along with its corresponding category, is displayed in a message box using the MsgBox function.

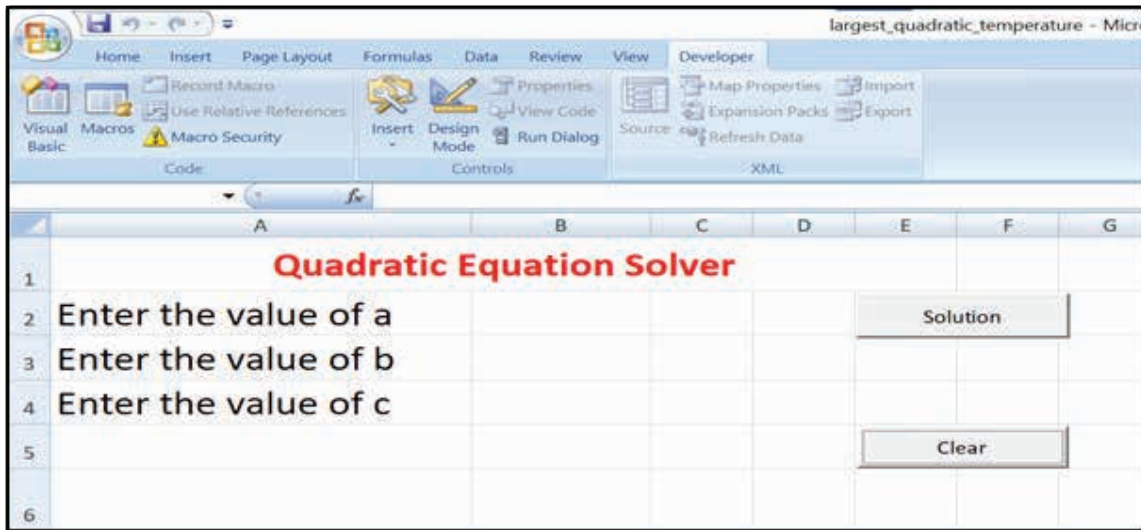
7 Error Handling:

- If the user enters invalid input (weight or height less than or equal to 0), an error message is displayed using MsgBox.

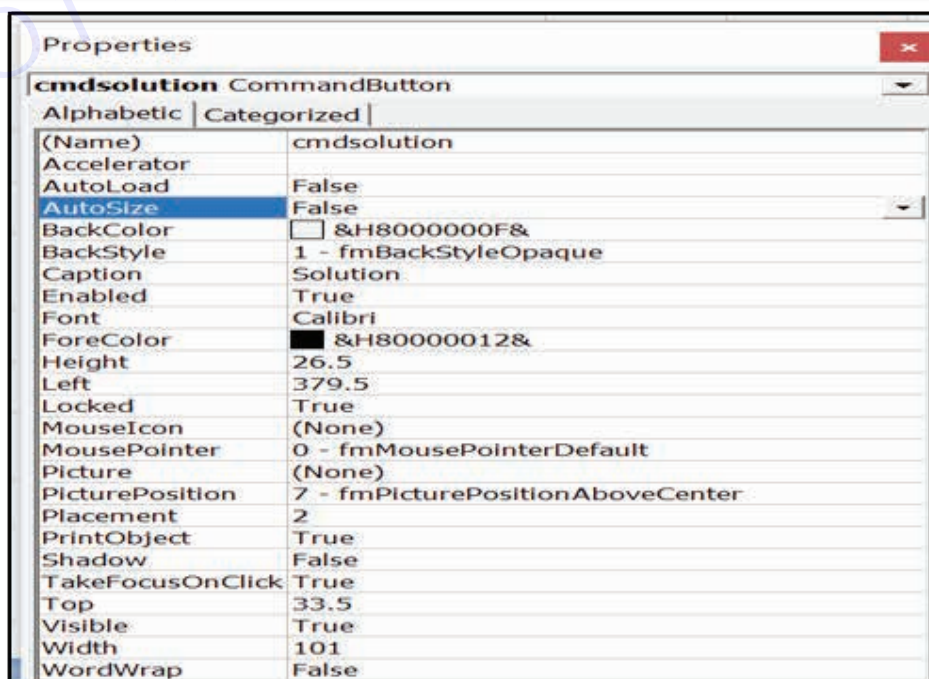
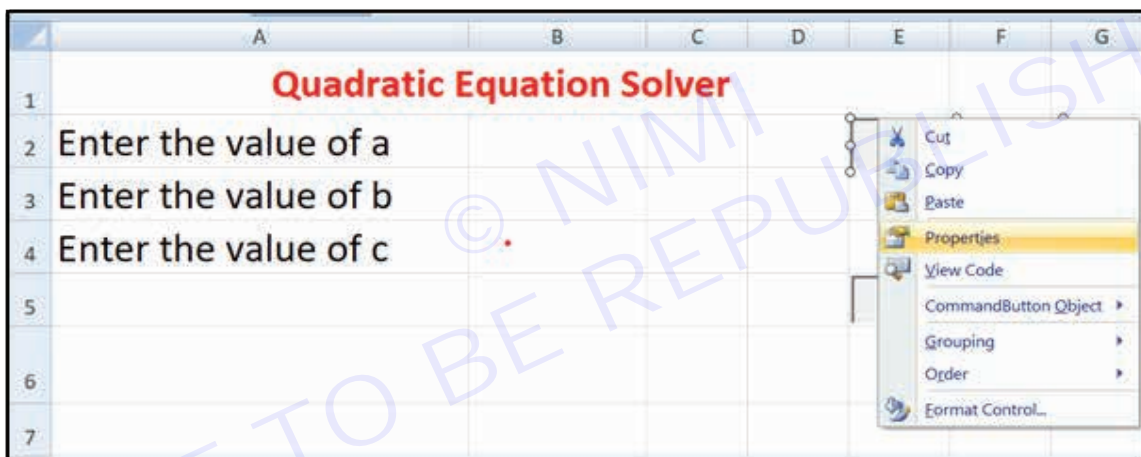
This subroutine provides a simple way to calculate BMI and categorize it, offering users insights into their weight status.

TASK 3: Quadratic Equation Solver

Step 1: Develop the interface



Step 2: Set the property of the controls



Property Box

S. No.	Control	Property	Value
1	CommndButton1	Name	cmdsolution
		Caption	Solution
2	CommandButton2	Name	cmdclear
		Caption	Clear

Step 3: Double click on the Clear button and write down the following code:

```
Private Sub cmdclear_Click()
Range("B2:B9,A6:A8") = " "
End Sub
```

Step 4: Double click on the Solution button and write down the following code:

```
Private Sub cmdsolution_Click()
Dim a As Integer, b As Integer, c As Integer
Dim d As Single, root1 As Single, root2 As Single
a = InputBox("Enter the value of a")
b = InputBox("Enter the value of b")
c = InputBox("Enter the value of c")
Range("B2") = a
Range("B3") = b
Range("B4") = c
d = b * b - 4 * a * c
Range("A6") = "Discriminant"
Range("B6") = Round(d, 2)
If d = 0 Then
root1 = -b / (2 * a)
Range("A7") = "Root is "
Range("B7") = Round(root1, 2)
Range("B9") = " Roots are real and Equal"
Elseif d > 0 Then
root1 = (-b + Sqr(d)) / (2 * a)
root2 = (-b - Sqr(d)) / (2 * a)
Range("A7") = "Root1 is "
Range("B7") = Round(root1, 2)
Range("A8") = "Root2 is "
Range("B8") = Round(root2, 2)
Range("B9") = " Roots are real and UnEqual"
Else
Range("B9") = " Roots are imaginary"
```

End If

End Sub

Step 5: Then save the workbook type as a 'Excel Macro- Enabled Workbook'

Step 6: Run the Macro and view the result.



Microsoft Excel

Enter the value of a

OK

Cancel

-1

This screenshot shows a Microsoft Excel dialog box titled "Microsoft Excel". The text "Enter the value of a" is displayed above a text input field. The input field contains the value "-1". To the right of the input field are two buttons: "OK" and "Cancel".



Microsoft Excel

Enter the value of b

OK

Cancel

0

This screenshot shows a Microsoft Excel dialog box titled "Microsoft Excel". The text "Enter the value of b" is displayed above a text input field. The input field contains the value "0". To the right of the input field are two buttons: "OK" and "Cancel".



Microsoft Excel

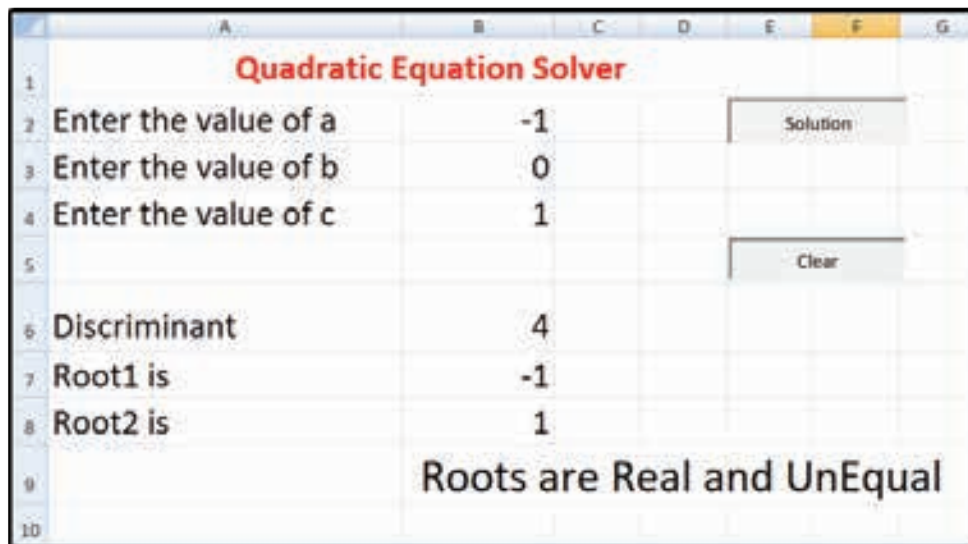
Enter the value of c

OK

Cancel

1

This screenshot shows a Microsoft Excel dialog box titled "Microsoft Excel". The text "Enter the value of c" is displayed above a text input field. The input field contains the value "1". To the right of the input field are two buttons: "OK" and "Cancel".



This VBA (Visual Basic for Applications) macro is designed to solve quadratic equations of the form $ax^2 + bx + c = 0$ and display the roots and nature of the roots (real and equal, real and unequal, or imaginary) in an Excel worksheet.

Here's an explanation of the macro:

1 Variable Declarations:

- Three integer variables a, b, and c are declared to store the coefficients of the quadratic equation.
- Three single precision variables d, root1, and root2 are declared to store the discriminant and roots of the equation.

2 User Input:

- The macro prompts the user to input the values of a, b, and c using InputBoxes.

3 Calculating the Discriminant:

- The discriminant d is calculated using the formula: $b^2 - 4ac$
- The calculated discriminant is rounded to 2 decimal places.

4 Displaying the Discriminant:

- The calculated discriminant is displayed in cell B6 of the Excel worksheet under the heading "Discriminant".

5 Determining the Roots:

- The macro checks the value of the discriminant d to determine the nature of the roots.
- If d is equal to 0, there is one real root, which is calculated and displayed.
- If d is greater than 0, there are two real roots, which are calculated and displayed.
- If d is less than 0, the roots are imaginary.

6 Displaying the Roots:

- The roots, along with their nature, are displayed in cells A7, B7 (for root 1), and A8, B8 (for root 2) of the Excel worksheet.
- If the roots are real and equal or real and unequal, the nature of the roots is displayed in cell B9.

Explanation Summary:

- The macro prompts the user for coefficients of a quadratic equation.
- It calculates the discriminant and determines the nature of the roots.
- The roots and their nature are displayed in an Excel worksheet.

This macro offers a handy tool for quickly solving quadratic equations and understanding their nature within Excel.

Related Exercises:

- 1 Develop a macro to perform temperature conversion(Celsius to Fahrenheit and Vice versa)
- 2 Develop a macro for interest calculation($I=PNR$)
- 3 Develop a macro to do Simple Arithmetic Operations

EXERCISE 71 : Perform form controls and create simple data entry form with macros

Objectives

At the end of this exercise you shall be able to

- create simple data entry form with macros.

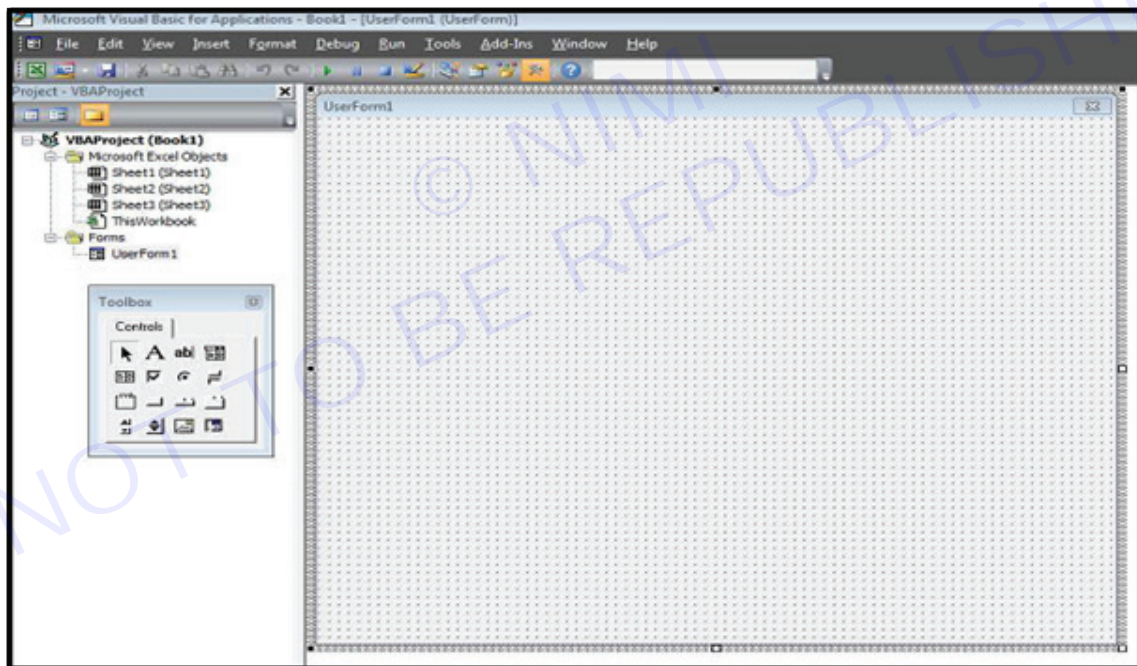
Procedure

User form is a customised interface, developed in VBA. It enables a user to interact, using a form, in an organised and logical manner or to enter data and retrieve data with an Excel Worksheet or Run a VBA application.

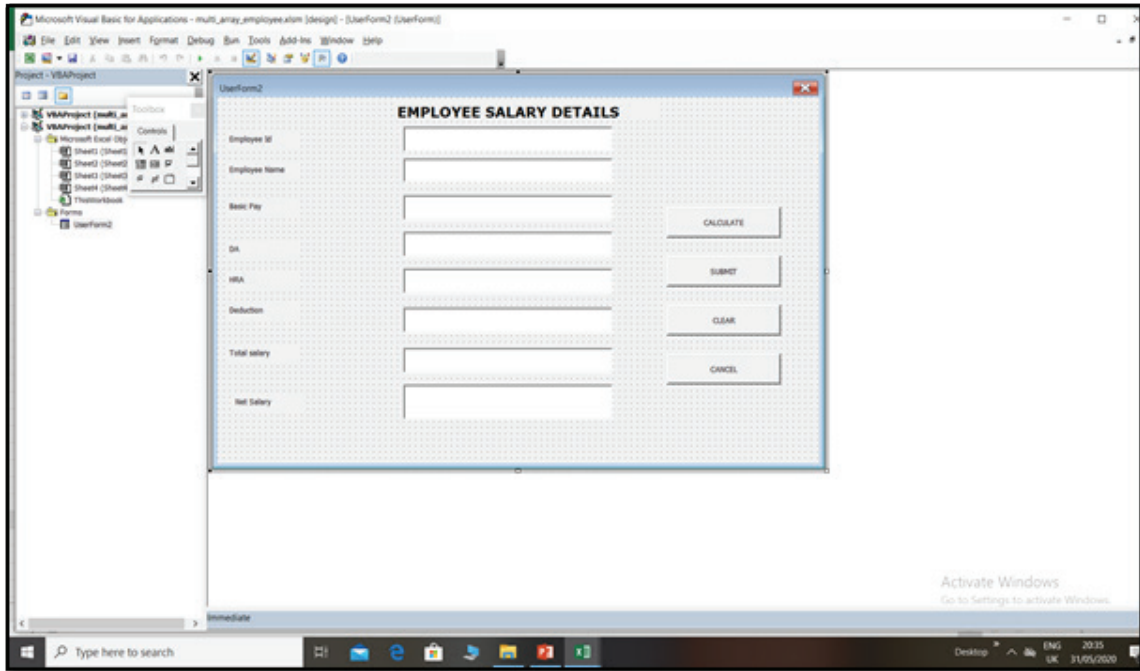
- User Forms acts as a container in which you add multiple controls, each of which has a specific use and associated properties.

TASK 1: Create a User Form – Employee Salary Details

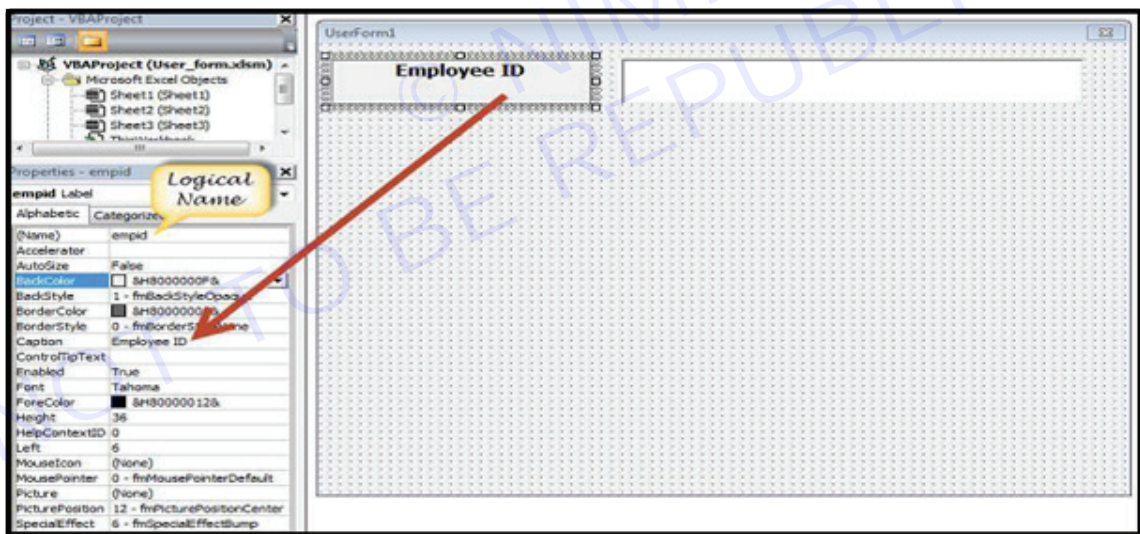
- **Step 1:** Navigate to VBA Window by pressing Alt+F11 and Navigate to “Insert” Menu and select “User Form”. Upon selecting, the user form is displayed as shown in the following screenshot.



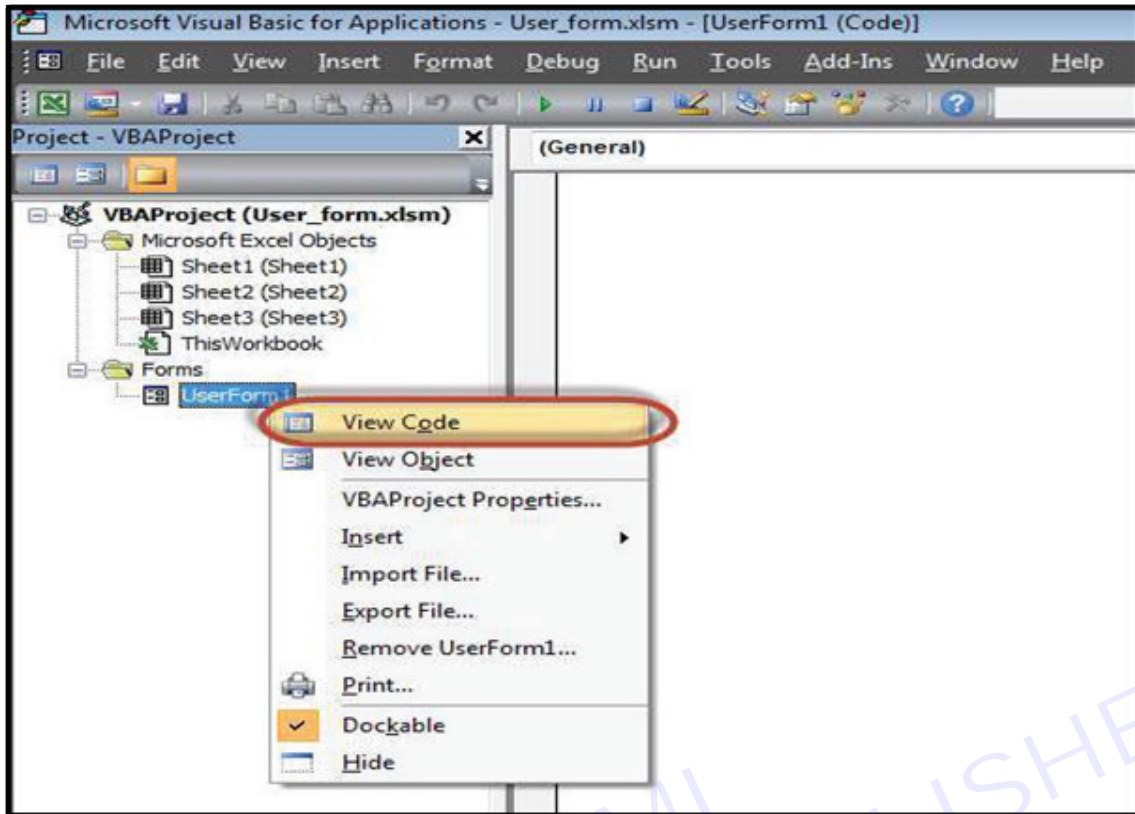
- **Step 2:** Design the forms using the given controls.



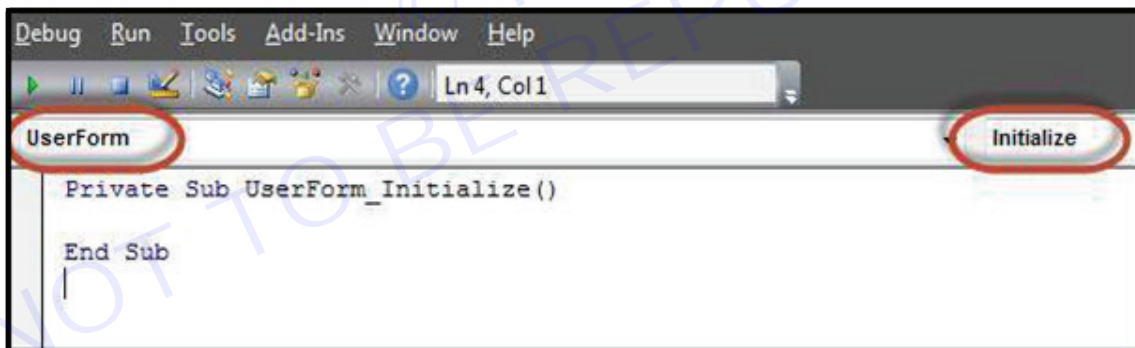
- **Step 3:** After adding each control, the controls have to be named through property settings. Caption corresponds to what appears on the form and name corresponds to the logical name that will be appearing when you write VBA code for that element.



- **Step 4:** Add the code for the form load event by performing a right-click on the form and selecting 'View Code'



- **Step 5:** Select 'Userform' from the objects drop-down and select 'Initialize' method as shown in the following screenshot.



- **Step 6:** Upon Loading the form, ensure that the text boxes are cleared.

```
Private Sub UserForm_Initialize()
```

```
txtid.Value = ""
```

```
txtname.Value = ""
```

```
txtbp.Value = ""
```

```
txtda.Value = ""
```

```
txthra.Value = ""
```

```
txtded.Value = ""
```

```
txtts.Value = ""
```

```
txtns.Value = ""
```

```
txtid.SetFocus
```

```
End Sub
```

when the user form is initialized, all text boxes' contents are cleared or set to default values, and the focus is set to the txtid text box. This ensures a clean and focused user interface when the form is opened, providing a clear starting point for user input or interaction.

- **Step 7:** Now add the code to the Calculate and Submit button. Upon clicking the submit button, the user should be able to add the values into the worksheet. The following VBA code seems to be part of a user interface for calculating salaries based on certain inputs such as employee ID, name, basic pay, and deductions.

```
Dim row As Long 'in general declaration
```

```
Private Sub cmdcalculate_Click()
```

```
Dim eid As Integer, ename As String, bp As Single, da As Single, hra As Single, ts As Single, ns As Single, ded As Single
```

```
Cells(2, 1).Value = "Employee Id"
```

```
Cells(2, 2).Value = "Employee Name"
```

```
Cells(2, 3).Value = "Basic Pay"
```

```
Cells(2, 4).Value = "DA"
```

```
Cells(2, 5).Value = "HRA"
```

```
Cells(2, 6).Value = "Deduction"
```

```
Cells(2, 7).Value = "Total Salary"
```

```
Cells(2, 8).Value = "Net Salary"
```

```
eid = txtid.Value
```

```
ename = txtname.Text
```

```
bp = txtbp.Value
```

```
da = bp * (107 / 100)
```

```
txtda.Text = da
```

```
hra = bp * (25 / 100)
```

```
txthra.Text = hra
```

```
ded = txtded.Value
```

```
txtts.Value = bp + hra + da
```

```
txtns.Value = txtts.Value - ded
```

```
End Sub
```

This VBA code seems to be part of a user interface for calculating salaries based on certain inputs such as employee ID, name, basic pay, and deductions. Let's break down the code and explain each part:

1 Variable Declarations:

- **row As Long:** This declares a variable named row as a Long data type. However, this variable is not used in the provided code snippet.
- **eid As Integer, ename As String, bp As Single, da As Single, hra As Single, ts As Single, ns As Single, ded As Single:** These lines declare several variables:
 - **eid** as an Integer to store Employee ID,
 - **ename** as a String to store Employee Name,
 - **bp, da, hra, ts, ns,** and **ded** as Single data types to store Basic Pay, Dearness Allowance, House Rent Allowance, Total Salary, Net Salary, and Deductions respectively.

2 Command Button Click Event (cmdcalculate_Click()):

- This subroutine is executed when the user clicks the "Calculate" button.

- The values from text boxes (**txtid**, **txtname**, **txtbp**, **txtded**) are assigned to respective variables (**eid**, **ename**, **bp**, **ded**).
- Dearness Allowance (**da**) is calculated as 107% of the Basic Pay (**bp**) and stored in the **da** variable.
- House Rent Allowance (**hra**) is calculated as 25% of the Basic Pay (**bp**) and stored in the **hra** variable.
- Total Salary (**ts**) is calculated as the sum of Basic Pay (**bp**), Dearness Allowance (**da**), and House Rent Allowance (**hra**) and displayed in the **txtts** textbox.
- Net Salary (**ns**) is calculated as the Total Salary (**ts**) minus Deductions (**ded**) and displayed in the **txtns** textbox.

3 Worksheet Cell Values:

- The lines setting values in Cells(2, 1) through Cells(2, 8) are meant to label the columns in the worksheet. These cells are likely the headers for the data being entered or calculated.

4 Displaying Calculated Values:

- The calculated values for Dearness Allowance (**da**) and House Rent Allowance (**hra**) are displayed in text boxes **txtda** and **txthra** respectively. However, these text boxes are not shown in the provided code snippet.

Overall, this code snippet captures the process of calculating and displaying employee salary details based on input provided through a user interface. It seems to be designed to work with Excel VBA given the references to cells and text boxes.

```
Private Sub cmdsubmit_Click()
```

```
Dim row As Long
```

```
    'Make Sheet4 active
```

```
    Sheet4.Activate
```

```
    'Determine emptyRow
```

```
row = WorksheetFunction.CountA(Range("A:A")) + 1
```

```
Cells(row, 1).Value = txtid.Value
```

```
Cells(row, 2).Value = txtname.Value
```

```
Cells(row, 3).Value = txtbp.Value
```

```
Cells(row, 4).Value = txtda.Value
```

```
Cells(row, 5).Value = txthra.Value
```

```
Cells(row, 6).Value = txtded.Value
```

```
Cells(row, 7).Value = txtts.Value
```

```
Cells(row, 8).Value = txtns.Value
```

```
End Sub
```

This VBA code is associated with the click event of a button named cmdsubmit. Here's what the code does:

1 Variable Declaration:

- **row As Long:** Declares a variable named row as a Long data type. This variable will be used to determine the row where the new data will be entered.

2 Activating Sheet4:

- **Sheet4.Activate:** This line makes sure that Sheet4 is the active sheet where the data will be entered.

3 Determining the Next Empty Row:

- **row = WorksheetFunction.CountA(Range("A:A")) + 1:** This line calculates the next empty row in column A of the active worksheet (Sheet4) by counting the number of non-empty cells in column A and adding 1. It uses the **CountA** function of the **WorksheetFunction** object.

4 Assigning Values to Cells:

- The subsequent lines assign values to cells in the active worksheet (Sheet4) based on the inputs provided in various text boxes (**txtid, txtname, txtbp, txtda, txthra, txtded, txtts, txtns**).
- Each value is assigned to a specific column in the determined row, allowing for the storage of employee information and salary details.

5 Data Entry:

- The data from the text boxes (**txtid, txtname, etc.**) are entered into the next empty row in the specified columns of Sheet4.

This code essentially allows users to enter employee details and their corresponding salary information into Sheet4 of the Excel workbook by clicking the "Submit" button (cmdsubmit). It automates the process of determining the next empty row and filling in the necessary information accordingly.

- **Step 8:** Add a method to close the form when the user clicks the Cancel button.

```
Private Sub cmdcancel_Click()
Unload Me
End Sub
```

The above VBA code is associated with the click event of a button named cmdcancel. Here's what the code does:

1 Unload Form:

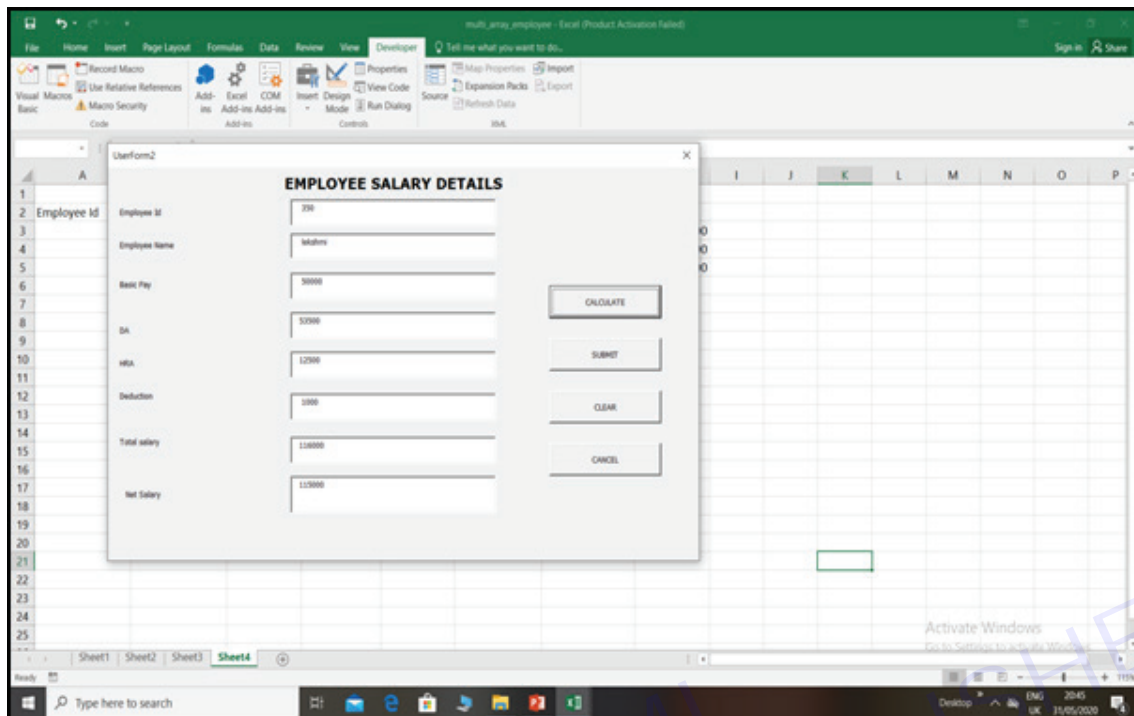
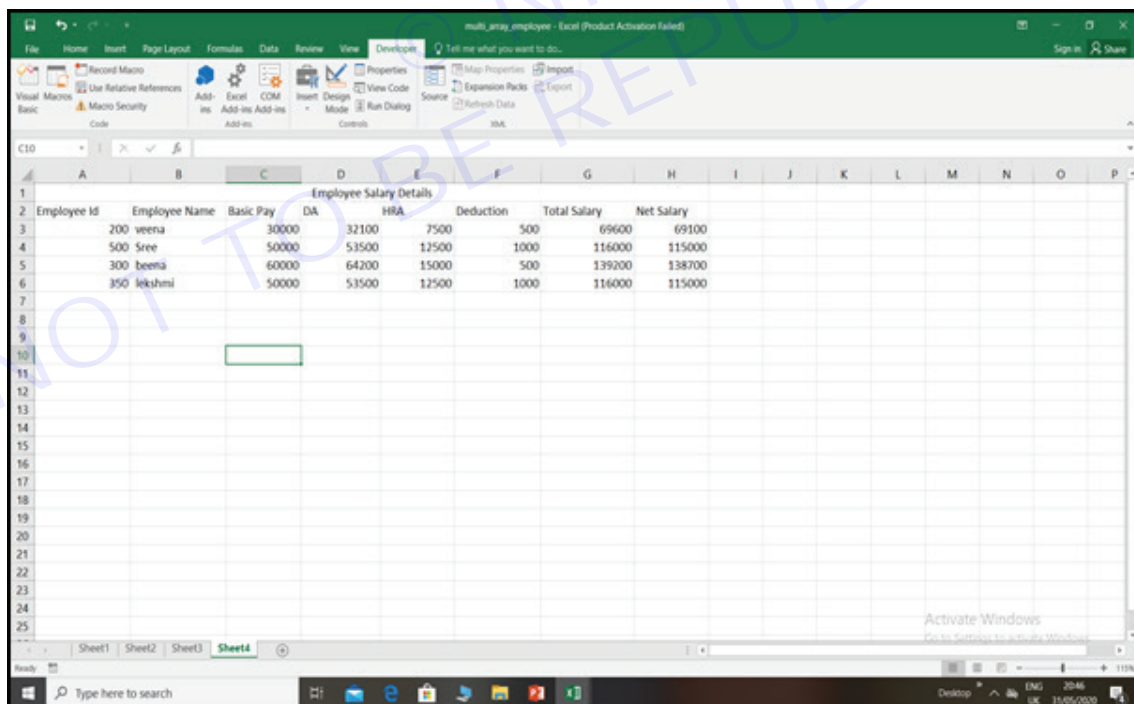
- **Unload Me:** This line unloads (closes) the user form associated with the VBA code. The **Me** keyword refers to the current instance of the form where the VBA code is located. **Unload Me** effectively closes the form when the user clicks the "Cancel" button (**cmdcancel**).

- **Step 9:** Add a method to clear the form when the user clicks the Clear button. Private Sub cmdclear_Click()

```
txtid.Value = ""
txtname.Value = ""
txtbp.Value = ""
txtda.Value = ""
txthra.Value = ""
txtded.Value = ""
txtts.Value = ""
txtns.Value = ""
txtid.SetFocus
End Sub
```

This action provides a convenient way for users to clear out all input fields and start fresh if needed. When the user clicks the "Clear" button (cmdclear), all the text boxes' contents are cleared, and the focus is set back to the txtid text box, ready for new input or interaction.

- **Step 10:** Execute the form by clicking the "Run" button. Enter the values into the form and click the 'Calculate' button and then click the 'Submit' button. Automatically the values will flow into the worksheet as shown in the following screenshot.

Output:**1 CLICK ON CALCULATE BUTTON****2 CLICK ON SUBMIT BUTTON****Related Exercise:**

- 1 Create a User Form – Student's Result Sheet (Hints: Design a form with Roll No, Name, Marks of 3 subjects(each subject out of 100), calculate the total marks, percentage and grade and display the details in the Excel sheet).Grade can be calculated based on the following conditions

If marks of any subject less than 35 then Grade = 'F' otherwise :

Percentage	Grade
≥ 90	A+
≥ 80 and < 90	A
≥ 70 and < 80	B+
≥ 60 and < 70	B
≥ 50 and < 60	C+
≥ 40 and < 50	C
≥ 35 and < 40	D+

- 2 Build a form to input student grades for different assignments and exams. Include fields for Assignment Scores, Exam Scores, and weights for each category. Calculate the overall grade based on weighted averages.
- 3 Create an Excel form for inventory management. Include fields for Item Name, Quantity In, Quantity Out, and a button to calculate the current inventory level. Display the calculated inventory level for each item.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 72 : Look up data by using functions

Objectives

At the end of this exercise you shall be able to

- lookup data by using functions VLOOKUP ,HLOOKUP & INDEX formula.

Procedure

Method 1: VLOOKUP

VLOOKUP is a powerful function in Excel used to search for a value in the first column of a range (table or array) and return a value in the same row from a column you specify.

Syntax:

```
=VLOOKUP(lookup_value, table_array, col_index_num, [range_lookup])
```

- **lookup_value:** The value you want to search for.
- **table_array:** The range of cells that contains the data you want to search in. The first column of this range should contain the lookup values.
- **col_index_num:** The column number in the table_array from which to return the value.
- **[range_lookup] (optional):** A logical value that specifies whether you want an exact match or an approximate match. If TRUE or omitted, it will find an approximate match. If FALSE, it will find an exact match.

TASK 1: Create a new dataset with columns ID, Product, and Price. Use the VLOOKUP function to find the price of a product with ID 201

Products Details

ID	Product	Price
101	Laptop	80000
102	Smartphone	50000
103	Camera	350000
104	Smartwatch	30000
201	Headphones	2000

Now, suppose you want to look up the Price of the Product with ID 201. You can use the **VLOOKUP** function.

In cell D2, you can enter the formula:

```
=VLOOKUP(A7,A3:C7,3,FALSE)
```

	A	B	C	D	E	F
1	Product Details					
2	ID	Product	Price			
3	101	Laptop	80000	=VLOOKUP(A7,A3:C7,3,FALSE)		
4	102	Smartphone	50000			
5	103	Camera	350000			
6	104	Smartwatch	30000			
7	201	Headphones	2000			
8						
9						

Explanation:

- lookup_value: This is the value you want to search for in the first column of the table array (in this case, cell A7).
- table_array: This is the range of cells that contains the data. It must include the column containing the lookup value and the column containing the data you want to retrieve (in this case, cells A3:C7).
- col_index_num: This is the column number in the table_array from which the matching value should be returned (in this case, 3, indicating the third column of the table_array).
- range_lookup: This is an optional argument that specifies whether to find an exact match or an approximate match. If set to FALSE, it will find an exact match.

In the given example, the VLOOKUP function searches for the value in cell A7 within the range A3:C7. It then returns the value from the third column (index 3) of the range if it finds an exact match. If no exact match is found, it returns an error or a value specified in the function.

After entering this formula, it will return the Price of the Product with ID 201.

	A	B	C	D
1	Product Details			
2	ID	Product	Price	
3	101	Laptop	80000	2000
4	102	Smartphone	50000	
5	103	Camera	350000	
6	104	Smartwatch	30000	
7	201	Headphones	2000	
8				

Related Exercises:

- 1 Create a new dataset with columns ID, Product, and Price, Department. Use the VLOOKUP function to find the price of a product with ID 104.

- 2 Create a dataset with columns EmployeeID, Name, Department, and Salary. Use VLOOKUP to find the salary of an employee with a given EmployeeID.
- 3 Create a dynamic table that allows users to input an EmployeeID, and VLOOKUP should automatically fetch and display the corresponding employee name, department, and salary.

Method 2: HLOOKUP

HLOOKUP is a function in Excel used to search for a value in the first row of a table (or array) and return a value in the same column from a row you specify. Here's how to use HLOOKUP in Excel:

Syntax:

```
=HLOOKUP(lookup_value, table_array, row_index_num, [range_lookup])
```

=HLOOKUP(lookup_value, table_array, row_index_num, [range_lookup])

- **lookup_value:** The value you want to search for.
- **table_array:** The range of cells that contains the data you want to search in. The first row of this range should contain the lookup values.
- **row_index_num:** The row number in the table_array from which to return the value.
- **[range_lookup] (optional):** A logical value that specifies whether you want an exact match or an approximate match. If TRUE or omitted, it will find an approximate match. If FALSE, it will find an exact match.

TASK 1: Create a dataset with columns Product, Samsung_Price, Apple_Price, and Realme_Price. Use HLOOKUP to find the price of a specific product in row 2.

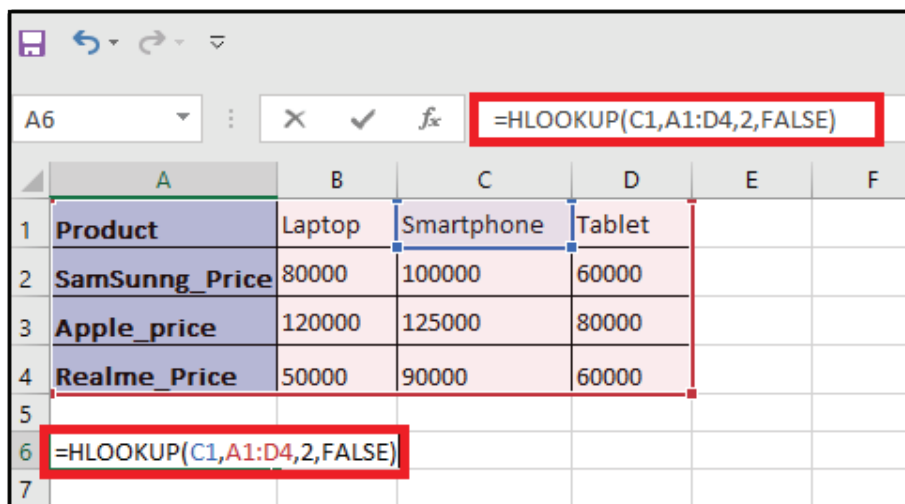
Step 1: Create the Dataset

Product	Laptop	Smartphone	Tablet
SamSung_Price	80000	100000	60000
Apple_price	120000	125000	80000
Realme_Price	50000	90000	60000

Step 2: Use HLOOKUP for Price Lookup

Assuming you want to find the price of the product "Smartphone," here is the HLOOKUP formula:

=HLOOKUP(C1,A1:D4,2,FALSE)



Explanation:

- **C1:** The value we are looking for in the first row of the table.
- **A1:D4:** The table array where the lookup value is located.
- **2:** The row number in the table from which to retrieve the value.
- **FALSE:** Exact match for the lookup value.

Step 3: Result

The formula would return the price of the “Smartphone” from row 2 of the table, which is Rs.100000/-

	A	B	C	D	E
1	Product	Laptop	Smartphone	Tablet	
2	SamSung_Price	80000	100000	60000	
3	Apple_price	120000	125000	80000	
4	Realme_Price	50000	90000	60000	
5					
6					
7					

Related Exercises:

- 1 Create a dataset with columns representing different cities (Chandigarh, Bhopal, Indore, Patna, Thiruvananthapuram, Kochi, Surat, Visakhapatnam) and rows representing temperature data for each month. Use HLOOKUP to find the temperature in Patna for the month of March.
- 2 Create a dataset with columns representing different months (January, February, March) and rows representing sales data for each product. Use HLOOKUP to find the sales of a specific product in January.

Method 3: Index Formula

The INDEX function in Excel is used to return the value of a cell in a specified row and column of a given range. It is particularly useful when you want to retrieve a specific value from a table of data.

The syntax of the INDEX function is as follows:

INDEX(array, row_num, [column_num])

- array: This is the range of cells from which you want to retrieve the value.
- row_num: This is the row number within the array from which to retrieve the value.
- column_num: (Optional) This is the column number within the array from which to retrieve the value. If omitted, INDEX returns the entire row specified by row_num.

TASK 1: Consider the following table representing the sales data for different products

Step 1: Create Data Set

Product Details

Product	January	February	March	April
Item A	100	120	80	150
Item B	75	90	110	85
Item C	120	100	130	95
Item D	90	80	75	110

Step 2: Use Index Formula

Using the INDEX function, find the sales value of "Item B" in February.

Product Details				
Product	January	February	March	April
Item A	100	120	80	150
Item B	75	90	110	85
Item C	120	100	130	95
Item D	90	80	75	110

Result:

The result of the formula is the sales value of "Item B" in February, which is 90.

Product Details				
Product	January	February	March	April
Item A	100	120	80	150
Item B	75	90	110	85
Item C	120	100	130	95
Item D	90	80	75	110
	90			

Related Exercises:

- 1 Create a data range in cells A1 to D10. Use the INDEX function to retrieve the value located in the 4th row and 3rd column.
- 2 Create a dataset in column A, starting from A1. Create a formula using INDEX to create a dynamic range that automatically expands as new data is added in column A.

EXERCISE 73 : Use advanced date functions

Objectives

At the end of this exercise you shall be able to

- use advanced date functions.

Procedure

TASK 1: Change Dates from/to Text

Using DATEVALUE Function

The DATEVALUE function in Excel converts a date represented as text into a serial number that Excel recognizes as a date.

Syntax of DATEVALUE function:

=DATEVALUE(date_text)

Formula	Serial Number
=DATEVALUE("28 March 2023")	45013
=DATEVALUE("12/10/2023")	45270
=DATEVALUE("23-June-23")	45100
=DATEVALUE("August 5, 2023")	45143

Note: Dates are stored in Excel as consecutive integers, and it is only the formatting of a cell that permits a number to be shown as a date. All dates are recorded as integers denoting the number of days from January 1, 1900 (number 1) to December 31, 9999 (number 2958465). You can change the serial number back to date format by pressing Ctrl+1.

TASK 2: Applying TEXT Function

The TEXT function is used to convert a numeric value, date, or time into a text string with a specified format.

Syntax of TEXT function:

=TEXT(value,format_text)

Dates	Dates as Text	Formula
5/10/2023	10-May-23	=TEXT(B5,"d-mmm-yy")
10/12/2023	12 October, 2023	=TEXT(B6,"dd mmmm, yyyy")
Monday, September 18, 2023	18/09/2023	=TEXT(B7,"dd/mm/yyyy")
6/10/2023	Saturday, Jun 10, 2023	=TEXT(B8,"dddd, mmm d, yyyy")
3/12/2021	2021, 12 March	=TEXT(B9,"yyyy, dd mmmm")

In the format text argument type your preferred formatting.

TASK 3: Fetch Dates

Using DAY Function

The DAY function extracts the day of the month from a given date and returns it as a numeric value.

Syntax of DAY function:

=DAY(date)

Dates	Day	Formula	Remarks
5/1/2024	1	=DAY(B5)	DAY function returns day number of a date as an integer from 1 to 31.
2024-10-03	3	=DAY(B6)	
Monday, September 18, 2023	18	=DAY(B7)	
	25	=DAY(DATE(2015,10,25))	
	21	=DAY(TODAY())	

TASK 4: Using MONTH Function

The MONTH function extracts the month from a given date and returns it as a numeric value.

Syntax of MONTH function:

=MONTH(date)

Dates	Month	Formula	Remarks
5/10/2023	5	=MONTH(B5)	MONTH function returns the month of a date as an integer ranging from 1 (January) to 12 (December).
2023-10-12	10	=MONTH(B6)	
Friday, July 21, 2023	7	=MONTH(B7)	
	3	=MONTH(DATE(2015,3,25))	
	2	=MONTH(TODAY())	

TASK 5: Applying YEAR Function

The YEAR function extracts the month from a given date and returns it as a numeric value.

Syntax of YEAR function:

=YEAR(date)

Date	Year	Formula	Remarks
5/10/2023	2023	=YEAR(B5)	YEAR function returns the year of a date as an integer from 1900 to 9999.
2019-10-12	2019	=YEAR(B6)	
Sunday, June 12, 2022	2022	=YEAR(B7)	
	2025	=YEAR("10-Mar-2025")	
	2015	=YEAR(DATE(2015,10,25))	
	2024	=YEAR(TODAY())	

TASK 6: Utilizing EOMONTH Function

The EOMONTH function finds the last day of the output month based on a specified number of months before or after a referenced date.

Syntax of EOMONTH function:

=EOMONTH(start_date, months)

Date	Output Date	Formula	Remarks
2/1/2024	5/31/2024	=EOMONTH(B5,3)	EOMONTH function returns the date of the last day of the month after adding or subtracting the specified number of months from the given date.
2020-05-03	2/29/2020	=EOMONTH(B6,-3)	
Tuesday, July 11, 2023	Monday, July 31, 2023	=EOMONTH(B7,0)	
	12/31/2015	=EOMONTH(DATE(2015,10,25),2)	
	2/29/2024	=EOMONTH(TODAY(),0)	

TASK 7: Applying WEEKDAY Function

The WEEKDAY function returns the day of the week for a given date and returns it as a numeric value.

Syntax of WEEKDAY function:

=WEEKDAY(date, [return_type])

Date	Day Serial No.	Formula	Remarks
5/10/2023	4	=WEEKDAY(B5)	Each day of the week is represented by a number from 1 to 7 in Excel and WEEKDAY function returns that number from the given date.
2018-10-12	5	=WEEKDAY(B6,2)	
Friday, September 18, 2020	2	=WEEKDAY(B7,14)	
	1	=WEEKDAY(DATE(2015,10,25),17)	
	6	=WEEKDAY(TODAY(),15)	

Return Type	Weekday Values
1 or omitted	1 through 7, Sunday to Saturday
2	1 through 7, Monday to Sunday
3	0 through 6, Monday to Sunday
11	1 through 7, Monday to Sunday
12	1 through 7, Tuesday - Monday
13	1 through 7, Wednesday - Tuesday
14	1 through 7, Thursday - Wednesday
15	1 through 7, Friday - Thursday
16	1 through 7, Saturday - Friday
17	1 through 7, Sunday to Saturday

TASK 8: Using WEEKNUM Function

The WEEKNUM function calculates the week number of a given date based on a specified numbering system.

Syntax of WEEKNUM function:

=WEEKNUM(date, [return_type])

The WEEKNUM function can be used in two different ways based on return type:

Way 1: Week 1 specifies the week which contains January 1st;

Way 2: Week 1 is the week that contains the first Thursday of the year.

Get the Week Number from Date Using WEEKNUM Function

Date	Week Serial No.	Formula	Remarks
5/10/2023	19	=WEEKNUM(B5)	WEEKNUM function returns the week number of the year for the referenced date as a integer from 1 to 53.
2018-10-12	42	=WEEKNUM(B6,12)	
Friday, September 18, 2020	38	=WEEKNUM(B7,21)	
	44	=WEEKNUM(DATE(2015,10,25),17)	
	8	=WEEKNUM(TODAY(),15)	

Return Type	Week Begins on	System
1 or omitted	Sunday	1
2	Monday	1
11	Monday	1
12	Tuesday	1
13	Wednesday	1
14	Thursday	1
15	Friday	1
16	Saturday	1
17	Sunday	1
21	Monday	2

TASK 9: Utilizing ISOWEEKNUM Function

The ISOWEEKNUM function gives the ISO week number of the year of a given date.

Syntax of ISOWEEKNUM function:

=ISOWEEKNUM(date)

Get the Week Number from Date Using ISOWEEKNUM Function

Date	Week Serial No.	Formula	Remarks
5/10/2023	19	=ISOWEEKNUM(B5)	ISOWEEKNUM function returns the International Organization for Standardization (ISO) week number of the year for the specified date.
2018-10-12	41	=ISOWEEKNUM(B6)	
Friday, September 18, 2020	38	=ISOWEEKNUM(B7)	
	43	=ISOWEEKNUM(DATE(2015,10,25))	
	8	=ISOWEEKNUM(TODAY())	



TASK 10: Applying DAYS360 Function

The DAYS360 function calculates the difference between two dates using a 360-day year, assuming each month has 30 days.

Syntax of DAYS360 function:

=DAYS360(start_date, end_date, [method])

Start Date	End Date	Difference	Formula	Remarks
25-May-21	21-Oct-28	2666	=DAYS360(B5,C5)	DAYS360 function calculates the number of days between two dates, based on 360-day year.
1/31/2023	12/3/2024	663	=DAYS360(B6,C6,TRUE)	
March 3, 2021	November 16, 2023	973	=DAYS360(B7,C7,FALSE)	

Note: Method is an optional argument to specify the day-count method. TRUE means the function counts European method and FALSE means the method will be the US method. The default is FALSE.

TASK 11: Applying EDATE Function

The EDATE function adds or subtracts a specified number of months to a given date and returns the resulting date.

Syntax of EDATE function:

=EDATE(start_date, months)

Start Date	End Date	Formula	Remarks
25-May-21	25-Mar-22	=EDATE(B5,10)	EDATE function returns the date of a future or past month where the day of the month is identical to the date being specified.
1/31/2023	4/30/2024	=EDATE(B6,15)	
March 3, 2021	April 3, 2020	=EDATE(B7,-11)	

EXERCISE 74 : Demonstrate advanced charts

Objectives

At the end of this exercise you shall be able to

- prepare the advanced charts and graphs in excel.

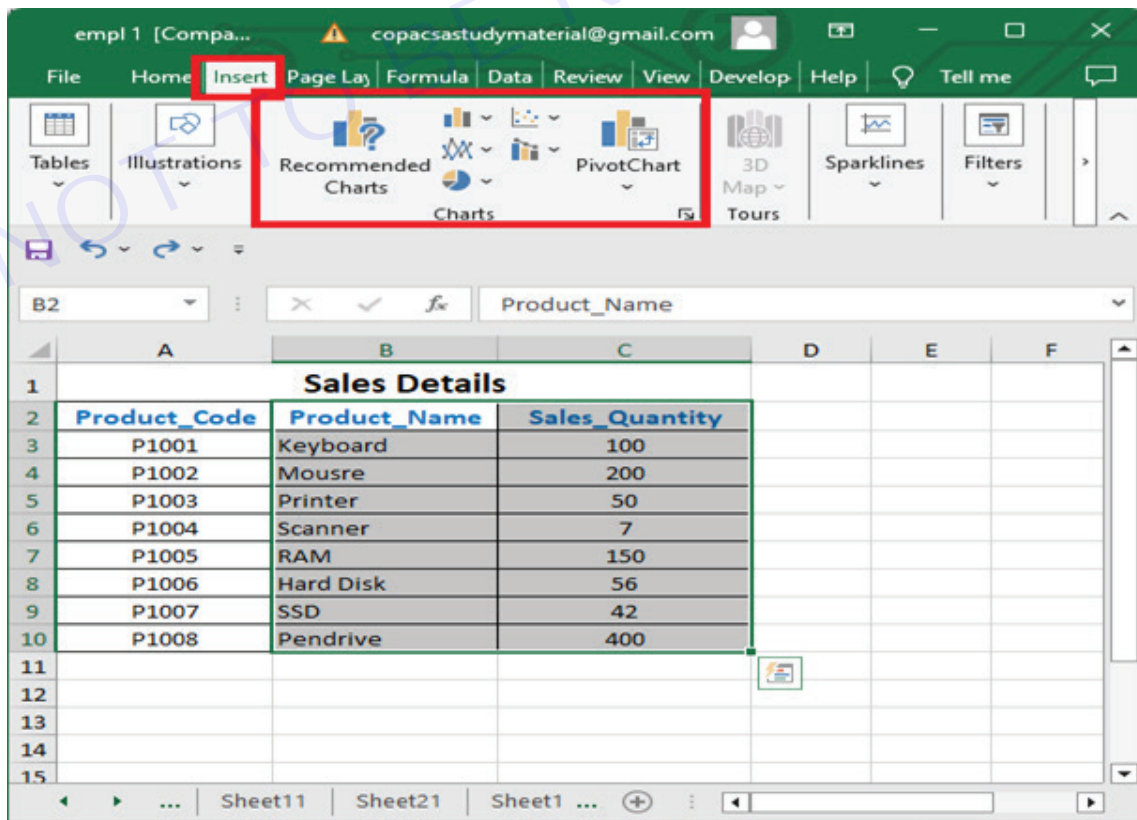
Procedure

TASK 1: Prepare 3D Chart based on the following table

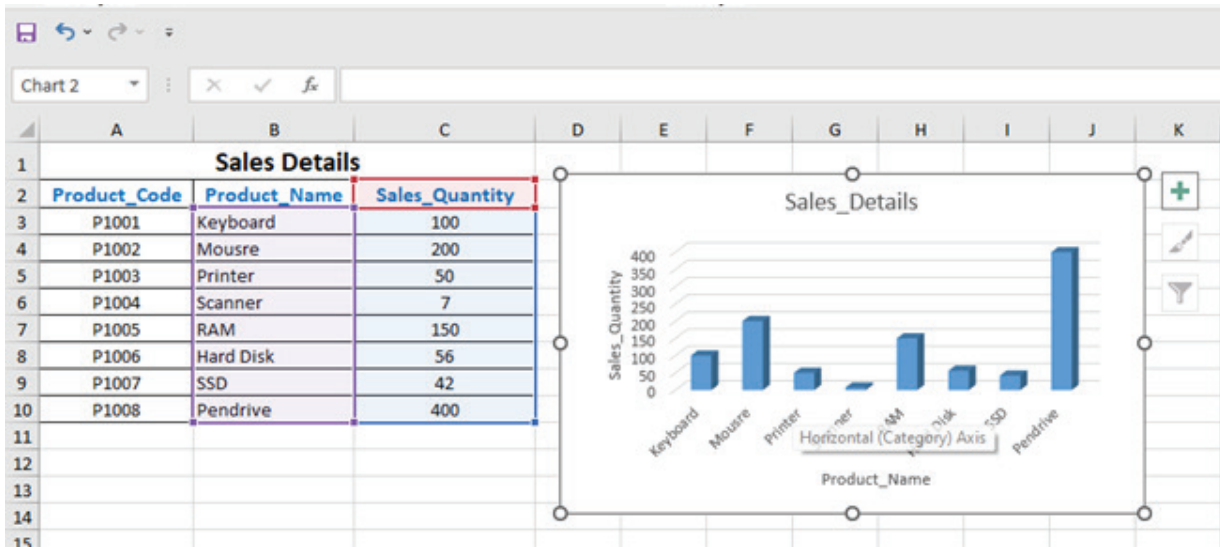
Step 1: First open the MS-EXCEL and then write your data there in the sheet.

Sales Details		
Product_Code	Product_Name	Sales_Quantity
P1001	Keyboard	100
P1002	Mousre	200
P1003	Printer	50
P1004	Scanner	7
P1005	RAM	150
P1006	Hard Disk	56
P1007	SSD	42
P1008	Pendrive	400

Step 2: Then select all your required data. Go to the insert option, from this you can insert any type of chart and graph according to your data.

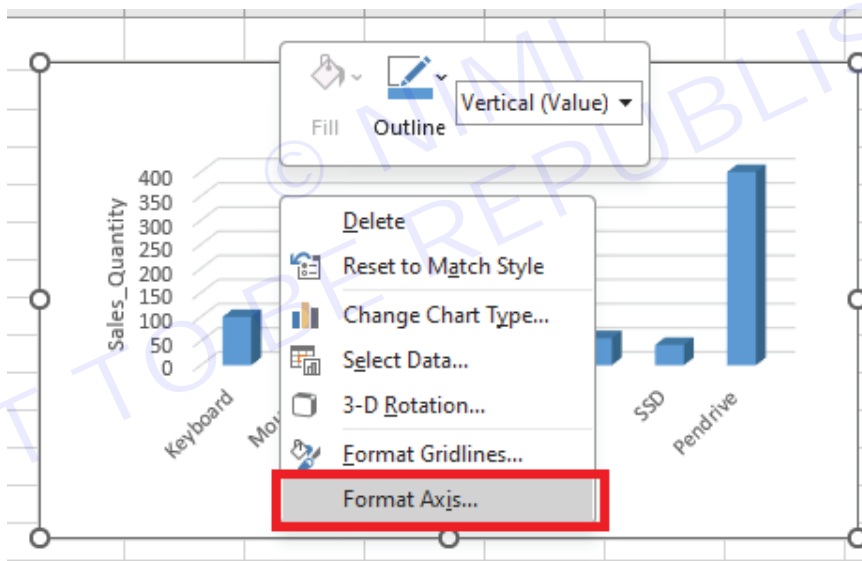


For example here selected a 3D column chart for our dataset. Excel will automatically create the following chart based on your data.

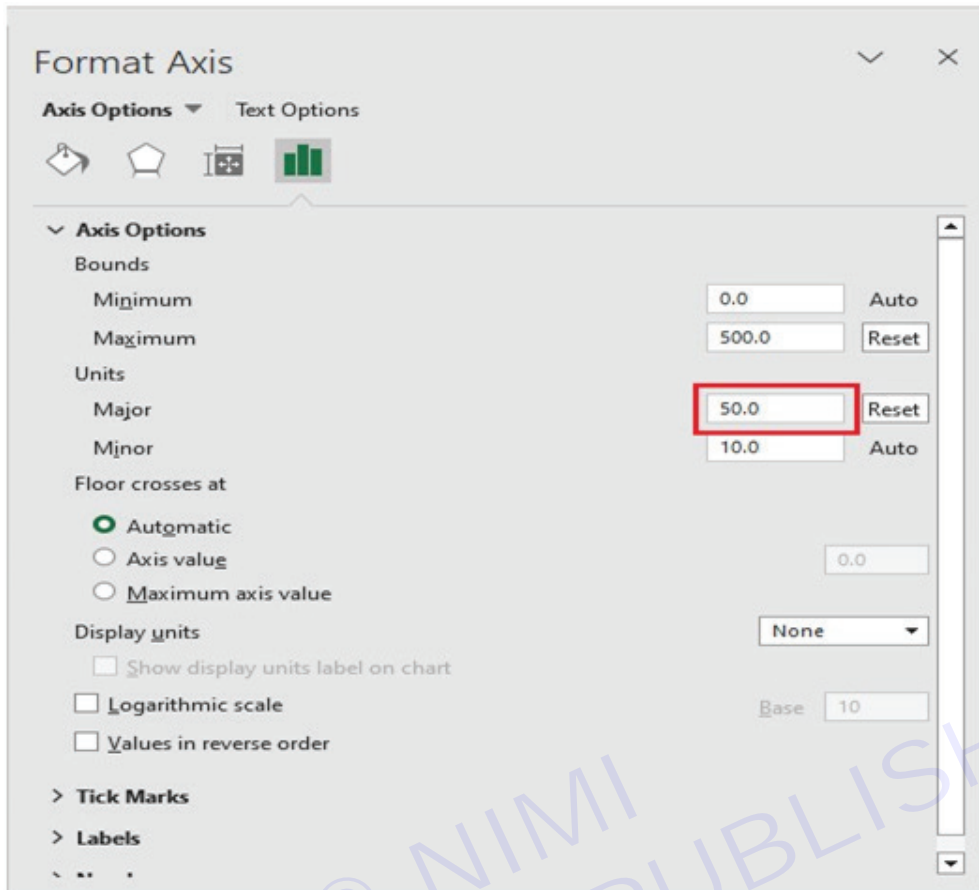


Step 3: Format Axis

In this step, we will format the chart axis. First, Select Axis, then Right-Click on it, and then select Format Axis.



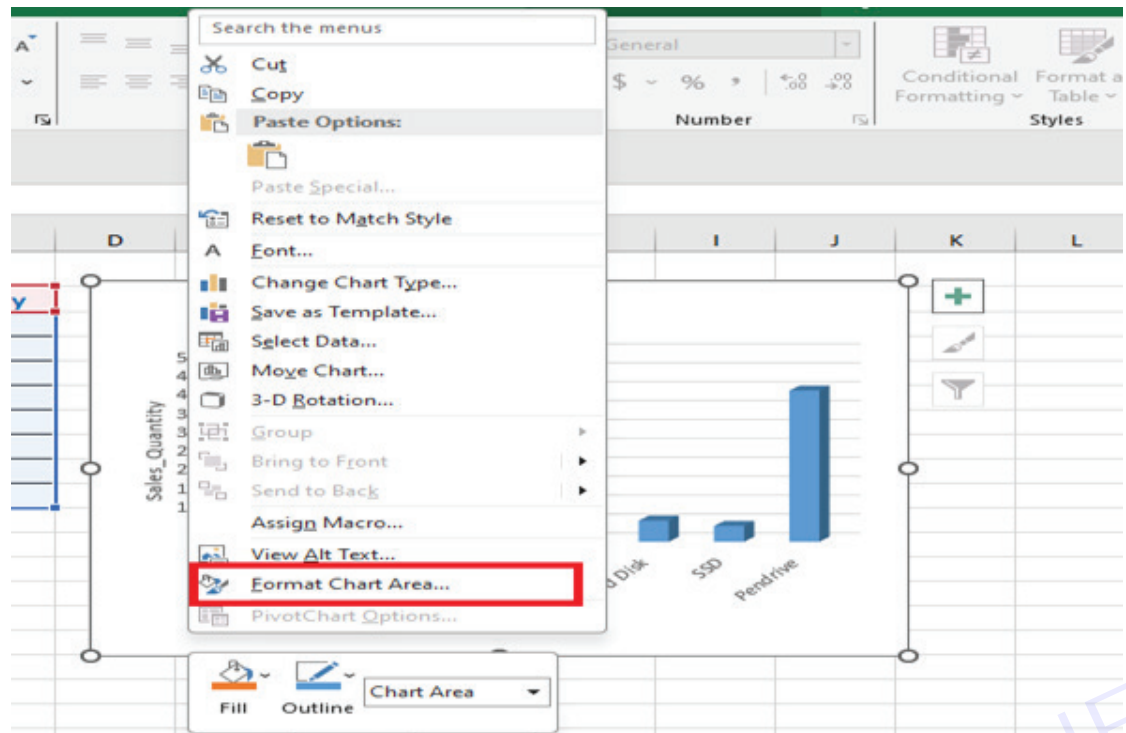
Once we click on the Format Axis option, Excel will automatically open a Format Axis Pane. Using the format axis pane, we can format our chart axis according to our requirements. Here, we are going to change units to display in **Major Unit with an increment of '50'**.



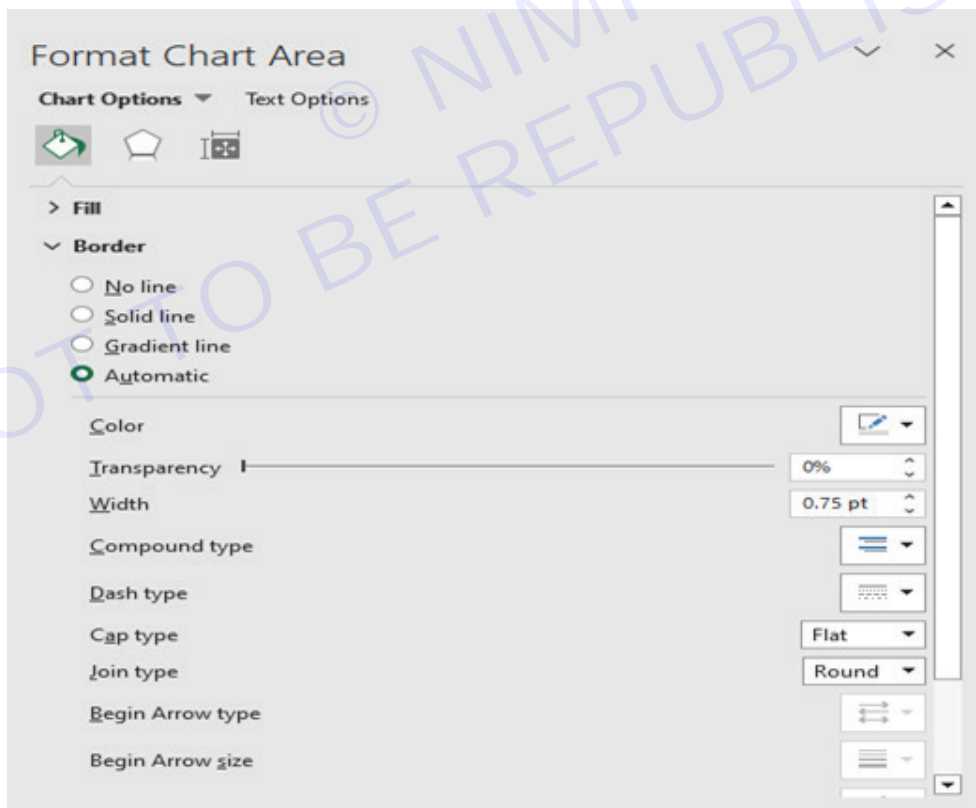
We will get the following output.



Step 4: Similarly you can format the chart area by Right Clicking anywhere inside the chart area and choose format chart area option as follows.



When selecting Format Chart Area option you will get a dialogue box as shown below.



From this dialogue box you can apply the required options in your chart.

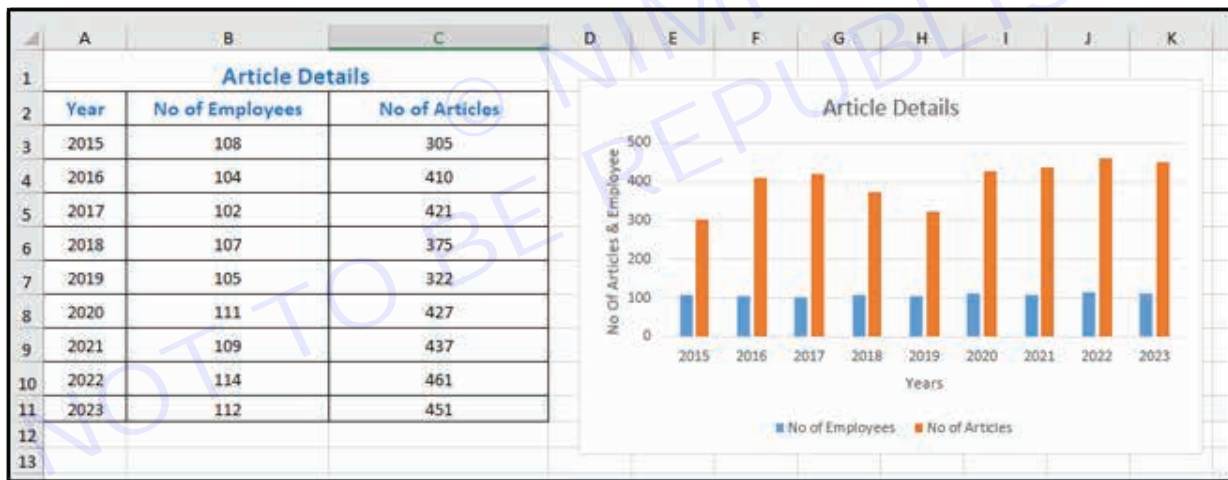
TASK 2: Create a Combination Chart

Combination of two or more chart types in a single chart is known as a combination chart. To create a combination chart, follow the steps

Step1: First open the MS-EXCEL and then write your data there in the sheet.

	A	B	C
1	Article Details		
2	Year	No of Employees	No of Articles
3	2015	108	305
4	2016	104	410
5	2017	102	421
6	2018	107	375
7	2019	105	322
8	2020	111	427
9	2021	109	437
10	2022	114	461
11	2023	112	451
12			

Step 2: Create a basic chart and here we are going to plot a column chart for our dataset.

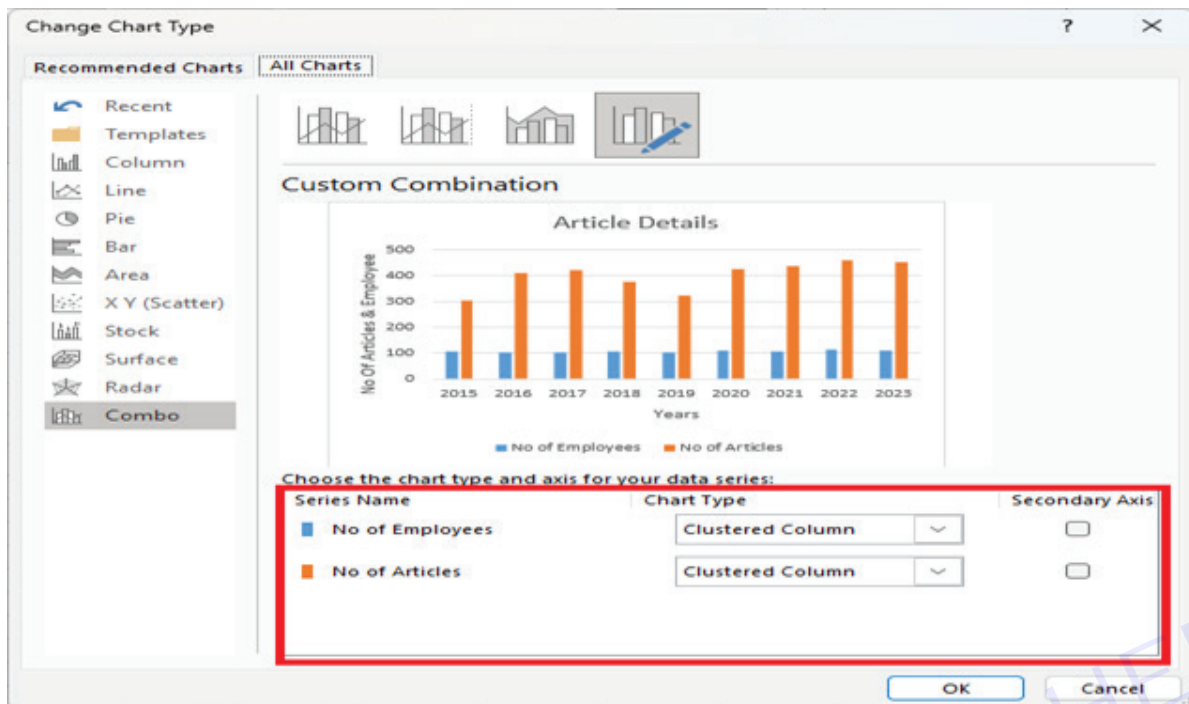


Step 3: We have created a basic chart and now it's time for complex graphs to play their role. Click either on the orange bars or blue bars.

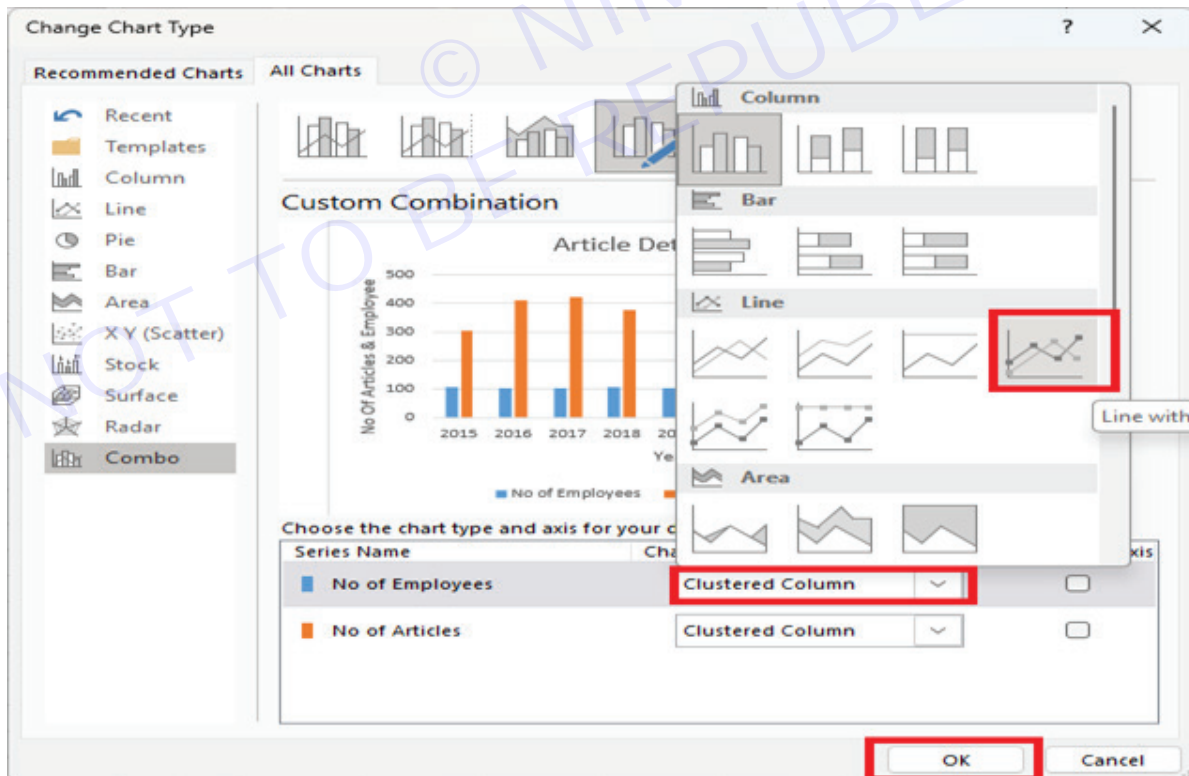
Note: Here, we are going to click on the blue bars which are representing the No of Employees. Go to the design tab, click on the change chart type.



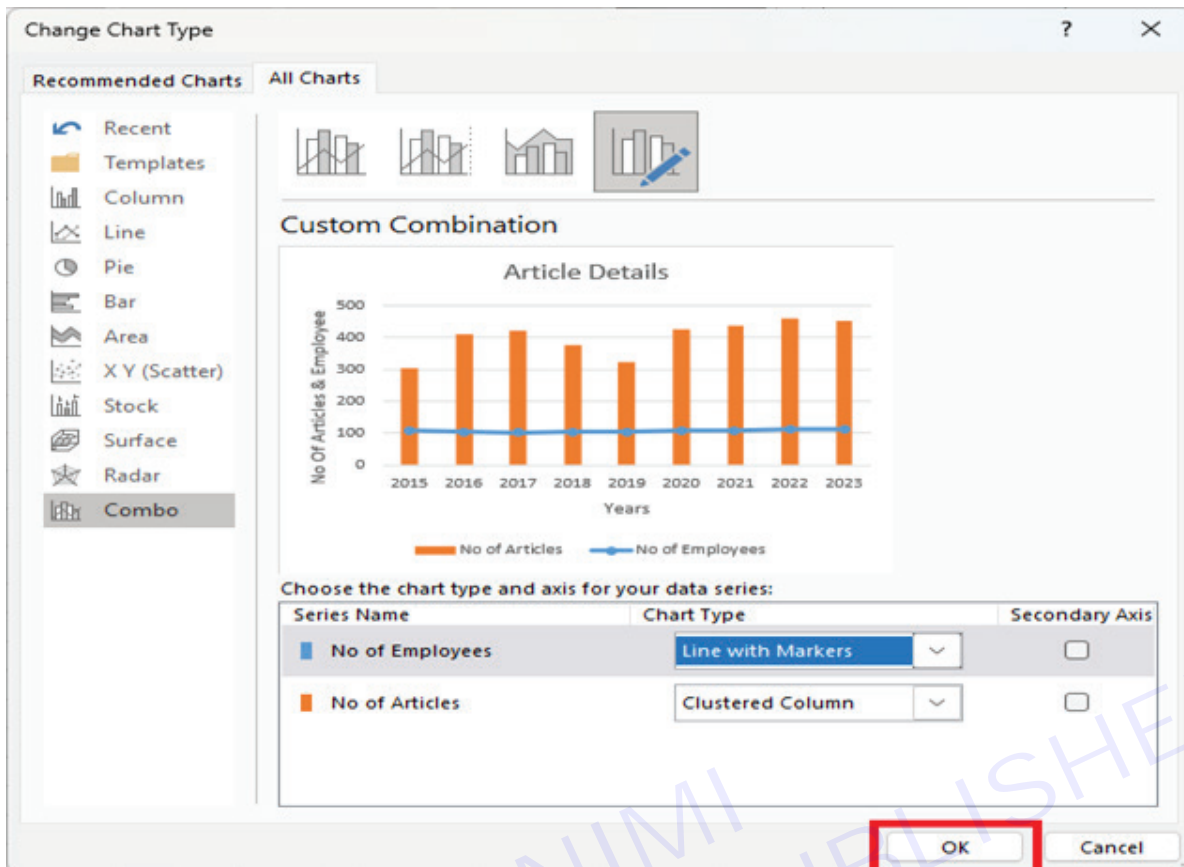
Step 4: A dialog window appears



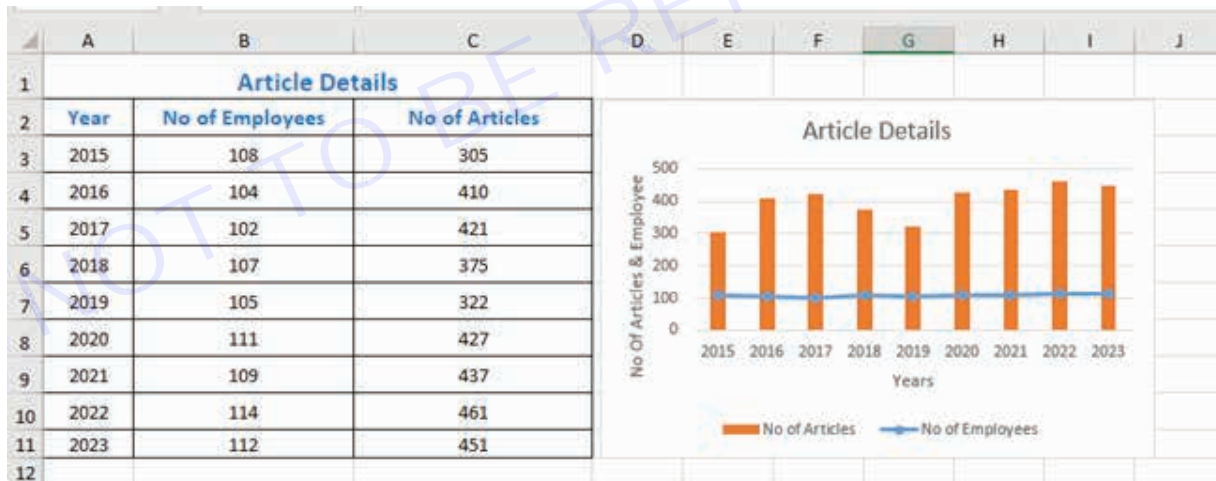
In that dialog window, navigate to combo from the left side of the panel and click on the employees chart type drop down menu.



Choose the first graph under the line and press ok.



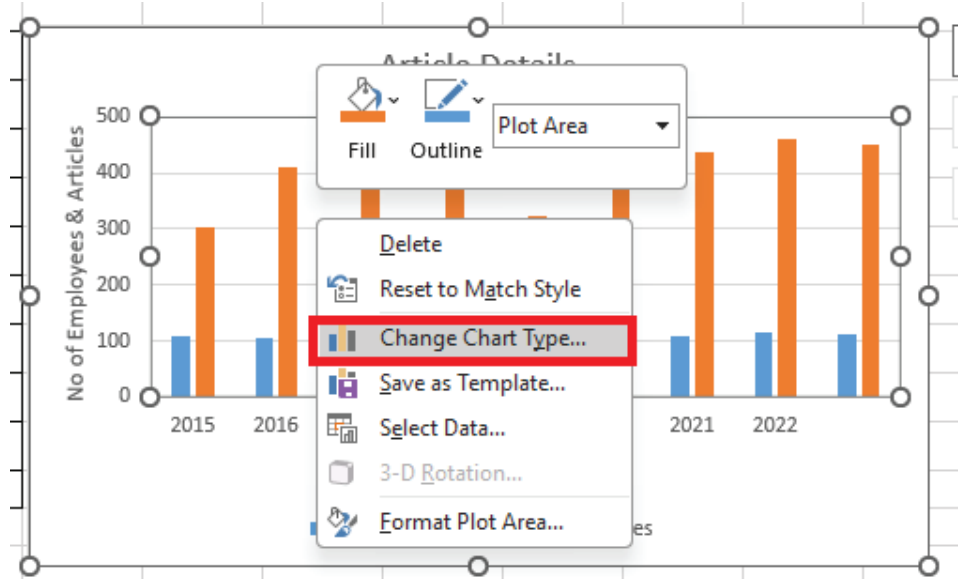
Once the steps are followed, the following output appears



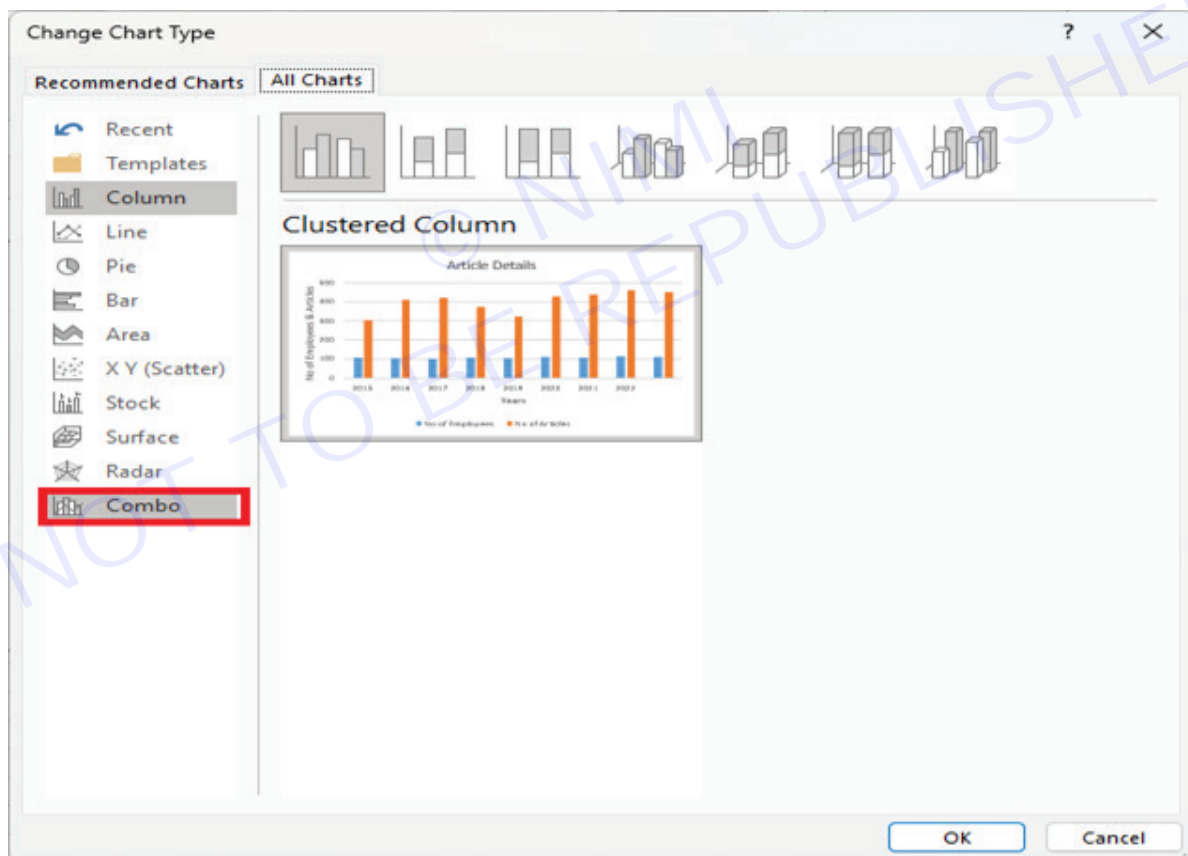
TASK 3: Dual-Axis Chart

The primary axis is the x-axis and the y-axis which is usually on the right side of the chart is known as the secondary axis. Now add a secondary axis to our Excel chart to make it more understandable and look presentable.

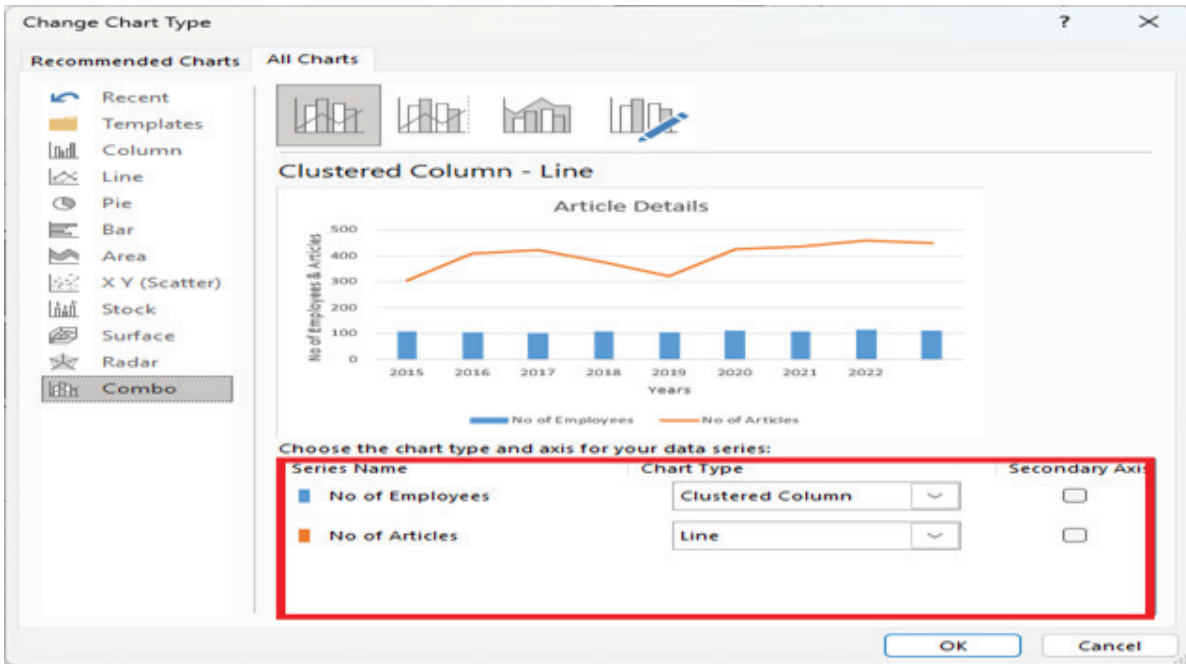
Step 1: Click on the chart. Go to the Design tab and click on the change chart type, a dialog window appears.



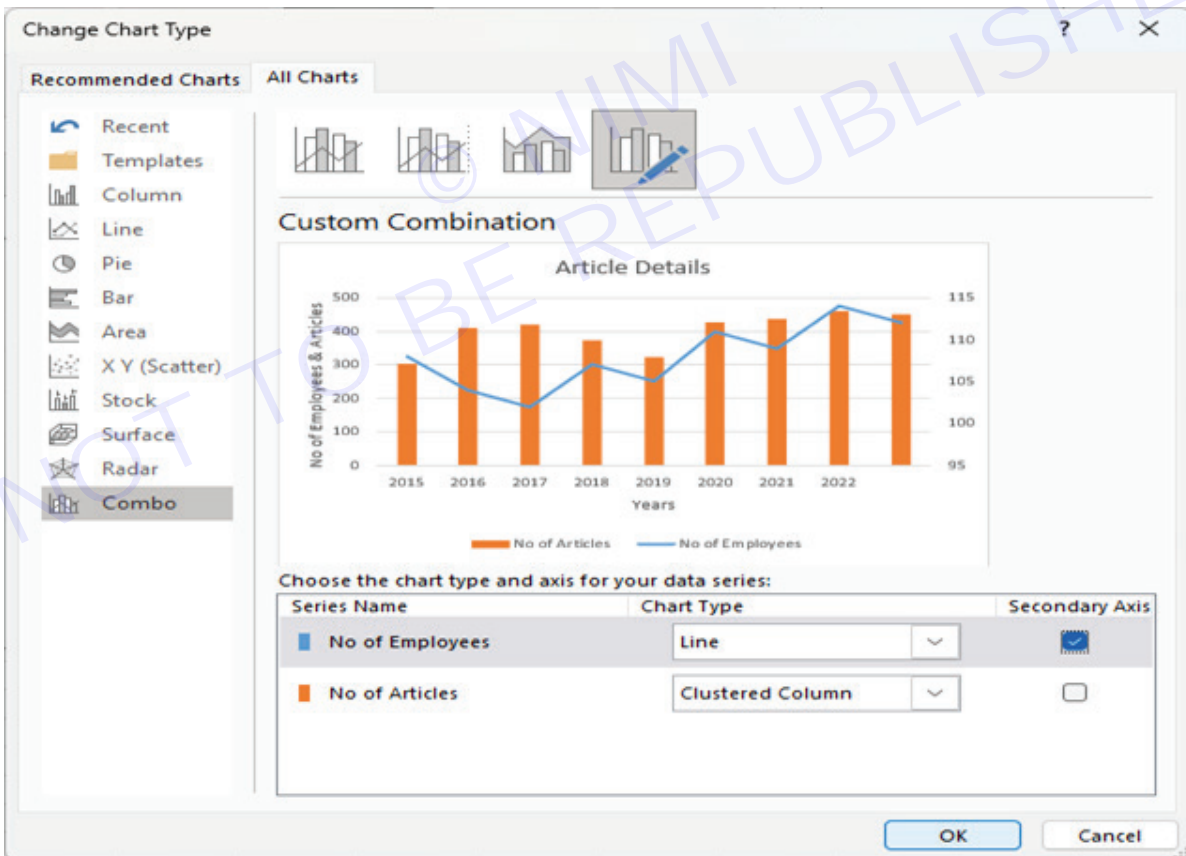
Click on the change chart type. Then the following dialogue box will appear



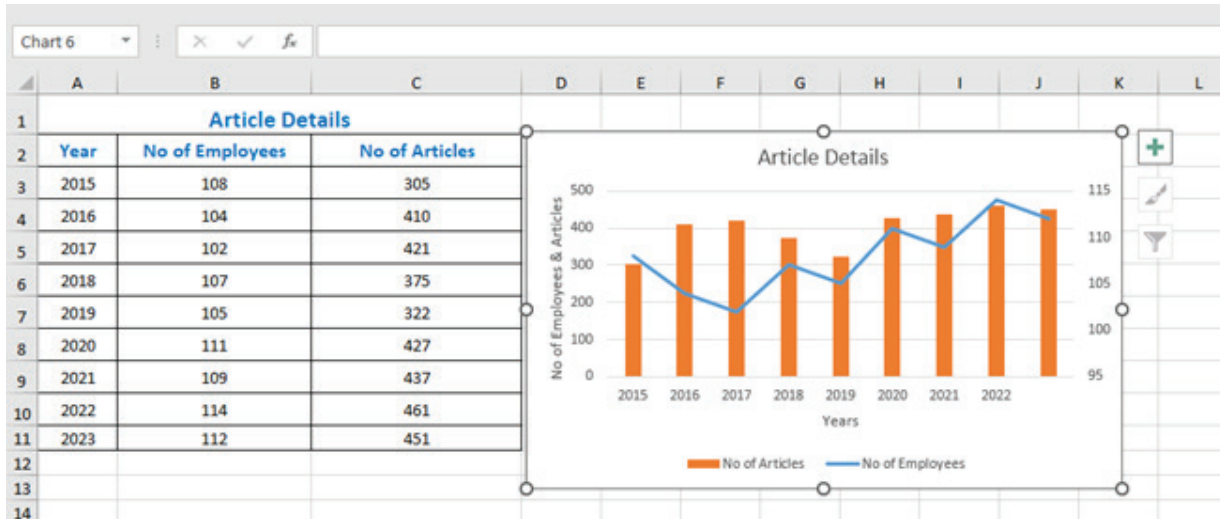
In that dialog window, navigate to combo from the left side of the panel and click on it. You will get the following screen



In the screen select the required chart type for No of Employees and No of Articles and check the Secondary Axis.



Finally, press ok to get the chart like this.



Note: To insert the upcoming charts, follow the steps

Step 1: Open a MS Excel worksheet and click on the “Insert” button from the menu bar.

Step 2: From the Insert tab, go to the “Charts” option, there you would find different types of charts. You can choose the desired chart from it.

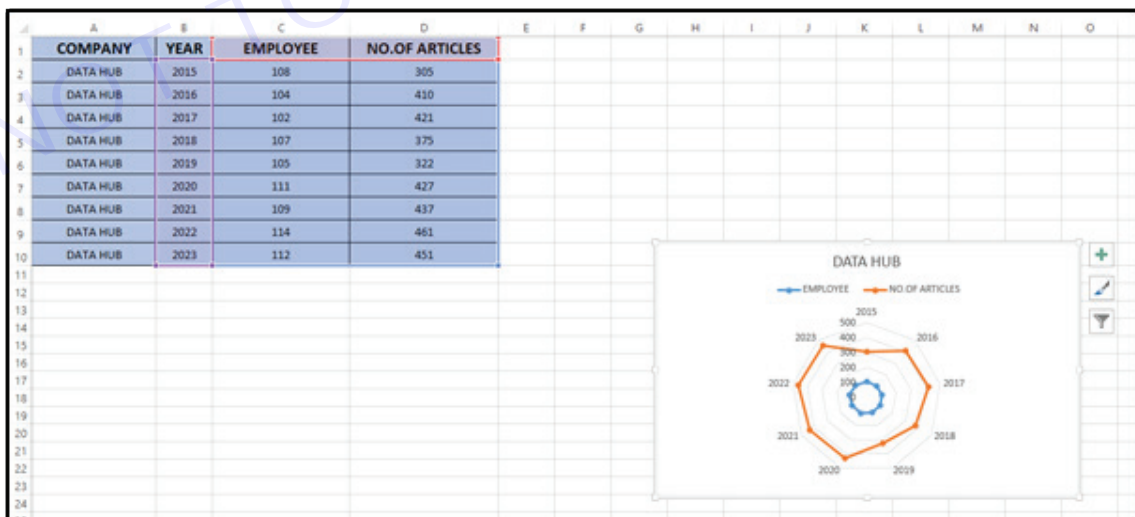
Note: Choose the chart as required. For example: Radar Chart, Filled Radar etc.

Step 3: Then we need to select the data for which the graph has to be plotted

Eg 1: Spider/Radar Chart

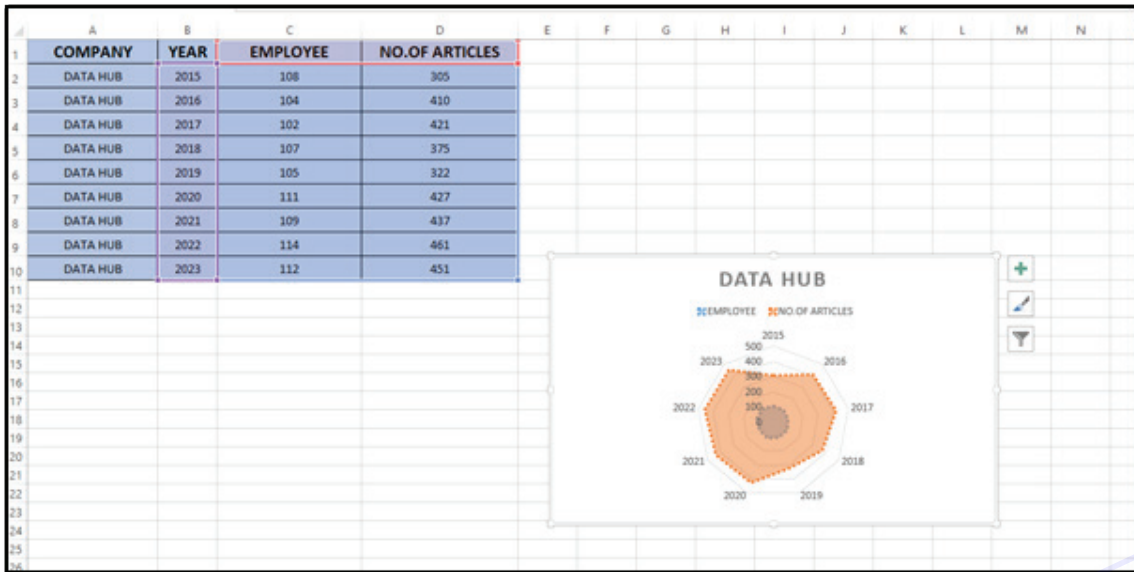
Displays multivariate data on a circular grid, with each axis representing a different variable. Spider charts are useful for comparing the performance of multiple entities across different metrics.

These charts represent the values relative to a centre point. Radar with Markers charts represents the markers for the individual points and Radar charts are represented without the markers for the individual points. The user uses the Radar and Radar with Marker charts when the categories are not directly comparable.



Eg 2: Filled Radar

In filled radar charts, the data series is filled with a colour. This chart represents the values relative to a centre point.



Related Exercise

Here's a simple table of a sales data containing ProductCode, product Name, QuantitySold and Rate:

Product Code	Product Name	Quantity Sold	Rate
ABC123	Widget A	100	500
XYZ456	Widget B	75	400
DEF789	Widget C	150	250
GHI012	Widget D	200	800
AHF213	Widget E	560	300
RFE033	Widget F	70	350
GF456	Widget G	220	340

Using the above sales data table containing Product codes, Product names, Quantities sold and Rate, create an advanced chart in Excel that visually compares the sales performance of the top 5 best-selling products. Ensure that the chart includes labels, appropriate formatting, and any necessary visual enhancements to effectively communicate the sales insights.

EXERCISE 75 : Demonstrate pivot tables

Objectives

At the end of this exercise you shall be able to

- make pivot tables for summarizing and analyzing data sheets.

Procedure

Demonstrate PivotTables

A PivotTable in Microsoft Excel is a powerful tool used for summarizing, analyzing, exploring, and presenting large amounts of data from various sources. It allows users to rearrange and summarize selected columns and rows of data into a more useful format without altering the original data set. Here's a step-by-step guide on how to create a PivotTable in Excel:

- 1 Prepare your data:** Ensure that your data is organized in rows and columns with a clear header row. There should be no blank rows or columns within the data set. Each column should have a header that describes the data it contains.
- 2 Select your data:** Click anywhere within the range of data you want to analyze.
- 3 Insert a PivotTable:**
 - Go to the "Insert" tab on the Excel ribbon.
 - Click on the "PivotTable" button. This will open the "Create PivotTable" dialog box.
- 4 Choose your data range:**
 - In the "Create PivotTable" dialog box, Excel will automatically detect the range of your data. Ensure that this range is correct.
 - You can also manually specify the data range if Excel doesn't detect it automatically.
- 5 Choose where to place your PivotTable:**
 - Decide whether you want the PivotTable to be placed in a new worksheet or an existing worksheet.
 - Select the location where you want your PivotTable to be placed and click "OK".
- 6 Design your PivotTable:**
 - Once the PivotTable is inserted, you'll see the PivotTable Field List pane on the right.
 - Drag and drop the fields from your data into the "Rows", "Columns", "Values", or "Filters" area, depending on how you want to summarize and analyze your data.
 - You can also apply functions (e.g., sum, count, average) to the values in the PivotTable by clicking on the drop-down arrow next to the field in the "Values" area and selecting "Value Field Settings".
- 7 Customize your PivotTable:**
 - You can further customize your PivotTable by formatting the cells, changing the layout, applying filters, sorting data, and more.
 - Experiment with different arrangements and configurations to see what best suits your analysis needs.
- 8 Refresh your PivotTable (if needed):**
 - If your source data changes, you can refresh your PivotTable to reflect those changes. Right-click anywhere in the PivotTable and select "Refresh".
- 9 Save your workbook:**
 - Once you're done creating and customizing your PivotTable, make sure to save your Excel workbook to preserve your work.

By following these steps, you can create and customize PivotTables in Excel to analyze your data more effectively.

TASK 1: Let's say we wanted to answer the question: **What is the amount sold by each salesperson?** for the sales data in the example below. Answering this question could be time consuming and difficult- each salesperson appears on multiple rows, and we would need to total all of their different orders individually. We could use the **Subtotal** command to help find the total for each salesperson, but we would still have a lot of data to work with.

Fortunately, a **PivotTable** can instantly **calculate** and **summarize** the data in a way that's both easy to read and manipulate. When we're done, the PivotTable will look something like this:

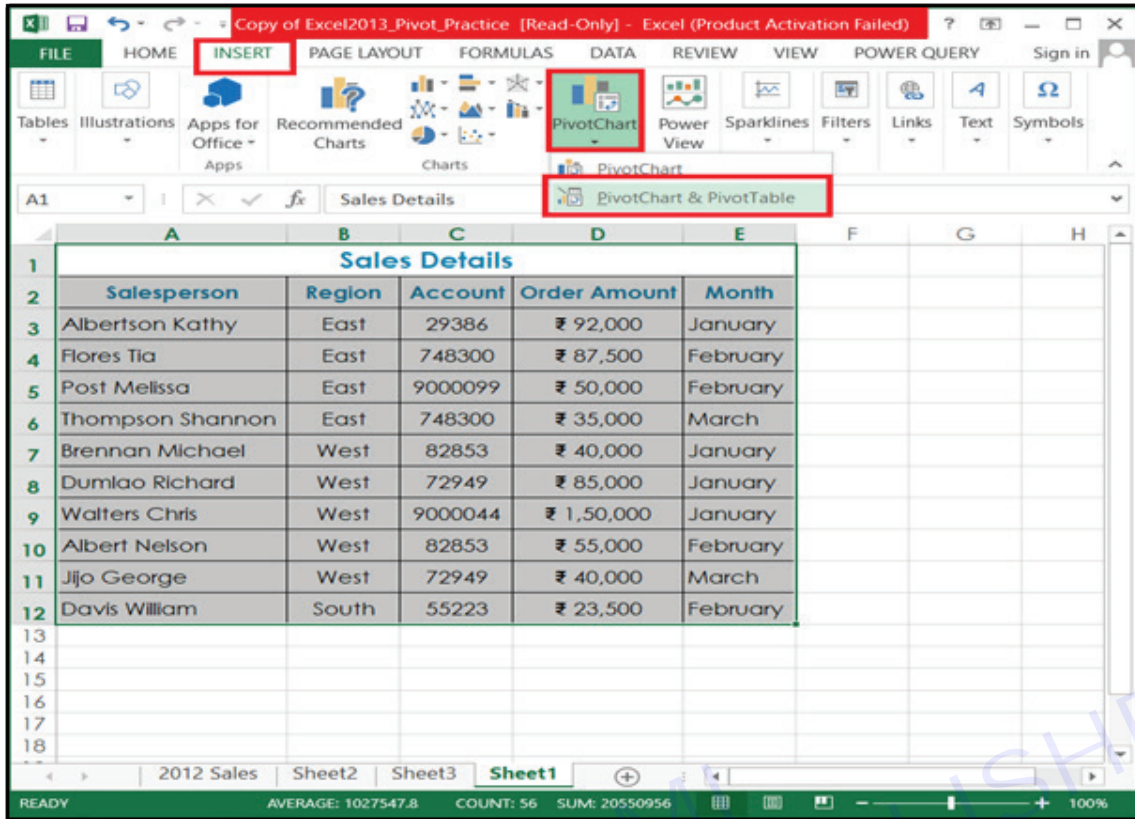
Once you've created a PivotTable, you can use it to answer different questions by rearranging-or **pivoting**-the data. For example, if we wanted to answer the question: **What is the total amount sold in each month?** we could modify our PivotTable to look like this:

To create a PivotTable:

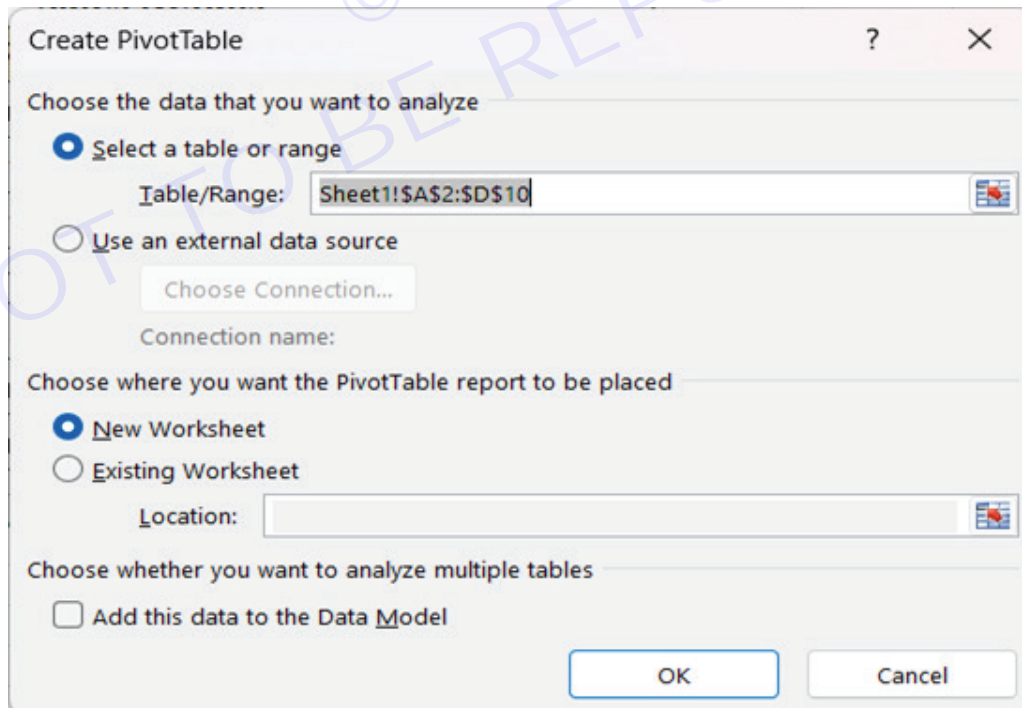
- 1 Select the **table** or **cells** (including column headers) containing the data you want to use.

Sales Details				
Salesperson	Region	Account	Order Amount	Month
Albertson Kathy	East	29386	₹ 92,000	January
Flores Tia	East	748300	₹ 87,500	February
Post Melissa	East	9000099	₹ 50,000	February
Thompson Shannon	East	748300	₹ 35,000	March
Brennan Michael	West	82853	₹ 40,000	January
Dumlao Richard	West	72949	₹ 85,000	January
Walters Chris	West	9000044	₹ 1,50,000	January
Albert Nelson	West	82853	₹ 55,000	February
Jllo George	West	72949	₹ 40,000	March
Davis William	South	55223	₹ 23,500	February

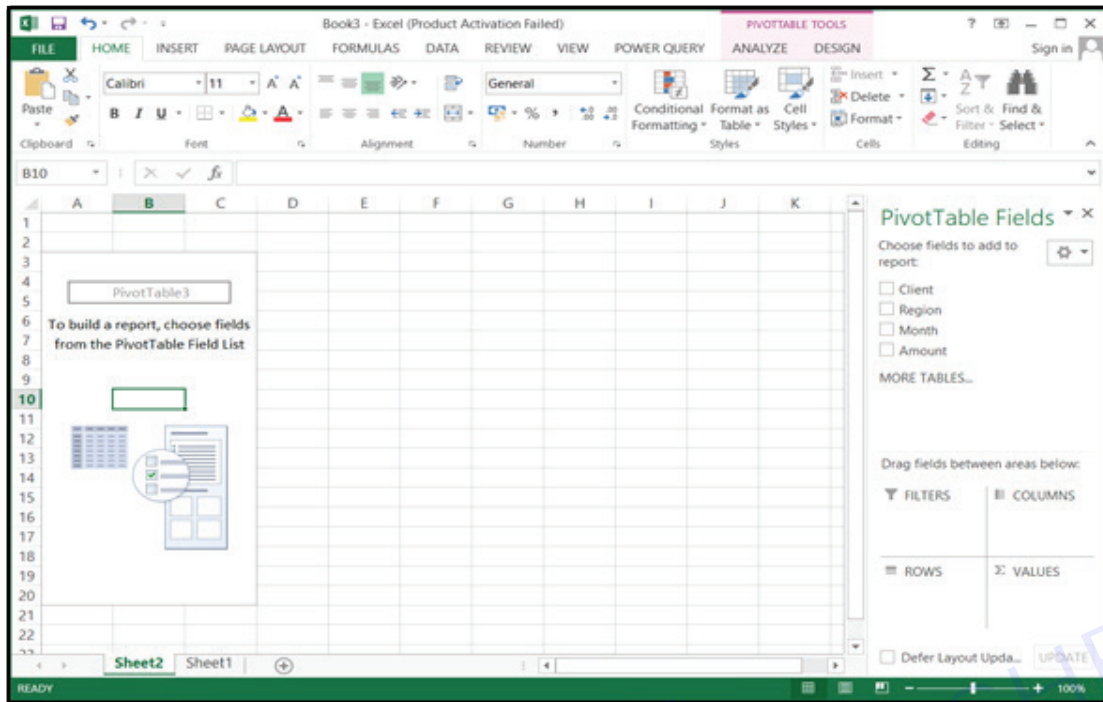
- 2 From the **Insert** tab, click the **PivotTable** command.



- 3 The **Create PivotTable** dialog box will appear. Choose your settings, then click **OK**. In our example, we'll use **Table1** as our source data and place the PivotTable on a **new worksheet**.



4 A blank **PivotTable** and **Field List** will appear on a new worksheet.



5 Once you create a PivotTable, you'll need to decide which **fields** to add. Each field is simply a **column header** from the source data. In the **PivotTable Field List**, check the box for each field you want to add. In our example, we want to know the total **amount** sold by each **salesperson**, so we'll check the **Salesperson** and **Order Amount** fields.



6 The selected fields will be added to one of the four areas below the Field List. In our example, the **Salesperson** field has been added to the **Rows** area, while the **Order Amount** has been added to the **Values** area. Alternatively, you can click, hold, and drag a field to the desired area.



- 7 The PivotTable will calculate and summarize the selected fields. In our example, the PivotTable shows the amount sold by each salesperson.

Row Labels	Sum of Order Amount
Albert Nelson	55000
Albertson Kathy	92000
Brennan Michael	40000
Davis William	23500
Dumlao Richard	85000
Flores Tia	87500
Jijo George	40000
Post Melissa	50000
Thompson Shannon	35000
Walters Chris	15000
Grand Total	658000

Just like with normal spreadsheet data, you can sort the data in a PivotTable using the Sort & Filter command in the Home tab. You can also apply any type of number formatting you want. For example, you may want to change the Number Format to Currency. However, be aware that some types of formatting may disappear when you modify the PivotTable.

	A	B	C
1			
2			
3	Row Labels	Sum of Order Amount	
4	Albert Nelson	55000	
5	Albertson Kathy	92000	
6	Brennan Michael	40000	
7	Davis William	23500	
8	Dumlao Richard	85000	
9	Flores Tia	Dumlao Richard (Salesperson)	7500
10	Jijo George	Row: Dumlao Richard	0000
11	Post Melissa	50000	
12	Thompson Shannon	35000	
13	Walters Chris	150000	
14	Grand Total	658000	
15			

If you change any of the data in your source worksheet, the PivotTable will not update automatically. To manually update it, select the PivotTable and then go to Analyze > Refresh.

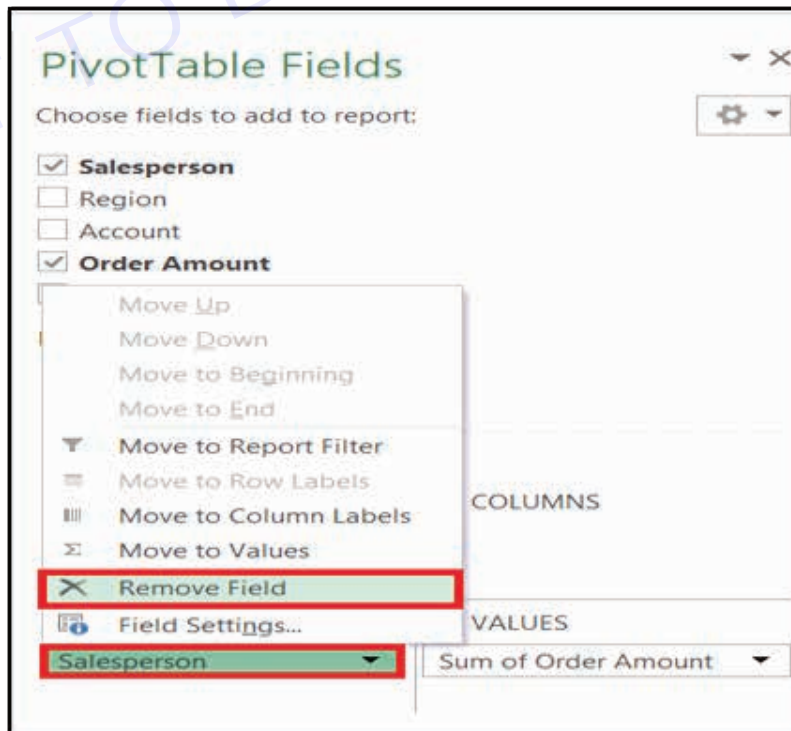
Pivoting data

One of the best things about PivotTables is that they can quickly pivot—or reorganize—data, allowing you to look at your worksheet data in different ways. Pivoting data can help you answer different questions and even experiment with the data to discover new trends and patterns.

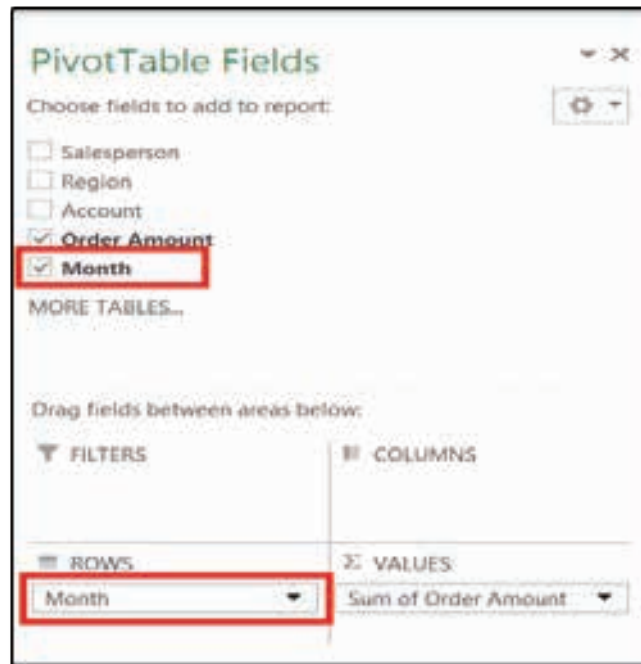
In our example, we used the PivotTable to answer the question: What is the total amount sold by each salesperson? But now we'd like to answer a new question: What is the total amount sold in each month? We can do this by simply changing the field in the Rows area.

To change the row:

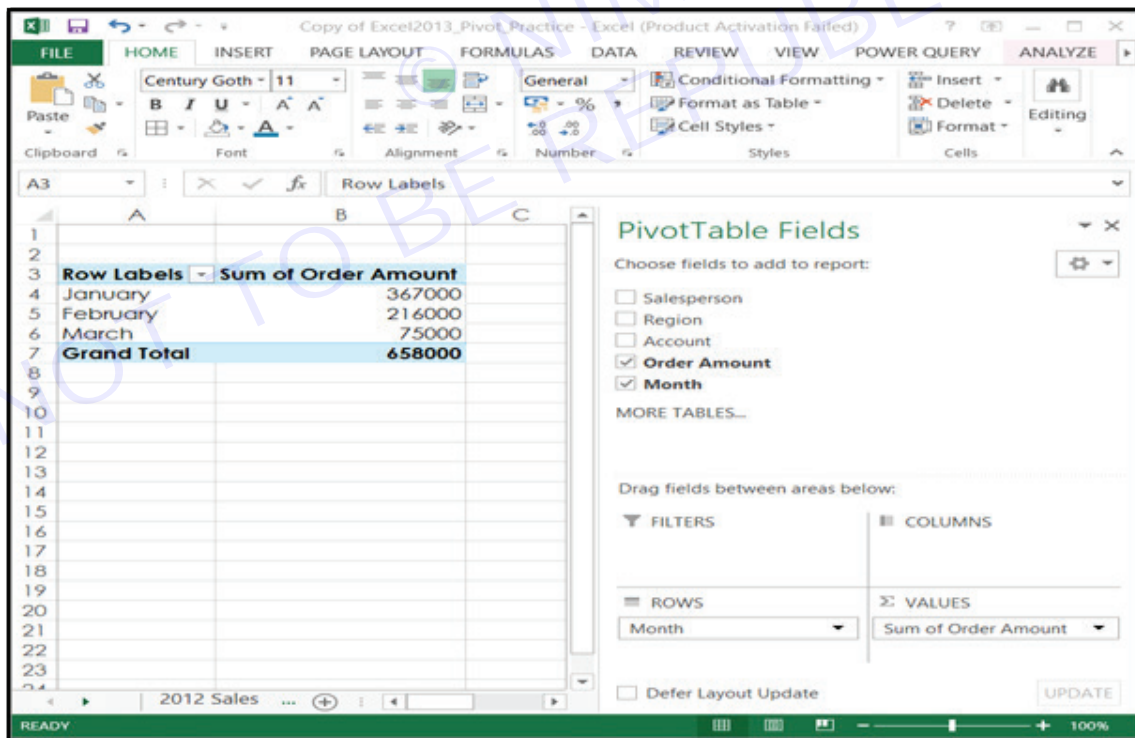
- 1 Click, hold, and drag any existing fields out of the Rows area. The field will disappear.



2 Drag a new field from the Field List into the Rows area. In our example, we'll use the Month field.



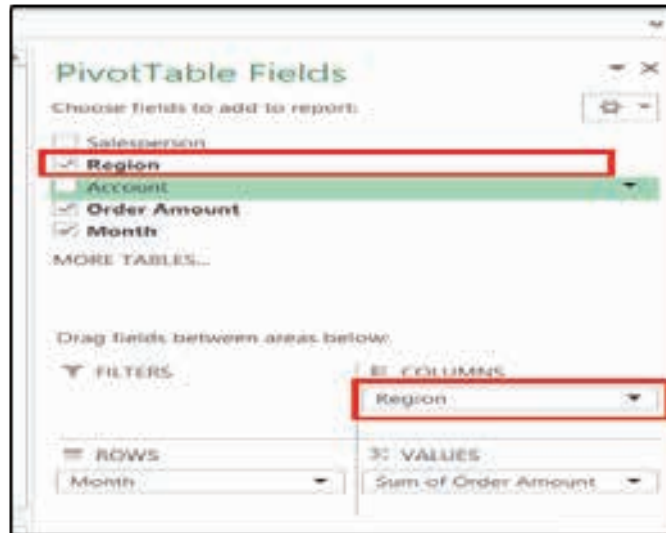
3 The PivotTable will adjust-or pivot-to show the new data. In our example, it now shows the total order amount for each month.



To add columns:

So far, our PivotTable has only shown one column of data at a time. In order to show multiple columns, you'll need to add a field to the Columns area.

1 Drag a field from the Field List into the Columns area. In our example, we'll use the Region field.



2 The PivotTable will include multiple columns. In our example, there is now a column for each region.

Sum of Order Amount	Column Labels			
Row Labels	East	South	West	Grand Total
January	92000	275000		367000
February	137500	23500	55000	216000
March	35000		40000	75000
Grand Total	264500	23500	370000	658000

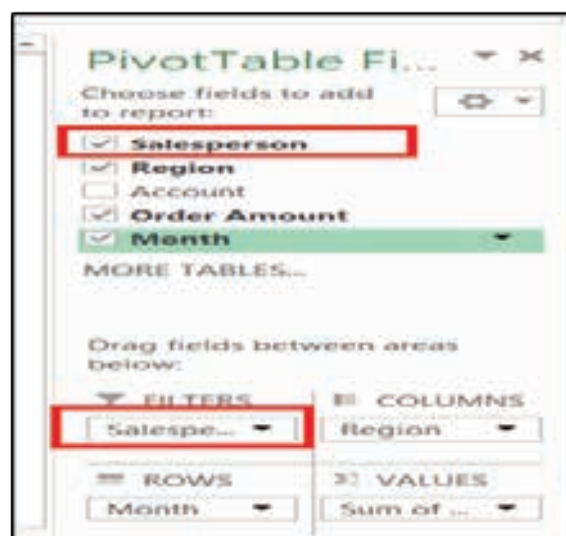
Filters

Sometimes you may want focus on just a certain section of your data. Filters can be used to narrow down the data in your PivotTable, allowing you to view only the information you need.

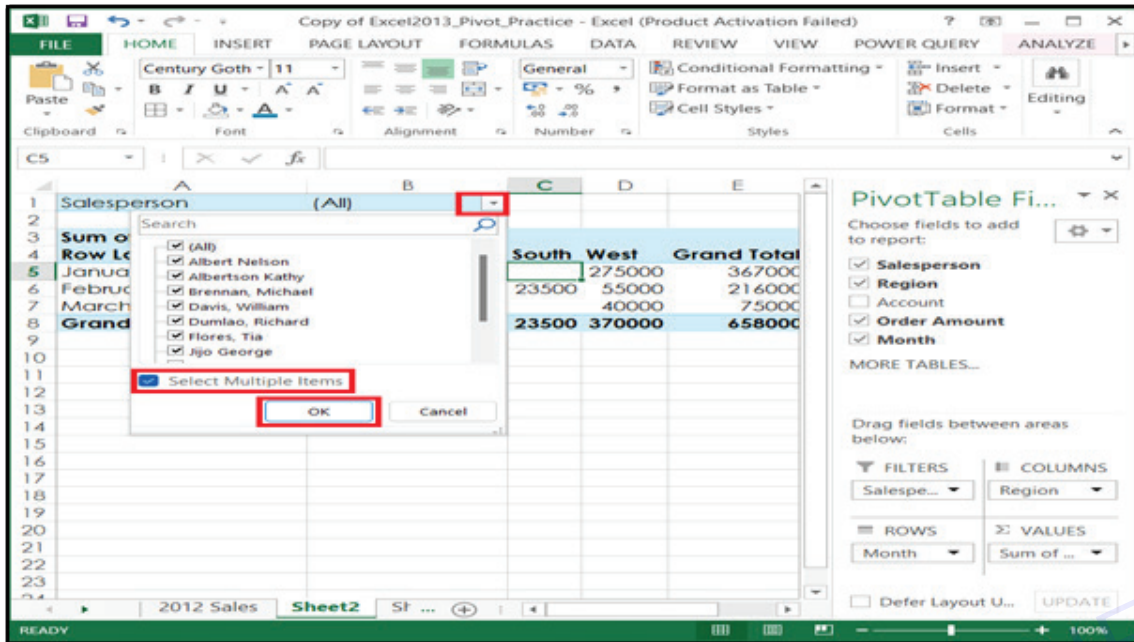
To add a filter:

In our example, we'll filter out certain salespeople to determine how they affect the total sales.

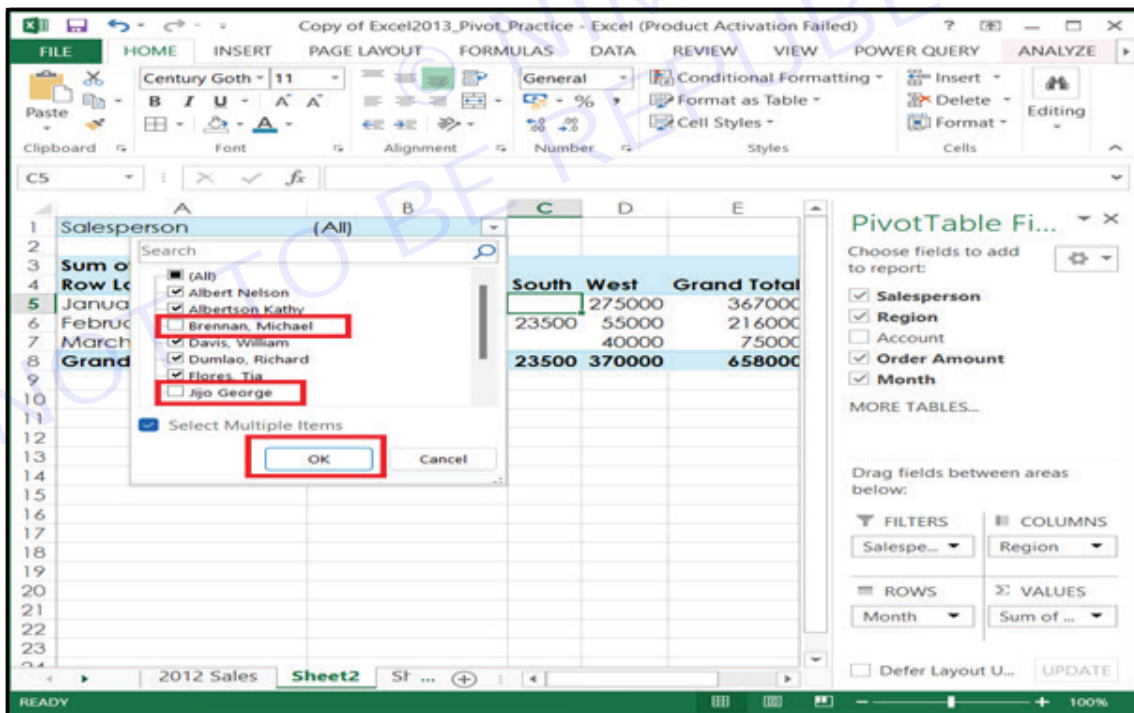
1 Drag a field from the Field List to the Filters area. In this example, we'll use the Salesperson field.



- 2 The filter will appear above the PivotTable. Click the drop-down arrow, then check the box next to Select Multiple Items.



- 3 Uncheck the box for any items you don't want to include in the PivotTable. In our example, we'll uncheck the boxes for a few different salespeople, then click OK.



- 4 The PivotTable will adjust to reflect the changes.

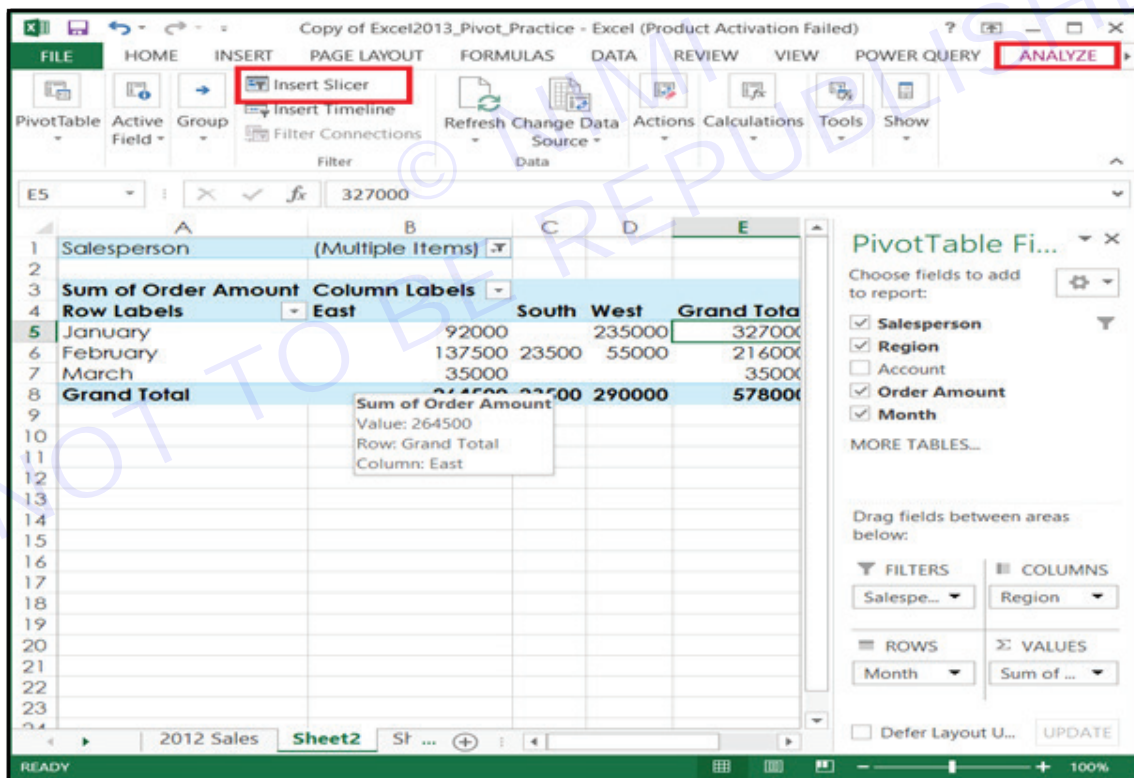
	A	B	C	D	E	F
1	Salesperson	(Multiple Items)				
2						
3	Sum of Order Amount	Column Labels				
4	Row Labels	East	South	West	Grand Total	
5	January	92000		235000	327000	
6	February	137500	23500	55000	216000	
7	March	35000			35000	
8	Grand Total	264500	23500	290000	578000	
9						

Slicers

Slicers make filtering data in PivotTables even easier. Slicers are basically just filters, but they're easier and faster to use, allowing you to instantly pivot your data. If you frequently filter your PivotTables, you may want to consider using slicers instead of filters.

To add a slicer:

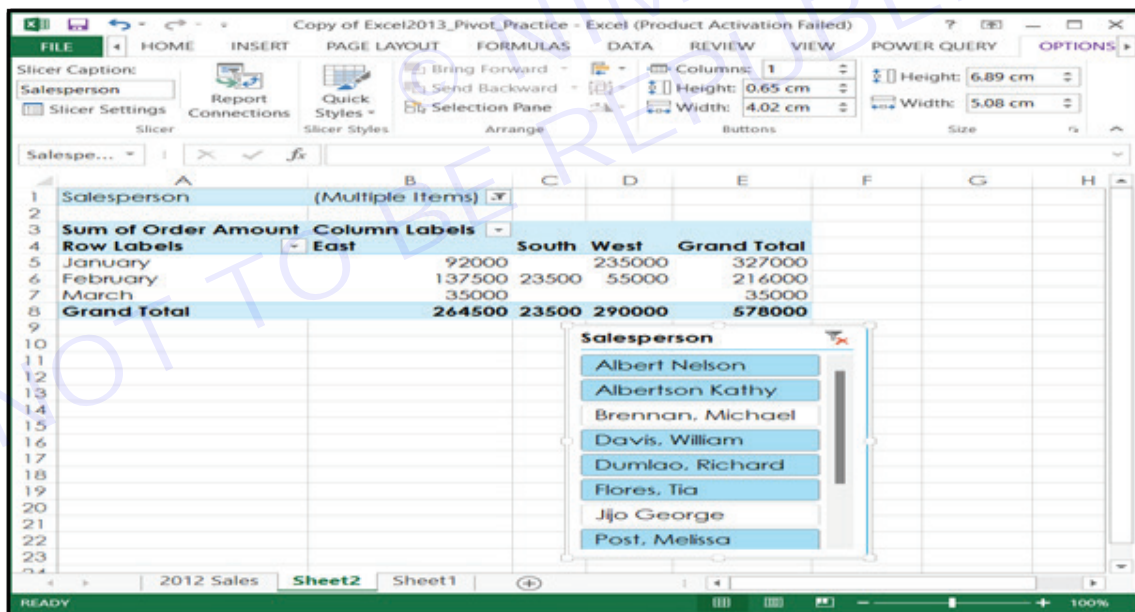
- 1 Select any cell in the PivotTable.
- 2 From the Analyze tab, click the Insert Slicer command.



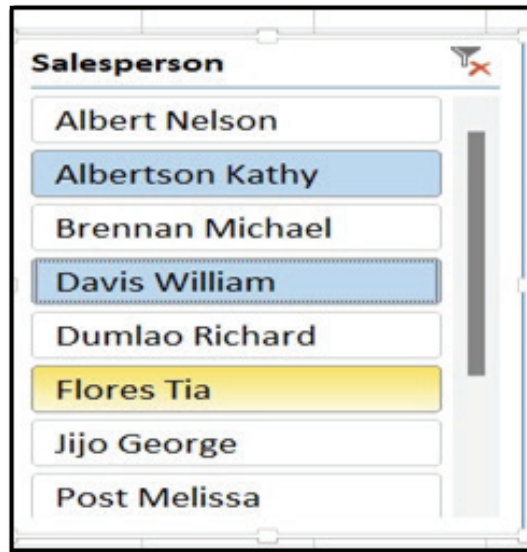
- 3 A dialog box will appear. Select the desired field. In our example, we'll select Salesperson, then click OK.



- 4 The slicer will appear next to the PivotTable. Each selected item will be highlighted in blue. In the example below, the slicer contains a list of all salespeople, and six of them are currently selected.

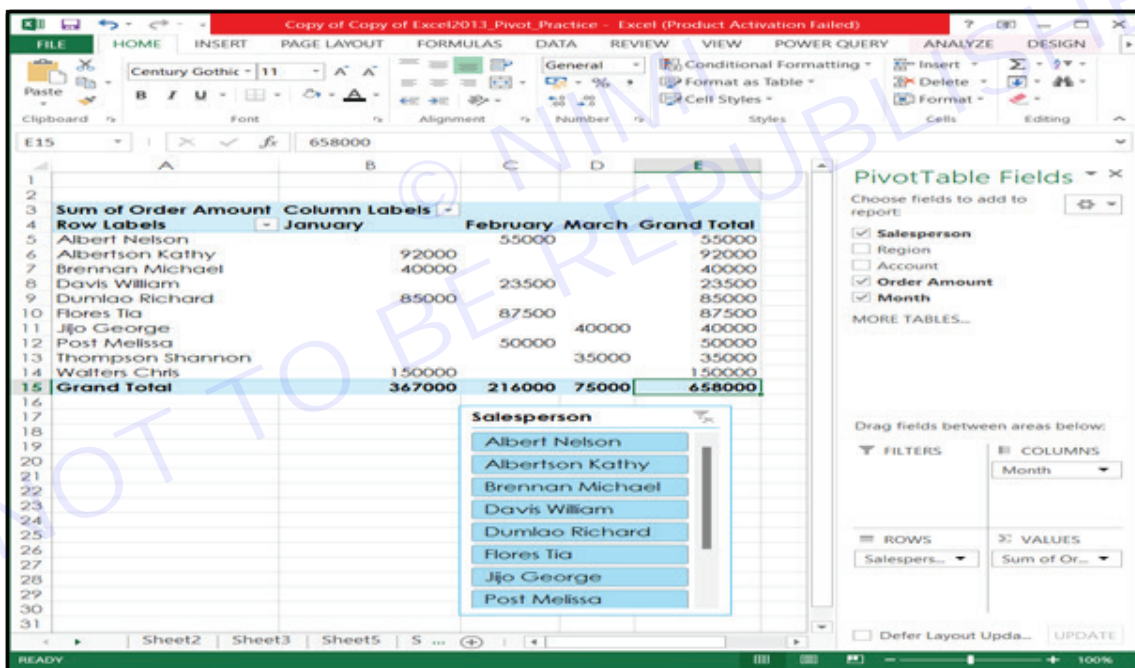


- 5 Just like filters, only selected items are used in the PivotTable. When you select or deselect items, the PivotTable will instantly reflect the changes. Try selecting different items to see how they affect the PivotTable. Press and hold the Ctrl key on your keyboard to select multiple items from a slicer.



You can also click the Filter icon in the top-right corner to select all items from the slicer at once.

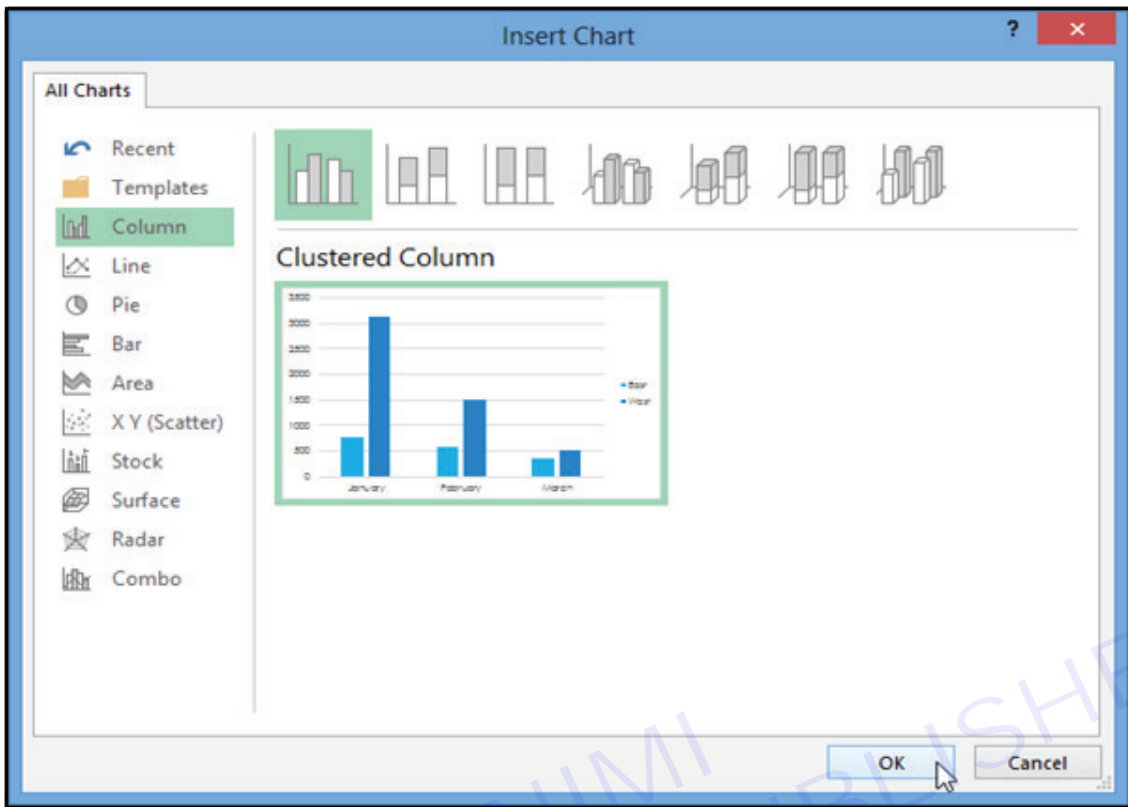
TASK 2: Create Pivot Chart from above table



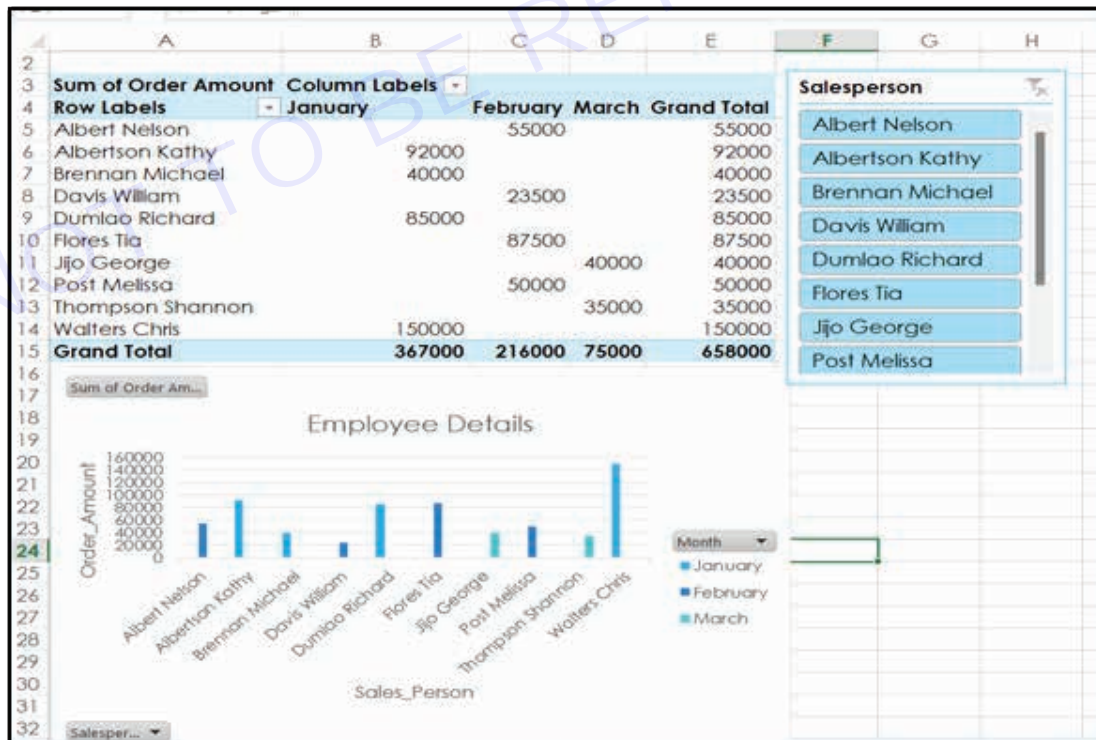
- 1 Select any cell in your PivotTable.



2 From the Insert tab, click the PivotChart command.

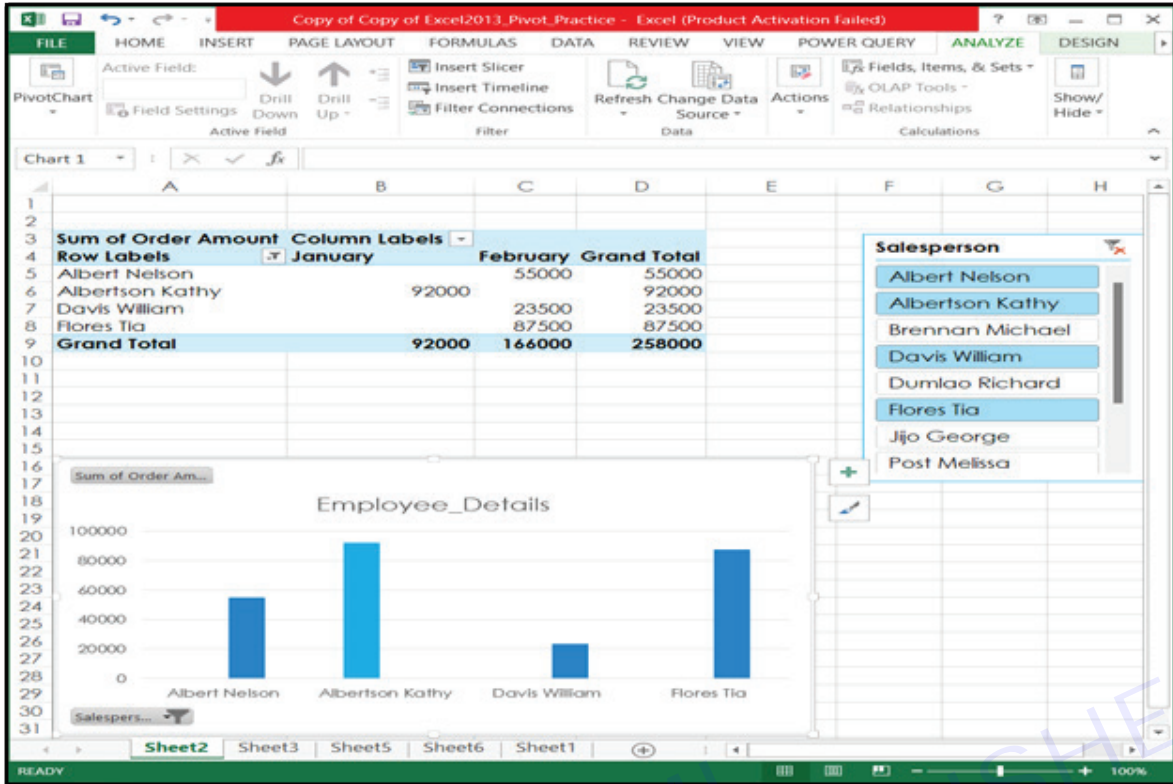


3 The Insert Chart dialog box will appear. Select the desired chart type and layout, then click OK.



4 The PivotChart will appear.

Try using slicers or filters to change the data that is displayed. The PivotChart will automatically adjust to show the new data.



TASK 3: By using the following product details data

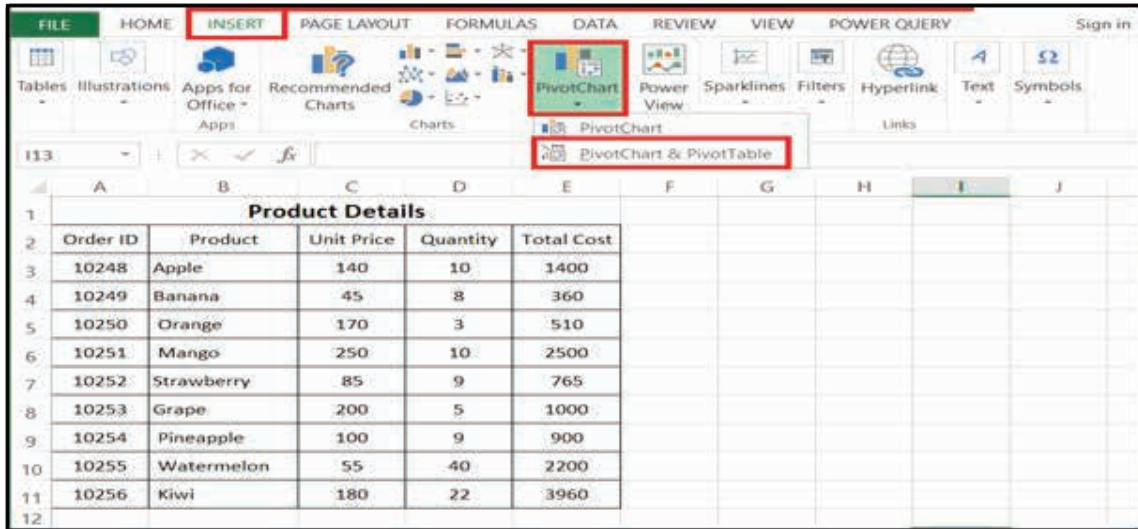
- a Create a Pivot Table to display the Order ID and Sum of Total Cost
- b Create a Pivot Chart based on that Pivot Table

To create a PivotTable:

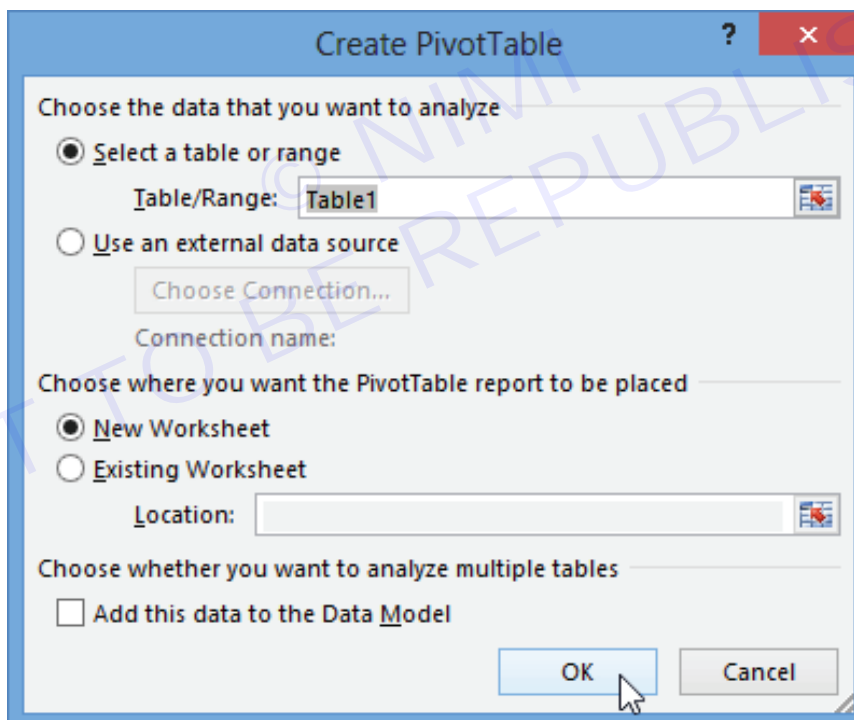
- 1 Select the table or cells (including column headers) containing the data you want to use.

	Product Details				
Order ID	Product	Unit Price	Quantity	Total Cost	
10248	Apple	140	10	1400	
10249	Banana	45	8	360	
10250	Orange	170	3	510	
10251	Mango	250	10	2500	
10252	Strawberry	85	9	765	
10253	Grape	200	5	1000	
10254	Pineapple	100	9	900	
10255	Watermelon	55	40	2200	
10256	Kiwi	180	22	3960	

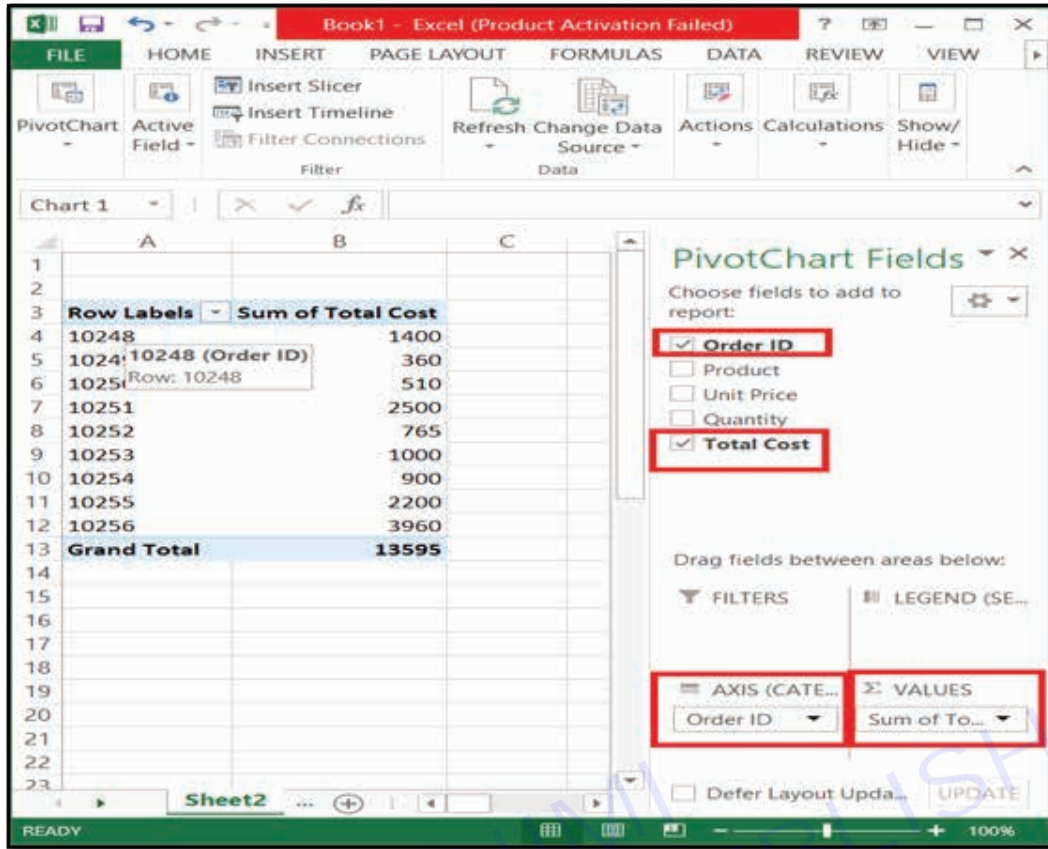
- 2 From the Insert tab, click the PivotTable command.



- 3 The Create PivotTable dialog box will appear. Choose your settings, then click OK. In our example, we'll use Table1 as our source data and place the PivotTable on a new worksheet.



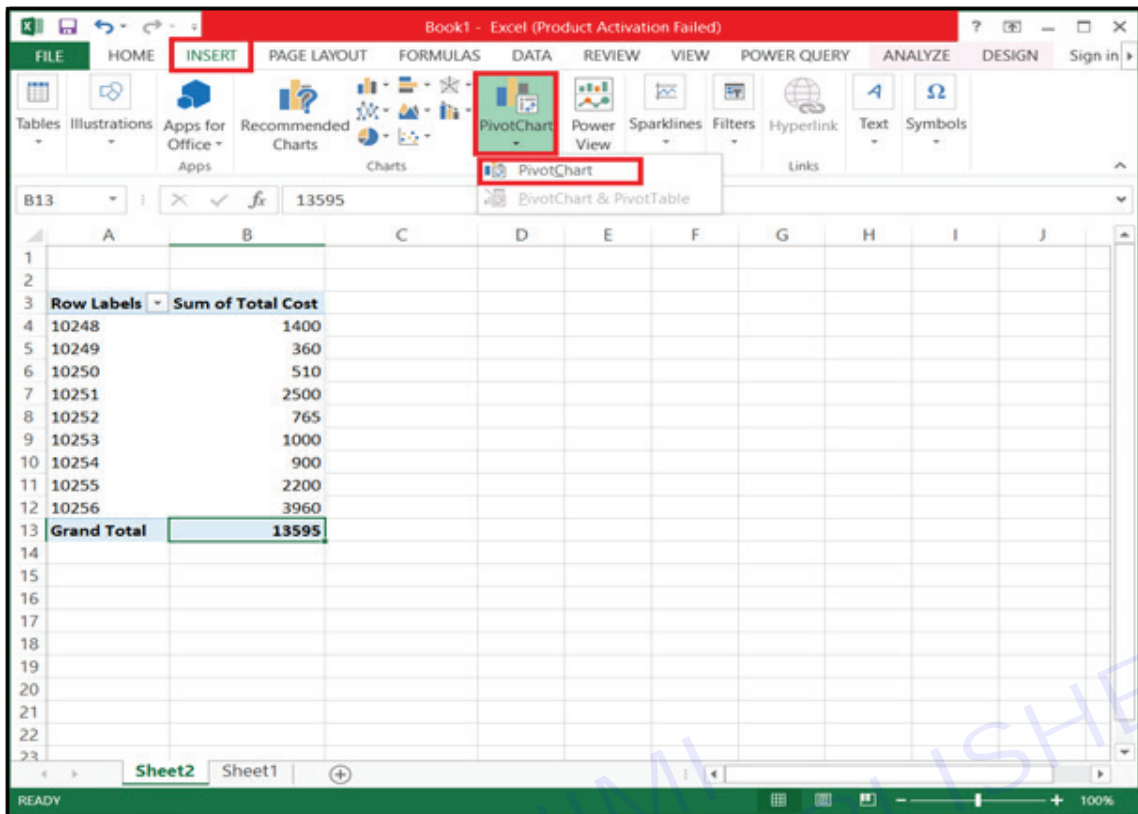
- 4 The selected fields will be added to one of the four areas below the Field List. In our example, the Order ID field has been added to the Rows area, while the Total Cost has been added to the Values area. Alternatively, you can click, hold, and drag a field to the desired area.



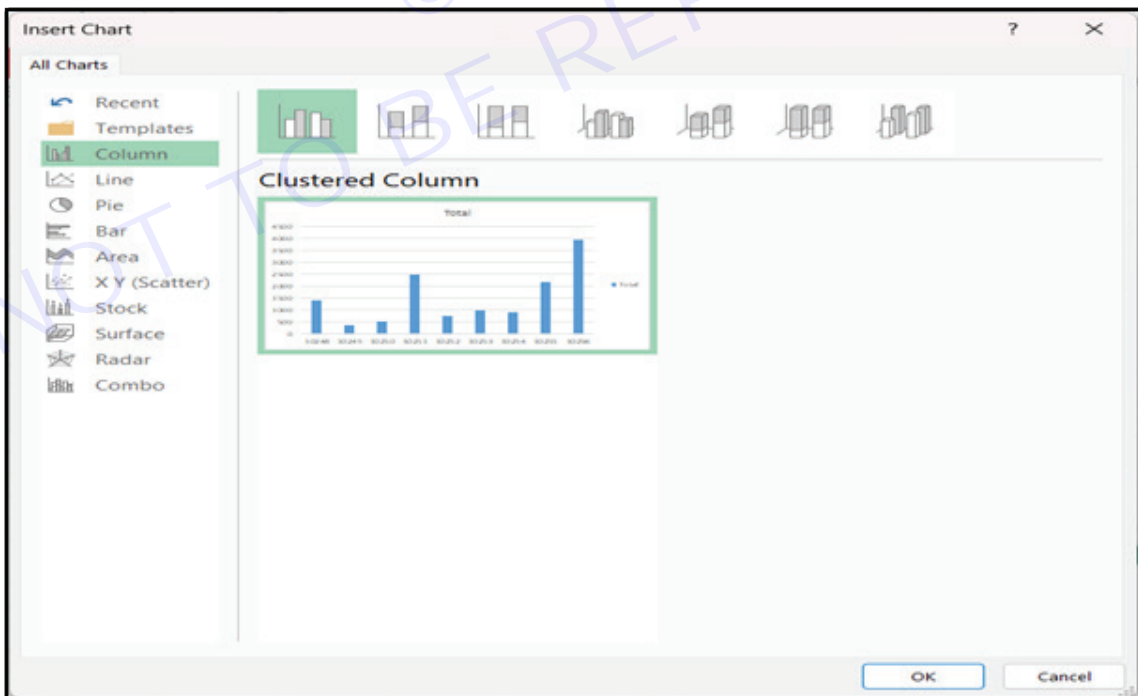
5 The PivotTable will calculate and summarize the selected fields.

1		
2		
3	Row Labels	Sum of Total Cost
4	10248	1400
5	10249	360
6	10250	510
7	10251	2500
8	10252	765
9	10253	1000
10	10254	900
11	10255	2200
12	10256	3960
13	Grand Total	13595
14		

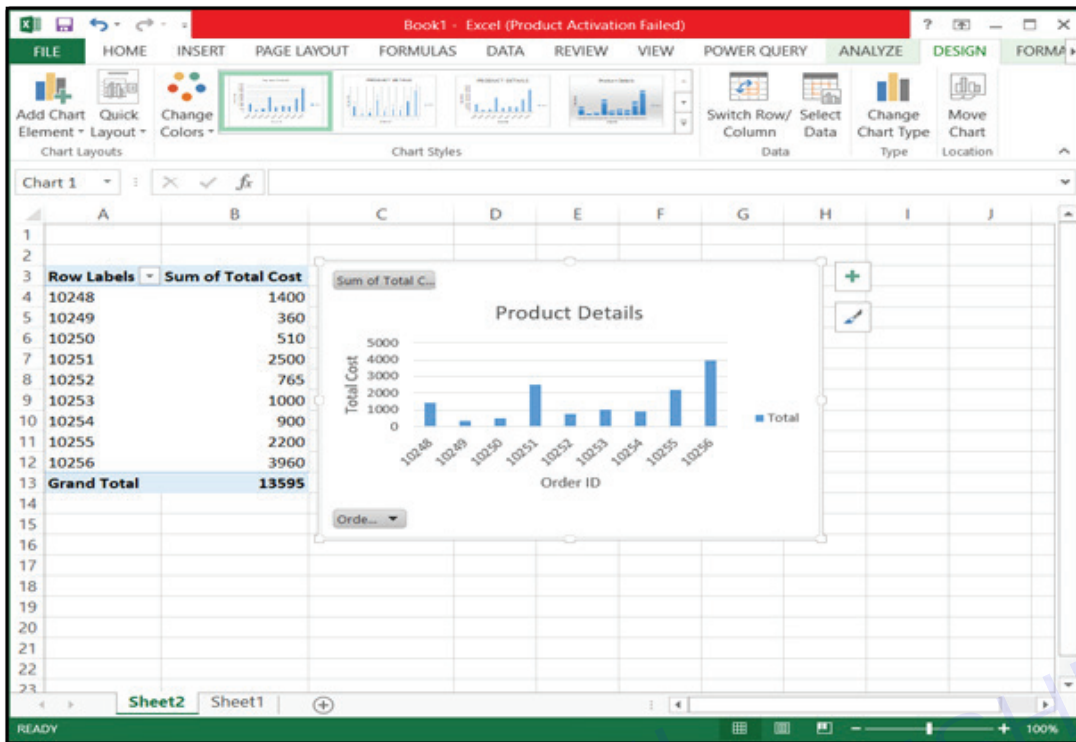
6 From the Insert tab, click the PivotChart command.



7 The Insert Chart dialog box will appear. Select the desired chart type and layout, then click OK.



8 The PivotChart will appear.



Related Exercise.

1 “By using the provided employee data, create a pivot table to analyze the total salary expenditure by department. Additionally, identify the department with the highest total salary expenditure and the average salary within that department.”

S.No.	EmployeeName	Designation	Department	Salary
1	John	Manager	Sales	50000
2	Alice	Assistant Manager	Marketing	40000
3	Bob	Sales Executive	Sales	36000
4	Emma	Marketing Analyst	Marketing	32000
5	David	Sales Associate	Sales	28000
6	Sarah	HR Assistant	HR	25000
7	Michael	Finance Manager	Finance	60000

Customize your pivot table further by applying filters, formatting, and rearranging fields as needed. Also prepare the Pivot Chart.

2 “By using the following product sales data, analyze the total sales revenue generated by each product. Additionally, visualize this data by creating both a pivot table and a pivot chart. Identify the product with the highest total sales revenue and its corresponding sales quantity.

Product Code	Product Name	Sales Quantity	Unit Price	Total Price
001	Widget A	100	\$10	\$1000
002	Widget B	150	\$8	\$1200
003	Widget C	200	\$15	\$3000
004	Widget D	75	\$20	\$1500
005	Widget E	120	\$12	\$1440
006	Widget F	90	\$18	\$1620

EXERCISE 76 : Demonstrate a power query, power query function. Invoking the power query function and combining queries. Organize the workbook queries

Objectives

At the end of this exercise you shall be able to

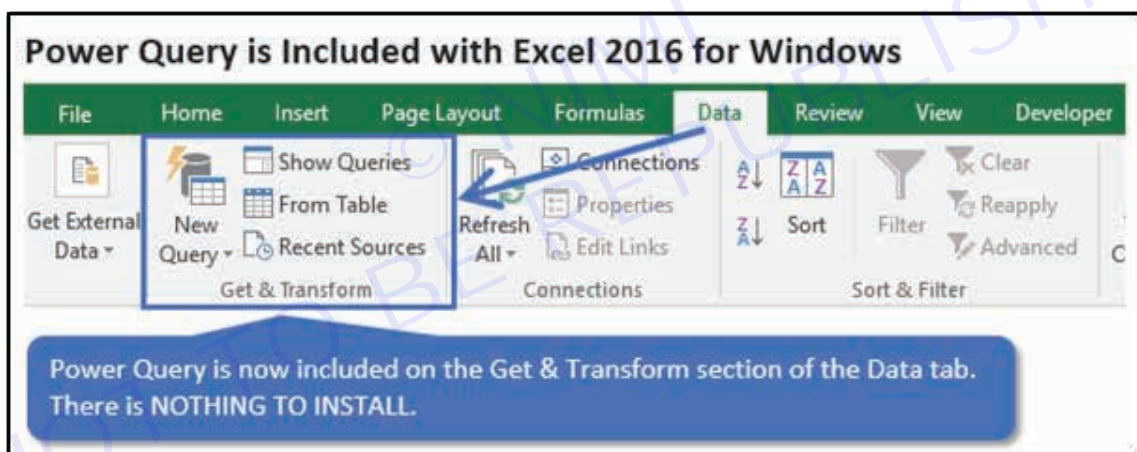
- install power query
- demonstrate a power query, power query function
- invoking the power query function and combining queries.

Procedure

How to Install Power Query

Excel 2016

Power Query is included with Excel 2016 (Office 365). It has been renamed and is now on the Data tab of the Ribbon in the Get & Transform section.



This means there is nothing to install. If you are using Excel 2016, go to the Data tab on the ribbon and press the New Query button to create a query and open the Power Query editor.

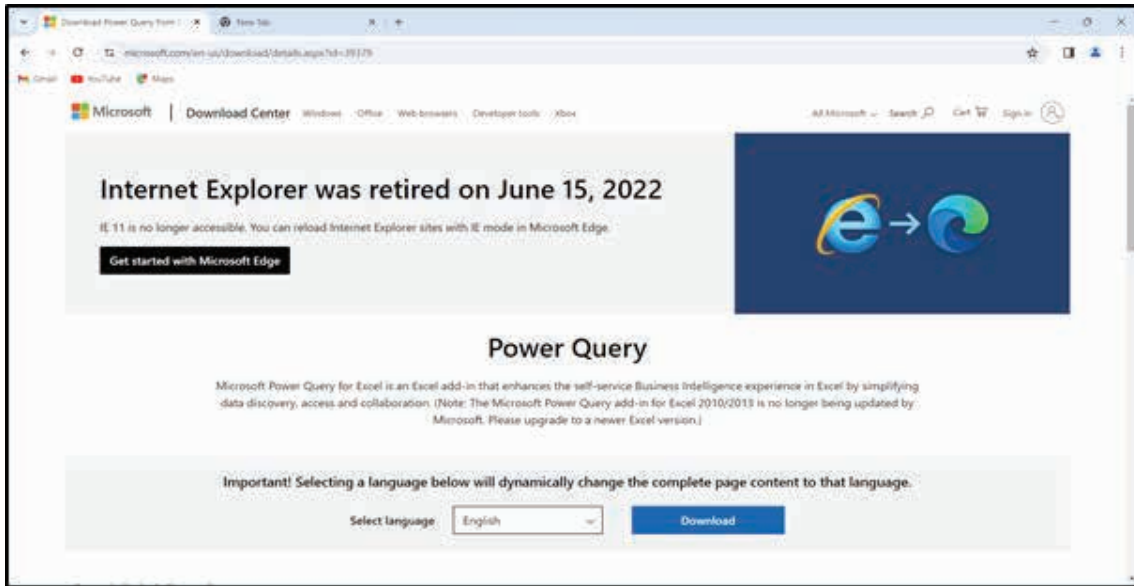
Power Query is available with all levels of Office 365 subscriptions.

Excel 2010 & 2013

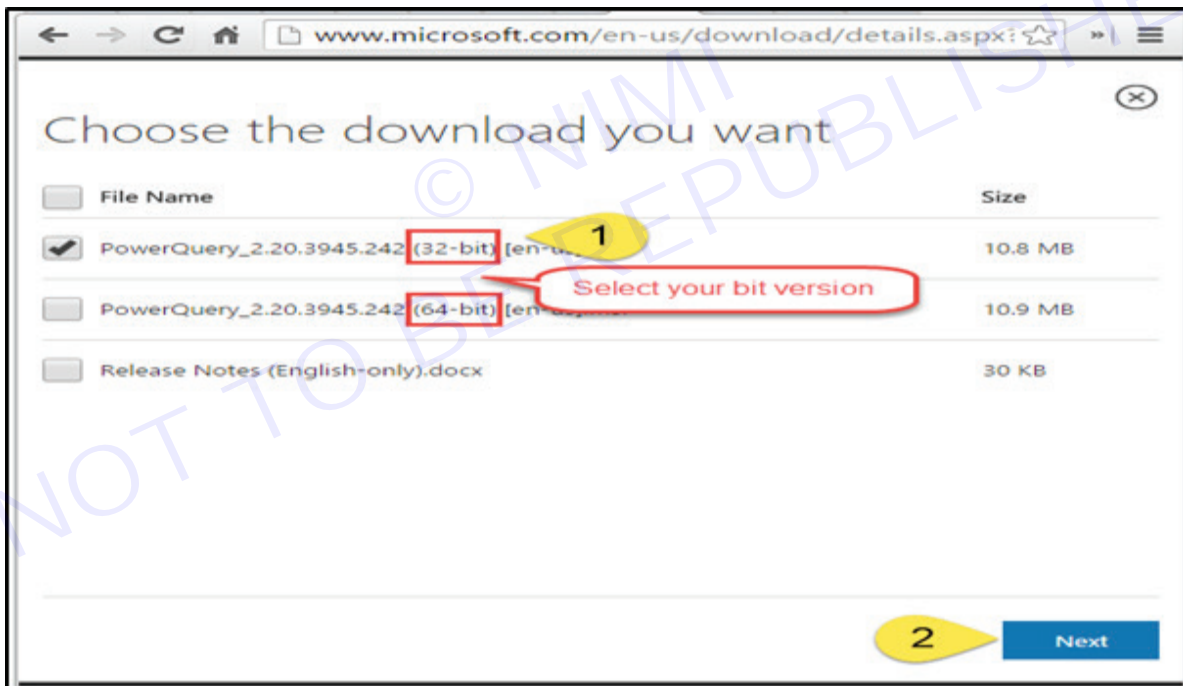
For Excel 2010 and 2013 you will need to download the Power Query add-in and install it.

The installation steps are about the same in Excel 2010 and 2013.

- 1 Close (exit) Excel completely.
- 2 Click the following link to go to the download page.
<https://www.microsoft.com/en-us/download/details.aspx?id=39379>
- 3 Click the Download button.

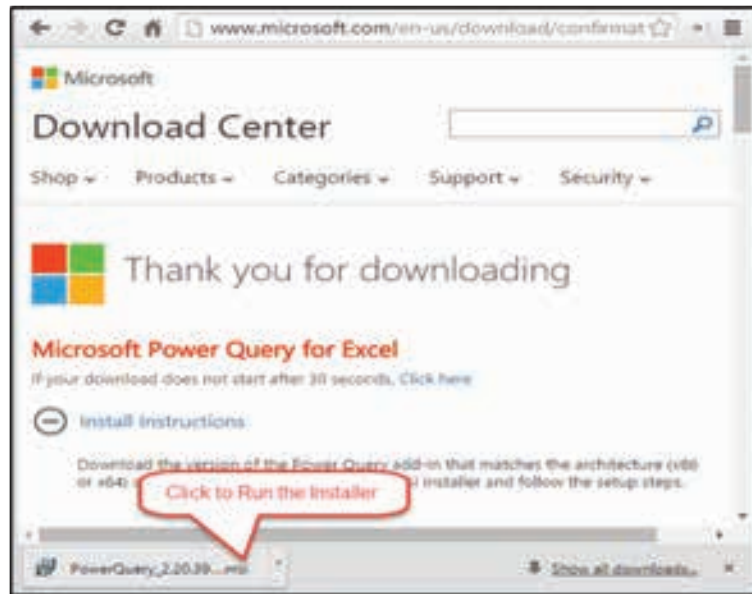


4 Click the checkbox for the bit version you are using. Most likely you will be using 32-bit.



5 Click Next

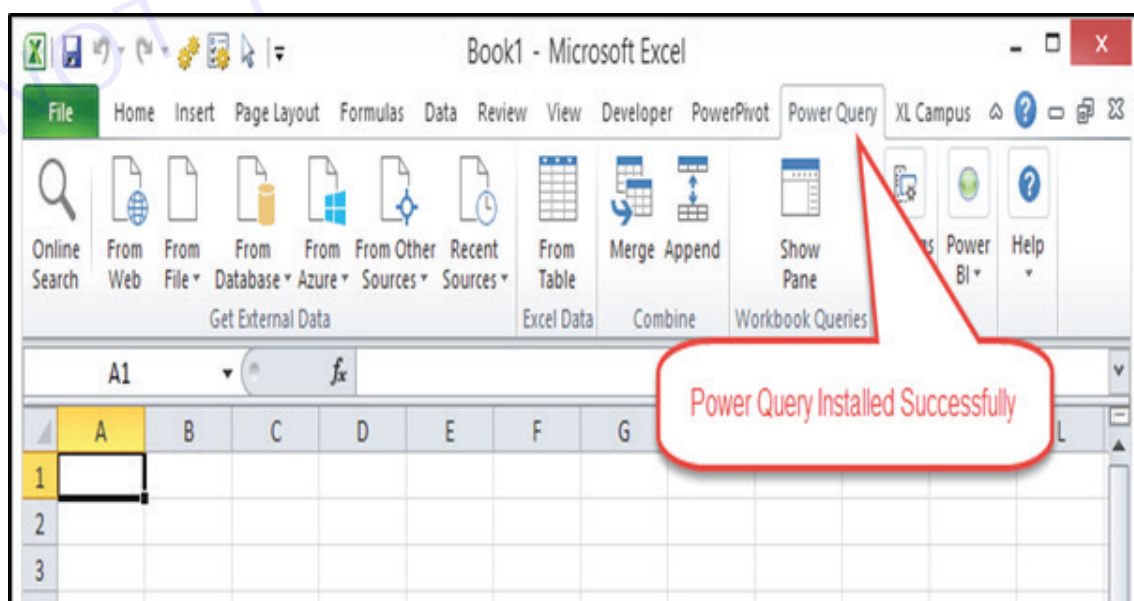
6 The add-in installation file will download. Click the file to run the installation.



7 The Setup Wizard window will open. Follow the steps to install Power Query.



8 Once the installation is complete, open Excel. You should now see the Power Query tab in the Ribbon.

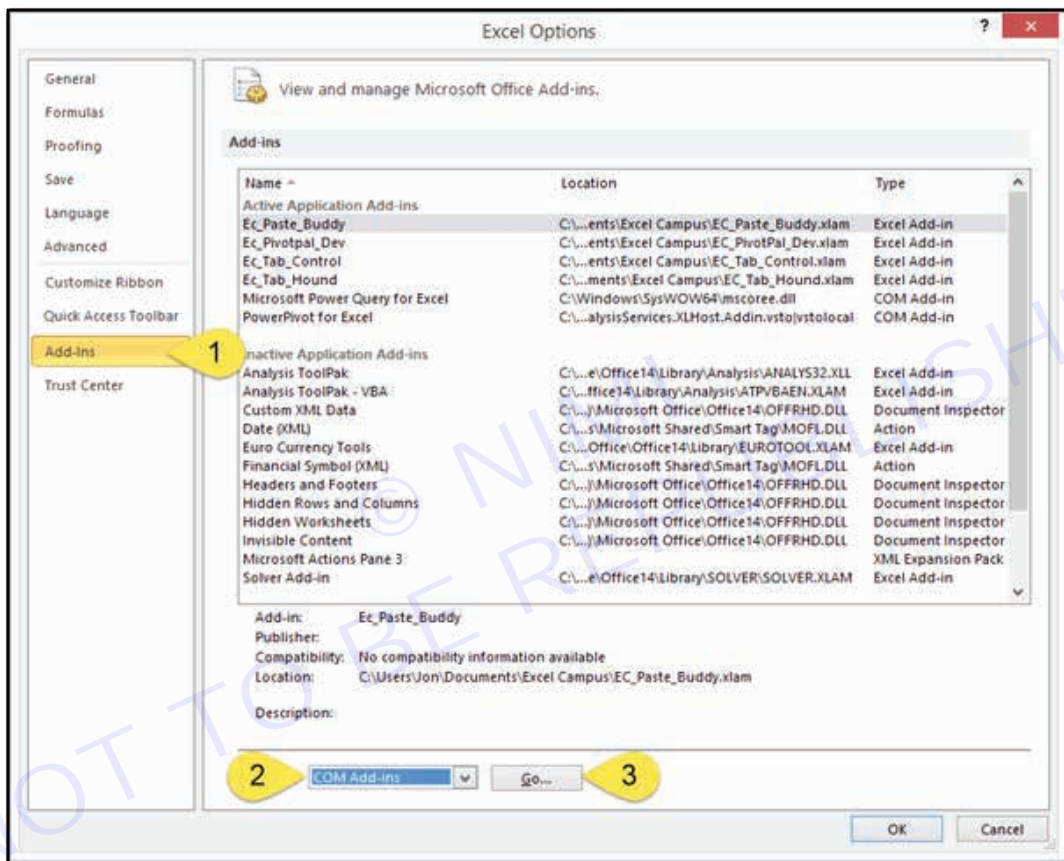


My Power Query Tab Disappeared

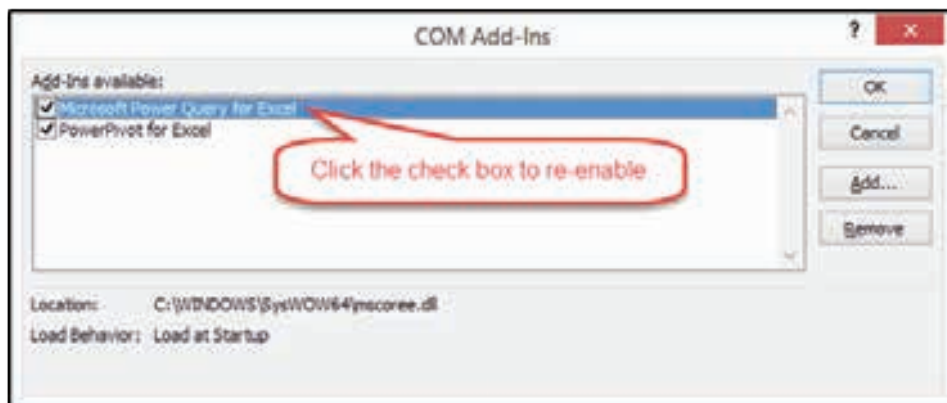
If your **Power Query** tab ever goes missing, you can usually re-enable the add-in by going to the COM Add-ins menu.

There are a few ways to get to the COM Add-ins menu.

- 1 File menu.
- 2 Click Options on left side menu.
- 3 Click Add-ins on left side menu.
- 4 Select COM Add-ins from the Manage drop-down.
- 5 Click the Go... button.



- 6 That will open the COM Add-ins Window. If the Power Query check box is not selected, just select it to reload the add-in.



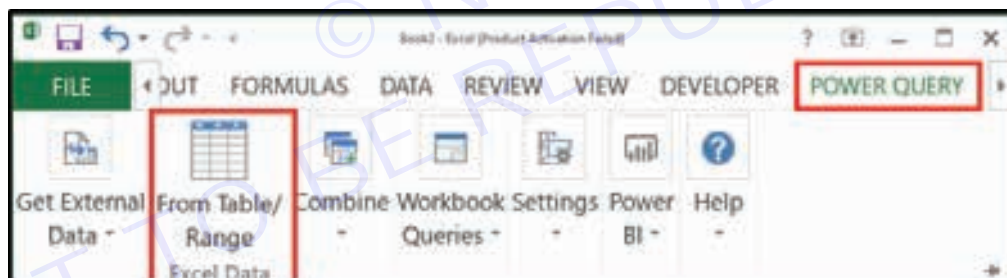
TASK 1: Create a simple Power Query program using a sample Excel file

1 Load Data:

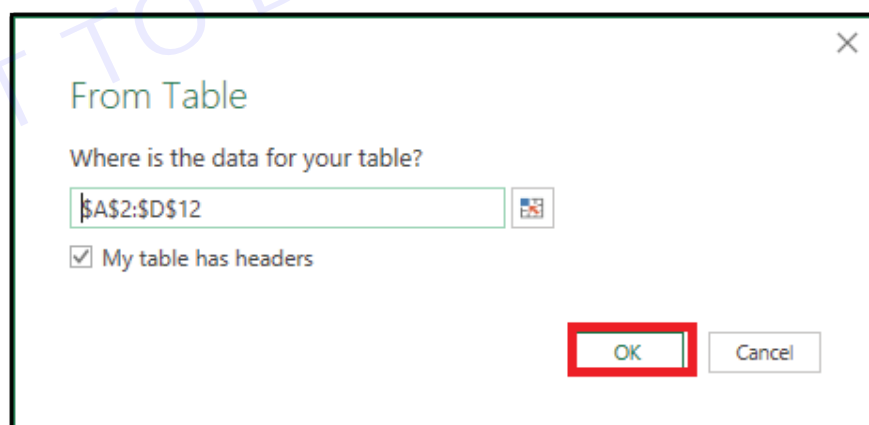
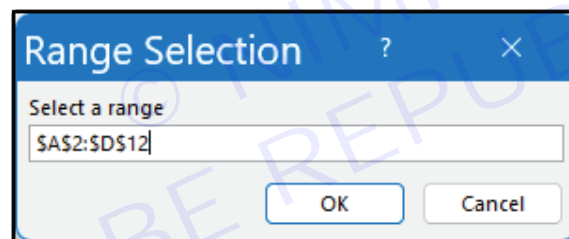
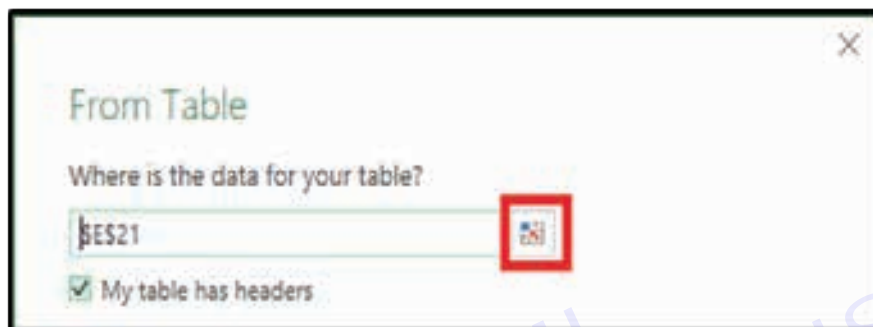
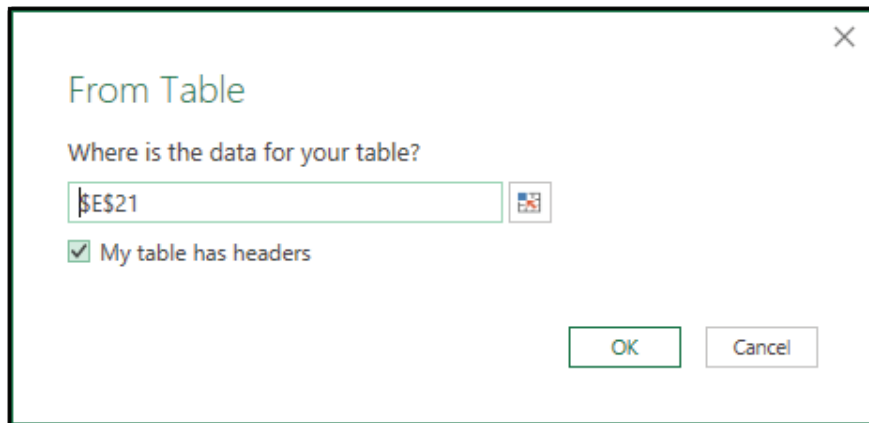
- Open Excel and create a new workbook.
- Enter some sample data into a worksheet.

INTEREST CALCULATION				
SI No	Name	Principal Amount	No of Years	
1	ANU.R	4700	5	
2	ARUNIMA.R	3500	4	
3	ARYA.A.R	1100	8	
4	ASWATHY.K	2500	3	
5	BISMINA.S	4000	2	
6	ISHA.S.S	4500	7	
7	KARTHIKA. B.B	1500	6	
8	RESHMA.R	5500	3	
9	RIYA.L	3800	4	
10	SREEDHA.A	6000	9	

- Go to the “Data” tab and click “Get Data” > “From Table/Range.”/ Go to the “Power Query” tab and click “From Table/Range.”

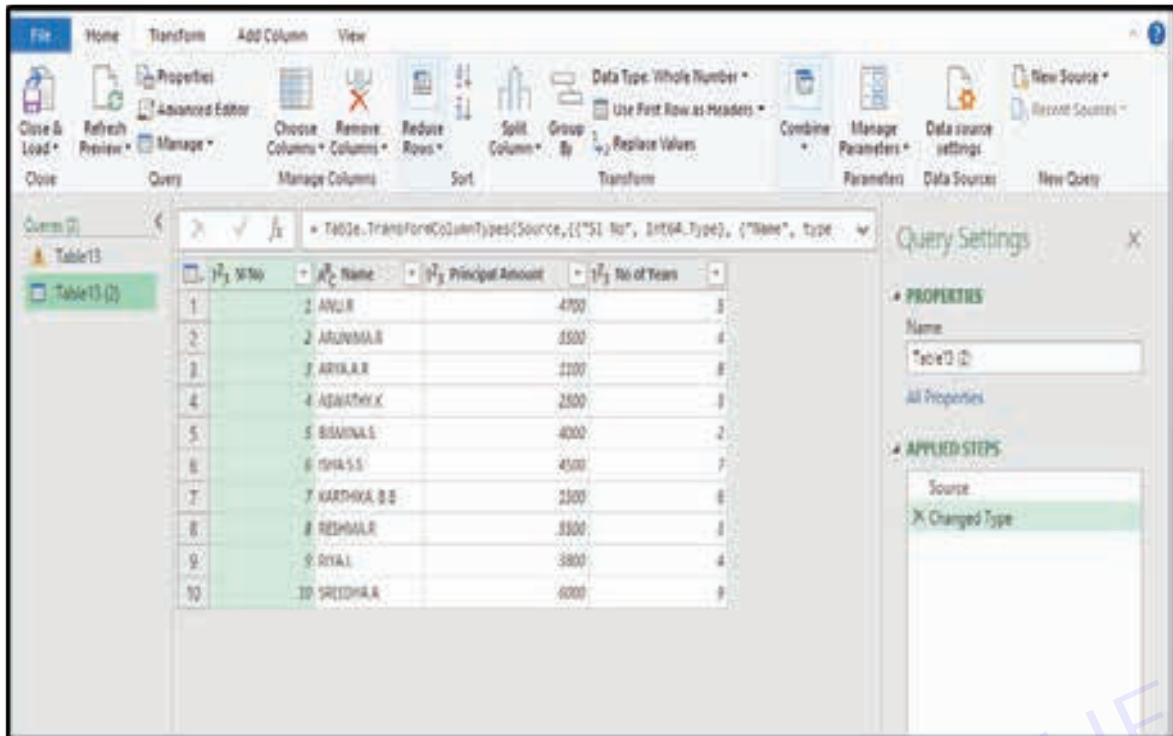


INTEREST CALCULATION				
SI No	Name	Principal Amount	No of Years	
1	ANU.R	4700	5	
2	ARUNIMA.R	3500	4	
3	ARYA.A.R	1100	8	
4	ASWATHY.K	2500	3	
5	BISMINA.S	4000	2	
6	ISHA.S.S	4500	7	
7	KARTHIKA. B.B	1500	6	
8	RESHMA.R	5500	3	
9	RIYA.L	3800	4	
10	SREEDHA.A	6000	9	



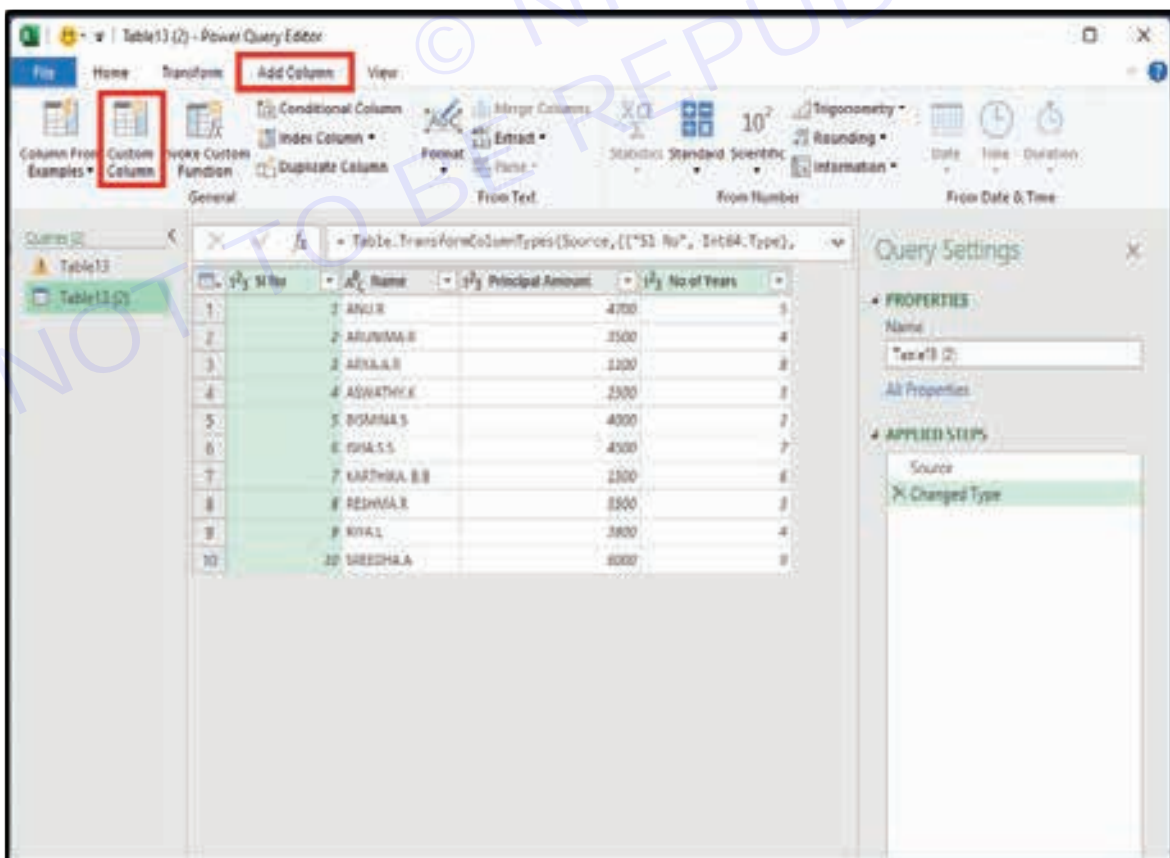
2 Power Query Editor:

- The Power Query Editor will open, displaying a preview of your data.
- You can see the “SL NO,” “NAME,” “Salary,” and “NO OF YEAR” columns.

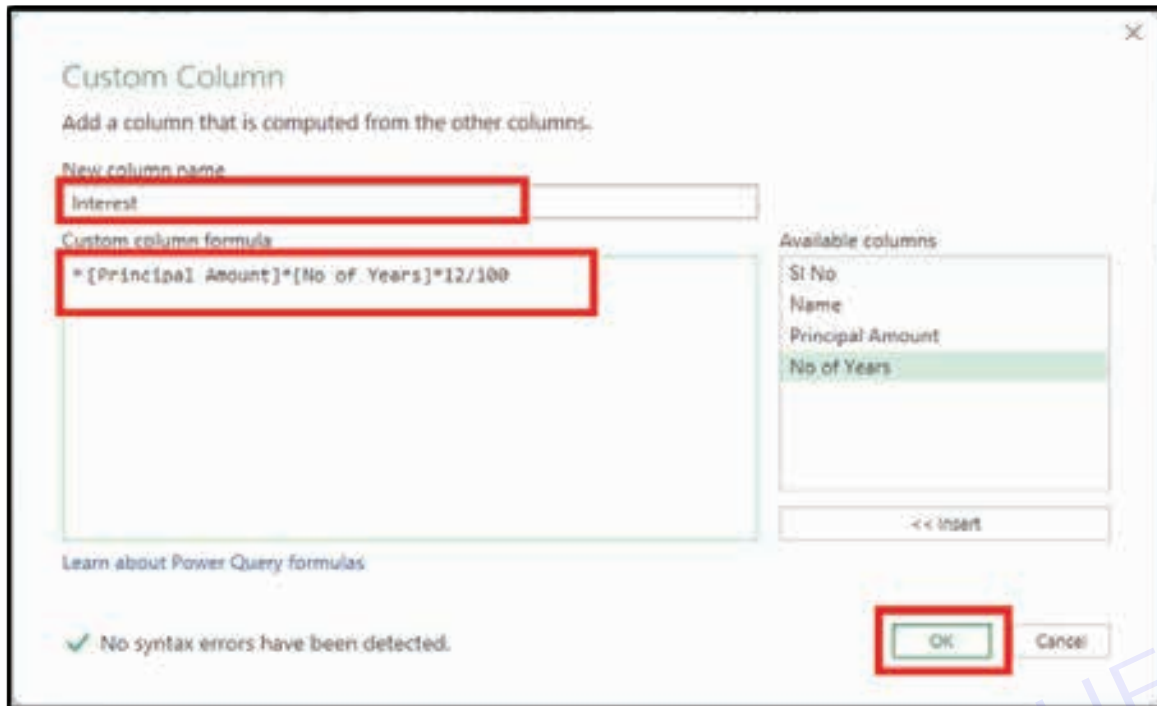


3 Transform Data:

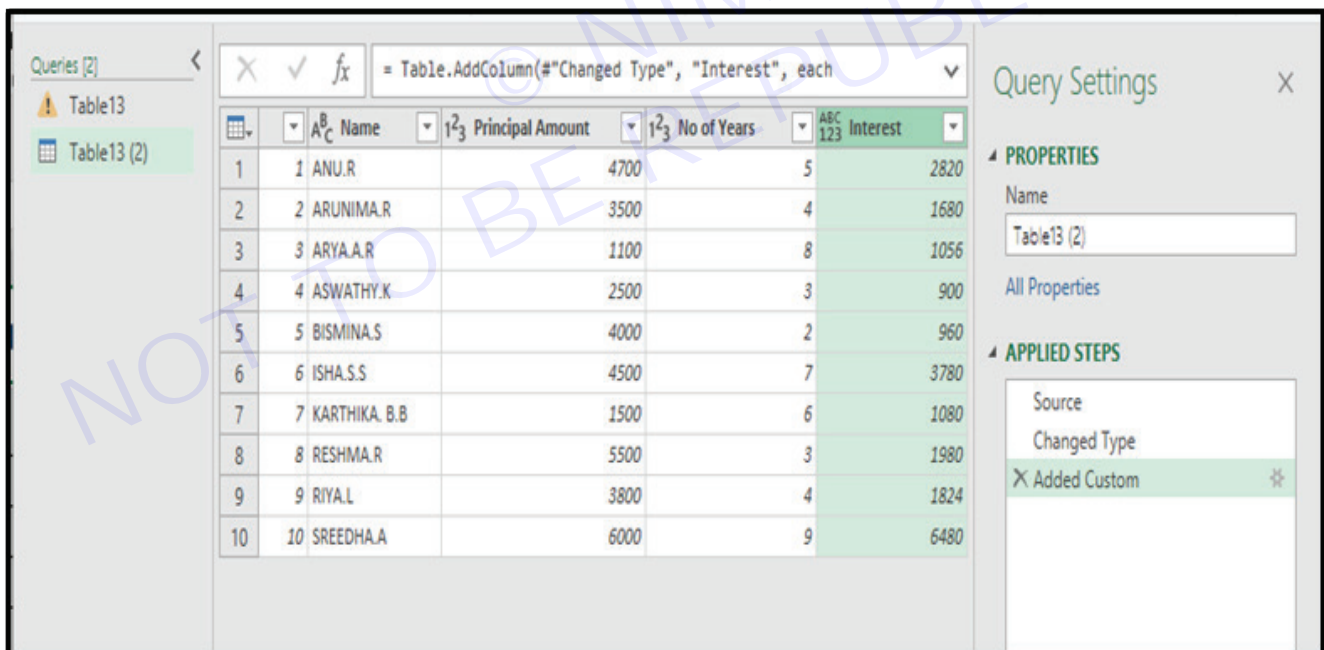
- Let's add a new column that calculates a bonus based on the Principal Amount. Click on "Add Column" > "Custom Column."



- Name the new column: "Interest"
- Enter the formula: $= [\text{Principal Amount}] * [\text{No of Years}] * 0.12$ (Assuming a 12% Interest)

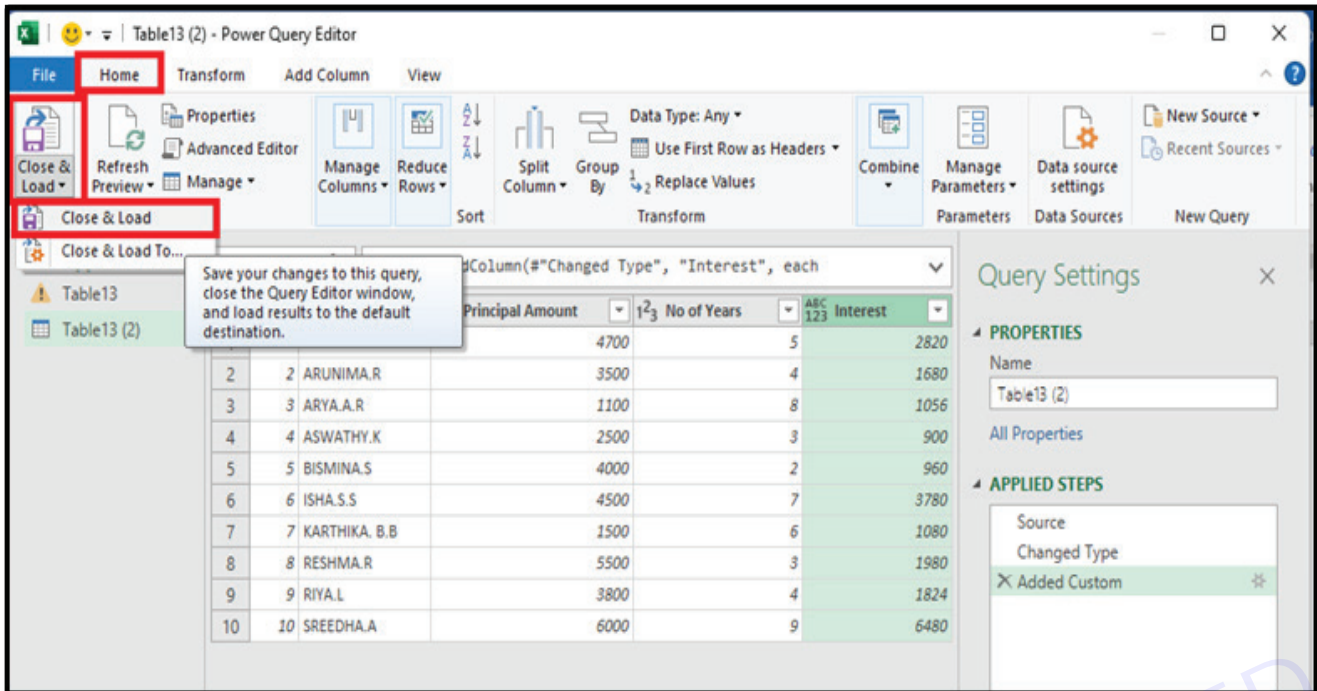


- Click "OK" to create the new column.



4 Load Transformed Data:

- Once you're satisfied with the transformations, click "Close & Apply" to load the data back into Excel.



- The transformed data will appear in your existing workbook.

	A	B	C	D	E	F
1	Sl No	Name	Principal Amount	No of Years	Interest	
2	1	ANU.R	4700	5	2820	
3	2	ARUNIMA.R	3500	4	1680	
4	3	ARYA.A.R	1100	8	1056	
5	4	ASWATHY.K	2500	3	900	
6	5	BISMINA.S	4000	2	960	
7	6	ISHA.S.S	4500	7	3780	
8	7	KARTHIKA. B.B	1500	6	1080	
9	8	RESHMA.R	5500	3	1980	
10	9	RIYA.L	3800	4	1824	
11	10	SREEDHA.A	6000	9	6480	

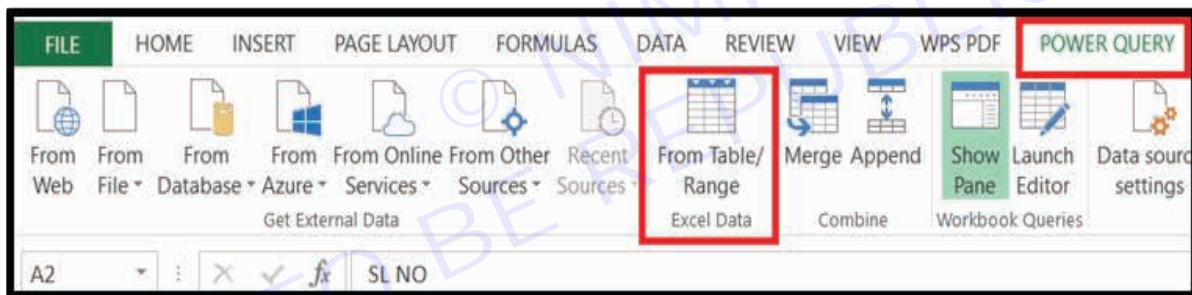
TASK 2: Create a simple Power Query program using a sample Excel file

Step 1: Load Data

- 1 Open Excel and create a new worksheet.
- 2 Enter some sample data in a table. For example:

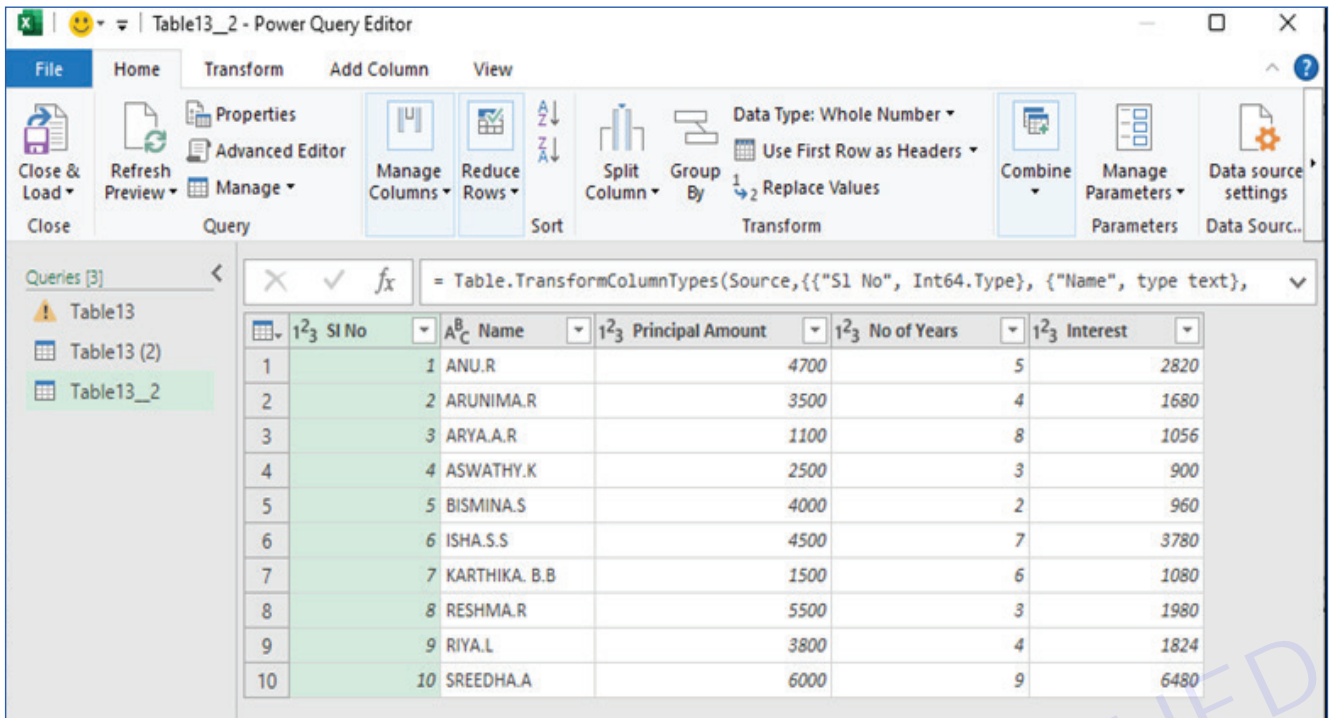
Interest Calculation					
Sl No	Name	Principal Amount	No of Years	Interest	
1	ANU.R	4700	5	2820	
2	ARUNIMA.R	3500	4	1680	
3	ARYA.A.R	1100	8	1056	
4	ASWATHY.K	2500	3	900	
5	BISMINA.S	4000	2	960	
6	ISHA.S.S	4500	7	3780	
7	KARTHIKA. B.B	1500	6	1080	
8	RESHMA.R	5500	3	1980	
9	RIYA.L	3800	4	1824	
10	SREEDHA.A	6000	9	6480	

- Select the data and Go to the “Data” tab and click “Get Data” > “From Table/Range.”/ Go to the “Power Query” tab and click “From Table/Range.”

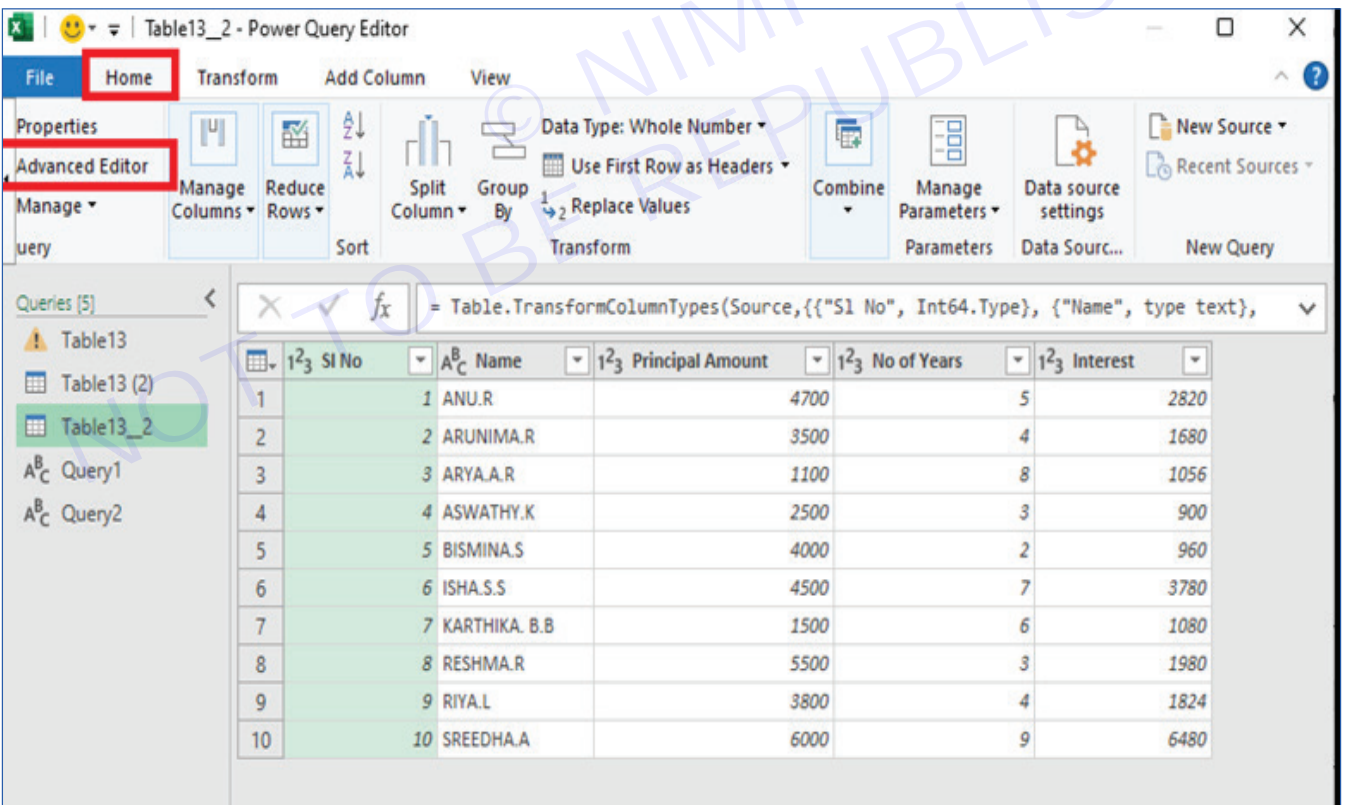


Interest Calculation					
Sl No	Name	Principal Amount	No of Years	Interest	
1	ANU.R	4700	5	2820	
2	ARUNIMA.R	3500	4	1680	
3	ARYA.A.R	1100	8	1056	
4	ASWATHY.K	2500	3	900	
5	BISMINA.S	4000	2	960	
6	ISHA.S.S	4500	7	3780	
7	KARTHIKA. B.B	1500	6	1080	
8	RESHMA.R	5500	3	1980	
9	RIYA.L	3800	4	1824	
10	SREEDHA.A	6000	9	6480	

- Then Power Query Editor will open. Here, you can see a preview of your data.



- Then Click “Advanced Editor” to open the script editor.





- Write the code



// Add a new column "Sum" by adding the values from the referenced cells

CombinedQuery = Table.AddColumn(Source, "Sum", each [Principal Amount] + [Interest])

- **Excel.CurrentWorkbook():** Function that references the entire current workbook.
- **{[Name="Table13_2"]}**: Accesses a specific table in the workbook named "Table13_2."
- **[Content]**: Retrieves the content of the "Table13_2" table.

Add a New Column "Sum"

// Add a new column "Sum" by adding the values from the referenced cells

CombinedQuery = Table.AddColumn(Source, "Sum", each [Principal Amount] + [Interest])

- **Table.AddColumn(Source, "Sum", each [Principal Amount] + [Interest]):** Adds a new column named "Sum" to the **Source** table.
- **each [Principal Amount] + [Interest]:** Defines the operation to be performed in each row of the new "Sum" column. It adds the values from the "Principal Amount" and "Interest" columns.

ABC 123 SI No	ABC 123 Name	ABC 123 Principal Amount	ABC 123 No of Years	ABC 123 Interest	ABC 123 Sum
1	1 ANU.R	4700	5	2820	7520
2	2 ARUNIMA.R	3500	4	1680	5180
3	3 ARYA.A.R	1100	8	1056	2156
4	4 ASWATHY.K	2500	3	900	3400
5	5 BISMINA.S	4000	2	960	4960
6	6 ISHA.S.S	4500	7	3780	8280
7	7 KARTHIKA. B.B	1500	6	1080	2580
8	8 RESHMA.R	5500	3	1980	7480
9	9 RIYA.L	3800	4	1824	5624
10	10 SREEDHA.A	6000	9	6480	12480

- Click "Close & Apply" to apply the changes and load the data into your Excel workbook or Power BI report.

The screenshot shows the Power Query Editor window for 'Table13_2'. The 'Home' tab is active, and the 'Close & Load' button is highlighted with a red box. The formula bar displays the M code: `Table.AddColumn(Source, "Sum", each[Principal Amount]+[Interest])`. Below the formula bar, the data table is visible, showing the same 10 rows of data with the 'Sum' column calculated.

	A	B	C	D	E	F	G	H
1	Sl No	Name	Principal Amount	No of Years	Interest	Sum		
2	1	ANU.R	4700	5	2820	7520		
3	2	ARUNIMA.R	3500	4	1680	5180		
4	3	ARYA.A.R	1100	8	1056	2156		
5	4	ASWATHY.K	2500	3	900	3400		
6	5	BISMINA.S	4000	2	960	4960		
7	6	ISHA.S.S	4500	7	3780	8280		
8	7	KARTHIKA. B.B	1500	6	1080	2580		
9	8	RESHMA.R	5500	3	1980	7480		
10	9	RIYA.L	3800	4	1824	5624		
11	10	SREEDHA.A	6000	9	6480	12480		
12								

This code assumes that you have a table named "Table13_2" in your workbook with columns named "Principal Amount" and "Interest." The result is a new table (**CombinedQuery**) that includes the original columns from "Table13_2" and an additional column named "Sum" containing the sum of the "Principal Amount" and "Interest" values for each row.

TASK 3 :

Create a table named "OrderDetails" containing columns for "Product", "Quantity", and "UnitPrice". Apply a 10% discount to the "UnitPrice" for orders where the quantity purchased is greater than or equal to 10.

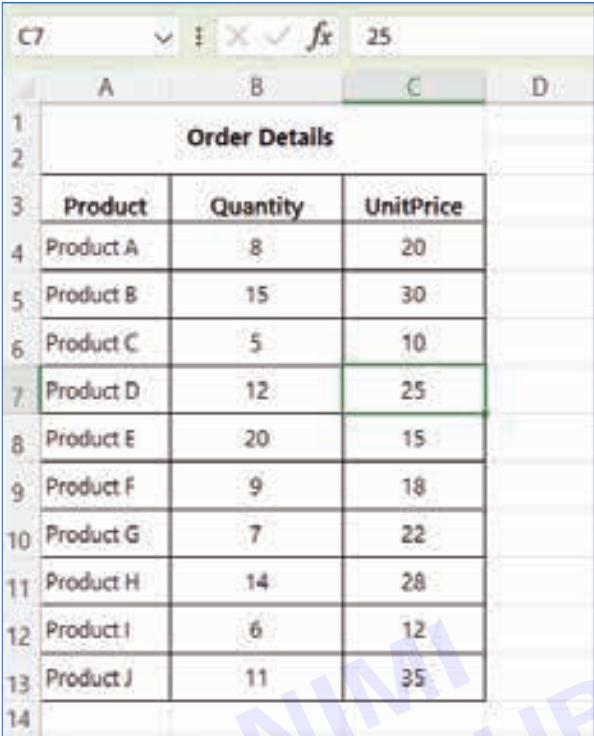
Order Details

Product	Quantity	Unit Price
Product A	8	20.00
Product B	15	30.00
Product C	5	10.00
Product D	12	25.00
Product E	20	15.00
Product F	9	18.00
Product G	7	22.00
Product H	14	28.00
Product I	6	12.00
Product J	11	35.00



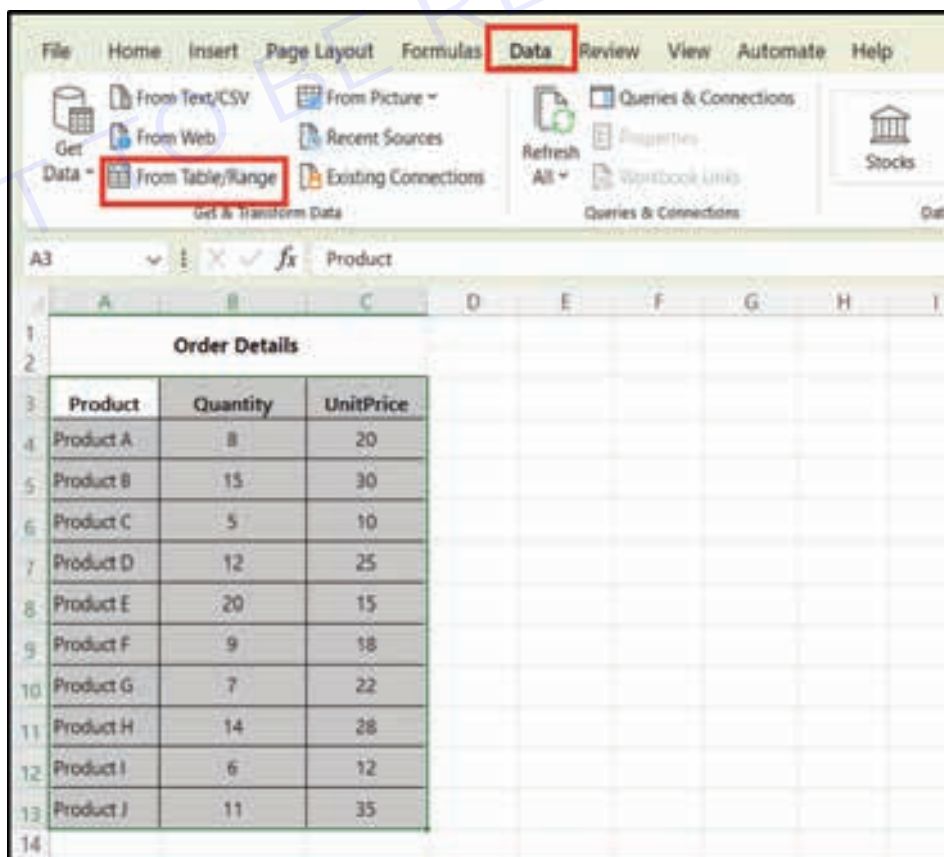
1 Load Data:

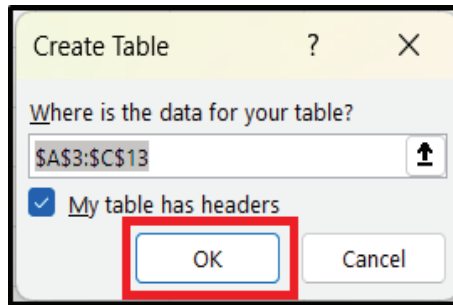
- Open Excel and create a new workbook.
- Enter some sample data into a worksheet.



Order Details		
Product	Quantity	UnitPrice
Product A	8	20
Product B	15	30
Product C	5	10
Product D	12	25
Product E	20	15
Product F	9	18
Product G	7	22
Product H	14	28
Product I	6	12
Product J	11	35

- Go to the “Data” tab and click “Get Data” > “From Table/Range.”/ Go to the “Power Query” tab and click “From Table/Range.”





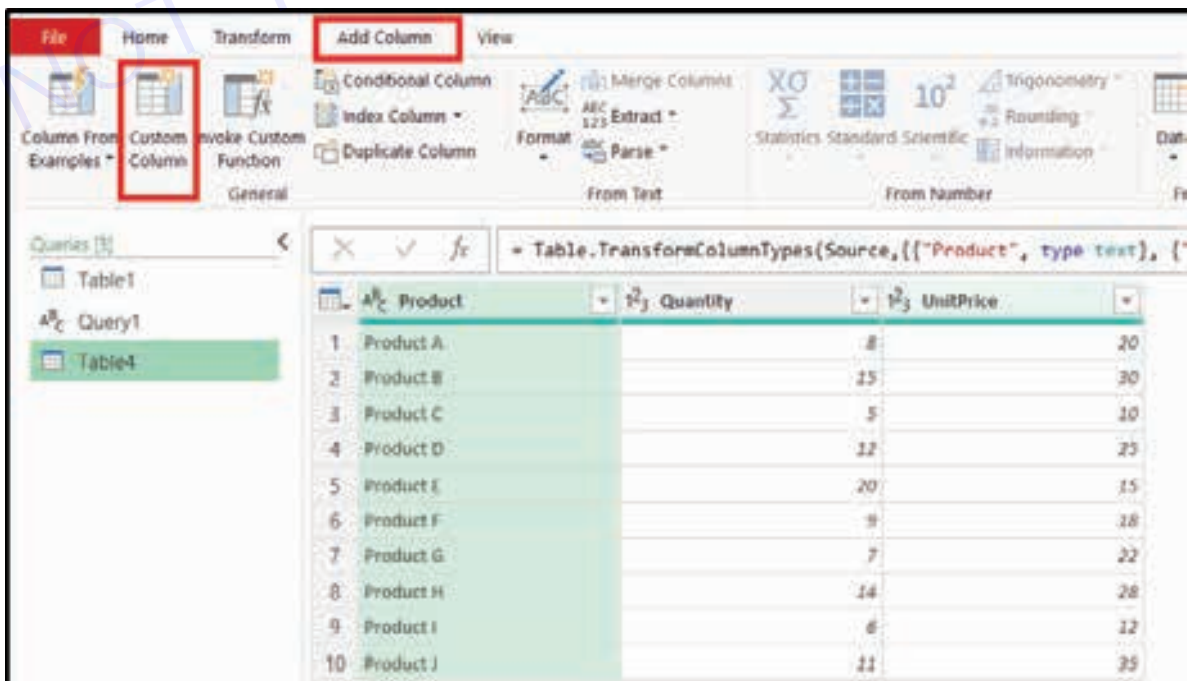
2 Power Query Editor:

- The Power Query Editor will open, displaying a preview of your data.
- You can see the “Product”, “Quantity,” and “Unit Price” columns.

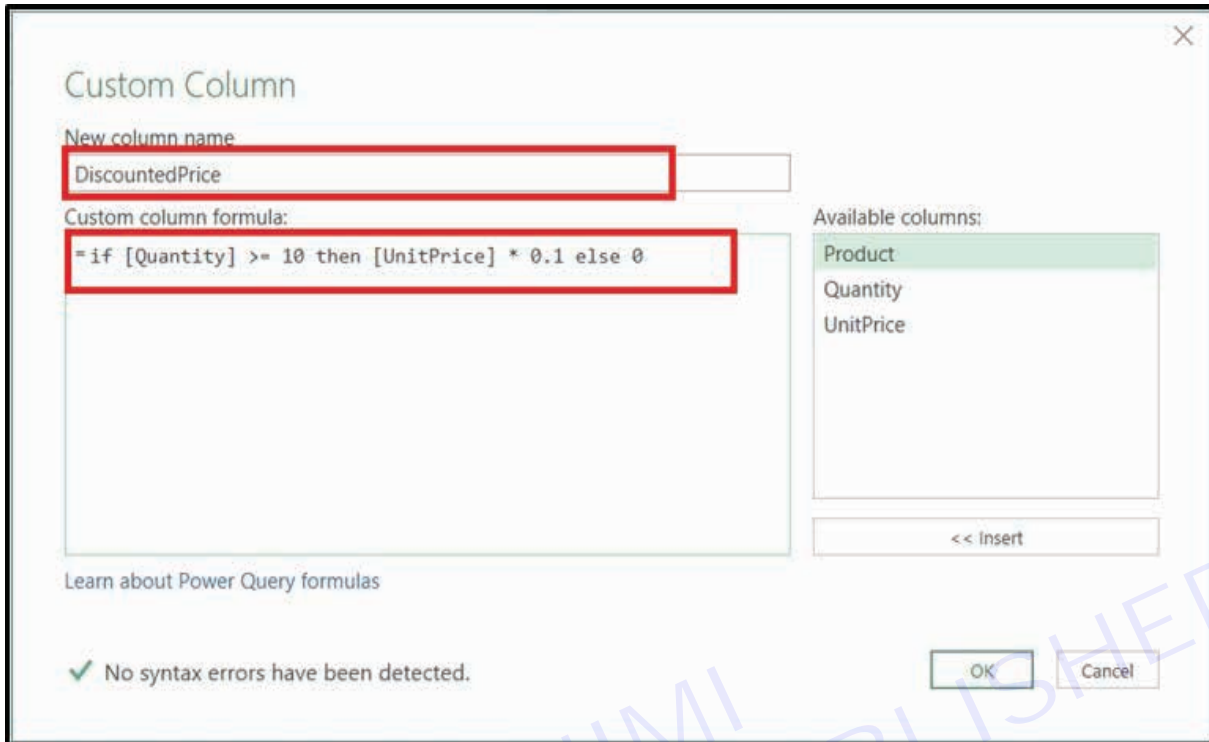
	Product	Quantity	UnitPrice
1	Product A	8	20
2	Product B	15	30
3	Product C	5	10
4	Product D	12	25
5	Product E	20	15
6	Product F	9	18
7	Product G	7	22
8	Product H	14	28
9	Product I	6	12
10	Product J	11	35

3 Transform Data:

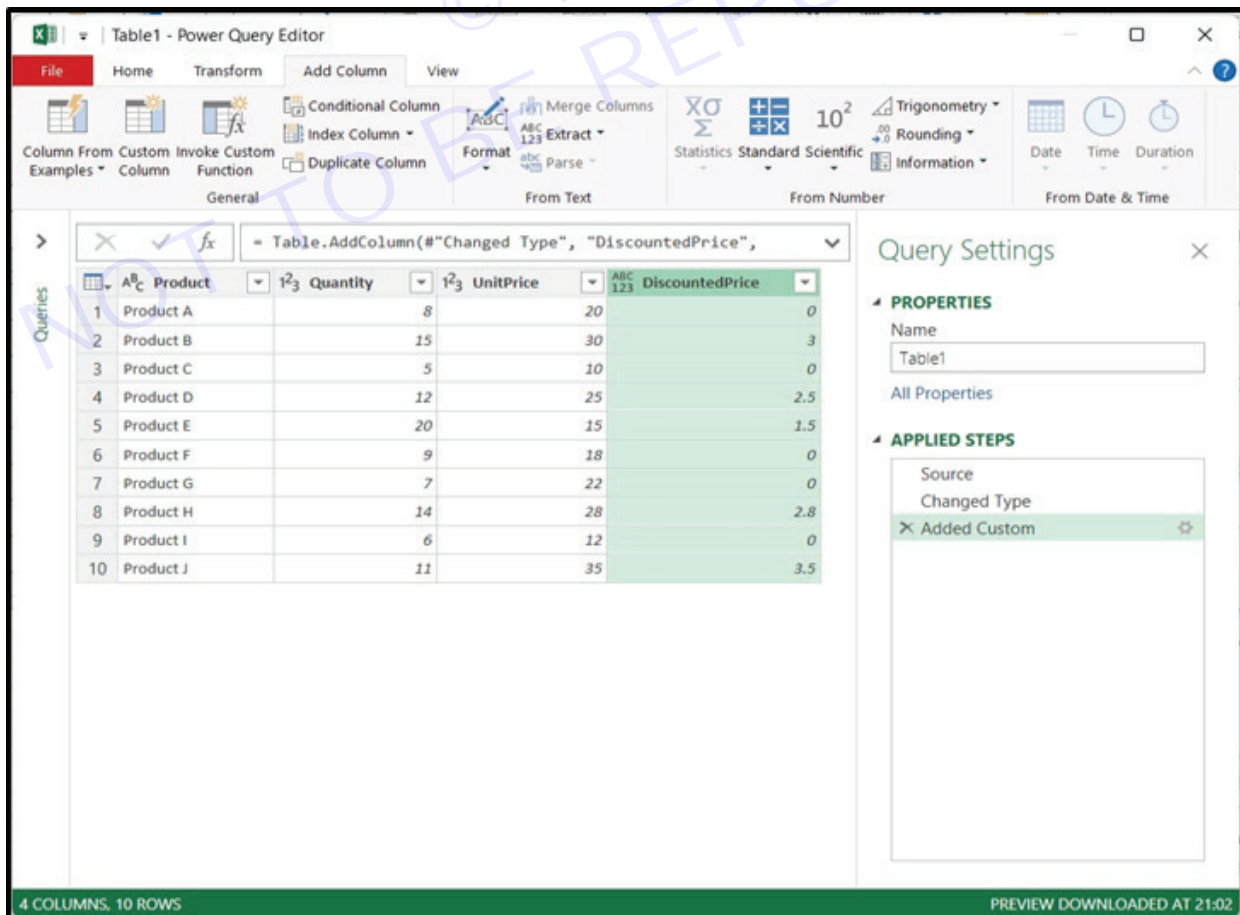
- Let’s add a new column that calculates a DiscountedPrice based on the Quantity. Click on “Add Column” > “Custom Column.”



- Name the new column: "DiscountedPrice"
- Enter the formula: = if [Quantity] >= 10 then [UnitPrice] * 0.1 else 0



- Click "OK" to create the new column.



- Let's add a new column that calculates Net Price based on the Quantity, Unit Price and Discounted Price. Click on "Add Column" > "Custom Column."

Custom Column

New column name

Custom column formula:

Available columns:

<< Insert

[Learn about Power Query formulas](#)

✓ No syntax errors have been detected.

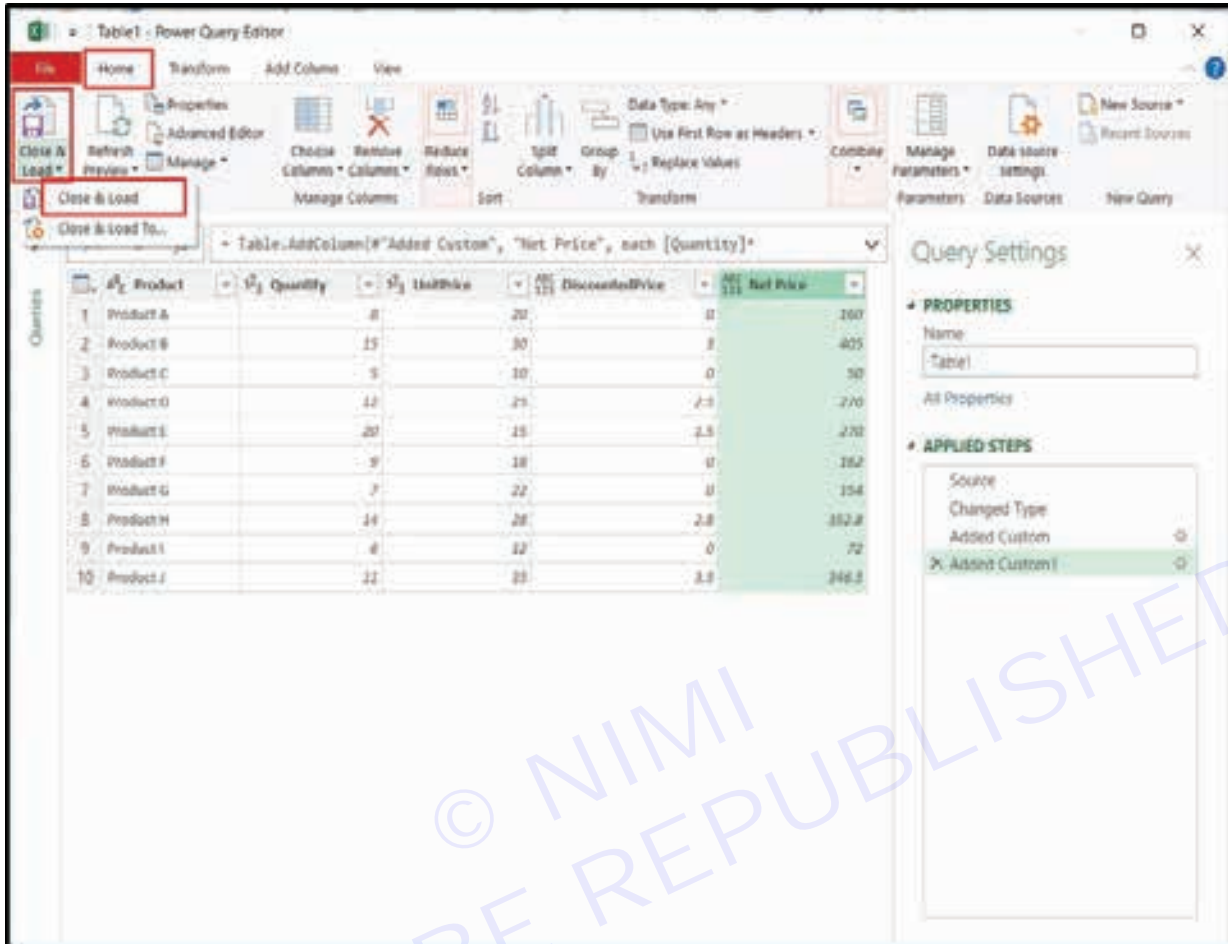
- Click "OK" again to create the new column.

= Table.AddColumn("#Added Custom", "Net Price", each [Quantity]*([UnitPrice]-[DiscountedPrice]))

	Product	Quantity	UnitPrice	DiscountedPrice	Net Price
1	Product A	8	20	0	160
2	Product B	15	30	3	405
3	Product C	5	10	0	50
4	Product D	12	25	2.5	270
5	Product E	20	15	1.5	270
6	Product F	9	18	0	162
7	Product G	7	22	0	154
8	Product H	14	28	2.8	352.8
9	Product I	6	12	0	72
10	Product J	11	35	3.5	346.5

4 Load Transformed Data:

- Once you're satisfied with the transformations, click "Close & Apply" to load the data back into Excel.



- The transformed data will appear in your existing workbook.

The screenshot shows an Excel spreadsheet with the following data:

	1	2	3	4	5	6
1	Product	Quantity	UnitPrice	DiscountedPrice	Net Price	
2	Product A	8	20	0	160	
3	Product B	15	30	3	405	
4	Product C	5	10	0	50	
5	Product D	12	25	2.5	270	
6	Product E	20	15	1.5	270	
7	Product F	9	18	0	162	
8	Product G	7	22	0	154	
9	Product H	14	28	2.8	352.8	
10	Product I	6	12	0	72	
11	Product J	11	35	3.5	346.5	
12						

TASK 4:

As an HR analyst at a company managing employee information, you have been provided with two tables in your Excel workbook: "EmployeeData" and "DepartmentData". The "EmployeeData" table contains details of employees, including their ID, name, and department ID, while the "DepartmentData" table includes information about departments, including their ID and name.

Your task is to perform the following operations:

- 1 Merge the "EmployeeData" and "DepartmentData" tables based on the department ID to create a consolidated dataset.
- 2 Add a new column in the consolidated dataset containing the name of the department for each employee.
- 3 Organize the workbook queries to ensure efficient management.

Employee Data Table:

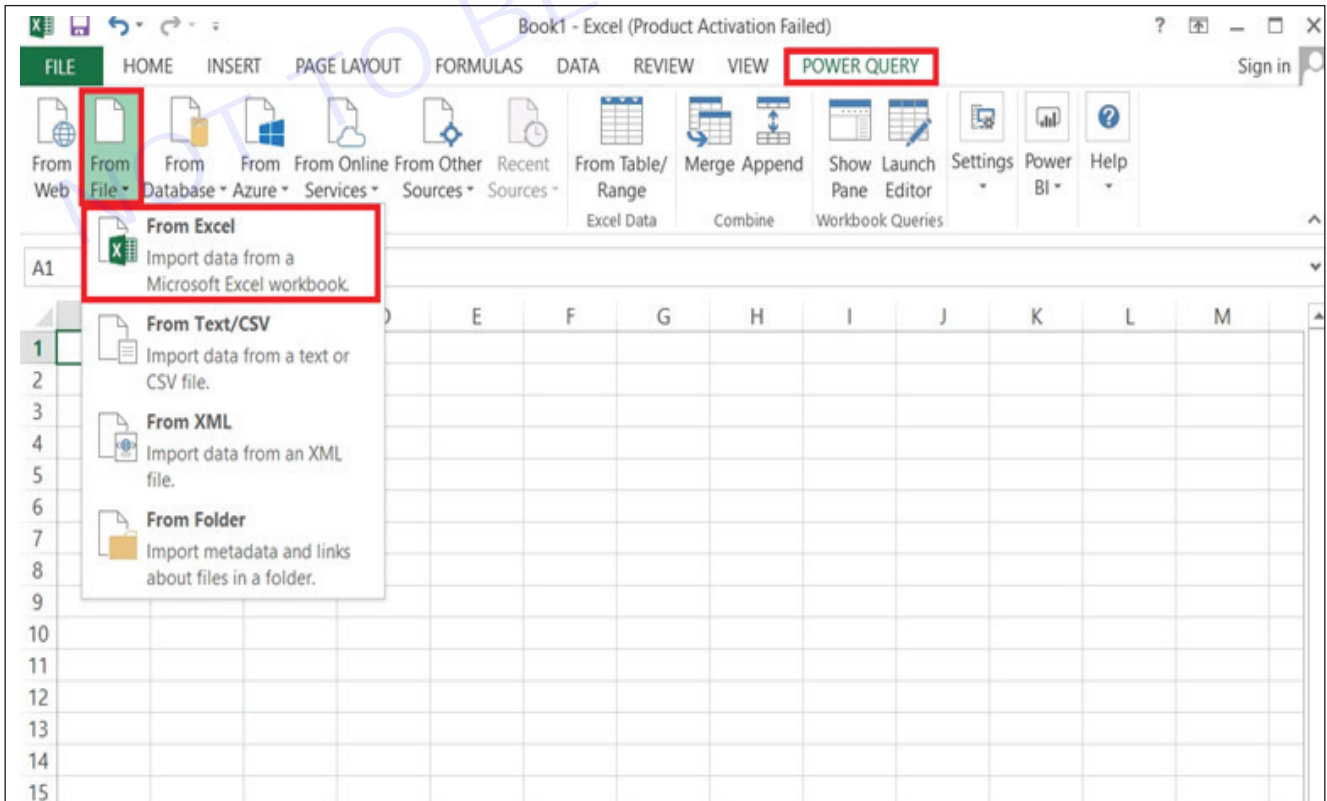
Employee ID	Name	Department ID
001	John Smith	101
002	Emily Brown	102
003	David Lee	101

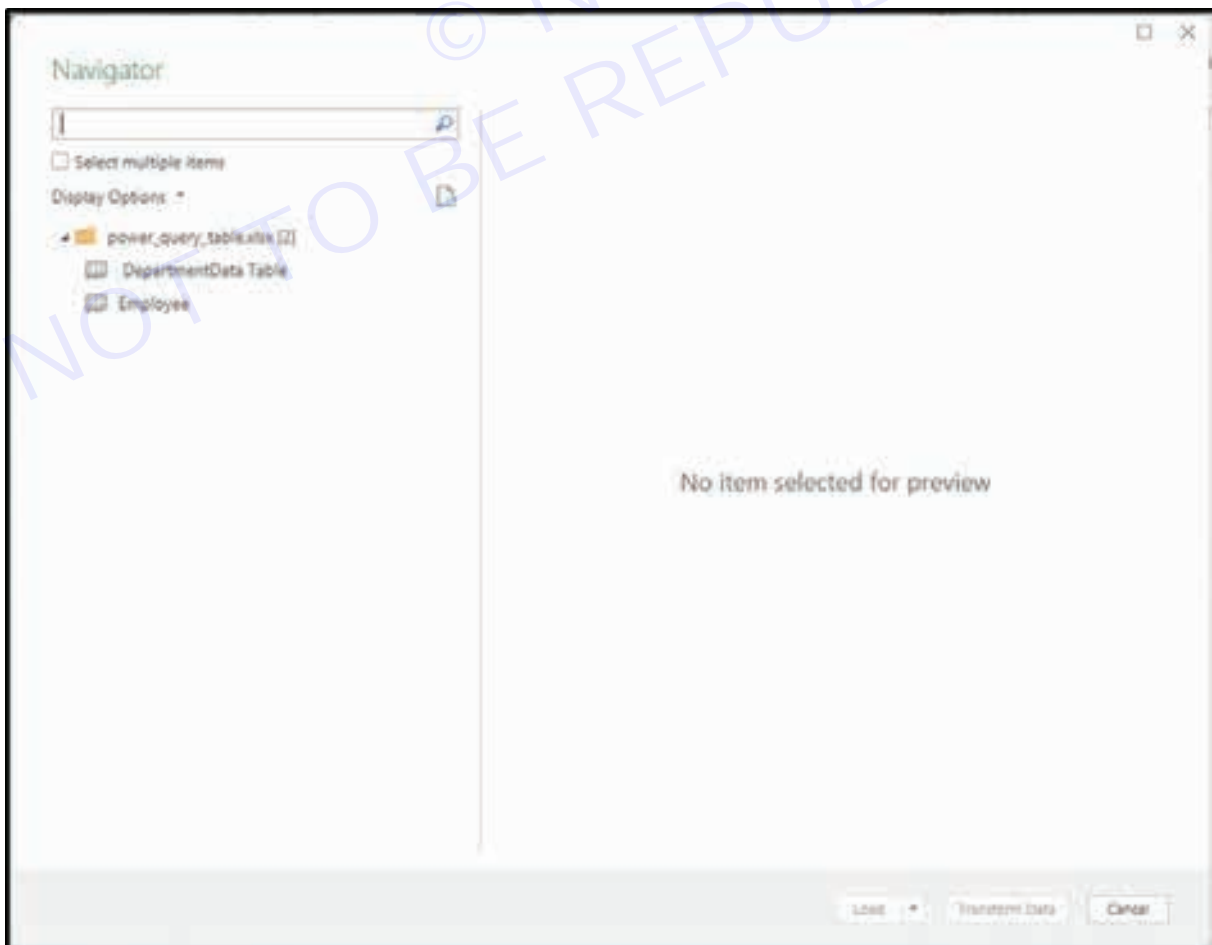
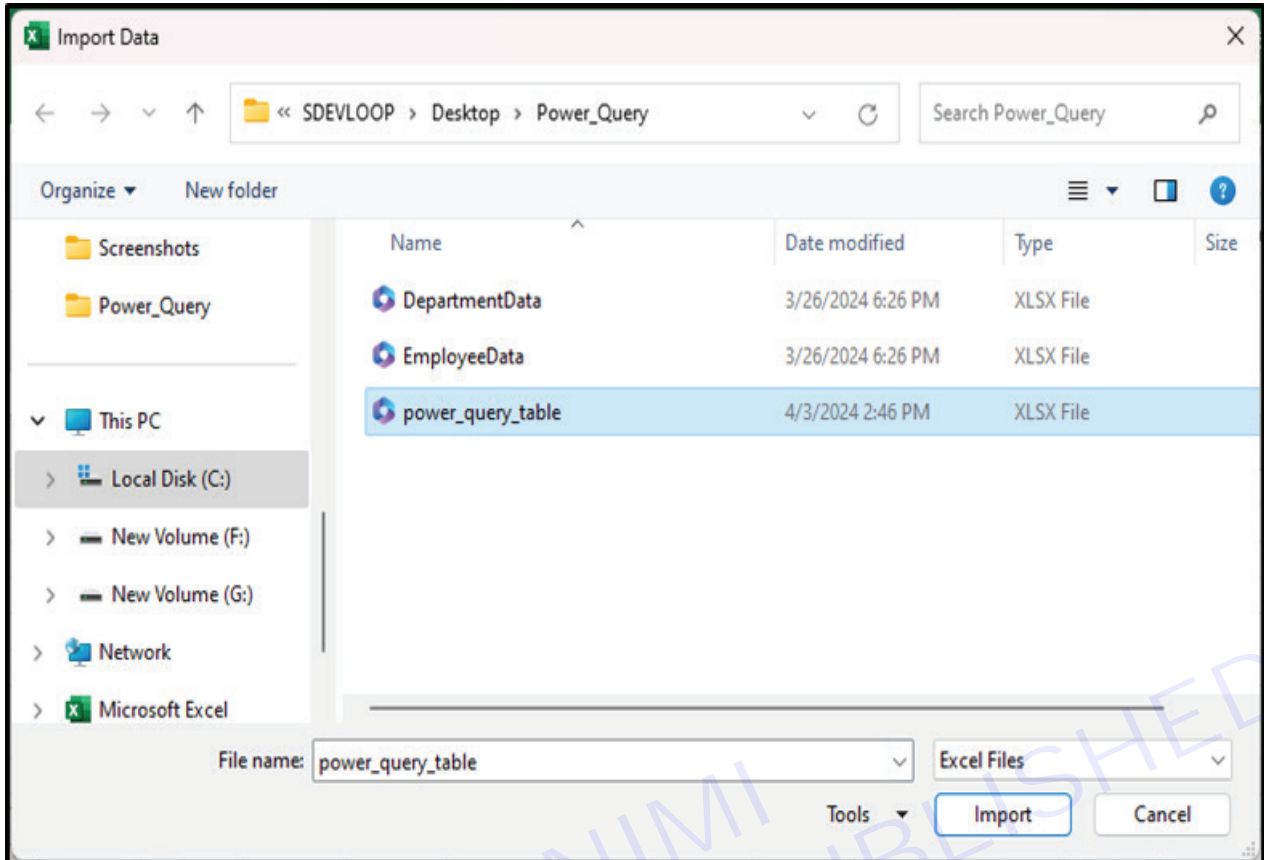
Department Data Table:

Department ID	Department Name
101	HR
102	Finance
103	Marketing

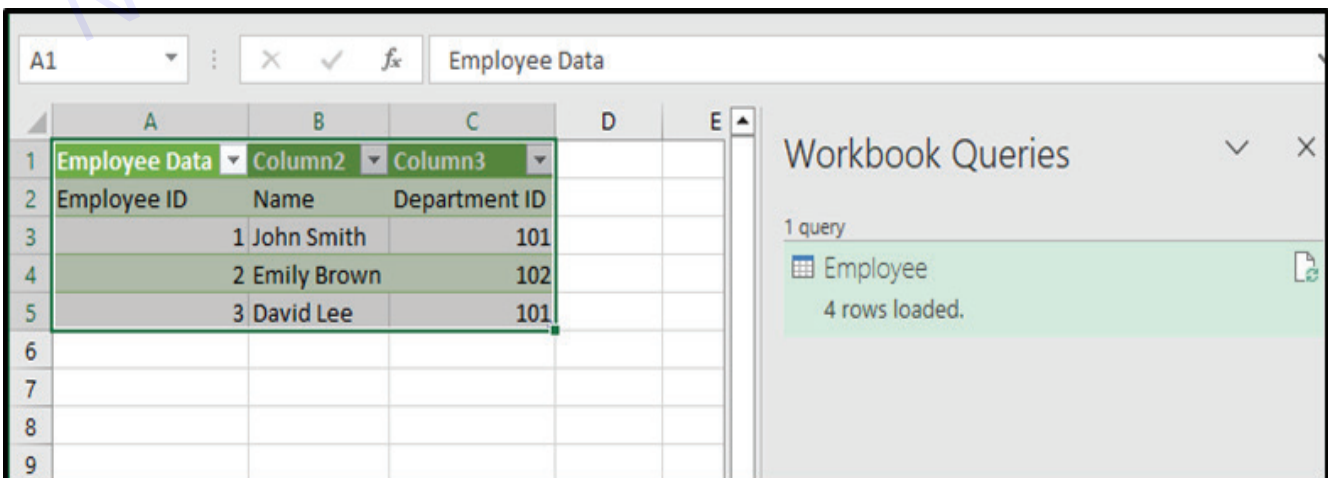
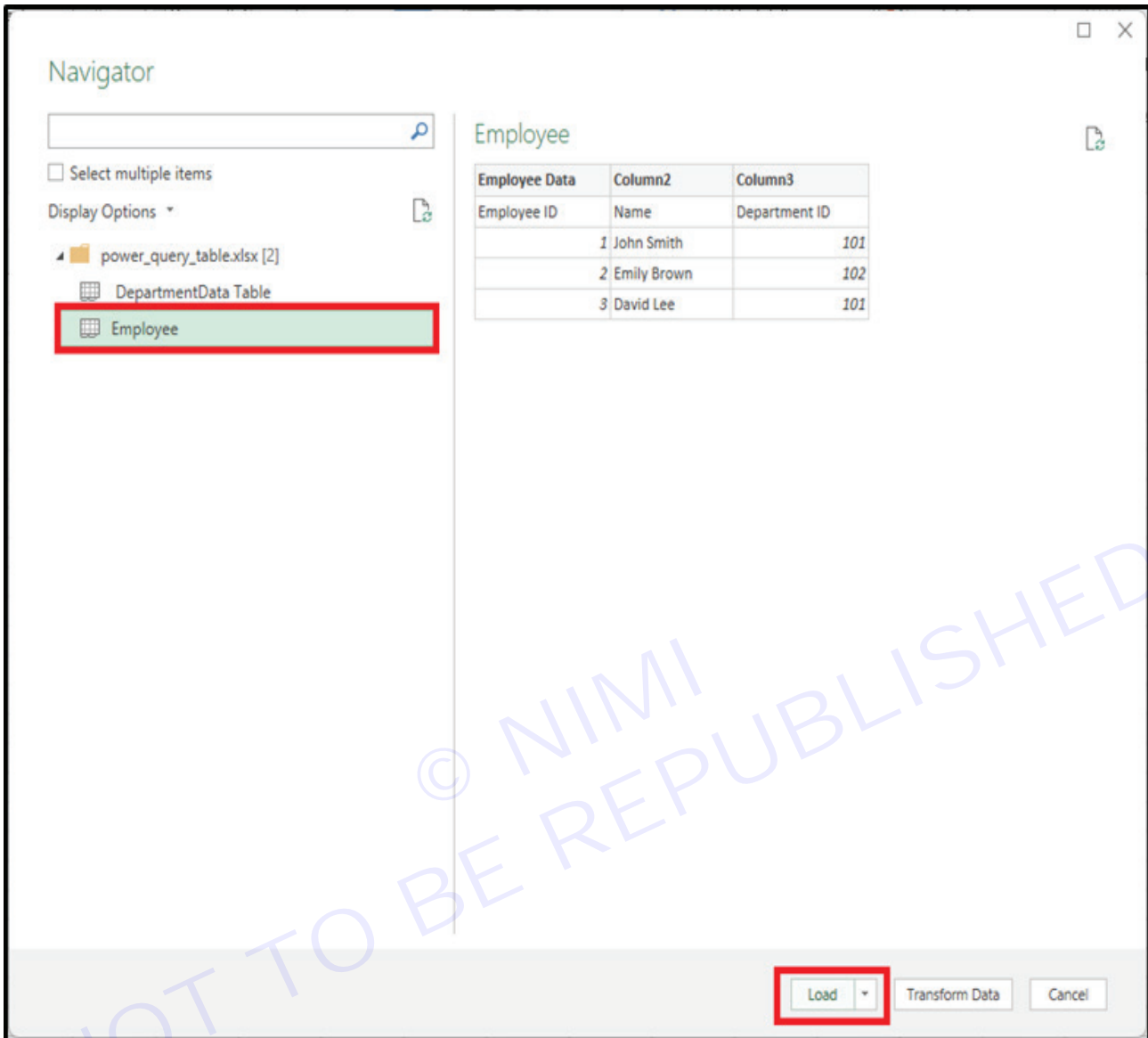
Step 1: Merge Tables

- 1 Open Microsoft Excel and navigate to the "Data" tab.
- 2 Click on "Get Data" > "From File" > "From Workbook" to import both "EmployeeData" and "DepartmentData" tables.



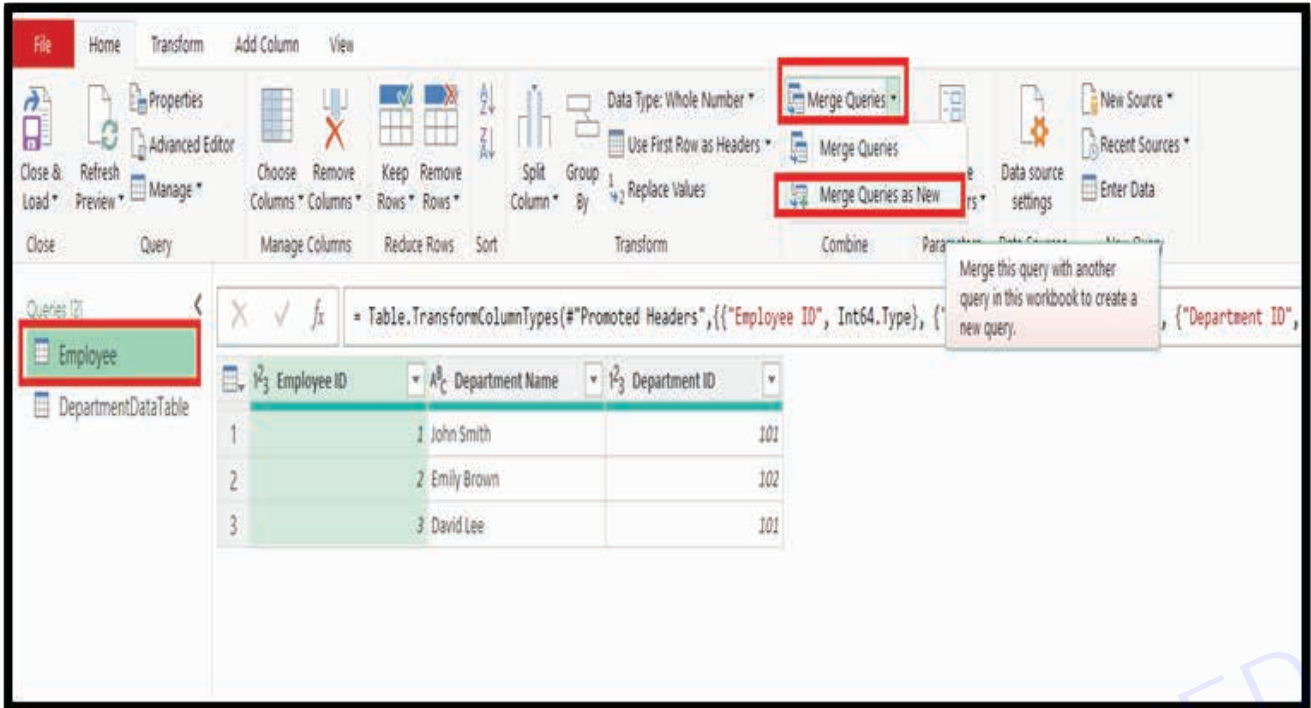


3 In the “Navigator” window, select both tables and click “Load” to load them into the Power Query Editor.

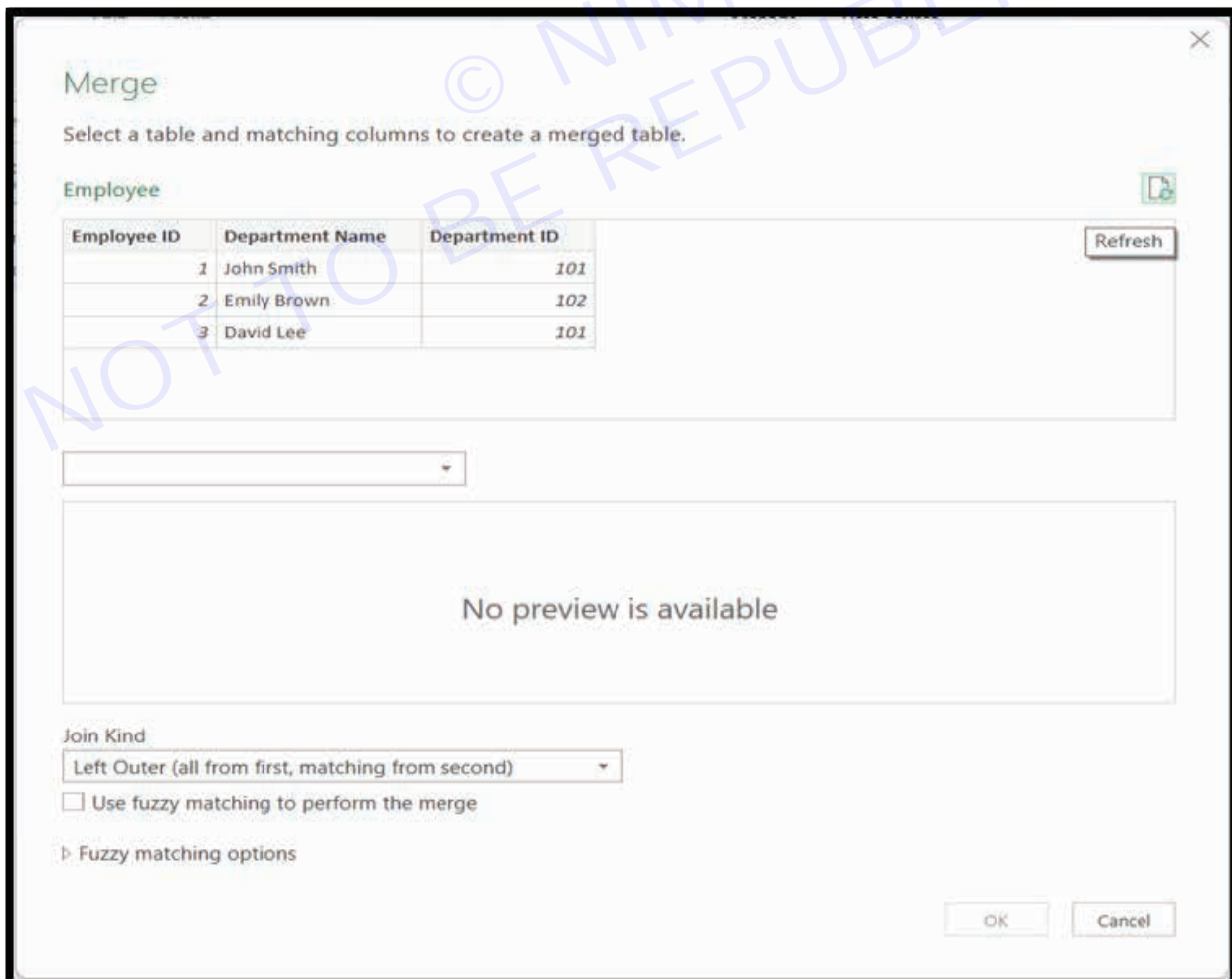


4 In the Power Query Editor, select the “EmployeeData” table.

5 Click on the “Merge Queries” dropdown menu and select “Merge Queries as New”.



- 6 Choose "DepartmentData" from the dropdown list and select "Department ID" as the matching column.
- 7 Select an appropriate join type, such as "Inner Join", and click "OK".



Merge

Select a table and matching columns to create a merged table.

Employee

Employee ID	Department Name	Department ID
1	John Smith	101
2	Emily Brown	102
3	David Lee	101

DepartmentDataTable

Department ID	Department Name
101	HR
102	Finance
103	Marketing

Join Kind
 Left Outer (all from first, matching from second)

Use fuzzy matching to perform the merge

▷ Fuzzy matching options

OK Cancel

Merge

Select a table and matching columns to create a merged table.

Employee

Employee ID	Department Name	Department ID
1	John Smith	101
2	Emily Brown	102
3	David Lee	101

DepartmentDataTable

Department ID	Department Name
101	HR
102	Finance
103	Marketing

Join Kind
 Inner (only matching rows)

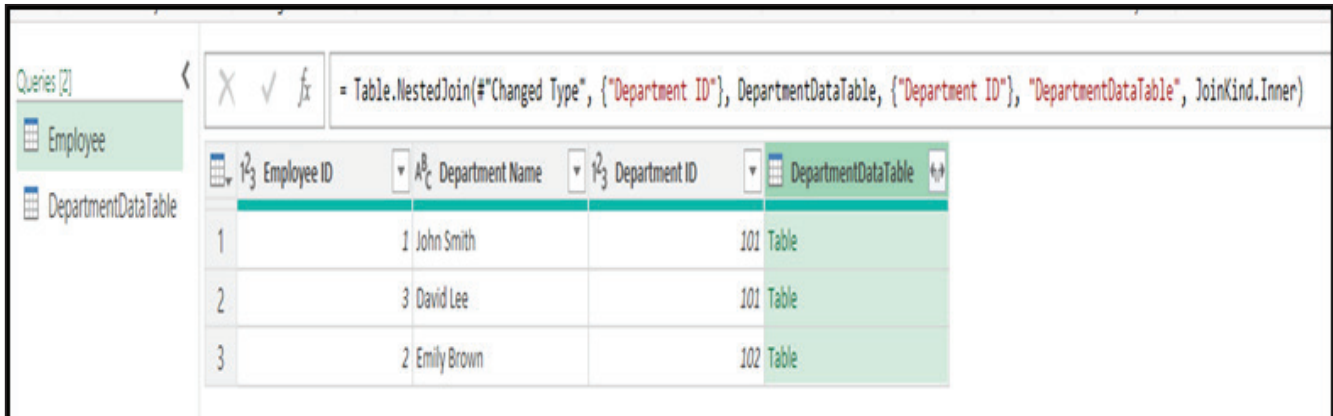
Use fuzzy matching to perform the merge

▷ Fuzzy matching options

✓ The selection matches 3 of 3 rows from the first table, and 2 of 3 rows from...

OK Cancel

8 The tables are merged based on the department ID



Related Exercises:

- 1 Create “Employee Data” table containing information about employees, including their EmployeeID, Name, Department, and Salary. Your task is to perform advanced filtering and parameterization to analyze employee bonuses based on salary levels.

Employee Data:

EmployeeID	Name	Department	Salary
101	John Smith	IT	60000
102	Jane Doe	HR	75000
103	Bob Johnson	Sales	80000
104	Alice Brown	IT	70000
105	Mark White	Sales	90000

TASKS :

- 1 Load the “Employee Data” into Power Query.
- 2 Create a parameter named “MinSalary” that represents the minimum salary for filtering.
- 3 Use the “MinSalary” parameter to dynamically filter the “Employee Data” table, keeping only the rows where the salary is greater than or equal to the parameter value.
- 4 Add a custom column named “Bonus” based on the following conditions:
 - If the salary is above 65000, assign a bonus of 5%.
 - If the salary is between 60000 and 65000, assign a bonus of 3%.
 - Otherwise, assign a bonus of 1%.
- 5 Load the final table into the Excel workbook.

Create “Sales Data” table with sample data, along with a Python script that performs combining and aggregating operations to analyze the total revenue for each product.

Sales Data Table:

OrderID	CustomerID	ProductID	Quantity	Revenue
1	101	201	2	100
2	102	202	1	50
3	103	201	3	150
4	104	203	2	120
5	105	202	1	50

TASKS :

- 1 Load the "Sales Data" into Power Query.
- 2 Combine this table with a reference table containing information about each product, such as the "ProductID" and "ProductCategory." Assume this reference table is named "ProductInfo."
- 3 Aggregate the combined data to calculate the total revenue for each product category.
- 4 Create a new column in the final table named "AverageRevenuePerUnit" that calculates the average revenue per unit sold for each product category.

You have a "Monthly Sales" table containing information about sales data over several months. Your task is to perform time series analysis to understand the trends in monthly sales.

Date	Sales Amount
2022-01-01	10000
2022-02-01	12000
2022-03-01	15000

TASKS :

- 1 Load the "Monthly Sales" into Power Query.
- 2 Create a new column named "Month" that extracts the month and year from the "Date" column.
- 3 Calculate the month-to-month percentage change in sales and add it as a new column.
- 4 Identify any months with significant increases or decreases in sales.

© NIMI
NOT TO BE REPUBLISHED

EXERCISE 77 : Demonstrate Power BI for simple data visualizations

Objectives

At the end of this exercise you shall be able to

- demonstrate power BI for simple data visualizations.

Procedure

Power BI is a powerful business analytics tool developed by Microsoft. It allows users to visualize and analyze data from various sources in interactive reports and dashboards.

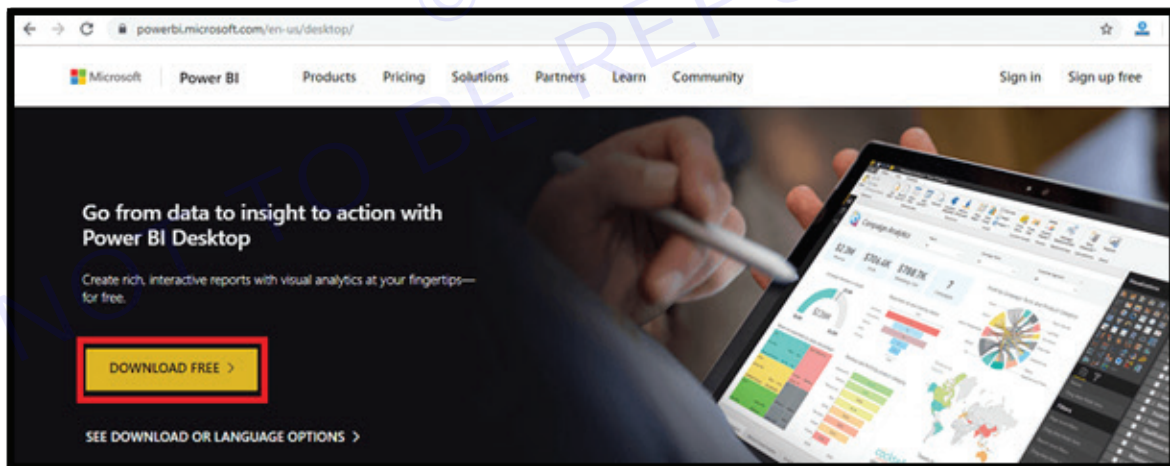
Download and Install Power BI Desktop

Here are some requirements of the system to download the Power BI Desktop:

- Window 7, window 8, window 8.1, window 10, and windows server 2008 R2, windows server 2012, windows server 2012 R2.
- It requires internet explorer 9 or higher.
- Power BI Desktop is available for both 32 bit and 64-bit platforms.
- Let's see the downloading process of the Power BI Desktop step by step:

Step 1: Click on the below link to directly download Power BI Desktop. <https://powerbi.microsoft.com/en-us/desktop/>

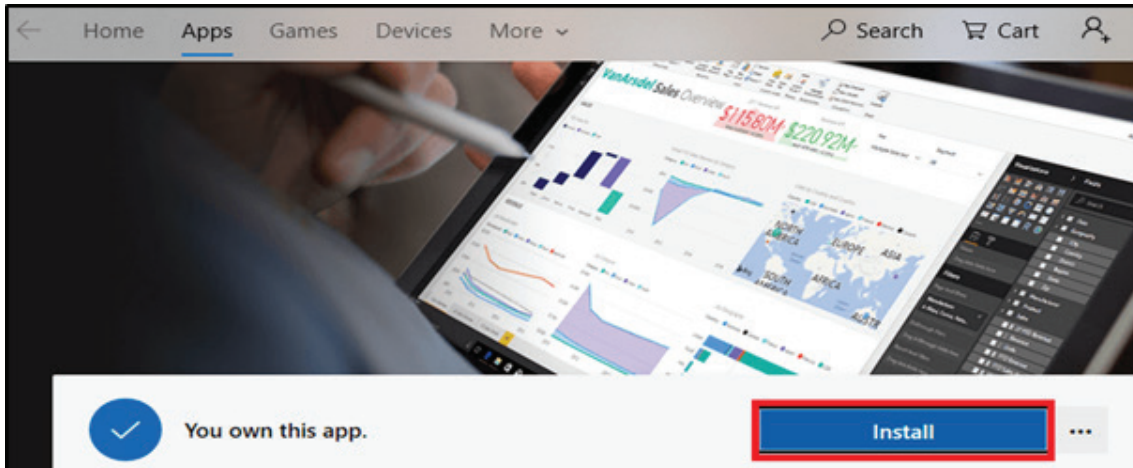
Step 2: Then click on the **Download Free** button.



Step 3: Now, you will redirect to a **Microsoft Store** and then select the **Get** button.



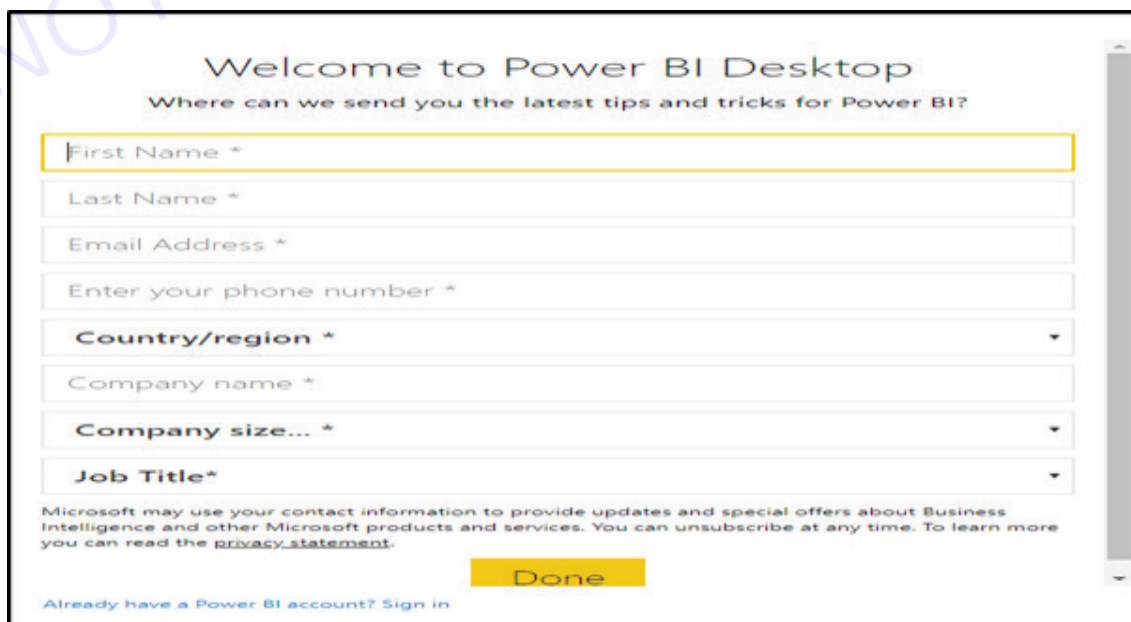
Step 4: Click on the **Install** button.



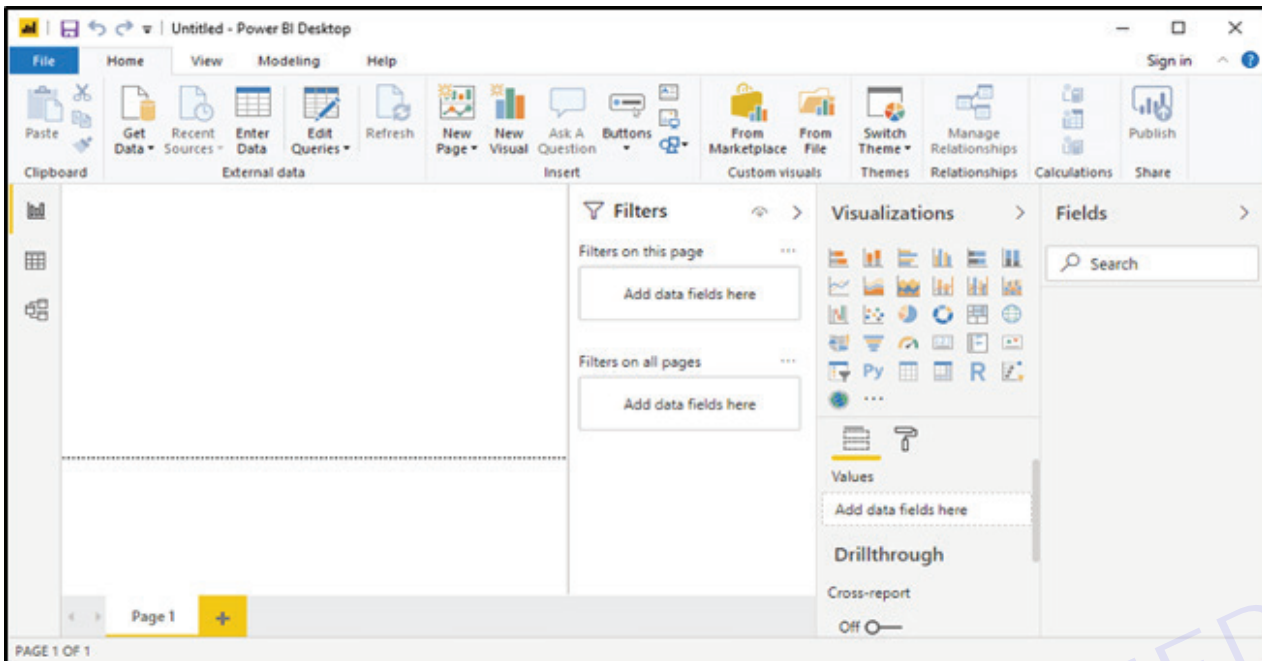
you can see the progress status of the Power BI Desktop on the screen.



Step 5: You can see “welcome to Power BI Desktop” screen and then register yourself on the desktop.



Step 6: When you run the Power BI desktop, it displayed the home page or welcome screen.

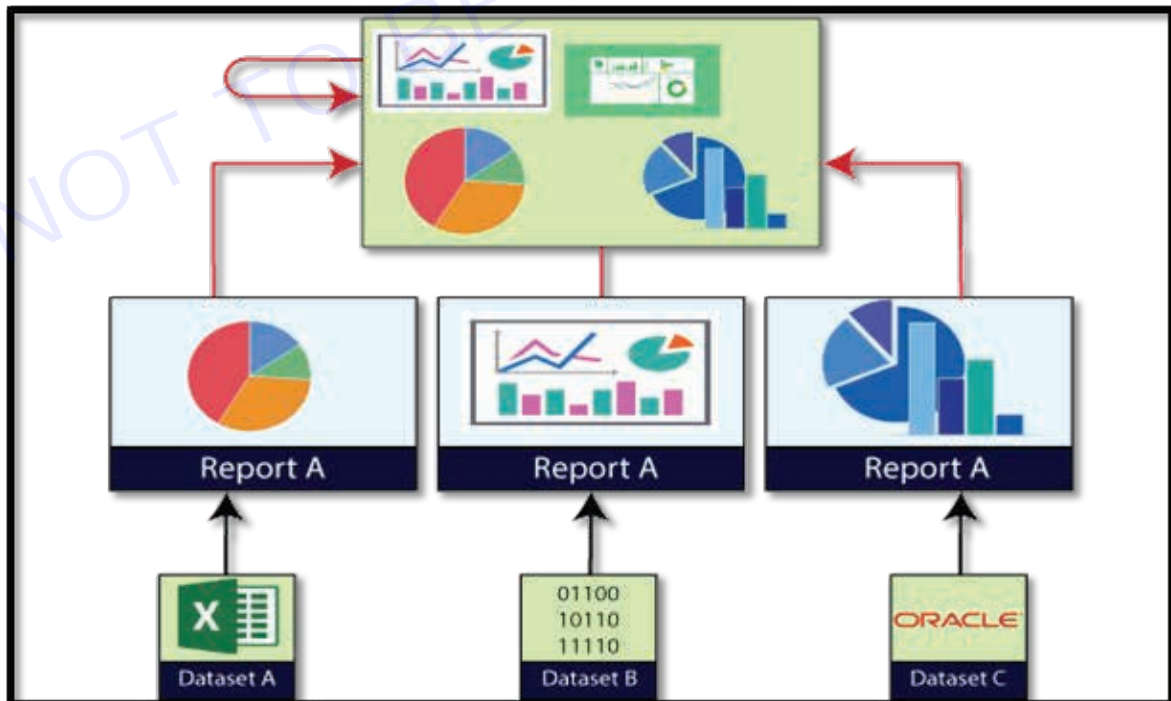


Power BI Dashboard

Power BI dashboard is a single page, also called a canvas that uses visualization to tell the story. It is limited to one page; therefore, a well-designed dashboard contains only the most essential elements of that story.

The visualizations visible on the dashboard are known as tiles. These tiles are pinned to the dashboard from reports. The visualizations on a dashboard come from reports, and each report is based on one data set.

A dashboard can combine on-premises and cloud-born data. And they are providing a consolidated view regardless of where the data lies.



Creating Dashboard in Power BI

We need to import one sample datasets of the Power BI and use it to create a new dashboard.

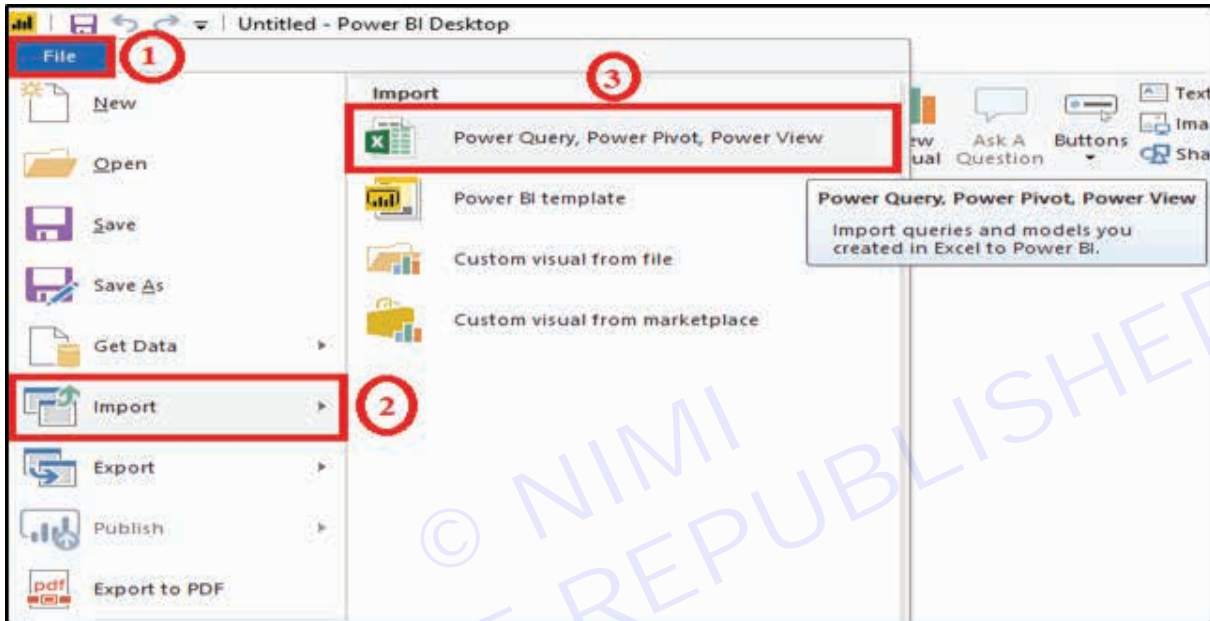
For example, suppose a sample such as **Procurement Analysis**. This sample is an excel workbook with two PowerView sheets.

When Power BI imports the workbook, it adds a dataset and a report to the workspace. Let’s see step by step.

Step 1: Open the Power BI Desktop and click on the **File** pane.

Step 2: Go to the **Import** option.

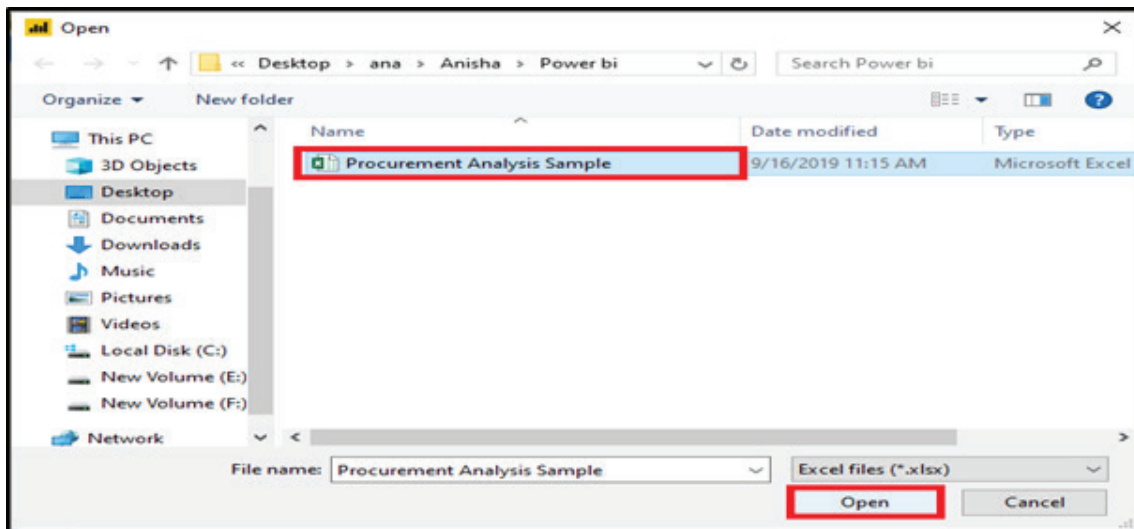
Step 3: And select the Excel dataset **file** to import the file.



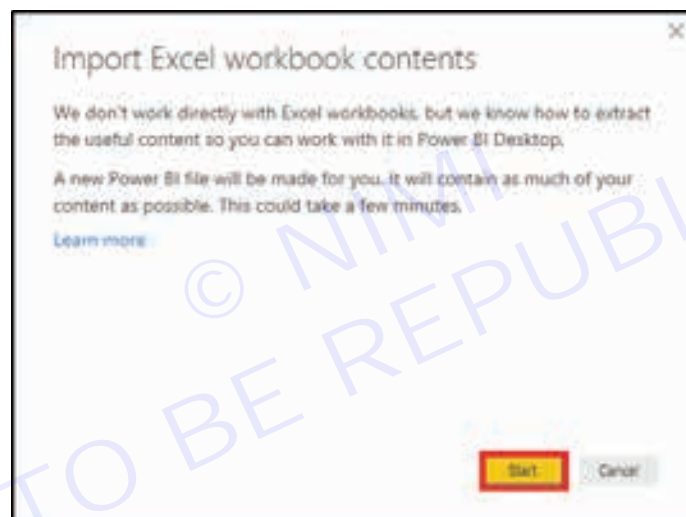
Step 4: Select the **procurement analysis** sample file. Eg: You can select the sample data set from the link <https://www.kaggle.com/datasets>

	A	B	C	D	E	F	G
1	Currency	Exchange Rate	Location	Vendor	Date	Invoice	
2	\$141,604	20	United States	Emily Davis	4/8/2016	E00530	
3	\$99,975	15	China	Theodore Dinh	11/29/1997	E04239	
4	\$163,099	21	United States	Luna Sanders	10/26/2006	E03496	
5	\$84,913	20	United States	Penelope Jordan	9/27/2019	E00549	
6	\$95,409	20	United States	Austin Vo	11/20/1995	E00163	
7	\$50,994	21	China	Joshua Gupta	1/24/2017	E00884	
8	\$119,746	21	United States	Ruby Barnes	7/1/2020	E04116	
9	\$41,336	22	United States	Luke Martin	5/16/2020	E04625	
10	\$113,527	22	United States	Easton Bailey	1/25/2019	E03680	
11	\$77,203	23	United States	Madeline Walker	6/13/2018	E04732	
12	\$157,333	23	United States	Savannah Ali	2/11/2009	E03484	
13	\$109,851	24	United States	Camila Rogers	10/21/2021	E00671	
14	\$105,086	24	United States	Eli Jones	3/14/1999	E02071	
15	\$146,742	25	China	Everleigh Ng	6/10/2021	E02206	
16	\$97,078	25	United States	Robert Yang	11/4/2017	E04545	
17	\$249,270	26	United States	Isabella Xi	3/13/2013	E00154	
18	\$175,837	26	United States	Bella Powell	3/4/2002	E03343	
19	\$154,828	27	United States	Camila Silva	12/1/2003	E00304	
20	\$186,503	27	United States	David Barnes	11/3/2013	E02594	
21	\$166,331	28	China	Adam Dang	7/9/2002	E00402	
22	\$146,140	28	Brazil	Elias Alvarado	1/9/2012	E01994	
23	\$151,703	29	United States	Eva Rivera	4/2/2021	E03549	
24	\$172,787	29	Brazil	Logan Rivera	5/24/2002	E03247	
25	\$49,998	30	United States	Leonardo Dixon	9/5/2019	E02074	
26	\$207,172	30	China	Mateo Her	3/2/2014	E04152	

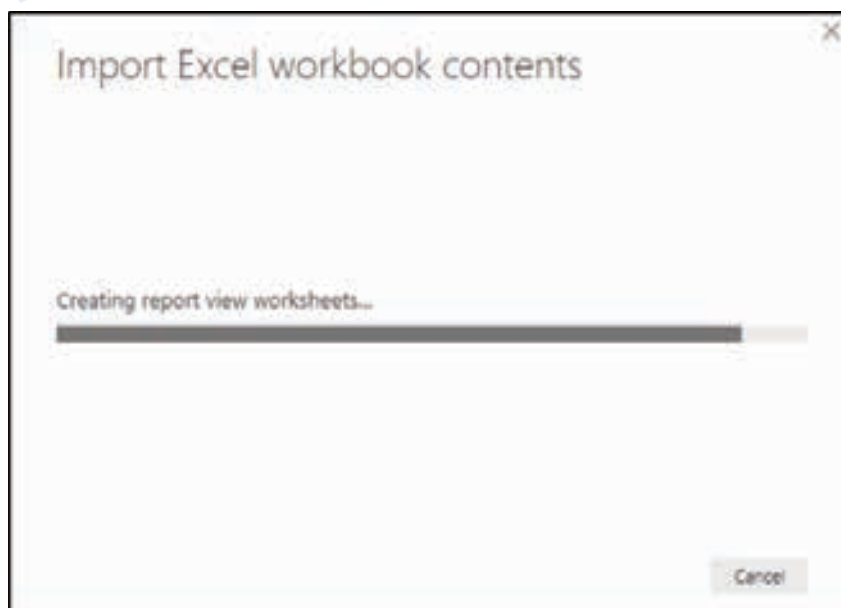
Step 5: And click on the **Open** button.



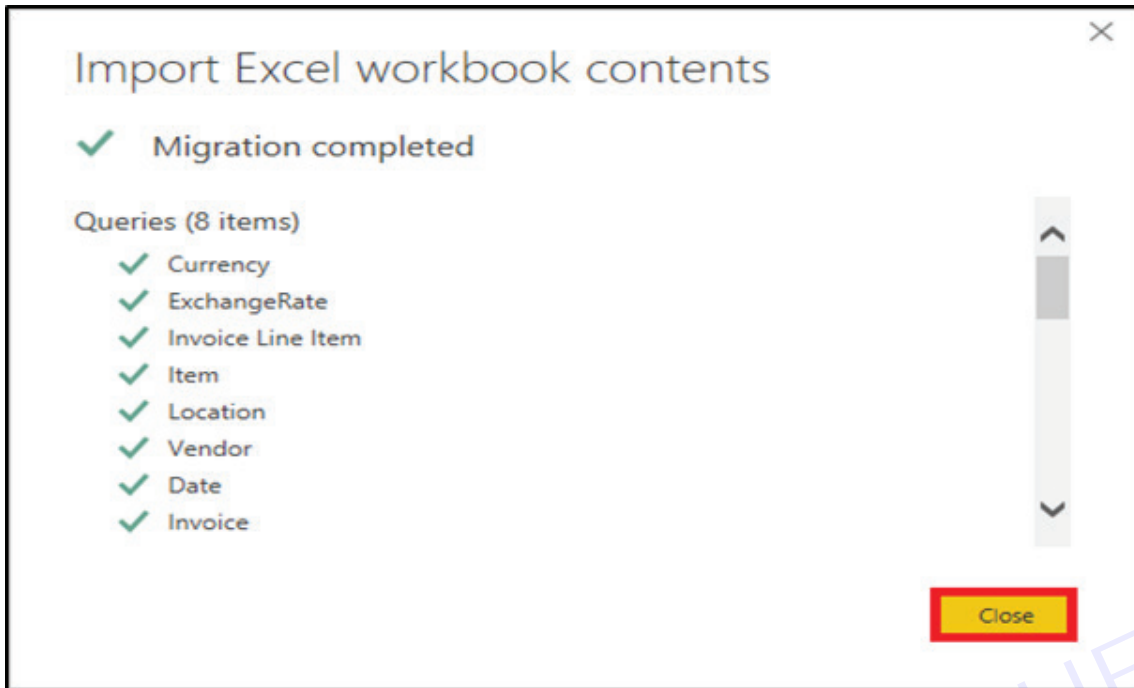
Step 6: For the exercise, select the **Start** button.



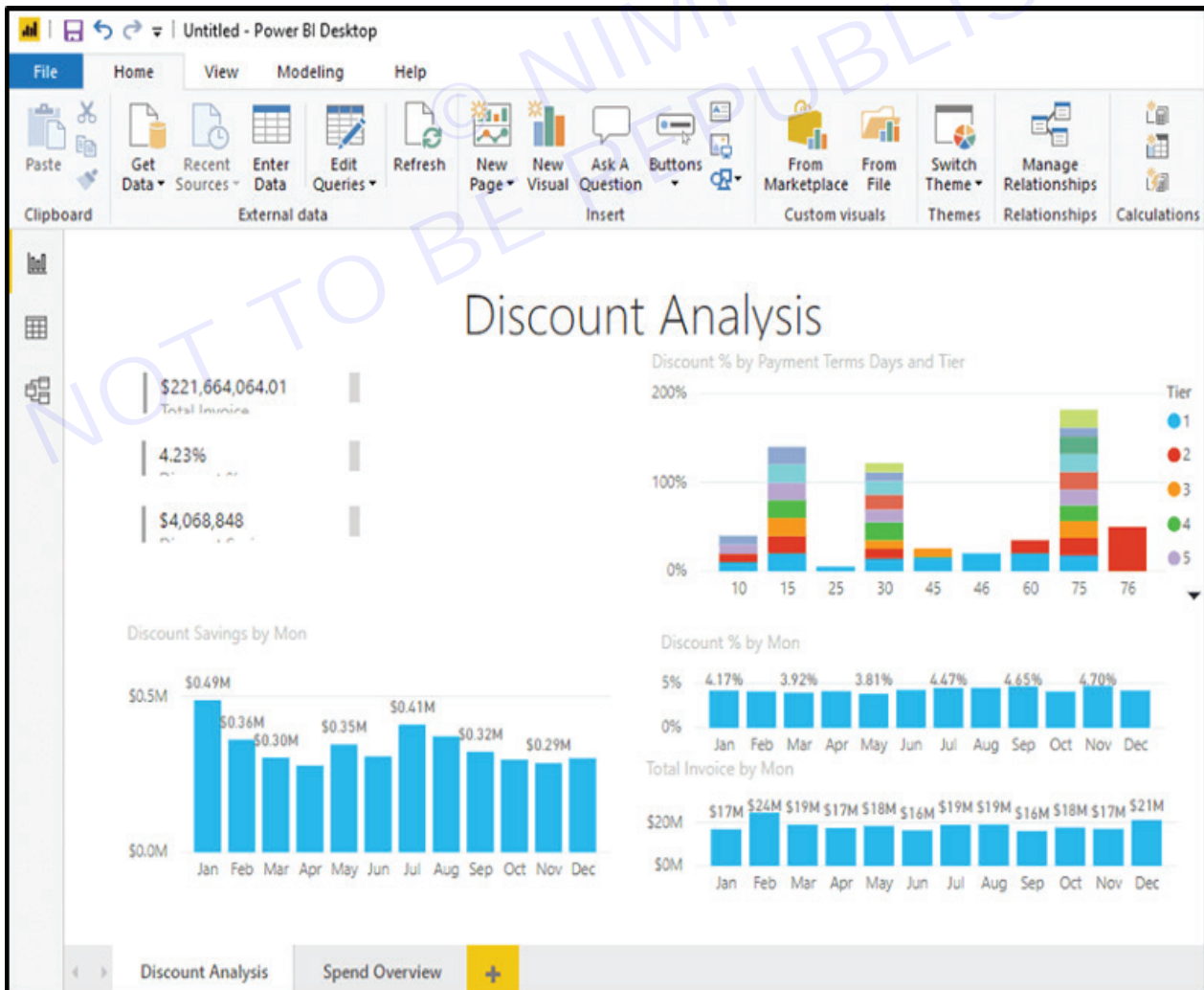
It starts import excel workbook and creating report view worksheets shown in the below screenshot.



Step 7: When the **completed** message appears, then select the **Close** button to dismiss it.



In the below screenshot, you can see the discount analysis of the imported dataset in the form of tiles.

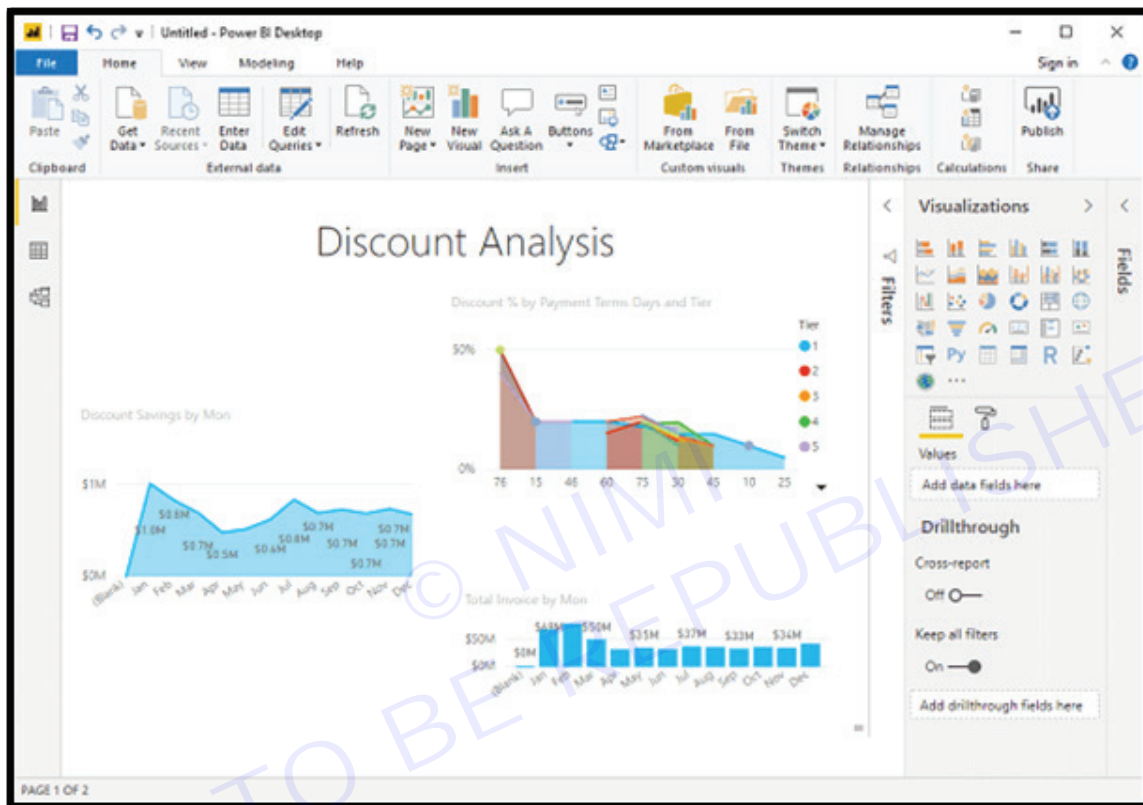


Power BI Reports

A Power BI **report** is a multi-perspective view into the dataset, with visualizations which represent different findings and insights from that dataset.

A report can have a single visualization or multiple visualizations. The visualizations in a report represent something like a dashboard does but serve a different purpose.

These visualizations are not static. These are highly interactive & highly customizable visualizations which update, as the underlying data changes. You can add and remove the data, change visualization types, and apply filters in your model to discover insights.



Related Exercises:

Simple Data Visualizations in Power BI

Scenario:

You have been provided with a dataset containing information about monthly sales for a retail business. The dataset includes columns such as Date, Product Category, Sales Amount, and Region.

TASKS:

1 Import Data:

- Load the provided dataset into Power BI.

2 Data Cleaning and Transformation:

- Perform any necessary data cleaning and transformation steps to ensure the data is suitable for analysis.

3 Create Visualizations:

- Design the following visualizations:
 - **Line Chart:** Display the trend of total sales over different months.

- **Bar Chart:** Compare sales amounts for different product categories.
- **Map:** Show the geographical distribution of sales using the Region information.

4 Implement Slicers:

- Add slicers to enable dynamic filtering. For example, create a slicer for the date range to view sales for a specific period.

5 Calculate Key Metrics:

- Create new calculated columns or measures to calculate important metrics, such as monthly growth rate or total sales.

6 Dashboard Creation:

- Assemble the visualizations on a Power BI dashboard for a comprehensive overview.

7 Interactive Elements:

- Implement interactive elements such as tooltips, drill-throughs, or bookmarks to enhance user experience.

Dataset Example:

Date	Product Category	Sales Amount	Region
2022-01-01	Electronics	10000	North
2022-02-01	Clothing	12000	South
2022-03-01	Home Goods	15000	East

© NIMI
NOT TO BE REPUBLISHED