### **1. What is the VBA code to enter your name in A1 Cell using Input Box and display a message box confirming the entry?**

vba

Copy code

Sub EnterNameAndDisplayMessage()

Dim userName As String

' Prompt user to enter name using Input Box

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

' Check if user entered a name

If userName <> "" Then

' Enter name in cell A1

Range("A1").Value = userName

' Display message box confirming name entry

MsgBox "Name '" & userName & "' has been entered."

Else

MsgBox "No name entered. Please try again."

End If

End Sub

### **2. What are UserForms in VBA? Why are they used? How do you fill a ListBox using a For loop?**

**UserForms** in VBA are custom dialog boxes used to create graphical interfaces for users. They serve to gather input, display information, and interact with users more effectively than standard message boxes or input prompts.

To fill a ListBox using a For loop:

vba

Copy code

Sub FillListBoxWithForLoop()

Dim i As Integer

Dim items As Variant

' Example items to add to ListBox

items = Array("Item 1", "Item 2", "Item 3", "Item 4", "Item 5")

' Clear existing ListBox contents

UserForm1.ListBox1.Clear

' Add items to ListBox using a For loop

For i = LBound(items) To UBound(items)

UserForm1.ListBox1.AddItem items(i)

Next i

' Show the UserForm

UserForm1.Show

End Sub

### **3. What is an array? How do you use an array in VBA to enter students and their marks from a table?**

An **array** in VBA is a collection of variables of the same type stored under a single variable name. It allows you to store multiple values of the same data type sequentially.

vba

Copy code

Sub EnterStudentsAndMarks()

Dim students(1 To 5, 1 To 2) As Variant ' Assuming 5 students and 2 columns (Name and Marks)

Dim i As Integer

' Example data

students(1, 1) = "John"

students(1, 2) = 85

students(2, 1) = "Alice"

students(2, 2) = 92

students(3, 1) = "Bob"

students(3, 2) = 78

students(4, 1) = "Sarah"

students(4, 2) = 95

students(5, 1) = "Michael"

students(5, 2) = 88

' Enter data into worksheet

For i = 1 To 5

Range("A" & i).Value = students(i, 1) ' Name

Range("B" & i).Value = students(i, 2) ' Marks

Next i

End Sub

### **4. How do you create a pie chart using VBA with specific formatting and colors?**

vba

Copy code

Sub CreatePieChart()

Dim ws As Worksheet

Dim chartObj As ChartObject

Dim rngData As Range

' Set worksheet

Set ws = ThisWorkbook.Sheets("Sheet1") ' Replace with your sheet name

' Define data range

Set rngData = ws.Range("A1:B5") ' Assuming data range

' Add a new chart object

Set chartObj = ws.ChartObjects.Add(Left:=100, Width:=375, Top:=75, Height:=225)

' Specify chart type and data

With chartObj.Chart

.ChartType = xlPie

.SetSourceData Source:=rngData

' Chart title formatting

.HasTitle = True

.ChartTitle.Text = "Pie Chart"

.ChartTitle.Font.Name = "Times New Roman"

.ChartTitle.Font.Size = 14

.ChartTitle.Font.Bold = True

' Series formatting (colors)

.SeriesCollection(1).Points(1).Format.Fill.ForeColor.RGB = RGB(255, 0, 0) ' Red

.SeriesCollection(1).Points(2).Format.Fill.ForeColor.RGB = RGB(0, 255, 0) ' Green

' Add more colors as needed

End With

End Sub

### **5. How do you create a pivot table using VBA to show sales for the year from the stationary category?**

vba

Sub CreatePivotTable()

Dim ws As Worksheet

Dim pt As PivotTable

Dim ptField As PivotField

Dim dataRange As Range

' Set worksheet

Set ws = ThisWorkbook.Sheets("Sheet1") ' Replace with your sheet name

' Define data range (assuming data starts from A1)

Set dataRange = ws.Range("A1").CurrentRegion

' Add pivot table to new worksheet

Set pt = ws.PivotTableWizard(SourceType:=xlDatabase, SourceData:=dataRange, TableDestination:=ws.Range("D1"), TableName:="SalesPivotTable")

' Add fields to pivot table

Set ptField = pt.PivotFields("Category")

ptField.Orientation = xlRowField

Set ptField = pt.PivotFields("Year")

ptField.Orientation = xlColumnField

Set ptField = pt.PivotFields("Sales")

ptField.Orientation = xlDataField

' Filter for stationary category

pt.PivotFields("Category").PivotFilters.Add Type:=xlCaptionEquals, Value1:="Stationary"

End Sub

### **6. Step-by-step procedure to protect your workbook using a password in VBA.**

1. **Open VBA Editor**: Press Alt + F11 to open the VBA Editor.
2. **Insert Module**: Insert a new module (Insert > Module).
3. **Enter Protection Code**: Enter the following code to protect the workbook with a password (replace "YourPassword" with your desired password):

vba

Copy code

Sub ProtectWorkbookWithPassword()

ThisWorkbook.Protect Password:="YourPassword"

End Sub

1. **Run the Code**: Run the macro by pressing F5 or Run > Run Sub/UserForm in the VBA Editor.

This protects the entire workbook with the specified password, limiting access to the workbook's structure and windows. Remember to store your password securely as it cannot be recovered if forgotten.