

## RSH901 Excel Tips & Tricks – Part 4

Improving your speed and efficiency in Excel by learning how to record and use macros.

### What is an Excel Macro?

If you have ever repeated tasks in Microsoft Excel, you can use a macro to automate those tasks. A macro is an action or a set of actions that you “record” and can run as many times as you need. While creating macros, you are essentially recording your mouse clicks and keystrokes, the keystrokes that you repeat every time you run/download, and then work within a monthly project ledger for instance.

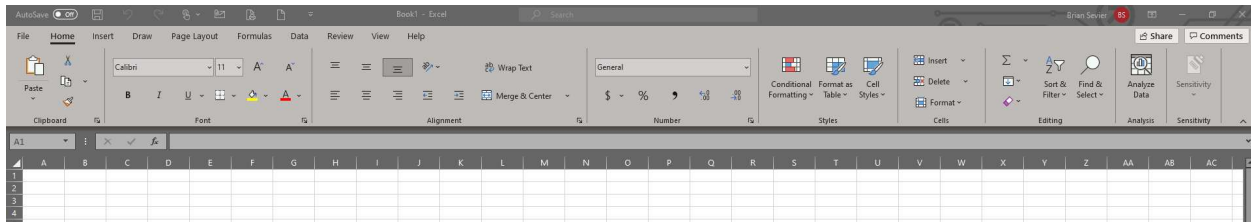
<https://support.microsoft.com/en-us/office/quick-start-create-a-macro-741130ca-080d-49f5-9471-1e5fb3d581a8>

First, how, and where do we manage macros within MS Excel? Depending on your version of Excel, Mac OS vs. Windows, and previous customizations to your ribbon, macros can be accessed in several places.

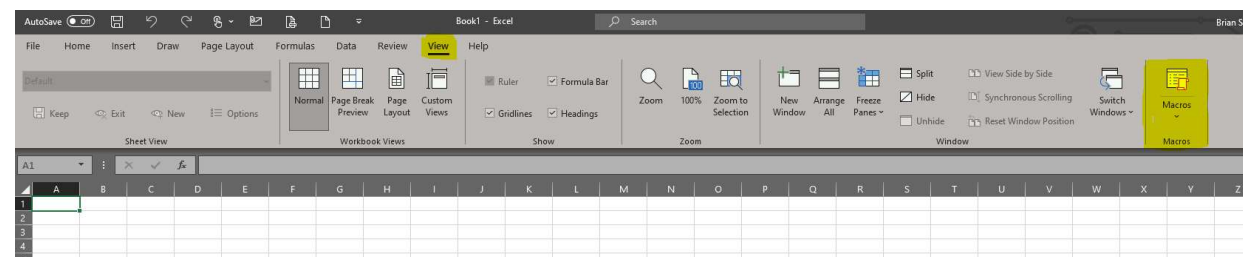
Open MS Excel (my version is the installed version of Excel that comes with the Enterprise Microsoft 365 Apps package, commonly installed at UF). You can find this information within any Microsoft Office application by choosing File → Account.



This is the layout of my ribbon, with menus (tabs) and commands.



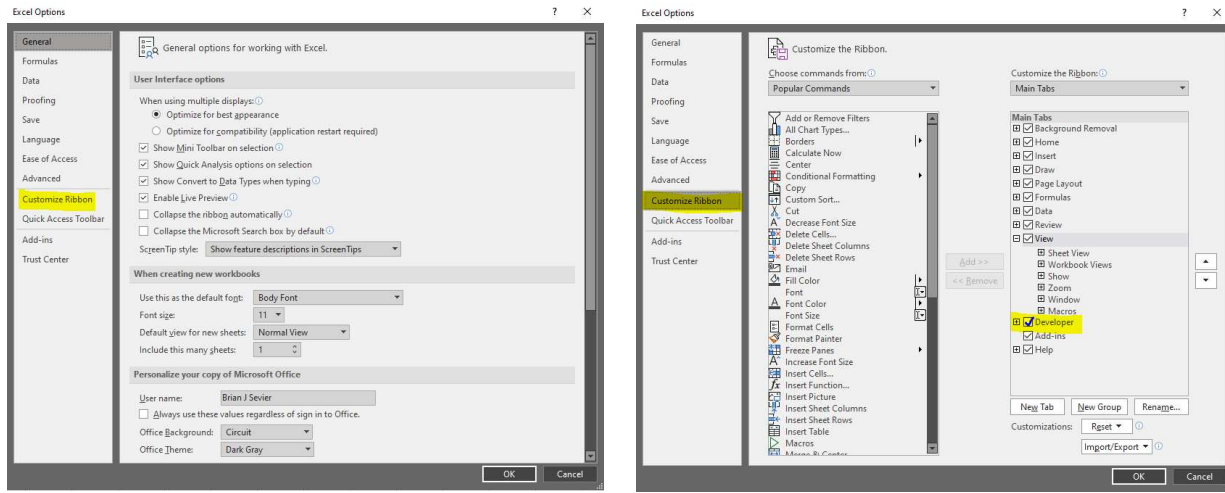
Choose the View menu, and you will see one location where Macros can be easily accessed. If you plan to just run previously saved macros, this is a convenient location to launch those macros.



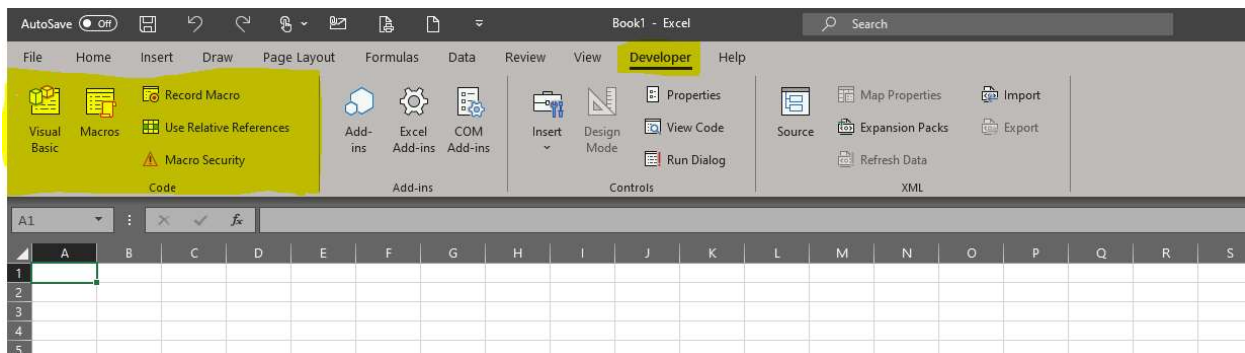
However, for today's exercise (and of course with the intention of using macros outside of today's workshop) it is recommended that you activate the DEVELOPER menu from your Excel menu options. This means we need to customize your ribbon if DEVELOPER does not already display in your menu structure.

*Go to FILE | OPTIONS (Bottom of the menu – lower left)*

Select Customize Ribbon on the left side of the Excel Options dialog box, then check the box for Developer on the right side of the Customize Ribbon Excel Options Dialog Box. Select OK.



Again, your menus and ribbon may still look different than mine displayed below, but you should now have a Menu tab for Developer. Select it and you can now see the highlighted region in the screen captured below on your ribbon.





## A Few Beginning Topics

Macros – it's the MS Excel version of “wash – rinse – repeat”

Reflecting on **Excel Tips and Tricks – Session 1** (and also the subsequent sessions...), we invested a lot of time in formatting, consistent column widths, page layout, and general formatting. If you recall, Macros are intended to automate functions that you repeat frequently.

For example:

- Perhaps you download a monthly project ledger in Excel from myUFL. Currently you go in manually sort, filter, format, and layout that Excel query into a version or format you prefer...repeatedly, whether its for on-screen review or printed reports to leadership.
- When working on a table of data, is your preferred preference to have bold column and row headers, grid lines, and maybe even color shaded areas? Do you use that format every time (or even most of the time) you produce an Excel output?

Other lessons that were introduced, also carry forward into the Macro conversation. Absolute versus Relative references. If you record Macros using an **absolute** reference, it will only automate those actions on that absolute cell (or range of cells). If we record Macros alternatively using a **relative** reference, it can be applied to any cell (or range cells) that are selected when you run that Macro. There are benefits (and drawbacks) to both depending on how you plan to use the Macro.

## How to record macros

For this Exercise, we will use the EXCEL file that was provided:

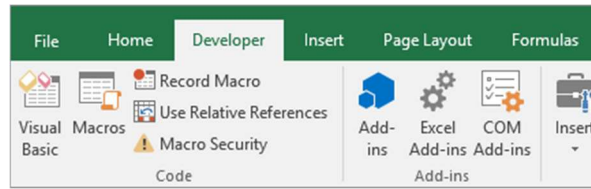
- **MS\_Excel\_Macro\_Exercise1.xls**

After you have opened that file, familiarize yourself with the layout, the data, and possibly your desired final presentation of that information. In previous class workshop sessions, we performed the following specific activities:

- Select All (CTRL+A) to choose the entire dataset.
- Double-click any column boundary to auto expand the columns to fit the width of the largest cell contents in that column.
- Sort expense accounts in order by date of the transaction.
- Subtotal the expense accounts, and grand total the subtotals.
- Format the columns, possibly bold, larger font, shaded/colored cell backgrounds.

OK so its time to give it a try.

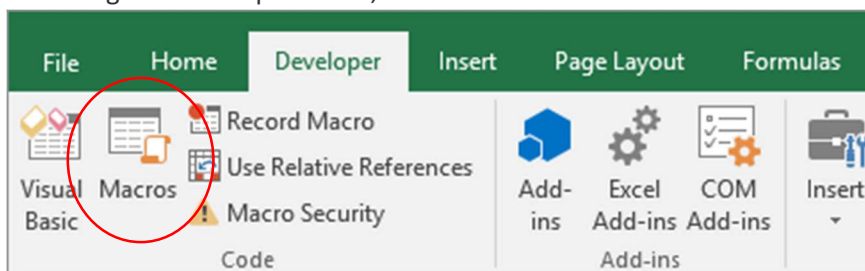
- 1) Using the **Data1** worksheet in the Exercise1 file, select the **Record Macro Button** – remember everything you do with your **KEYBOARD** or **MOUSE** becomes part of the recorded Macro. Plan your steps accordingly!!!



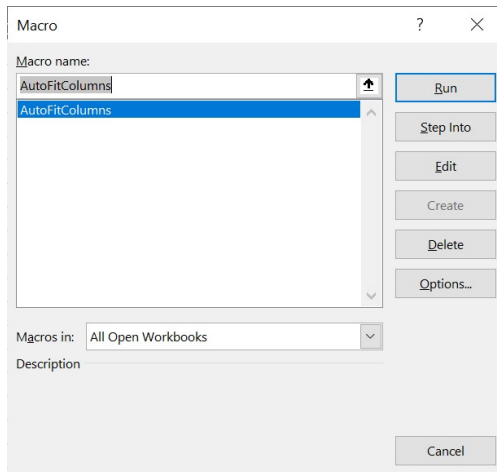
- 2) Optionally, you can **enter a name for the macro in the Macro name box**, but for now let's skip the part where you can enter a shortcut key in the Shortcut key box or a description in the Description box
  - a. Name it **AutoFitColumns**
- 3) Then select **OK to start recording**.
- 4) Your Excel application may inform you that you cannot use AutoSave functionality in the .XLS file format – that's OK!
  - a. Select **STOP RECORDING**
  - b. It will prompt you to save the file in a new format and choose **.XLSM – Macro-Enabled Excel Workbook**.
- 5) Then select **OK to start recording**.
- 6) Select All (CTRL+A) or use your mouse to highlight the entire area in order to choose the entire dataset (top to bottom, left to right)
- 7) Double-click any column boundary to auto expand the columns to fit the width of the largest cell contents in that column.
- 8) Stop Recording.

So, what did it do? It recorded us selecting all of the data on the worksheet, and then scaling the width of the columns to autofit the widest data value in any column. So what? Let's try it again now that it's recorded.

- 1) Select the **Data1 (test)** worksheet in the Exercise1 workbook.
- 2) Still using the Developer menu, select **Macros**



- 3) You should have a Macro Dialog Box that looks like this and choose the AutoFitColumns Macro and select Run.



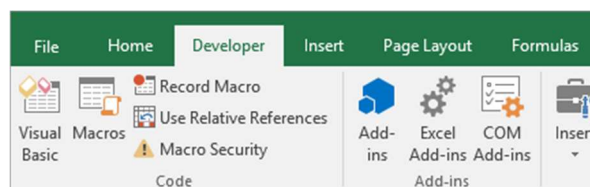
- 4) It should perform the same Auto Fit Columns exercise we went through while recording our steps a moment ago. Eureka – you just made a Macro.

Now this was a simple exercise, and no you didn't actually save yourself any mouse or keyboard steps per se...but if we started adding more steps to the Macro we can begin to appreciate the efficiency the Macro can give us.

In this next example, we are going to use the same excel output, but we are going to work from worksheet **Data2**. This time we are going to capture more steps in the recording of this macro. Specifically, this macro will perform all of the formatting steps that we would do every time we download that raw output or query from myUFL. So that means we need to be very careful about the order we choose for formatting and changing the layout. This next example will only work for spreadsheets that contain the exact same column header names!!!

OK so its time to give it a try.

- 1) Using the **Data2** worksheet in the Exercise1 file, select the **Record Macro Button** – remember everything you do with your **KEYBOARD** or **MOUSE** becomes part of the recorded Macro. Plan your steps accordingly!!!



- 2) Optionally, you can **enter a name for the macro in the Macro name box**, but for now lets skip the part where you can enter a shortcut key in the Shortcut key box or a description in the Description box
  - a. Name it **LedgerFormat**
- 3) Then select **OK to start recording**.



- 4) Your Excel application may inform you that you cannot use AutoSave functionality in the .XLS file format – that's OK!
  - a. Select STOP RECORDING
  - b. It will prompt you to save the file in a new format and choose **.XLSM – Macro-Enabled Excel Workbook**.
- 5) Then select **OK to start recording**.
- 6) Choose Column Header M, set it to CURRENCY number formatting.
- 7) Choose Row Header 1, set it to BOLD font style.
- 8) Select All (CTRL+A) or use your mouse to highlight the entire area in order to choose the entire dataset (top to bottom, left to right)
  - a. Double-click any column boundary to auto expand the columns to fit the width of the largest cell contents in that column.
- 9) While data is all selected, go to your DATA menu, choose SORT;
  - a. Sort first by ACCOUNT
  - b. Sort second by ACCTNG DATE
  - c. Hit ok/sort.
- 10) Stop Recording.

OK you just recorded your second Macro! Let's go ahead and test it now.

In the same workbook, now navigate to worksheet **Data2 (test)**, and choose the DEVELOPER menu. Select the Macros icon, and this time we want to choose the FormatLedger macro and hit run. In just seconds it should format your worksheet to the format you recorded during the Macro.

### Macro Naming Tips

Macro names must begin with a letter and cannot contain spaces, symbols, or punctuation marks. After the first letter, you can use more letters, numbers, or the underscore character, but the maximum length is 80 characters.

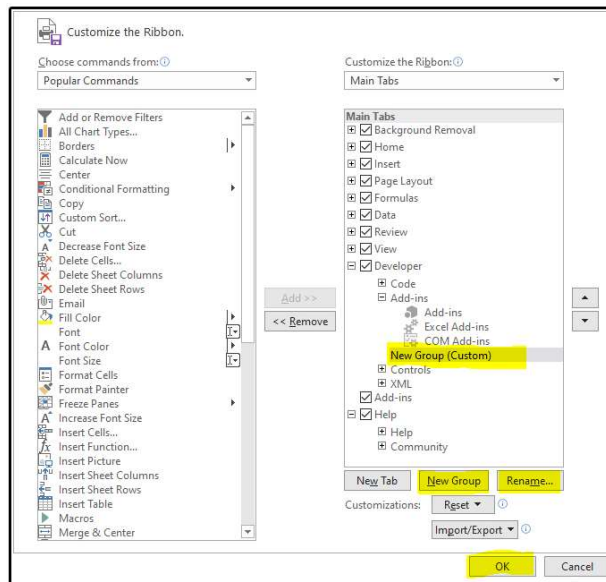
Keep macro names short (but descriptive), especially if you record a lot of macros, so you can easily identify them in the Macro Dialog Box.

### Best Practices for Using Macros

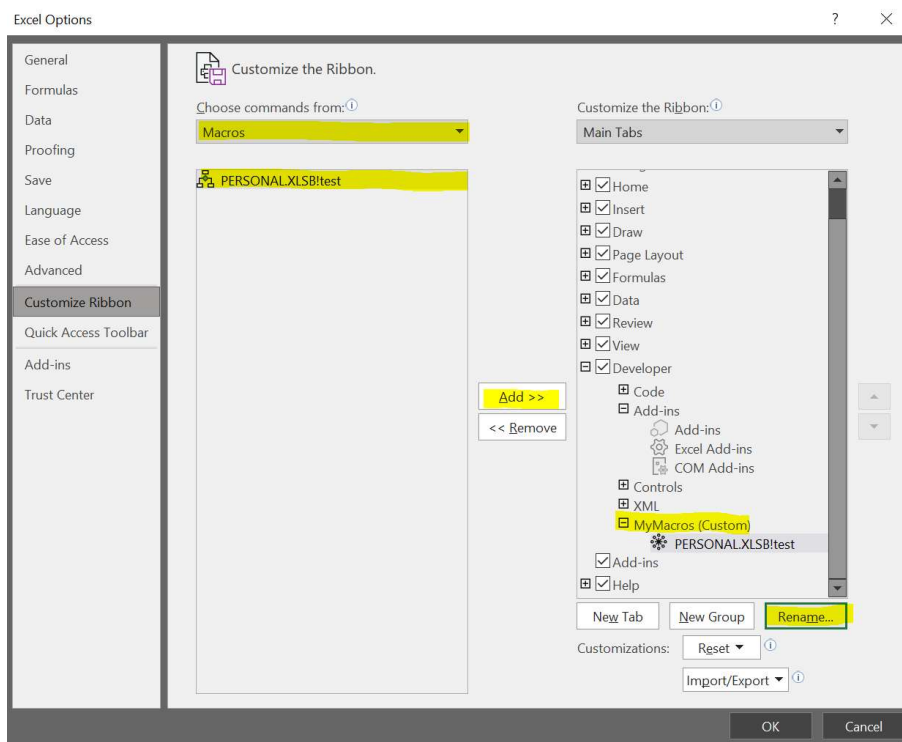
- 1) You cannot UNDO or CTRL+Z anything run via a macro
  - a. Until you have tested your macro sufficiently, make extra copies of your raw data before running the macro, because once the macro runs....you break it, your buy it!
- 2) Always begin at Home
  - a. Have your mouse positioned at cell A1 before starting the Record Macro process
- 3) Make macros RELATIVE not ABSOLUTE
  - a. Wrt the last best practice, when using relative macros, remember to start at the Home Position, as each relative cell reference is based on the starting cell reference!
  - b. Wrt to absolute macros, they have a purpose but if the data extends beyond the absolute range, your macro either wont work (debugging required) or it will miss some of the data in the absolute range.



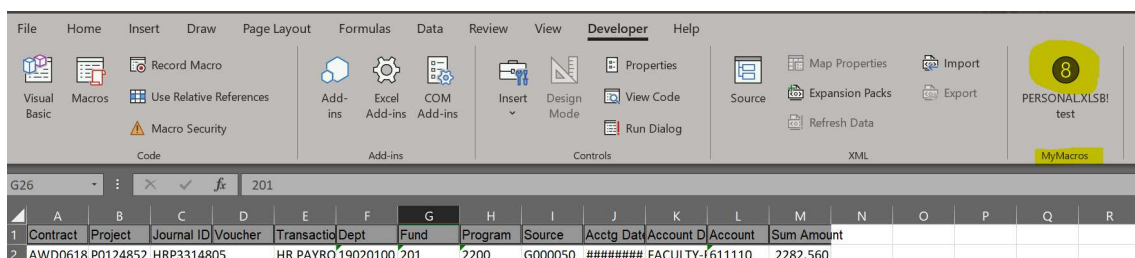
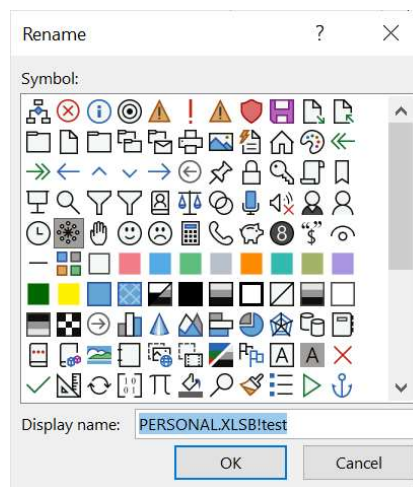
- 4) Always navigate with directional keys
  - a. Key board directional keys are not recorded by the macro, but if you scroll down a page or use your mouse wheel, it will record in the macro that the scroll was adjusted.
- 5) Keep macros small and specific
  - a. Modularize or compartmentalize your macros...if something doesn't work in a macro, it is easier to debug them when they are trying to rebuild the world in an Excel worksheet
- 6) Saving the spreadsheet (.xlsm) macro-enabled workbook
  - a. Once you have recorded a macro in a workbook, that workbook must now be saved as a Macro-enabled workbook (.xlsm)
  - b. After today's session, you should consider saving your final macros (after testing of course) to your Personal Macro Workbook – this means your macros are not just embedded in the single worksheet you are actively working on, but instead become a fixture embedded in EVERY excel workbook you open.
- 7) Add a macro button to a custom group on the Ribbon menu, suggest adding it to the DEVELOPER menu so it is embedded to other Macro functionality.
  - a. File → Options
  - b. Customize Ribbon Dialog Box



- c. **Expand** the Developer Menu, **insert** a New Group, **shift** it to the same location as shown above, **rename** it "MyMacros".
- d. Now add an icon for your custom macro while still in the Customize Ribbon Dialog Box.



- e. In the Choose Commands From drop-down, **select** Macros, **select** your new Macro, **select** the MyMacros custom Ribbon group, **press** Add, then **select** Rename, and **choose** and Icon that you like on the Ribbon to represent your new Macro. Press OK twice.







There are other valuable items to review regarding how you can use Macros. Macros are actually recorded through the user interface of Excel, but what is actually happening behind the scenes is a captured set of activities displayed as “code” in the Visual Basic console.

Time permitting we will review the code base that was recorded for our macros, purely for familiarity, and then share references and other materials if you want to dive deeper.

Best search terms in Google....wrt getting help on Excel VBA topics:

“MSDN Excel VBA”

#### References and sources for content in this handout:

<https://support.microsoft.com/en-us/office/quick-start-create-a-macro-741130ca-080d-49f5-9471-1e5fb3d581a8>

<https://www.myexcelonline.com/blog/101-excel-macros-examples/>

<https://www.bluepecantraining.com/portfolio/tips-and-tricks-for-excel-macro-recording/>

<https://www.pcworld.com/article/2880353/5-essential-tips-for-creating-excel-macros.html>

<https://trumpexcel.com/excel-macro-examples/>

<https://www.youtube.com/watch?v=AJOSb8E8Q5w>