

Excel Assignment - 21

- 1. Write a VBA code to enter your name in A1 Cell using Input Box and once you enter the name display a message box that says the name has been entered.**
- 2. What are Userforms? Why are they used? How to fill a list box using for loop.**
- 3. What is an array? Write a VBA code to enter students and their marks from the below table.**
- 4. Use the following data to create a pie chart using VBA code. Use Font - 'Times new Roman', Size -14, Bold, Title - Piechart' and you are per to use colours as per your taste.**
- 5. Check the dataset in the link given below and create a pivot table using VBA showing the sales for the year from stationary category.**

<https://docs.google.com/spreadsheets/d/1IRSEnmgz8Ro276-GslknRNk0zlrB5CZH1YrnT71kqFM/edit?usp=sharing>

6. Write step by step procedure to protect your workbook using a password.

ANSWERS;

1. Enter Name in A1 Cell and Display Message:

```
Sub EnterNameAndDisplayMessage()
```

```
    Dim userName As String
```

```
    ' Prompt for user input using Input Box
```

```
    userName = InputBox("Enter your name:", "Enter Name")
```

```
    ' Check if the user entered a name
```

```
    If userName <> "" Then
```

```
' Enter the name in cell A1
```

```
Range("A1").Value = userName
```

```
' Display a message box
```

```
MsgBox "Name has been entered: " & userName
```

```
Else
```

```
' Display a message if no name is entered
```

```
MsgBox "No name entered. Please try again."
```

```
End If
```

```
End Sub
```

2. UserForms and Filling a ListBox Using For Loop:

- **UserForms:** UserForms in VBA are custom dialog boxes that allow you to create interactive forms for data input or display.

```
Sub FillListBoxWithForLoop()  
    ' Create a UserForm  
    Dim myForm As Object  
    Set myForm = VBA.UserForms.Add
```

```

' Add a ListBox to the UserForm
Dim myListbox As MSForms.ListBox
Set myListbox = myForm.Controls.Add("Forms.ListBox.1")

' Set properties for the ListBox
With myListbox
    .ColumnCount = 1 ' Number of columns in the ListBox
    .ColumnWidths = "100" ' Width of columns in the ListBox
    .RowSource = "" ' Clear any existing data source
End With

' Fill ListBox using For Loop
Dim i As Integer
For i = 1 To 10
    myListbox.AddItem "Item " & i
Next i

' Show the UserForm
myForm.Show
End Sub

```

3. Array and Entering Students and Marks:

- **Array:** An array is a collection of variables of the same type.

```

Sub EnterStudentsAndMarks()
' Declare an array to store student names and marks
Dim students(1 To 3, 1 To 2) As Variant
Dim i As Integer

' Enter data into the array
students(1, 1) = "Student1"
students(1, 2) = 90
students(2, 1) = "Student2"
students(2, 2) = 85
students(3, 1) = "Student3"
students(3, 2) = 92

```

```

' Enter array data into the worksheet
For i = 1 To 3
    Cells(i, 1).Value = students(i, 1) ' Student Name
    Cells(i, 2).Value = students(i, 2) ' Marks
Next i
End Sub

```

4. Create a Pie Chart:

```

Sub CreatePieChart()
    ' Create a new chart sheet
    Charts.Add
    ActiveChart.ChartType = xlPie

    ' Set chart properties
    With ActiveChart
        .SetElement (msoElementChartTitleAboveChart)
        .ChartTitle.Text = "Pie Chart"
        .ChartTitle.Font.Size = 14
        .ChartTitle.Font.Name = "Times New Roman"
        .ChartTitle.Font.Bold = True
    End With

    ' Modify the chart colors as per your taste
    ActiveChart.SeriesCollection(1).Points(1).Interior.Color = RGB(255, 0, 0)
    ActiveChart.SeriesCollection(1).Points(2).Interior.Color = RGB(0, 255, 0)
    ' Add more color modifications as needed
End Sub

```

5. Create Pivot Table from External Data:

- Import the necessary library references in the VBA editor:

- Microsoft ActiveX Data Objects Library (for database connectivity).
- Microsoft Scripting Runtime (for working with files and folders).

- Use the following VBA code:

- Sub CreatePivotTableFromExternalData()
- ' Set up connection to Google Sheets using the provided link
- Dim conn As Object
- Set conn = CreateObject("ADODB.Connection")
- conn.Open "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=https://docs.google.com/spreadsheets/d/1IRSEnmgz8Ro276-GslknRNk0zlrB5CZH1YrnT71kqFM/edit?usp=sharing;Extended Properties='\"Excel 12.0 Xml;HDR=YES\"';"
-
- ' Create a Pivot Table
- Dim ws As Worksheet
- Set ws = Sheets.Add
- ActiveWorkbook.PivotTableWizard SourceType:=xlExternal, SourceData:=conn.Execute("SELECT * FROM [Sheet1\$]").GetRows, TableDestination:=ws.Range("A1")
-
- ' Set up Pivot Table fields
- With ws.PivotTables(1).PivotFields("Category")
- .Orientation = xlRowField
- .Position = 1
- End With
-
- With ws.PivotTables(1).PivotFields("Year")
- .Orientation = xlColumnField
- .Position = 1
- End With

-
- With ws.PivotTables(1).PivotFields("Sales")
- .Orientation = xlDataField
- .Function = xlSum
- .NumberFormat = "#,##0.00"
- End With
- End Sub

6. Protect Workbook Using Password:

1. Open the Excel workbook you want to protect.
2. Press **ALT + F11** to open the VBA editor.
3. Insert a new module (**Insert > Module**) and paste the following code:

```
Sub ProtectWorkbookWithPassword()

    ' Specify your desired password

    Dim password As String

    password = "YourPasswordHere"

    ' Protect the workbook with a password

    ThisWorkbook.Protect Password:=password

End Sub
```

1. Close the VBA editor.
2. Run the macro by pressing **ALT + F8**, select **ProtectWorkbookWithPassword**, and click **Run**.

