|  |  |
| --- | --- |
| Excel file name | EmployeeData\_LoopTraining |
|  |  |
|  |  |
| FOR Loop – Add Bonus to All Active Employees | Sub AddBonusUsingForLoop()  Dim i As Integer  For i = 2 To 101 ' Assuming headers in row 1 and 100 records  If Cells(i, 5).Value = "Active" Then  Cells(i, 4).Value = Cells(i, 4).Value + 2000  End If  Next i  End Sub |
| FOR EACH Loop – Highlight Resigned Employees | Sub HighlightResignedUsingForEach()  Dim cell As Range  For Each cell In Range("E2:E101")  If cell.Value = "Resigned" Then  cell.EntireRow.Interior.Color = RGB(255, 204, 204)  End If  Next cell  End Sub |
| DO WHILE Loop – Count Total Resigned | Sub CountResignedDoWhile()  Dim i As Integer  Dim count As Integer  i = 2  count = 0  Do While Cells(i, 1).Value <> ""  If Cells(i, 5).Value = "Resigned" Then  count = count + 1  End If  i = i + 1  Loop  MsgBox "Total Resigned Employees: " & count  End Sub |
| DO UNTIL Loop – Stop at First Finance Resignation | Sub FindFirstFinanceResigned()  Dim i As Integer  i = 2  Do Until Cells(i, 1).Value = ""  If Cells(i, 3).Value = "Finance" And Cells(i, 5).Value = "Resigned" Then  MsgBox "First Finance resigned employee: " & Cells(i, 2).Value  Exit Do  End If  i = i + 1  Loop  End Sub |
|  |  |
| File name | VBA\_Conditional\_IF\_Training |
| Conditional Statement using IF | Sub CategorizePerformance()  Dim ws As Worksheet  Dim lastRow As Long  Dim i As Long  Set ws = ThisWorkbook.Sheets("Sheet1")    ' Add header  ws.Cells(1, 5).Value = "Performance Category"    ' Find last row with data  lastRow = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row    ' Loop through records  For i = 2 To lastRow  If ws.Cells(i, 4).Value >= 85 Then  ws.Cells(i, 5).Value = "Excellent"  ElseIf ws.Cells(i, 4).Value >= 70 Then  ws.Cells(i, 5).Value = "Good"  ElseIf ws.Cells(i, 4).Value >= 50 Then  ws.Cells(i, 5).Value = "Average"  Else  ws.Cells(i, 5).Value = "Needs Improvement"  End If  Next i  End Sub |
| Erorr handling | Sub CategorizePerformanceWithErrorHandling()  Dim ws As Worksheet  Dim lastRow As Long  Dim i As Long  Dim score As Variant  Set ws = ThisWorkbook.Sheets("Sheet1")    ws.Cells(1, 5).Value = "Performance Category"  lastRow = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row    For i = 2 To lastRow  On Error GoTo ErrorHandler    score = ws.Cells(i, 4).Value    ' Check if score is numeric  If Not IsNumeric(score) Then GoTo ErrorHandler    ' Apply IF logic  If score >= 85 Then  ws.Cells(i, 5).Value = "Excellent"  ElseIf score >= 70 Then  ws.Cells(i, 5).Value = "Good"  ElseIf score >= 50 Then  ws.Cells(i, 5).Value = "Average"  Else  ws.Cells(i, 5).Value = "Needs Improvement"  End If    GoTo ContinueLoop    ErrorHandler:  ws.Cells(i, 5).Value = "Error in score"  Resume ContinueLoop    ContinueLoop:  Next i  MsgBox "Performance categorization completed!", vbInformation  End Sub |
|  |  |
| Using Formulae in VBA | Sub WriteFormulaInVBA()  Dim ws As Worksheet  Dim lastRow As Long  Dim i As Long  Set ws = ThisWorkbook.Sheets("Sheet1")  lastRow = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row    ' Header  ws.Cells(1, 6).Value = "Formula Result"  ' Write formula in each row of column F  For i = 2 To lastRow  ws.Cells(i, 6).Formula = "=IF(D" & i & ">=85,""Excellent"",IF(D" & i & ">=70,""Good"",IF(D" & i & ">=50,""Average"",""Needs Improvement"")))"  Next i  MsgBox "Formulas added in Column F", vbInformation  End Sub |
|  |  |
| Arrays | Sub FilterHighScorersToNewSheet()  Dim wsSource As Worksheet, wsTarget As Worksheet  Dim scoreData As Variant  Dim outputData() As Variant  Dim i As Long, j As Long  Dim lastRow As Long  ' Set source sheet  Set wsSource = ThisWorkbook.Sheets("Sheet1")  lastRow = wsSource.Cells(wsSource.Rows.Count, 1).End(xlUp).Row    ' Read data (columns A to D)  scoreData = wsSource.Range("A2:D" & lastRow).Value    ' Initialize output array with same columns, max rows = input size  ReDim outputData(1 To UBound(scoreData), 1 To 4)  j = 1 ' Row counter for output    ' Filter high scorers  For i = 1 To UBound(scoreData)  If IsNumeric(scoreData(i, 4)) And scoreData(i, 4) > 85 Then  outputData(j, 1) = scoreData(i, 1) ' EmpID  outputData(j, 2) = scoreData(i, 2) ' Name  outputData(j, 3) = scoreData(i, 3) ' Department  outputData(j, 4) = scoreData(i, 4) ' Score  j = j + 1  End If  Next i  ' Create or clear "High Scorers" sheet  On Error Resume Next  Application.DisplayAlerts = False  Worksheets("High Scorers").Delete  Application.DisplayAlerts = True  On Error GoTo 0    Set wsTarget = ThisWorkbook.Sheets.Add  wsTarget.Name = "High Scorers"  ' Header  wsTarget.Range("A1:D1").Value = Array("EmpID", "Name", "Department", "Score")    ' Output filtered data  If j > 1 Then  wsTarget.Range("A2").Resize(j - 1, 4).Value = outputData  MsgBox (j - 1) & " high scoring employees copied to 'High Scorers' sheet.", vbInformation  Else  MsgBox "No high scorers found.", vbExclamation  End If  End Sub |
|  |  |
| Form Control: Button | Assign the above macro to the button |
| Form Control: Combo Box | Sub FilterByDepartment()  Dim ws As Worksheet  Dim dept As String  Dim lastRow As Long  Dim i As Long  Set ws = ThisWorkbook.Sheets("Sheet1")  dept = ws.Range("G1").Value ' Cell linked to ComboBox  lastRow = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row  ws.Range("A1:D" & lastRow).AutoFilter Field:=3, Criteria1:=dept  End Sub |
|  |  |
| Conditional Formatting | Sub ApplyConditionalFormatting()  Dim ws As Worksheet  Dim lastRow As Long  Dim rng As Range  Set ws = ThisWorkbook.Sheets("Sheet1")  lastRow = ws.Cells(ws.Rows.Count, 4).End(xlUp).Row ' Find last row in column D  Set rng = ws.Range("D2:D" & lastRow) ' Score range  ' Clear previous formatting  rng.FormatConditions.Delete  ' Rule 1: Green for >=85  With rng.FormatConditions.Add(Type:=xlCellValue, Operator:=xlGreaterEqual, Formula1:="85")  .Interior.Color = RGB(144, 238, 144) ' Light Green  .Font.Bold = True  End With  ' Rule 2: Yellow for 70–84  With rng.FormatConditions.Add(Type:=xlCellValue, Operator:=xlBetween, Formula1:="70", Formula2:="84")  .Interior.Color = RGB(255, 255, 153) ' Light Yellow  End With  ' Rule 3: Red for <70  With rng.FormatConditions.Add(Type:=xlCellValue, Operator:=xlLess, Formula1:="70")  .Interior.Color = RGB(255, 102, 102) ' Light Red  End With  MsgBox "Conditional formatting applied!", vbInformation  End Sub |
|  |  |
| Sorting (score in desc order) | Sub SortByScoreDescending()  Dim ws As Worksheet  Dim lastRow As Long  Set ws = ThisWorkbook.Sheets("Sheet1")  lastRow = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row  ' Apply sort on Score (Column D)  ws.Sort.SortFields.Clear  ws.Sort.SortFields.Add Key:=ws.Range("D2:D" & lastRow), \_  SortOn:=xlSortOnValues, Order:=xlDescending, DataOption:=xlSortNormal  With ws.Sort  .SetRange ws.Range("A1:D" & lastRow)  .Header = xlYes  .MatchCase = False  .Orientation = xlTopToBottom  .Apply  End With  MsgBox "Sorted by Score (High to Low)", vbInformation  End Sub |
|  |  |
| Filter records where Department = "IT" and Score ≥ 70. | Sub FilterITHighScores()  Dim ws As Worksheet  Dim lastRow As Long  Set ws = ThisWorkbook.Sheets("Sheet1")  lastRow = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row  ' Clear any previous filters  If ws.AutoFilterMode Then ws.AutoFilterMode = False  ' Apply filter  ws.Range("A1:D" & lastRow).AutoFilter Field:=3, Criteria1:="IT" ' Department  ws.Range("A1:D" & lastRow).AutoFilter Field:=4, Criteria1:=">=70" ' Score  MsgBox "Filtered: IT Department with Score >= 70", vbInformation  End Sub |
|  |  |
| User def functions |  |
| >= 85 → "Excellent"  >= 70 → "Good"  >= 50 → "Average"  < 50 → "Needs Improvement" | Function GetPerformance(score As Double) As String  If score >= 85 Then  GetPerformance = "Excellent"  ElseIf score >= 70 Then  GetPerformance = "Good"  ElseIf score >= 50 Then  GetPerformance = "Average"  Else  GetPerformance = "Needs Improvement"  End If  End Function |
|  |  |
| Bonus Eligibility |  |
|  | Function IsEligibleForBonus(score As Double, department As String) As String  If score >= 80 And department <> "HR" Then  IsEligibleForBonus = "Eligible"  Else  IsEligibleForBonus = "Not Eligible"  End If  End Function |
| File | VBA\_Data\_Consolidation\_Regions |
| Data consolidation means combining data from multiple sheets, workbooks, or ranges into a single summary sheet. It’s a very practical skill for Excel users in HR, sales, finance, etc. | Sub ConsolidateEmployeeData()  Dim wsTarget As Worksheet, wsSource As Worksheet  Dim wb As Workbook  Dim lastRow As Long, pasteRow As Long  Dim sheetNames As Variant  Dim i As Long  Set wb = ThisWorkbook  sheetNames = Array("Region1", "Region2", "Region3")  ' Create/Clear target sheet  On Error Resume Next  Application.DisplayAlerts = False  wb.Sheets("MasterSheet").Delete  Application.DisplayAlerts = True  On Error GoTo 0  Set wsTarget = wb.Sheets.Add  wsTarget.Name = "MasterSheet"  wsTarget.Range("A1:D1").Value = Array("EmpID", "Name", "Department", "Score")  pasteRow = 2  ' Loop through source sheets  For i = LBound(sheetNames) To UBound(sheetNames)  Set wsSource = wb.Sheets(sheetNames(i))  lastRow = wsSource.Cells(wsSource.Rows.Count, 1).End(xlUp).Row  ' Copy data excluding header  wsSource.Range("A2:D" & lastRow).Copy wsTarget.Range("A" & pasteRow)  pasteRow = wsTarget.Cells(wsTarget.Rows.Count, 1).End(xlUp).Row + 1  Next i  MsgBox "Consolidation complete! Check 'MasterSheet'", vbInformation  End Sub |