

```
Private Sub btnExportPDF_Click()
    On Error GoTo ErrHandler
```

```
    Dim wsTemp As Worksheet
    Dim pdfPath As String
    Dim i As Long, j As Long
    Dim li As ListItem
    Dim saveDialog As FileDialog
    Dim lastRow As Long
```

' تحقق من وجود بيانات

```
    If Me.lstReportDetails.ListItems.Count = 0 Then
        MsgBox "لا توجد بيانات لتصديرها", vbExclamation
        Exit Sub
    End If
```

' إنشاء ورقة مؤقتة لتجهيز التقرير

```
    Application.ScreenUpdating = False
    Application.DisplayAlerts = False
    On Error Resume Next
    Set wsTemp = ThisWorkbook.Worksheets("TempReport")
    On Error GoTo 0
    If wsTemp Is Nothing Then
        Set wsTemp = ThisWorkbook.Worksheets.Add
        wsTemp.name = "TempReport"
    Else
        wsTemp.Cells.Clear
    End If
```

' إعداد العناوين

```
    wsTemp.Range("A1:E1").Value = Array("التاريخ", "النوع", "البيان", "المبلغ", "ملاحظات")
    wsTemp.Rows(1).Font.Bold = True
    wsTemp.Rows(1).Font.Size = 12
    wsTemp.Rows(1).Interior.Color = RGB(200, 200, 255)
```

' تعبئة البيانات من ListView

```
    For i = 1 To Me.lstReportDetails.ListItems.Count
        Set li = Me.lstReportDetails.ListItems(i)
        wsTemp.Cells(i + 1, 1).Value = li.Text
        For j = 1 To li.ListSubItems.Count
            wsTemp.Cells(i + 1, j + 1).Value = li.ListSubItems(j).Text
        Next j
    Next i
```

' تنسيق الأعمدة

```
wsTemp.Columns("A:E").AutoFit
wsTemp.PageSetup.Orientation = xlPortrait
wsTemp.PageSetup.Zoom = False
wsTemp.PageSetup.FitToPagesWide = 1
wsTemp.PageSetup.FitToPagesTall = False
wsTemp.PageSetup.CenterHorizontally = True
```

' نافذة اختيار مكان حفظ PDF

```
Set saveDialog = Application.FileDialog(msoFileDialogSaveAs)
With saveDialog
    .Title = "اختر مكان حفظ التقرير كـ PDF"
    .InitialFileName = "تقرير" & Format(Now, "yyyy-mm-dd_hh-mm") & ".pdf"
    .FilterIndex = 2
    If .Show <> -1 Then GoTo Cleanup
    pdfPath = .SelectedItems(1)
    If LCase(Right(pdfPath, 4)) <> ".pdf" Then pdfPath = pdfPath & ".pdf"
End With
```

' تصدير PDF

```
wsTemp.ExportAsFixedFormat Type:=xlTypePDF, Filename:=pdfPath,
Quality:=xlQualityStandard, _
IncludeDocProperties:=True, IgnorePrintAreas:=False, OpenAfterPublish:=True
```

```
MsgBox "تم حفظ التقرير بنجاح في " & pdfPath, vbInformation
```

Cleanup:

```
Application.DisplayAlerts = False
wsTemp.Delete
Application.DisplayAlerts = True
Application.ScreenUpdating = True
Exit Sub
```

ErrorHandler:

```
MsgBox "حدث خطأ أثناء التصدير " & Err.Description, vbCritical
Resume Cleanup
End Sub
```

Private Sub btnGenerateReport_Click()

```
On Error GoTo ErrorHandler
Application.ScreenUpdating = False
```

```

Dim fromDate As Date, toDate As Date
Dim wsS As Worksheet, wsR As Worksheet, wsE As Worksheet
Dim arrS As Variant, arrR As Variant, arrE As Variant
Dim i As Long
Dim totalSales As Double, totalReturns As Double, totalExpenses As Double
Dim li As MSComctlLib.ListItem
Dim reportType As String
Dim trxDate As Date
Dim rowsCount As Long

```

' اقرأ نوع التقرير وتحويل التواريخ بأمان إن وُجدت

```

reportType = Trim(Me.cmbReportType.Value)
If Trim(Me.txtFromDate.Value) <> "" Then fromDate = CDate(Me.txtFromDate.Value)
If Trim(Me.txtToDate.Value) <> "" Then toDate = CDate(Me.txtToDate.Value)

```

```

Set wsS = ThisWorkbook.Sheets("Sales")
Set wsR = ThisWorkbook.Sheets("Returns")
Set wsE = ThisWorkbook.Sheets("Expenses")

```

```

Me.lstReportDetails.ListItems.Clear
totalSales = 0: totalReturns = 0: totalExpenses = 0

```

' ----- المبيعات -----

```

rowsCount = wsS.Cells(wsS.Rows.Count, "A").End(xlUp).Row

```

```

If rowsCount >= 2 Then

```

```

    arrS = wsS.Range("A2", wsS.Cells(rowsCount, "I")).Value

```

```

    If IsArray(arrS) Then

```

```

        For i = LBound(arrS, 1) To UBound(arrS, 1)

```

' تحقق أن العمود الثاني يحتوي على تاريخ صالح

```

            If IsDate(arrS(i, 2)) Then

```

```

                trxDate = CDate(arrS(i, 2))

```

```

                If IsInDateRange(trxDate, fromDate, toDate, reportType) Then

```

```

                    totalSales = totalSales + val(arrS(i, 7)) ' إجمالي الفاتورة = عمود G

```

```

                    Set li = Me.lstReportDetails.ListItems.Add(, , Format(trxDate, "yyyy-mm-dd"))

```

```

                    li.SubItems(1) = "مبيعات"

```

```

                    li.SubItems(2) = CStr(arrS(i, 4)) ' اسم المنتج

```

```

                    li.SubItems(3) = Format(val(arrS(i, 7)), "0.00")

```

```

                    li.SubItems(4) = "-" ' تصنيف غير متوفر هنا

```

```

                End If

```

```

            End If

```

```

        Next i

```

```

    End If

```

End If

'----- المرتجعات -----'

```
rowCount = wsR.Cells(wsR.Rows.Count, "A").End(xlUp).Row
If rowCount >= 2 Then
    arrR = wsR.Range("A2", wsR.Cells(rowCount, "I")).Value ' H بدل | التعديل هنا: العمود ؟ '
    If IsArray(arrR) Then
        For i = LBound(arrR, 1) To UBound(arrR, 1)
            If IsDate(arrR(i, 3)) Then
                trxDate = CDate(arrR(i, 3))
                If IsInDateRange(trxDate, fromDate, toDate, reportType) Then
                    totalReturns = totalReturns + val(arrR(i, 9)) ' التعديل هنا أيضاً: العمود التاسع ؟ '
                    Set li = Me.lstReportDetails.ListItems.Add(, , Format(trxDate, "yyyy-mm-dd"))
                    li.SubItems(1) = "مرتجع"
                    li.SubItems(2) = CStr(arrR(i, 5)) ' اسم الصنف '
                    li.SubItems(3) = Format(val(arrR(i, 9)), "0.00") ' القيمة من العمود '
                    li.SubItems(4) = CStr(arrR(i, 8)) ' H السبب (العمود) '
                End If
            End If
        Next i
    End If
End If
```

'----- المصروفات -----'

```
rowCount = wsE.Cells(wsE.Rows.Count, "A").End(xlUp).Row
If rowCount >= 2 Then
    arrE = wsE.Range("A2", wsE.Cells(rowCount, "E")).Value
    If IsArray(arrE) Then
        For i = LBound(arrE, 1) To UBound(arrE, 1)
            If IsDate(arrE(i, 1)) Then
                trxDate = CDate(arrE(i, 1))
                If IsInDateRange(trxDate, fromDate, toDate, reportType) Then
                    totalExpenses = totalExpenses + val(arrE(i, 4)) ' المبلغ = D عمود '
                    Set li = Me.lstReportDetails.ListItems.Add(, , Format(trxDate, "yyyy-mm-dd"))
                    li.SubItems(1) = "مصروف"
                    li.SubItems(2) = CStr(arrE(i, 3)) ' بيان الصرف '
                    li.SubItems(3) = Format(val(arrE(i, 4)), "0.00")
                    li.SubItems(4) = CStr(arrE(i, 5)) ' التصنيف '
                End If
            End If
        Next i
    End If
End If
```

End If

' ----- عرض النتائج ----- '

```
Me.lblSalesTotal.Caption = Format(totalSales, "#,##0.00")
Me.lblReturnsTotal.Caption = Format(totalReturns, "#,##0.00")
Me.lblExpensesTotal.Caption = Format(totalExpenses, "#,##0.00")
Me.lblNetProfit.Caption = Format(totalSales - (totalReturns + totalExpenses), "#,##0.00")
```

```
Application.ScreenUpdating = True
Exit Sub
```

ErrorHandler:

```
Application.ScreenUpdating = True
MsgBox "حدث خطأ: " & Err.Number & " - " & Err.Description, vbCritical
End Sub
```

```
Private Sub UserForm_Initialize()
Application.ScreenUpdating = False
Me.cmbReportType.Clear
Me.cmbReportType.AddItem "يومي"
Me.cmbReportType.AddItem "شهري"
Me.cmbReportType.AddItem "فترة محددة"
Me.cmbReportType.AddItem "كل العمليات"
```

```
With Me.lstReportDetails
.View = lvwReport
.Gridlines = True
.FullRowSelect = True
.ColumnHeaders.Clear
.ColumnHeaders.Add , , "80", "التاريخ"
.ColumnHeaders.Add , , "80", "النوع"
.ColumnHeaders.Add , , "180", "البيان"
.ColumnHeaders.Add , , "80", "المبلغ"
.ColumnHeaders.Add , , "100", "التصنيف"
End With
```

```
Call ClearReportsFields
Application.ScreenUpdating = True
End Sub
```

```
Private Sub ClearReportsFields()
Me.txtFromDate.Value = ""
Me.txtToDate.Value = ""
```

```

Me.cmbReportType.Value = ""
Me.lstReportDetails.ListItems.Clear
Me.lblSalesTotal.Caption = "0"
Me.lblReturnsTotal.Caption = "0"
Me.lblExpensesTotal.Caption = "0"
Me.lblNetProfit.Caption = "0"
End Sub

```

```

Private Sub btnClear_Click()
    Call ClearReportsFields
End Sub

```

' دالة مرنة لفحص ما إذا كانت trxDate داخل النطاق حسب النوع

```

Private Function IsInDateRange(ByVal trxDate As Date, ByVal fromDate As Variant, ByVal
toDate As Variant, ByVal mode As String) As Boolean

```

```

    Dim f As Date, t As Date
    Dim sMode As String

```

```

    On Error GoTo ErrHandler
    IsInDateRange = False

```

' نمط موحد (صغير الحروف، نزيل المسافات)

```

sMode = LCase(Trim(mode))

```

' قبول مرادفات عربية وإنجليزية

```

Select Case sMode
    Case "اليومي", "daily", "يومي"
        sMode = "daily"
    Case "الشهري", "monthly", "شهري"
        sMode = "monthly"
    Case "محدد", "المحدد", "custom", "فترة محددة", "period"
        sMode = "period"
    Case "كل العمليات", "all", "all operations", "allops"
        sMode = "all"
    Case Else

```

' إذا لم يحدد نوع، اعتبر كل العمليات

```

        sMode = "all"
End Select

```

' تحويل from/to إن كانت صالحة، وإلا نعلمهما كقيمة غير معرفة

```

If IsDate(fromDate) Then f = CDate(fromDate) Else f = 0
If IsDate(toDate) Then t = CDate(toDate) Else t = 0

```

```

Select Case sMode

```

Case "daily"

' يوم محدد: إذا user أعطى fromDate نستخدمه، وإلا نستخدم اليوم الحالي

If f = 0 Then f = Date

IsInDateRange = (DateValue(trxDate) = DateValue(f))

Case "monthly"

' شهر محدد: نستخدم month/year من fromDate إن وُجد، وإلا نستخدم الشهر الحالي

If f = 0 Then

IsInDateRange = (Month(trxDate) = Month(Date) And Year(trxDate) = Year(Date))

Else

IsInDateRange = (Month(trxDate) = Month(f) And Year(trxDate) = Year(f))

End If

Case "period"

' فترة محددة: إذا لا تاريخ بداية ولا نهاية => قبول كل التواريخ

If f = 0 And t = 0 Then

IsInDateRange = True

Elseif f <> 0 And t = 0 Then

' من تاريخ معين وحتى نهاية الزمن (أي بعد أو يساوي f)

IsInDateRange = (trxDate >= f)

Elseif f = 0 And t <> 0 Then

' حتى تاريخ معين

IsInDateRange = (trxDate <= t)

Else

' من f إلى t (شاملة)

IsInDateRange = (trxDate >= f And trxDate <= t)

End If

Case "all"

IsInDateRange = True

Case Else

IsInDateRange = True

End Select

Exit Function

ErrorHandler:

' لو حصل خطأ في التحويل نعتبر عدم التطابق

IsInDateRange = False

End Function

```
Private Sub btnClose_Click()  
    Unload Me  
End Sub
```

===== TextBox تصحيح التاريخ المكتوب داخل ===== '

```
Private Function FixDateInput(txt As MSForms.TextBox) As Date
```

```
    Dim val As String, d As Date  
    On Error GoTo InvalidDate
```

```
    val = Trim(txt.Value)  
    If val = "" Then Exit Function
```

' نحاول اكتشاف التنسيقات الشائعة

```
    val = Replace(val, "\", "/")  
    val = Replace(val, "-", "/")
```

' تحويل آمن

```
    d = CDate(val)
```

' إعادة تنسيقه تلقائيًا في المربع

```
    txt.Value = Format(d, "yyyy-mm-dd")
```

```
    FixDateInput = d  
    Exit Function
```

```
InvalidDate:
```

```
    MsgBox "تاريخ غير صالح", vbExclamation, "الرجاء إدخال تاريخ صحيح في الخانة"
```

```
    txt.Value = ""
```

```
    FixDateInput = 0
```

```
End Function
```

```
Private Sub txtFromDate_AfterUpdate()
```

```
    Call FixDateInput(Me.txtFromDate)
```

```
End Sub
```

```
Private Sub txtToDate_AfterUpdate()
```

```
    Call FixDateInput(Me.txtToDate)
```

```
End Sub
```