

Microsoft Excel 2003

Macros



About Macros

I'm sure you have used Excel enough to know that there are certain tasks that you seem to repeat over and over again. I'm sure you have thought, there must be a way to automate these repeated tasks so that tasks can be repeated with minimal interaction by you. For this, you need to create a Macro. A macro is a huge time saving feature that enables you record common or repetitive tasks and replay these tasks by "running" the macro when needed.

In Excel, rather than writing each step as code, you record a macro similar to recording a show on your VCR—except you are the show! In other words, you start recording and Excel records each mouse click and keystroke you make as you perform the task. When you are done, you stop recording. Voila! Your macro is ready and is now available. Run the macro to repeat the same commands. Should you make a mistake, you can either re-record the macro or change it by using the Visual Basic for Applications (VBA) Editor.

Think through the steps that you want to record in a new macro.

Examples of Macros:

1. Insert a desired TU logo and set page setup.
2. Format column headings and data.
3. Sort data in multiple columns.
4. Find and Replace text.

Macros

1. About Macros
2. Set Up Excel to Allow Macros
3. Open a Workbook that Contains Macros
4. Run Macro
5. Record a Macro
6. About Shortcut Keys
7. Modify a Macro
8. About Visual Basic for Applications (VBA)
9. Create a Custom Toolbar and a Custom Toolbar Button
10. Customize the Toolbar Macro Button
11. Create a Worksheet Macro Button
12. Delete the Custom Toolbar
13. Delete a Macro

Set Up Excel to Allow Macros

Spreadsheets can contain macros that perform various automated tasks. Although most macros are harmless and very useful, some can contain malicious macro viruses. Macro viruses can destroy data and so Excel lets you decide to disable or enable macros through the uses of Security Settings. There are four levels of security ranging from Low to Very High. Setting the security level is done on the individual workstation and is a personal decision. By default the Security Level is High. If the Excel macro security level is set to Very High, most, to all, macros will be disabled. You may want to change the Security Level before you open spreadsheets from different sources.

In the following exercise, you will set the Security level to Medium.

1. Open Excel.
2. Select **Tools > Macro > Security** from the menu bar.
3. In the Security dialog box, select **Medium** (Figure 1, 1).
4. When ready, click **OK** (Figure 1, 2) to close the Security dialog box.

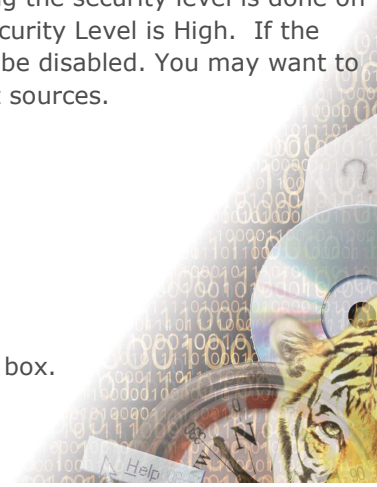
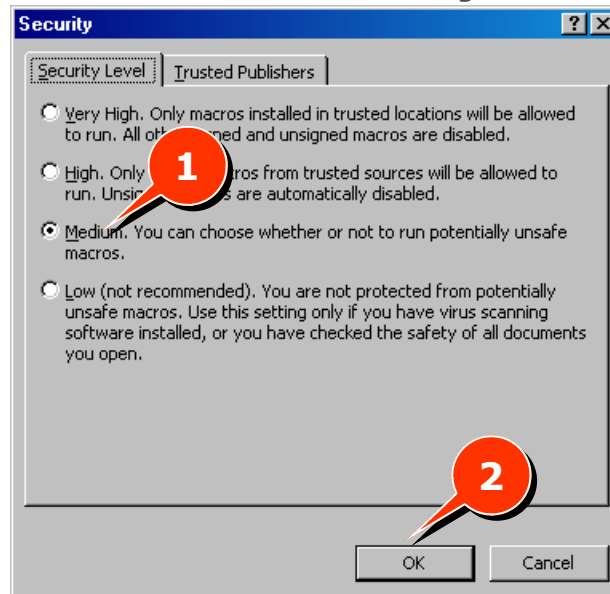


Figure 1



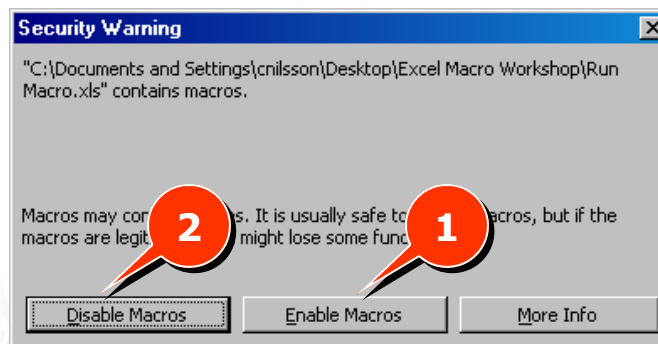
Open a Workbook that Contains Macros

When you open or reopen a file that contains a macro, you may see the Excel Security Warning (Figure 2). Excel is warning you that the file you are about to open contains macros. Because macros contain programming code, they have the potential to produce harmful effects on your system. If the Security Level is set to High (by default), Microsoft Excel will turn off the ability to run macros.

If you are sure that the macros in the file are perfectly harmless (perhaps you created them yourself, or the file came from someone you know) then click on "Enable Macros" (Figure 2,1). to open the file and be able to use the macros within the workbook.

If you want to open the file but do not want the macros to be able to run, click on "Disable Macros" (Figure 2,2). This option will open the file in "Read Only" mode, which means you can view the file, make changes and save it under a different name if you so chose.

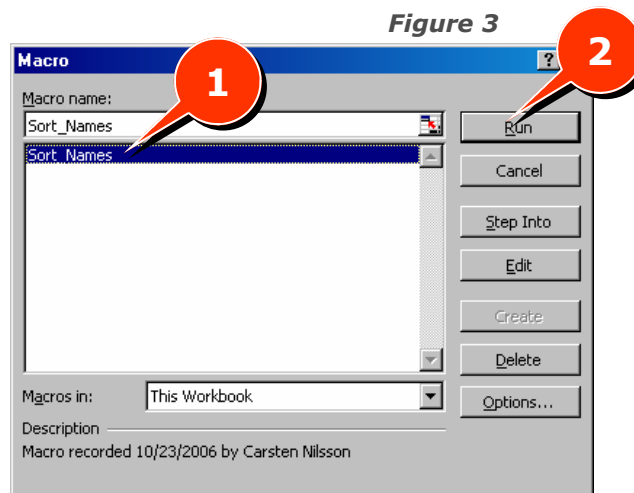
Figure 2



Run an Existing Macro

The following demonstrates how to run an existing macro. Once you have run a macro, please notice the Undo button is not available. If you must undo the macro changes, close the file and do not save changes.

1. Open the desired file.
2. When prompted, select **Enable Macros**.
3. Select **Tools > Macro > Macros** from the menu bar.
4. In the **Macro** dialog box, select the desired macro (Figure 3, 1), and then click **Run** (Figure 3, 2).



Record a Macro

The following demonstrates how to record a simple macro that inserts a TU logo located in the Workshop Files folder on the Desktop. Once the logo is inserted, you will also record sizing the logo to a proper size and place the logo in the upper right corner of the worksheet.

1. Open the desired file and display the desired worksheet
2. Select **Tools > Macro > Record New Macro** from the menu bar.
3. In the **Macro Name** field (Figure 4, 1), type a desired macro name, e.g. "Insert_Logo".

Note: A macro name can be up to 255 characters. It must be unique and may not include any spaces in the name.

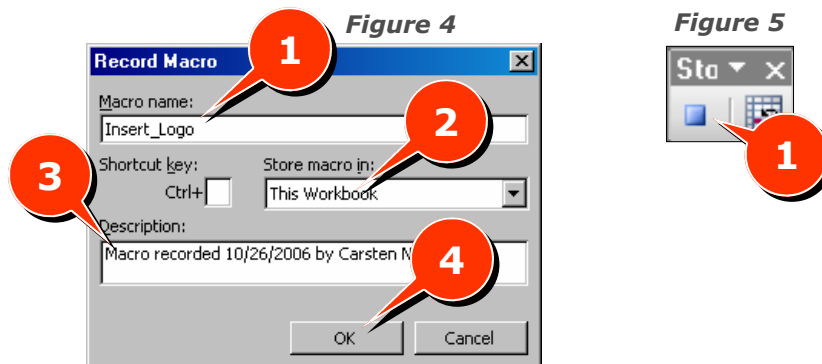
4. Stored the macro in "This Workbook" (Figure 4, 2).

Note: By default, the macro will be stored and available only in the current workbook; if appropriate, open the Store macro in drop-down list and select either New Workbook (to store and use the macro only in a new empty workbook) or Personal Macro Workbook (to make the macro available in all workbooks).

5. In the **Description** field (Figure 4, 3), type a desired description, e.g. "This macro inserts the TU Logo".
6. Click the **OK** button (Figure 4, 4) to begin recording.

- Note: The word "**Recording**" will appear on the status bar to remind you that all keystrokes and mouse actions are now being recorded. Depending on how your system is configured, a Stop Recording toolbar may also appear in the window. If you make a mistake, simply correct it as you normally would and continue; both the mistake and its correction will become part of the macro, and may be edited out later if desired.
- Perform the necessary action to complete the macro, e.g. insert the TU logo from My Pictures or a network location.
- When ready, click the STOP button (Figure 5, 1) on the Stop Recording toolbar.

Note: if the toolbar is not visible, select Tools > Macro > Stop Recording.



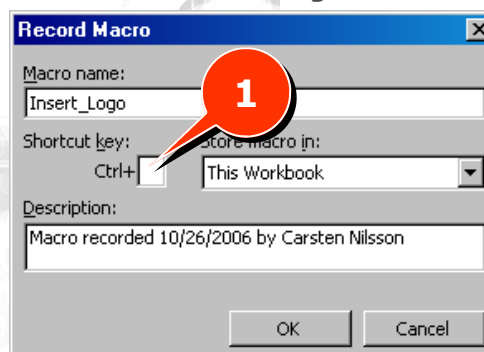
About Shortcut Keys

As an alternative to running a macro from the Tools > Macro menu, you can also assign a keyboard shortcut key (also known as a "hot key"). When you record a new macro, you have the option of assigning a shortcut key to it while the **Record Macro** dialog box is displayed. To assign a shortcut key, click the **Shortcut key** text box and either type a letter key (to produce a Ctrl shortcut key) or type a letter key while holding down the Shift key (to produce a Ctrl-Shift shortcut key).

If you have already recorded the macro and you can assign or change its shortcut key by doing the following:

- Select **Tools > Macro > Macros** from the menu bar.
- In the **Macro** dialog box, select the name of the macro and click the **Options** button.
- In the Shortcut Key field, type the desired shortcut key (Figure 6, 1) as indicated above.
- When ready, click the **OK** button to close the Macro Options dialog box.
- Click the **OK** button to close the Macro dialog box.

Figure 6




Modify a Macro

Visual Basics for Applications (VBA) is the programming language used to create macros in the Microsoft Office 2003 applications. When you record a macro, Excel translates the keystrokes and mouse clicks into VBA code and creates and stores the macro for you.

Macros are grouped together in larger VBA code blocks called Modules. An Excel file can contain one or more modules, and a module can contain one or more macros. Since a macro is a set of organized steps, a macro is also called a Procedure. All macro code can be viewed in the Code window (Figure 7, 1). In the VBA, in order to see the macro code, you must first double-click the Modules folder under the file name in the Project Explorer (Figure 7, 2). Each Module can have many macros/procedures. Each macro/procedure begins with a "Sub" and end with an "End Sub" statement. All text displayed in black is the actual macro code and any text displayed in green are comments and will not be executed. To switch from the VBA back to Excel, you can either close the VBA window or click the View Microsoft Excel button (Figure 7, 3)

The following demonstrates how to start up the VBA Editor.

1. Select Tools > Macro > Visual Basic Editor.

 **Shortcut:** You can also press ALT+F11 on the keyboard.

Objects to be aware of in the Visual Basic Editor:

 **Need Help?** While in the VBA, press F1 on the keyboard to startup the Help Screen.

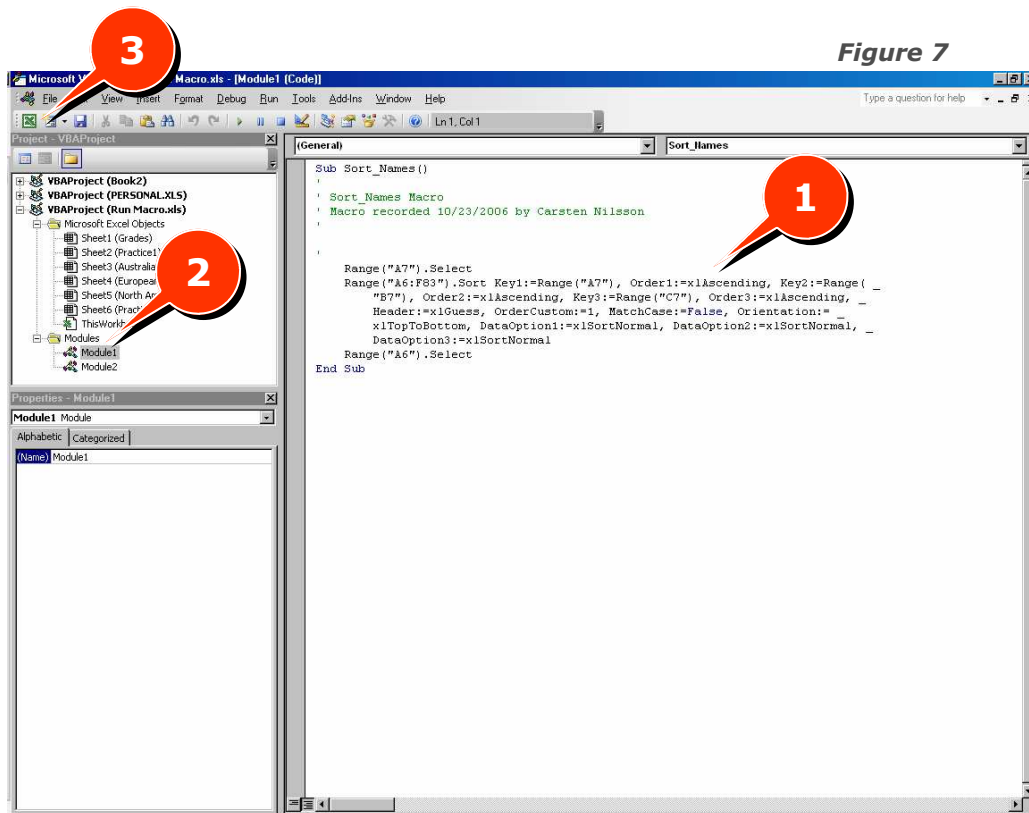


Figure 7

Create a Custom Toolbar and a Custom Toolbar Button

As an alternative to running a macro from the Tools > Macro menu, you can also create a custom button and place it on any of the existing toolbars or on a custom toolbar. In the next exercise, we will create a new macro toolbar, then place a new macro toolbar button on the toolbar. Once the button is created, we will change its appearance and screen tip.

To Create a Custom Toolbar:

1. Select **View > Toolbars > Customize**.
2. In the **Customize** dialog box, select the **Toolbars** tab.
3. On the **Toolbars** tab, click the **New...** button.
4. In the **New Toolbar** dialog box, type **"My Macros"** (Figure 8, 1) and click **OK** (Figure 8, 2).
5. Click **Close** to close the Customize dialog box, or go on to the next section to add a custom toolbar button.

Figure 8



To Create a Custom Toolbar icon:

1. Select **View > Toolbars > Customize**.
2. In the **Customize** dialog box, select the **Commands** tab.
3. Under **Categories**, scroll down to **"Macros"** (Figure 9, 1).
4. Under **Commands**, select and drag the **Custom Button** icon (Figure 9, 2) onto any existing toolbar or the new "My Macros" toolbar created in the previous section.
5. Right-click the new custom button and select **Assign Macro**.
6. In the **Assign Macro** dialog box, select the desired macro (Figure 10, 1) and click **OK** (Figure 10, 2).

Figure 9

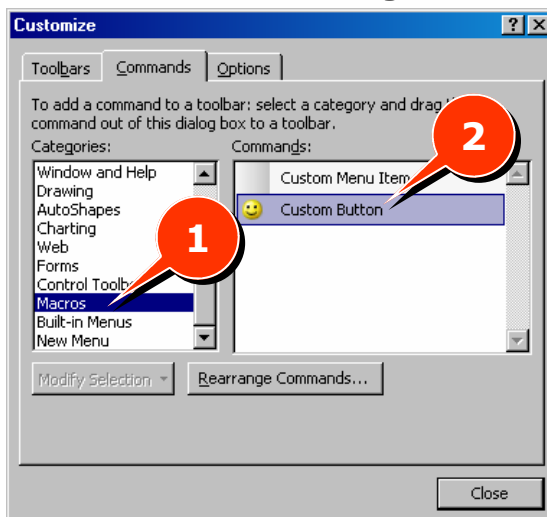
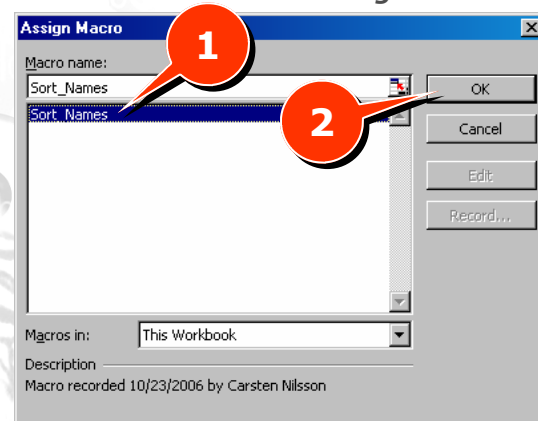


Figure 10



Customize the Button Image and Tool Tip

1. While the Customize dialog box is open, right-click the new macro button and select **Change Button Image**.
2. Select any desired image for the new macro button (Figure 11).
3. Right-click the new macro button again and select the text in the **Name** field, and type a desired button name, e.g. "Insert TU Logo".
4. Click away from the shortcut menu to close the menu.
5. Click the **Close** button in the Customize dialog box to close the dialog box.

Note: The macro can now be executed using the custom toolbar button.

Figure 11



Create a Worksheet Macro Button

A convenient way to run a macro is to create a Command Button on the actual worksheet. You can create a command button using the Forms toolbar. The Forms toolbar is not turned on by default, but can easily be turned on if needed.

To Turn on the Forms Toolbar

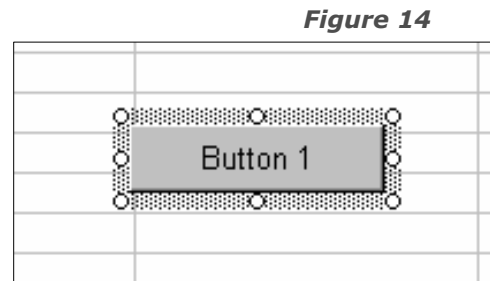
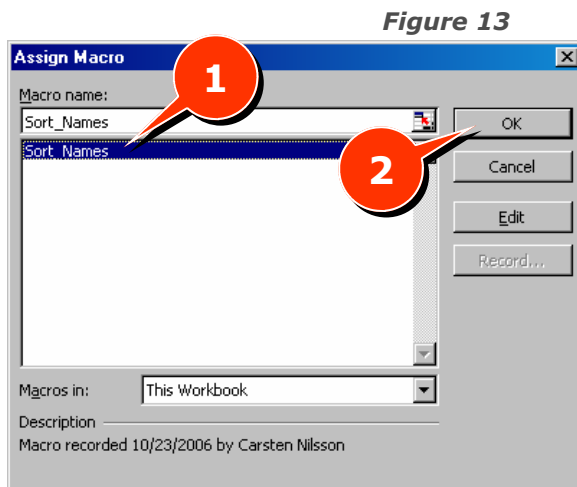
1. Open the desired file.
2. Select **View > Toolbars > Forms** to display the Forms toolbar.



Figure 12

To Create a Command Button on the Worksheet

1. On the **Forms** toolbar, click the **Button** icon (Figure 12, 1).
2. Click and hold the mouse button and drag the pointer on the worksheet to indicate the location and shape of the command button you wish to create.
3. Release the mouse button and the command button will be created and the Assign Macro dialog box displays.
4. In the **Assign Macro** dialog box, select the desired macro (Figure 13, 1), and click the **OK** (Figure 13, 2) button.
5. Select the text ("Button 1") on the button face (Figure 14) and type a desired new button name.
6. Click away from the in the worksheet, but not on the command button, to de-select the command button.



Delete the Custom Toolbar

1. Select **View > Toolbars > Customize** from the menu bar.
2. In the Customize dialog box, select the **Toolbars** tab.
3. Scroll to the bottom of the Toolbars section and select the desired toolbar.
4. Click the **Delete** button.
5. Click **OK** to confirm deletion of the custom toolbar.
6. Close the Customize dialog box.

Delete an Existing Macro

1. Open the desired workbook.
2. Select **Tools > Macro > Macros** from the menu bar.
3. In the **Macro** dialog box, select the desired macro, and then click **Delete**.
4. Click **Yes**, to confirm deletion of the macro.