

Excel VBA and Macros

Data analysis obtains maximum benefit from VBA because users can automate and clean their data and create reports through effective data processing. While there are several limitations of this tool, it remains vital for analysts since it enhances the capabilities of Excel programs for companies. The process of mastering VBA will improve your data analysis ability, no matter where you stand in terms of experience

Definition of Macros:

A single instruction that expands automatically into a set of instructions to perform a particular task.

Understand with an example:

I have an Excel file for Kevin's cookie company open in Microsoft Excel. Orders Fulfilled k columns me jo orders abhi deliver nhi huye, ham chahty k unki formatting different ho, like colors and font styles, different from those orders who are fulfilled. Ab hm aik single kaliye tou khud select kr k colors change kr skty, font change kr skty, lkn 4-5 kaliye kya krein, time waste. Agr mein chahou tou mein cell ko select kr format painter ko select kr k b Next cell pr paste kr skti, but still it takes time 3 clicks.

→ AB yha pr hm aik shortcut key, Aik button pr click krein taky Excel in sb steps ko automate kr de.

→ Ye aik simple example this lkn agr ap excel me kaam krti again and again koi task krti ho well that's an excellent candidate to turn into a macro.

→ We are going to use something called Macro Recorder to turn all these steps, or all of these clicks, into a code called VBA, and that's a programming language.

→ Jb hm aik macro bna lety hain, tou isy aik shortcut key, aik button ya menu se access kr skty haon.

Summary:

If you're annoyed by running repetitive tasks again and again, you can automate all of them using macros.

Let's start with how to record a basic macro that will do all of the formatting and font styling,

Go to view → Macros → Record Macro

1-Aik tab open kry ga, Change the name to "HighlightNewOrders" → You cannot include any spaces in the name.

2- You can specify a shortcut key combination to run your macro.

3- Place to store Macro: "This workbook" se yemacro sirf isi workbook me accessible hna, "Personal" se hm kisi b workbook se macro access kr skty.

4- You can also type a description about the macro.

5- Ok, kr k hm cells select kry gy, Bold, Italic, Underline aur cells k color change kry gy. Aur phr macro close kr dena.

6- An isy open krny kliye ya tou shortcut key press krna, Ya View se macros me ja k run krna ya,

Right click under ribbon → Customize Quick Access Toolbar → More Commands → Popular Commands → Macros -> HighlightNewOrders

7- Now, save the file as “Macro Enabled Workbook.”

8- Hm jb b macro run krty tou ye brbr usi row ko highlight krta jisy hmny macro record krty kiya tha, Ye aik absolute reference ha, → Hm ye existing macro delete krein gy aur again view → Relative reference ko click kr k then Record Macro ko click krna. “It applies formatting relative to where the starting position is.”

9- If you want to see the code you just created, Go to View -> Macros -> View Macros -> Edit, Another way to press Alt + F11. And also press to go back.

10- MS Visual basic ki tab open ho gi. By default, Code Window opens, Agr nhi hti tou Views se ja k open kro

11- On the left-hand side, we have Project Explorer, If you don't see then open it from View.

12- Here we can see all of our Excel objects. I can see my sheets and workbooks.

13- Down below, we can module, and ***Ever Macro is a Module.***

14- Down below in the left corner, Properties show hti related to any objects.

15- Now Understand the code:

```
"Sub HighlightNewOrders()
'
' HighlightNewOrders Macro
'
' Keyboard Shortcut: Ctrl+Shift+A
'
Selection.Font.Italic = True
Selection.Font.Underline = xlUnderlineStyleSingle
Selection.Font.Bold = True
Selection.Font.Italic = False
Selection.Font.Underline = xlUnderlineStyleNone
Selection.Font.Underline = xlUnderlineStyleSingle

Selection.Font.Italic = True

With Selection.Interior
    .Pattern = xlSolid
    .PatternColorIndex = xlAutomatic
    .ThemeColor = xlThemeColorAccent6
    .TintAndShade = 0.399975585192419
    .PatternTintAndShade = 0
End With
End Sub"
```

- “Sub ” This at the start of the code stands for Subroutine. Every Macro is a Subroutine.
- And this entire package between sub is a Unit.
- Code with a single quote and in green text is a comment.
- Most of the time, we create macros just by recording. Read the code every time after creating for better understanding.
- Jo Macros personal workbook me hty hm unhy direct open kr kr edit nhi kr skty, ALT+F11 press k open kr skty editing kaliye ya phr direct Visual Basic me open krna.

If Condition:

- Ab hm ye chahty k agar cells highlighted nhi tou highlight kr de, Agr Highlighted ha tou remove kr dein.
- Hm, phly removehighlight ka macro record krein gy then isy first highlighted Macro k code me if statement k sath add kr dein gy.

Strategy:

Simply code independent pieces and put them together in a single macro.

For LOOP:

- Ab hm ye chahty k Macro hmari aik row nhi blky Multiple row ko highlight kry jb hm usy run krein, tou iskaliye hmy **for loop** chlani ho gi. → Hmari 4 unfulfilled order rows ha tou hm Counter 1 to 4 chlaye gy, then End Sub se phly Next counter likhna.
- Ab yha hmny for loop me count khud specify kiya, but agar hm isy dynamic rkhny, so the user can choose. For that, we can use the input box back with the Visual Basic editor to take input from the user.

Do While loop

- When we don't know how many times it should run. This loop will continue running until a certain condition is met.
- Code :

```

"Sub HiglightRowsWithDoWhile()
'
'HighlightRowsWithDoWhile Macro
'

Do While ActiveCell <> ""
    ActiveCell.Range("A1:E1").Select
    Selection.Font.Bold = True
    Selection.Font.Italic = True
    Selection.Font.Underline = xlUnderlineStyleSingle
    With Selection.Interior

```

```

.Pattern = xlSolid
.PatternColorIndex = xlAutomatic
.ThemeColor = xlThemeColorAccent5
.TintAndShade = 0.399975585192419
.PatternTintAndShade = 0
End With
ActiveCell.Offset(1, 0).Range("A1:E1").Select -> Ye must likhna, ow dowhile last
blank row b chlaye gi.
Loop
End Sub
"

```

Do Until loop

Until the active cell is empty. “Do Until IsEmpty(ActiveCell)”.

Another Complex example :

Hmny apne existing kevin cookie k table ko power query me open kr k unpivot kr dena isay hamare first column me heading names aur second me values ho gi. Tou isy wapis proper format krne k liye ham Macro use karein ge. Yha a hmy **Absolute reference** use krna ha,

- Start recording macro
- Macro name is TransposeHeaders
- unique column names 5 ha unhy select kr k copy krna, aik blank cell select kr k, Ribbon me paste ka dropdown open kr k transpose select krna. Stop Recording
- Ab hmy record kaliye macro bnana. Ye second macro ha, last me hm dono ko combine kry gy.
- Hm record column k phly cell ko select kry gy with **absolute reference** kiun k hmy data whi se shuru hna. Usky bd **Relative Reference on** krna. Usky bd first 5 values select krni, header k neechy cell select kr k paste se special character se Transpose select krna, baki next 2,3 records kaliye b yhi krna.Taky hm asaani se loop lga sky. Macro stop kr dena. Ab jo hmny paste kiya is sb ko select kr k delete kr k dono macro chlany. TransposeHeaders and TransposeOrders.
- Hmny jo 3 Orders records select kiye thy ab unko loop lgani.
- Do While loop lgani.
- Jb hm TransposeOrders ka macro chlana tou record me spaces ani, ye spaces remove krny kaliye hmny aik aur macro bnana.
- Hm absolute reference k sath RemoveBlankRows ka macro bnana, kiun k hm janty k blank rows ki range kiya ha. Kin specific columns me hmari blank rows ha.
- Relative Refernce off → Name is RemoveBlankRows. → Columns selected jin me data ha. Home tab → Find and Search → Blank selected → right click → delete → Shift cells up → Recording off.
- Ab hm in teeno macros ko combine krna, Simply, first macro ka “End Sub” remove, 2nd ka start sub aur End Sub then 3rd ka b start sub remove.

- Jesy hi macro chla tou screen pr sb steps show huye, isy slow ho jata process tou hmy **Screen Updating ko Turn off** krna.
- Application.ScreenUpdating = False, AB hmara code chly ga, and we won't actually see it running on the screen.

You Can't Undo A Macro.