

Personal Financial Reconciliation VBA CODES

REFRESH CLEANED DATA VBA CODE

```
Sub Refresh_CleanData()  
,  
  
' Refresh_Clean_Data Macro  
' It refreshes newly added clean data from raw file and shows a confirmation message  
,  
  
    ActiveWorkbook.RefreshAll  
  
    ' Show message box when refresh is complete  
    MsgBox " Refresh completed successfully!"  
End Sub
```

IMPORT NEW TRANSACTION VBA CODE

```
Sub AppendNewTransactions_TableSafe()  
    Dim wsClean As Worksheet, wsEDC As Worksheet  
    Dim lo As ListObject  
    Dim lastCleanRow As Long, i As Long  
    Dim dict As Object, dt As String  
    Dim lr As ListRow  
    Dim src As Variant  
    Dim added As Long  
    Dim lastRow As Long
```

On Error GoTo CleanFail

Application.ScreenUpdating = False

Application.EnableEvents = False

Set wsClean = ThisWorkbook.Sheets("Cleaned_Data")

Set wsEDC = ThisWorkbook.Sheets("EDC")

' Try to get the EDC table if it exists

On Error Resume Next

Set lo = wsEDC.ListObjects("EDC") ' table named EDC (if present)

If lo Is Nothing Then Set lo = wsEDC.ListObjects(1) ' else first table on the sheet

On Error GoTo 0

' Build a dictionary of existing dates in EDC

Set dict = CreateObject("Scripting.Dictionary")

If Not lo Is Nothing Then

 If Not lo.DataBodyRange Is Nothing Then

 For i = 1 To lo.DataBodyRange.Rows.Count

 dt = CStr(lo.DataBodyRange.Cells(i, 1).Value) ' table col 1 == sheet col B

 If Len(dt) > 0 Then dict(dt) = True

 Next i

 End If

Else

 lastRow = wsEDC.Cells(wsEDC.Rows.Count, 2).End(xlUp).Row ' sheet col B

 If lastRow < 12 Then lastRow = 11

 For i = 12 To lastRow

 dt = CStr(wsEDC.Cells(i, 2).Value) ' column B

 If Len(dt) > 0 Then dict(dt) = True

 Next i

End If

```

' Loop Cleaned_Data and append rows whose DATE (col A) is not in EDC
lastCleanRow = wsClean.Cells(wsClean.Rows.Count, 1).End(xlUp).Row
For i = 2 To lastCleanRow
    dt = CStr(wsClean.Cells(i, 1).Value) ' Cleaned_Data col A = Date
    If Len(dt) > 0 Then
        If Not dict.Exists(dt) Then
            ' Read 1x8 source row (A:H)
            src = wsClean.Cells(i, 1).Resize(1, 8).Value

            If Not lo Is Nothing Then
                ' Append into the table (safe)
                Set lr = lo.ListRows.Add
                lr.Range.Cells(1, 1).Resize(1, 8).Value = src ' fills table cols 1..8 (B..I on sheet)
            Else
                ' No table: insert into the sheet at row 12
                wsEDC.Rows(12).Insert Shift:=xlDown
                wsEDC.Cells(12, 2).Resize(1, 8).Value = src ' B:I
            End If

            dict(dt) = True
            added = added + 1
        End If
    End If
Next i

' Sort by Date (B) newest → oldest
If Not lo Is Nothing Then
    If Not lo.DataBodyRange Is Nothing And lo.DataBodyRange.Rows.Count > 1 Then
        With lo.Sort
            .SortFields.Clear
            .SortFields.Add Key:=lo.ListColumns(1).DataBodyRange, _

```

```

        SortOn:=xlSortOnValues, Order:=xlDescending, DataOption:=xlSortNormal

        .Header = xlYes

        .Apply

    End With

End If

Else

    lastRow = wsEDC.Cells(wsEDC.Rows.Count, 2).End(xlUp).Row

    If lastRow >= 12 Then

        With wsEDC.Sort

            .SortFields.Clear

            .SortFields.Add Key:=wsEDC.Range("B12:B" & lastRow), _

                SortOn:=xlSortOnValues, Order:=xlDescending, DataOption:=xlSortNormal

            .SetRange wsEDC.Range("B12:R" & lastRow)

            .Header = xlNo

            .Apply

        End With

    End If

End If

MsgBox "Append complete. New rows added: " & added, vbInformation, "Process Complete"

CleanExit:

    Application.EnableEvents = True

    Application.ScreenUpdating = True

Exit Sub

CleanFail:

    MsgBox "Append failed: " & Err.Description, vbExclamation, "Error"

    Resume CleanExit

    ' Clear Clean_Data after append

    Call ClearCleanData

```

End Sub

BALANCE VALIDATION

```
Sub BalanceValidation()
```

```
    Dim tbl As ListObject
```

```
    Dim lastRow As Long
```

```
    Dim ws As Worksheet
```

```
    Dim startRow As Long
```

```
    ' Set worksheet and table
```

```
    Set ws = ThisWorkbook.Sheets("EDC")
```

```
    Set tbl = ws.ListObjects("Table3") ' Use the correct table name
```

```
    ' Start processing from row 12
```

```
    startRow = 12
```

```
    ' Find the last row of the table
```

```
    lastRow = tbl.Range.Rows.Count + tbl.Range.Row - 1
```

```
    ' Only continue if we have data beyond the start row
```

```
    If lastRow >= startRow Then
```

```
        ' Fill EXPECTED BALANCE (Column J)
```

```
        ws.Range("J" & startRow).FormulaR1C1 = _
```

```
            "=IF(RC[-5]="&"D"&"", RC[-1]-RC[-2], IF(RC[-5]="&"C"&"", RC[-1]+RC[-2], ""Check Type""))"
```

```
        ws.Range("J" & startRow).AutoFill _
```

```
            Destination:=ws.Range("J" & startRow & ":J" & lastRow)
```

```

' Fill VERIFICATION (Column K)

ws.Range("K" & startRow).FormulaR1C1 = _
    "=IF(ABS(RC[-2]-RC[-1])<0.01, ""True"", ""False"")"
ws.Range("K" & startRow).AutoFill _
    Destination:=ws.Range("K" & startRow & ":K" & lastRow)

' Conditional formatting for "False" results
With ws.Range("K" & startRow & ":K" & lastRow)
    .FormatConditions.Delete
    .FormatConditions.Add Type:=xlTextString, String:="False", TextOperator:=xlContains
    .FormatConditions(1).Interior.Color = vbRed
End With

MsgBox "☒ Balance validation completed!", vbInformation, "Validation Complete"

Else
    MsgBox "No rows found to validate starting from row " & startRow & ".", vbExclamation, "No
Data"
End If
End Sub

```

LINK EDC TO THE CHEQUES SHEET

```

Sub LinkEDCtoCheque()

    Dim wsEDC As Worksheet, wsCheque As Worksheet
    Dim tblEDC As ListObject, tblCheque As ListObject
    Dim lastEDCRow As Long, lastChequeRow As Long
    Dim i As Long
    Dim chequeNum As String
    Dim foundCell As Range

    ' Set worksheets and tables
    Set wsEDC = ThisWorkbook.Sheets("EDC")
    Set wsCheque = ThisWorkbook.Sheets("Cheques")
    Set tblEDC = wsEDC.ListObjects("Table3") ' EDC table
    Set tblCheque = wsCheque.ListObjects("Table4") ' Cheque table

    ' Find last data rows inside each table
    lastEDCRow = tblEDC.Range.Rows.Count + tblEDC.Range.Row - 1
    lastChequeRow = tblCheque.Range.Rows.Count + tblCheque.Range.Row - 1

    ' Loop through EDC Cheque Numbers (Column L in Table3)
    For i = tblEDC.Range.Row + 1 To lastEDCRow
        chequeNum = Trim(wsEDC.Cells(i, "L").Value)

        If chequeNum <> "" Then
            ' Search Cheque table Column B for match
            Set foundCell = wsCheque.Range("B" & tblCheque.Range.Row + 1 & ":B" & lastChequeRow) _
                .Find(What:=chequeNum, LookIn:=xlValues, LookAt:=xlWhole)

            If Not foundCell Is Nothing Then
                ' Example: copy Cheque Date from Column C in Table4 to Column M in Table3
                wsEDC.Cells(i, "M").Value = wsCheque.Cells(foundCell.Row, "C").Value
            End If
        End If
    Next i
End Sub

```

```

        End If
    Next i

    MsgBox "? Linking between EDC and Cheque completed using Cheque Number.", vbInformation,
    "Link Complete"

End Sub

```

SEARCH EDC BOTTON VBA CODE

```

Sub SearchEDC()
    Dim ws As Worksheet
    Dim lastRow As Long, lastCol As Long
    Dim searchText As String
    Dim cell As Range, searchRange As Range
    Dim firstAddress As String

    Set ws = ThisWorkbook.Sheets("EDC")
    searchText = Trim(ws.Range("D4").Value)

    ' Find last used row & column
    lastRow = ws.Cells(ws.Rows.Count, 2).End(xlUp).Row ' starting from col B
    lastCol = ws.Cells(11, ws.Columns.Count).End(xlToLeft).Column ' based on row 11 header

    ' Clear old highlights
    ws.Range("B12", ws.Cells(lastRow, lastCol)).Interior.ColorIndex = xlNone

```



```

If searchText <> "" Then
    Set searchRange = ws.Range("B12", ws.Cells(lastRow, lastCol))
    With searchRange
        Set cell = .Find(What:=searchText, LookAt:=xlPart, MatchCase:=False)
        If Not cell Is Nothing Then
            firstAddress = cell.Address
            Do
                cell.Interior.Color = vbYellow
                Set cell = .FindNext(cell)
            Loop While Not cell Is Nothing And cell.Address <> firstAddress
        Else
            MsgBox "No matches found in EDC.", vbInformation, "Search Complete"
        End If
    End With
Else
    MsgBox "Please enter a search value in D4.", vbExclamation, "No Search Value"
End If
End Sub

```

CHEQUE SEARCH VBA BOTTOM

```

Sub SearchCHEQUES()
    Dim ws As Worksheet

```

```

Dim lastRow As Long, lastCol As Long

Dim searchText As String

Dim cell As Range, searchRange As Range

Dim firstAddress As String


Set ws = ThisWorkbook.Sheets("CHEQUES")

searchText = Trim(ws.Range("D5").Value)


' Find last used row & column

lastRow = ws.Cells(ws.Rows.Count, 2).End(xlUp).Row
lastCol = ws.Cells(11, ws.Columns.Count).End(xlToLeft).Column


' Clear old highlights

ws.Range("B12", ws.Cells(lastRow, lastCol)).Interior.ColorIndex = xlNone


If searchText <> "" Then

    Set searchRange = ws.Range("B12", ws.Cells(lastRow, lastCol))

    With searchRange

        Set cell = .Find(What:=searchText, LookAt:=xlPart, MatchCase:=False)

        If Not cell Is Nothing Then

            firstAddress = cell.Address

            Do

                cell.Interior.Color = vbYellow

                Set cell = .FindNext(cell)

            Loop While Not cell Is Nothing And cell.Address <> firstAddress

        Else

            MsgBox "No matches found in CHEQUES.", vbInformation, "Search Complete"

        End If

    End With

Else

    MsgBox "Please enter a search value in D5.", vbExclamation, "No Search Value"

```

```
End If  
End Sub
```

CLEAR CLEANED DATA

```
Sub ClearCleanData()  
    Dim ws As Worksheet  
    Set ws = ThisWorkbook.Sheets("Cleaned_Data")  
  
    ' Clear all rows except headers (row 1)  
    ws.Range("A2:H" & ws.Cells(ws.Rows.Count, 1).End(xlUp).Row).ClearContents  
  
    MsgBox "? Clean_Data sheet cleared successfully! / ¡Datos limpiados correctamente!",  
    vbInformation  
End Sub
```