

Macros 1

Every action performed in Excel runs a little bit of code that can be recorded into a macro. In this chapter, we will enable macro recording, create a worksheet, and run a macro to duplicate the actions of creating a worksheet.

Practice Spreadsheet

Open a blank workbook for this chapter.

Enable Developer Toolbar

First, the Developer toolbar must be enabled to have access to macro functions. The toolbar is enabled in the Excel Options menu.

1. Go to the **File** tab in the top-left corner of Excel.
2. Press the **Options** button to open the menu.
3. Select **Customize Ribbon**.
4. Press the checkbox for **Developer** in the list on the right to enable the toolbar. (See **Figure 28.1**)
5. Press **OK**.

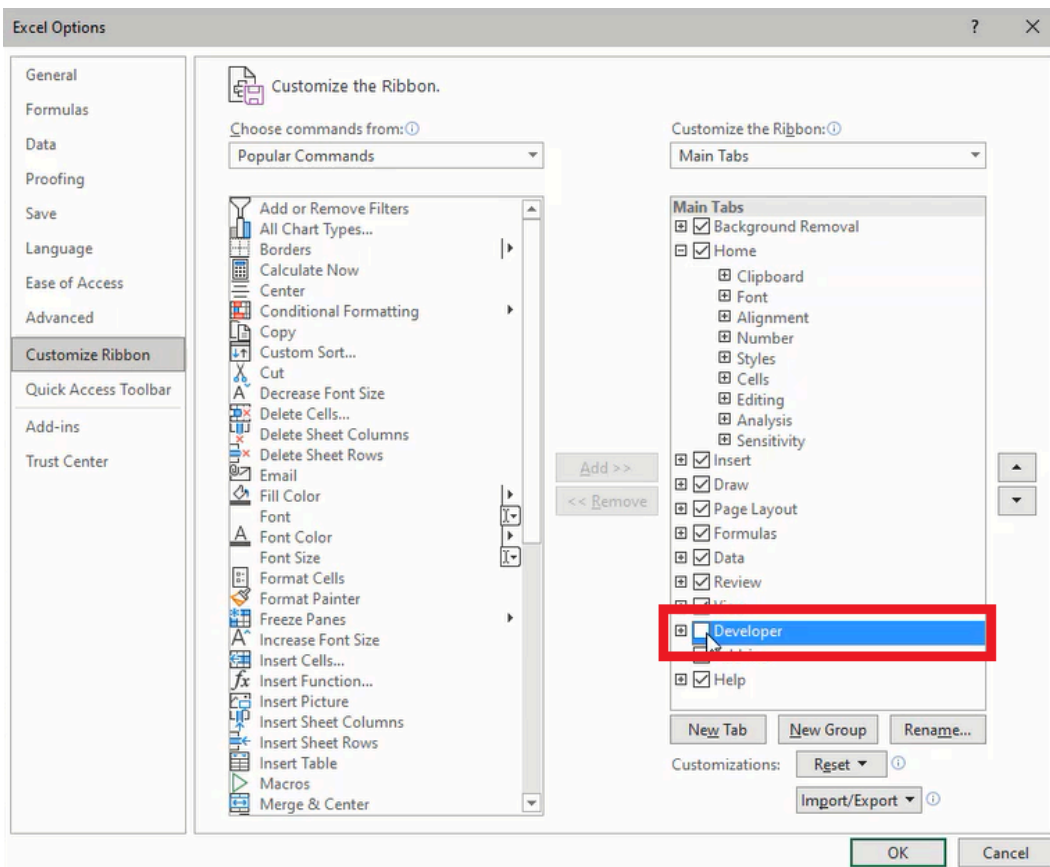


Figure 28.1

Create a Macro

We need to design a timesheet for a company wherein employees can record their hours by week. A new timesheet needs to be created for each week, so we need Excel to generate a duplicate sheet at the press of a button. A macro can perform the task by recording our actions in creating the first sheet, then we can run the macro every time a new sheet is needed.

1. Go to the **Developer** tab.
2. Press the **Record Macro** button.
3. Type a name in the **Macro name** field.
 - a. *Note:* The macro name cannot contain spaces or special characters.
4. (Optional) Type a shortcut key in the **Shortcut key** field.
 - a. Combining the Shift key and a letter key is recommended. Excel uses many preset shortcuts (for example, Ctrl + S is used for the Save function) that may interfere with your macro on the same shortcut.
5. Select **This Workbook** in the **Store macro in** field.
 - a. The Personal Macro Workbook and New Workbook options have distinct purposes. However, the This Workbook option will likely be the most effective for collaborative reasons.
6. Type a **Description** for other users to understand the macro.
7. Press **OK** to begin recording.

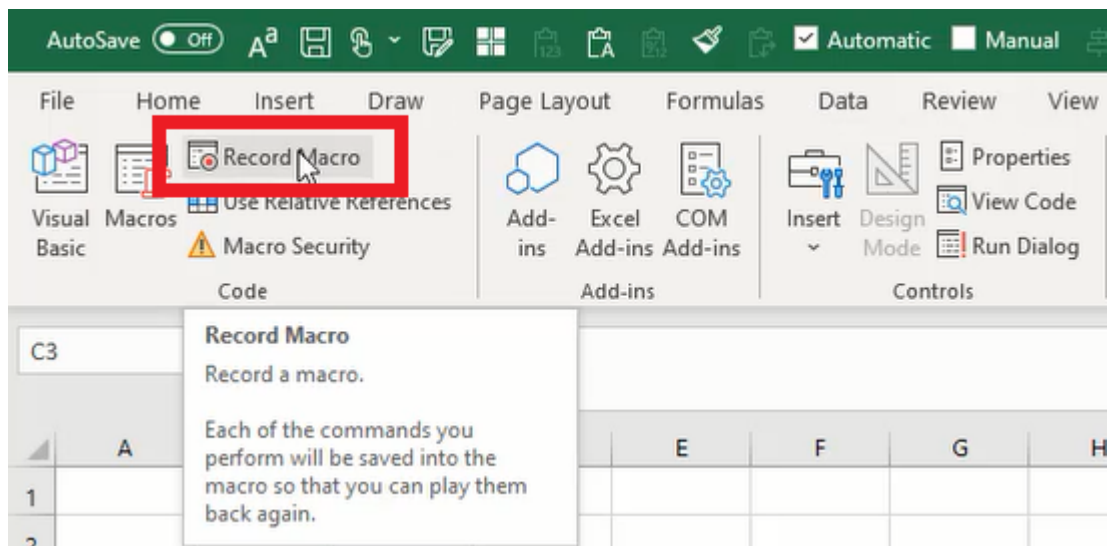


Figure 28.2

Record Actions to a Macro

Upon pressing OK, every action performed in Excel will be recorded in the macro. We will use the time to create and format a new spreadsheet to be the employee's timesheet.

1. Press the plus symbol (+) on the bottom edge of Excel to create a new spreadsheet.
2. Increase Zoom of the spreadsheet to 145%.
 - a. Use the Zoom slider in the bottom right.
 - b. Alternatively, go to the **View** tab, select **Zoom**, and type **145** into the Custom field.
3. Resize column A to a width of **4**.
4. Resize column B to a width of **20**.
5. Resize columns C–G and I to a width of **14**.
6. Resize columns H and J to a width of **4**.
7. Select and Hide all remaining columns after column J.
8. Select and Hide all remaining rows after row 17.
9. Type **Time Sheet** in cell **B2**.
10. Format cells **B2–I2** as **Heading 1**.
11. Type **Weekly hours worked by employee** in cell **B3**.
12. Format cells **B3–I3** as **Explanatory Text**.
13. Type **Monday** in cell **C4** and use the auto-fill function to fill the remaining weekdays across cells D4–G4.
14. Format cells **C4–G4** as **Heading 3**.
15. Type **Employee1** in cell **B5** and use the auto-fill function to fill cells B6–B14.
 - a. For this practice, we'll use a placeholder for the names and only include 10 employees.
16. Type **Total** in cell **I4**.
17. Type **Total** in cell **B16**.
18. Insert SUM formulas into the Total row and column to add the respective cells in the row or column.
19. Go to the **View** tab and deselect **Gridlines**.
20. Select employee and total rows and apply a light gray in between and bottom border lines.
21. Select cell **C5** and return to the dashboard spreadsheet. (See **Figure 28.3**)
22. Go to the **Developer** tab and press **Stop Recording**.

	A	B	C	D	E	F	G	H	I	J
1										
2			Time sheet							
3			<i>Weekly hours worked by employee</i>							
4			Monday	Tuesday	Wednesday	Thursday	Friday		Total	
5		Employee1								0
6		Employee2								0
7		Employee3								0
8		Employee4								0
9		Employee5								0
10		Employee6								0
11		Employee7								0
12		Employee8								0
13		Employee9								0
14		Employee10								0
15										
16		Total	0	0	0	0	0	0		0
17										

Figure 28.3

Run a Macro

The completed macro will perform all of the same steps used to create the timesheet, including a new spreadsheet. To run the macro, go to the **Developer** tab and press the **Macros** button. A menu will open with a list of macros. Select the appropriate macro and press **Run**. (See Figure 28.4)

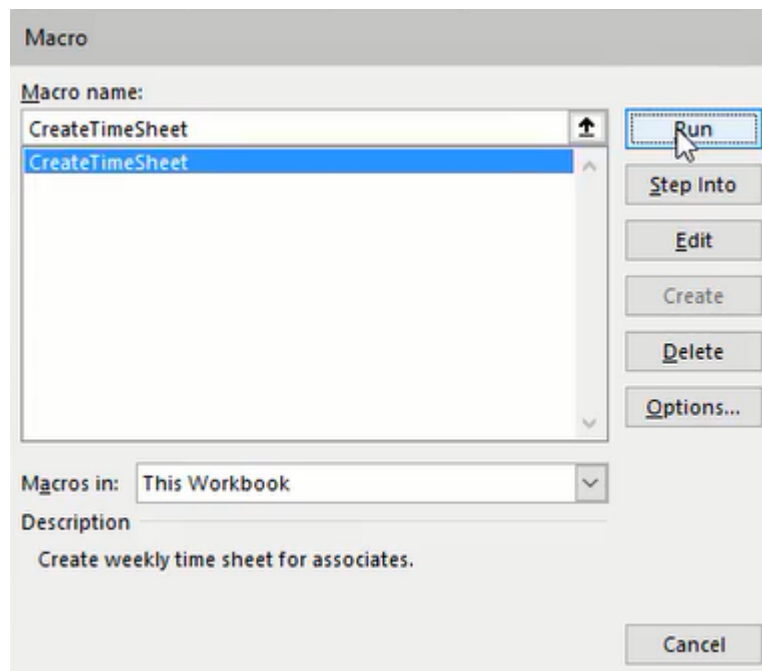


Figure 28.4

We can make finding and using the macro simpler for other users by creating a button on the dashboard spreadsheet. There are two methods to create a button. The first is to go to the **Developer** tab, press the **Insert** button, and select the first rectangular box icon under Form Controls for a simple shape button. A window will open and prompt for a macro to be assigned. Right-click the button and select **Edit Text** to name and identify the button. (See **Figure 28.5**)

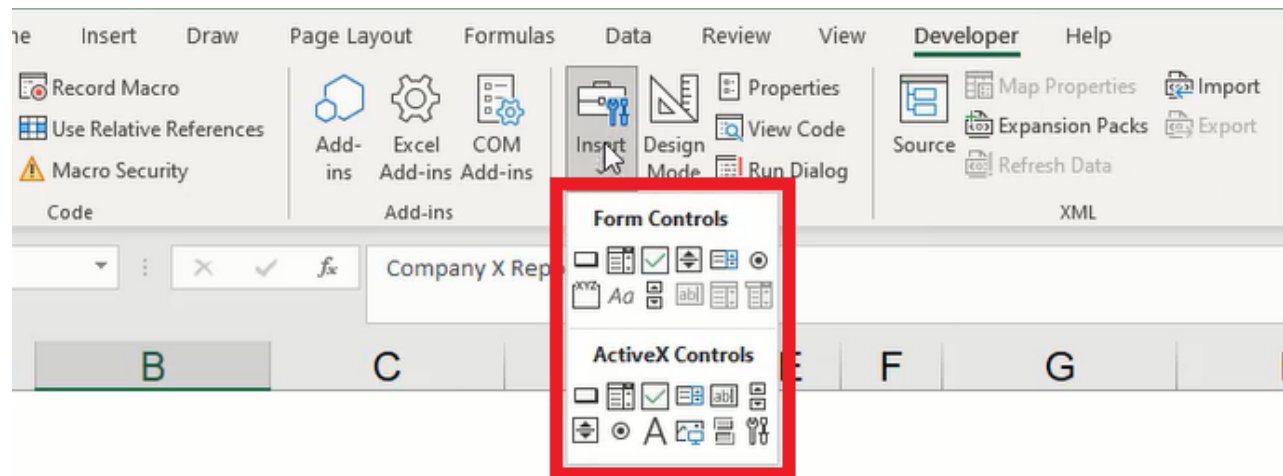


Figure 28.5

The second method is to go to the **Insert** tab, press the **Shapes** button, and select the desired shape. Draw the shape into a cell or multiple cells. Edit the text to identify the button. Right-click on the button and select **Assign Macro...** Choose the appropriate macro and press **OK**. (See **Figure 28.6**)

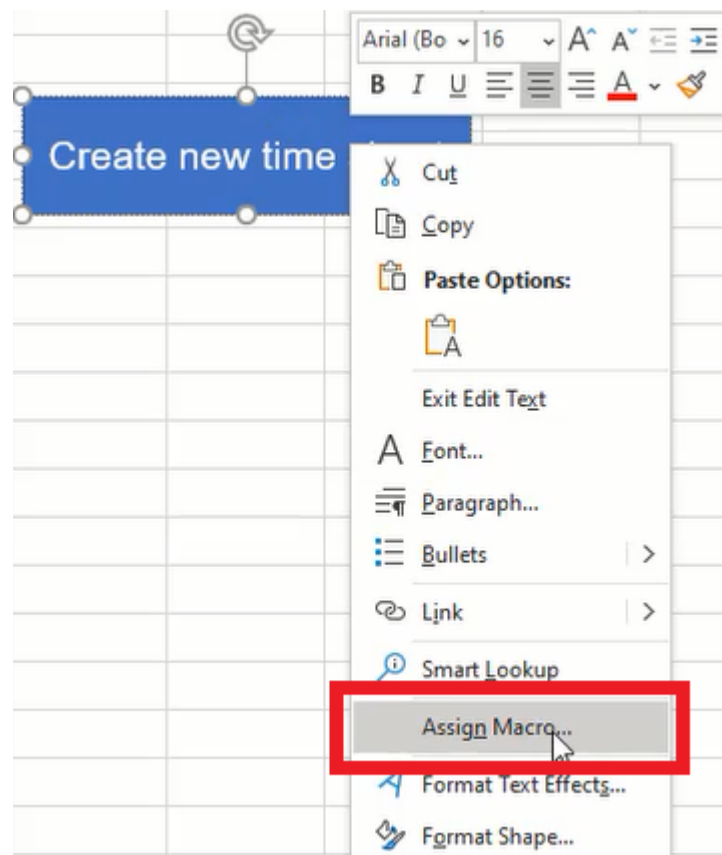


Figure 28.6

The new button will run the assigned macro to effortlessly create additional timesheets.

Review Macro Code

Recorded macros can be edited by pressing the **Visual Basic** button in the **Developer** tab. Visual Basic is a simple coding application that enables users to edit and create macros for Excel. The application will be explored in more detail in the next chapter.

Save a Macro-Enabled Workbook

It is important to note that a workbook with macros will need to be saved as a specific file type: Excel Macro-Enabled Workbook (.xlsm). If the workbook is not saved as macro-enabled, the associating macro(s) will be lost.

Additionally, macro-enabled workbooks will open with macros disabled and a security notice will be displayed under the ribbon toolbar. Macros are disabled by default because they could potentially contain harmful computer code. You should only open a macro-enabled workbook from a source that you trust. (See **Figure 28.7**)

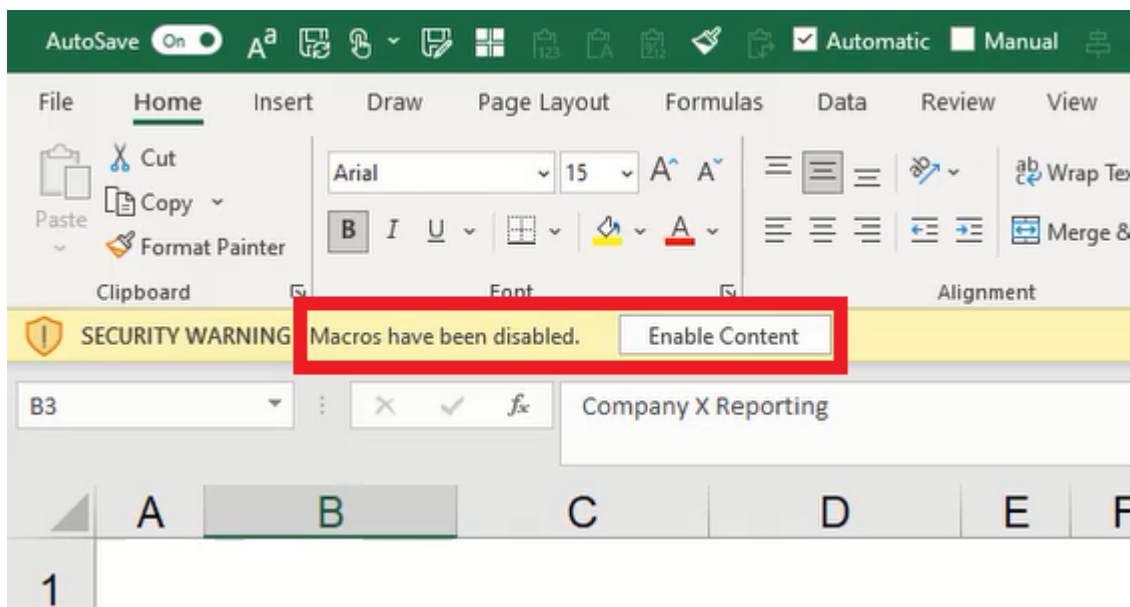


Figure 28.7

Supplemental Resource



This content is provided to you freely by BYU-I Books.

Access it online or download it at https://books.byui.edu/bus_115_business_app/macro_1.

