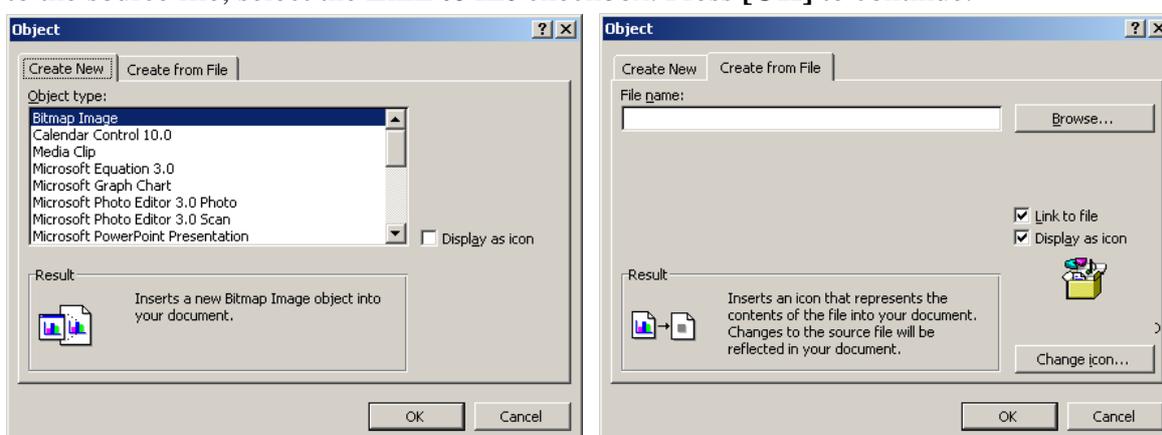


# 1. Integrating Excel with Other Applications

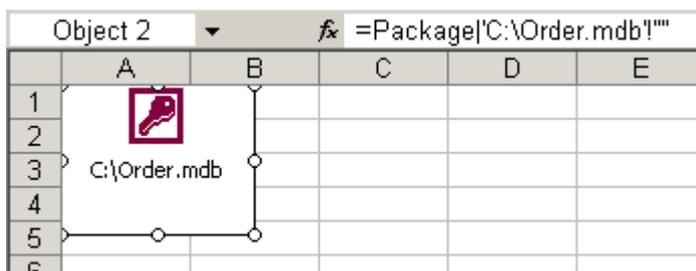
## 1.1 Linking and Embedding

You can display a linked object or embedded object in a workbook exactly as it appears in the source program or as an icon. If the workbook will be viewed online, and you don't intend to print the workbook, you can display the object as an icon. This minimizes the amount of display space it occupies. Viewers who want to display the information can double-click the icon.

1. Select **Insert** → **Object**.
2. You can either create a new object, or directly create it from a file. If you want to link the data to the source file, select the **Link to file** checkbox. Press **[OK]** to continue.



3. The new object will be embedded to your worksheet..



## 1.2 Using Hyperlinks

A hyperlink is a link from a document that opens another page or file when you click it. The destination is frequently another Web page, but it can also be a picture, or an e-mail address, or a program. The hyperlink itself can be text or a picture.

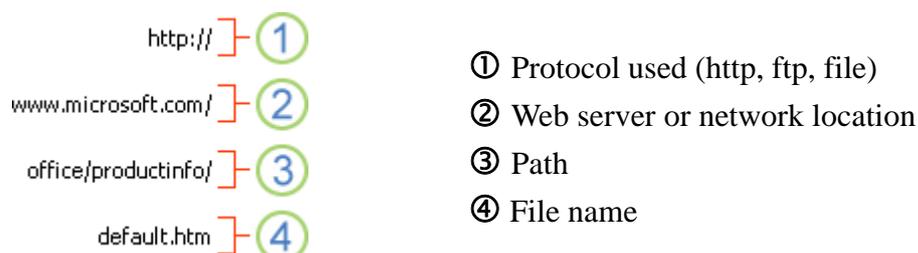
When a site user clicks the hyperlink, the destination is shown in a Web browser, opened, or run, depending on the type of destination. For example, a hyperlink to a page shows the page in the Web browser, and a hyperlink to an AVI file opens the file in a media player

You can use hyperlinks to do the following:

- Navigate to a file or Web page on a network, intranet, or Internet
- Navigate to a file or Web page that you plan to create in the future
- Send an e-mail message
- Start a file transfer, such as downloading or an FTP process

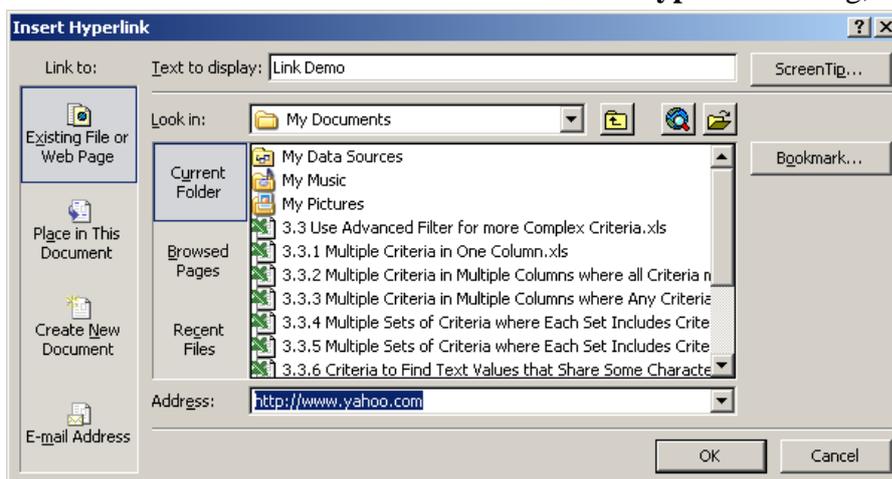
When you point to text or a picture that contains a hyperlink, the pointer becomes a hand, indicating that the text or picture is something that you can click.

A URL (Uniform Resource Locator) contains a protocol, FTP, or FILE, a Web server or network location, and a path and file name. The following illustration defines the parts of the URL:



To create a hyperlink, follow these step:

1. On a worksheet, click the cell where you want to create a hyperlink. You can also select an object (such as a picture or an element in a chart) that you want to use to represent the hyperlink.
2. Select **Insert** → **Hyperlink**.
3. Enter the URL in the **Address** box of the **Insert Hyperlink** dialog, and press [OK] to confirm.



4. The hyperlink is created for the specified cell. A tooltip will be displayed once you move your mouse on that cell

|   | A   | B | C | D |
|---|---|---|---|---|
| 1 | Link Demo   |   |   |   |
| 2 | http://www.yahoo.com/ - Click once to follow. Click and hold to select this cell. |   |   |   |
| 3 |   |   |   |   |
| 4 |   |   |   |   |

## 1.3 Using Excel Data in Word and PowerPoint Documents

Its many presentation features notwithstanding, Microsoft Excel is at heart an analytical tool. When it comes time to organize information and present it to others, you're likely to turn to two other programs in the Microsoft Office System, Word and PowerPoint. Naturally, these programs are designed to work hand-in-glove with Excel, so you can easily do your number crunching in Excel and transfer the results to Word or PowerPoint when you need to make a verbal or visual presentation

### 1.3.1 Pasting an Excel Table from the Clipboard

If you copy an Excel worksheet range to the Clipboard and then paste that range into Word, the smart-tag menu that appears below the lower-right corner of the pasted data, shown in Figure 23-1, provides quick access to the formatting options that Microsoft thinks you're most likely to want. These options are as follows:

- **Keep Source Formatting.** Word receives the data as a block of HTML and creates a table, preserving the fonts, alignment properties, numeric formatting, text color, and shading of your original. In most, but not all, cases, this option (the default) is an adequate way to create a table in Word that matches the appearance of your Excel data. After you have performed the paste, you can use commands on Word's Table menu to modify such characteristics as column widths and the position of the table within surrounding text.
- **Match Destination Table Style.** The data becomes a table in Word, but Word formats it as if you had created the table directly in Word. Numeric formatting and character styles that you used in Excel are preserved, but in most other respects your data will look as though you typed it in Word rather than pasting it from Excel. If your default font in Word is Times New Roman, for example, your table will appear in Times New Roman, regardless of what font you used in Excel. Use this option if you want to make the table's appearance blend in maximally with the rest of your document. Note, however, that if your data in Excel uses non-default alignments, you'll probably need to do some formatting after it arrives in Word. As with the default choice (Keep Source Formatting), Word's Table menu is available to help you with column widths, text-flow options, and so on.
- **Keep Text Only.** If you choose this option, Word does not create a table. Instead, it simply pastes each cell's contents in the current default font, separating cells with single tab characters. You might find this option useful if you're simply copying a single column from Excel. Where multiple columns are involved, the Keep Text Only choice usually produces a misaligned hash in Word.
- **Apply Style or Formatting.** This option behaves like the Keep Source Formatting option, but causes Word to display the Styles and Formatting task pane, allowing you to apply styles or character formatting immediately.

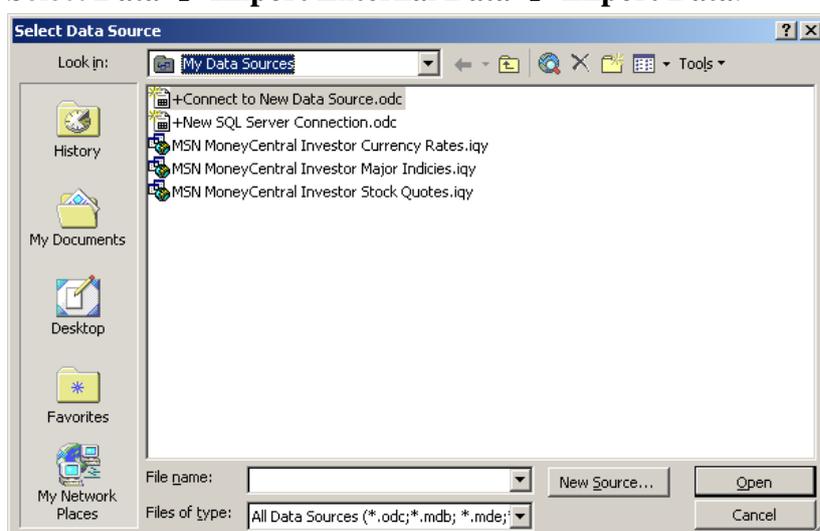
## 2. Working with External Data

You can compile your data into Excel by retyping or by copying and pasting. But importing is most effective when you're working with large amounts of data that would be time-consuming to enter manually or too large to copy and paste. In addition, if you import your data, Excel can automatically update your reports and summaries whenever the original source database is updated.

### 2.1 Import External Data Files

You can use these procedures to import data from most data sources, including OLAP databases.

#### 1. Select **Data** → **Import External Data** → **Import Data**.



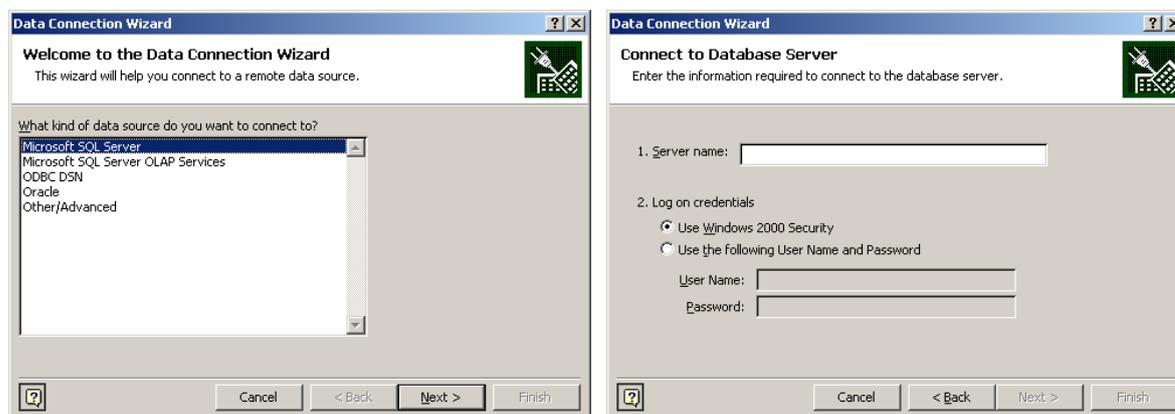
#### 2. You can either open and open an existing data source or create a new one

##### A) Open an existing data source

- In the **Select Data Source** dialog box, select a data source from **My Data Sources**, and then click [**Open**].

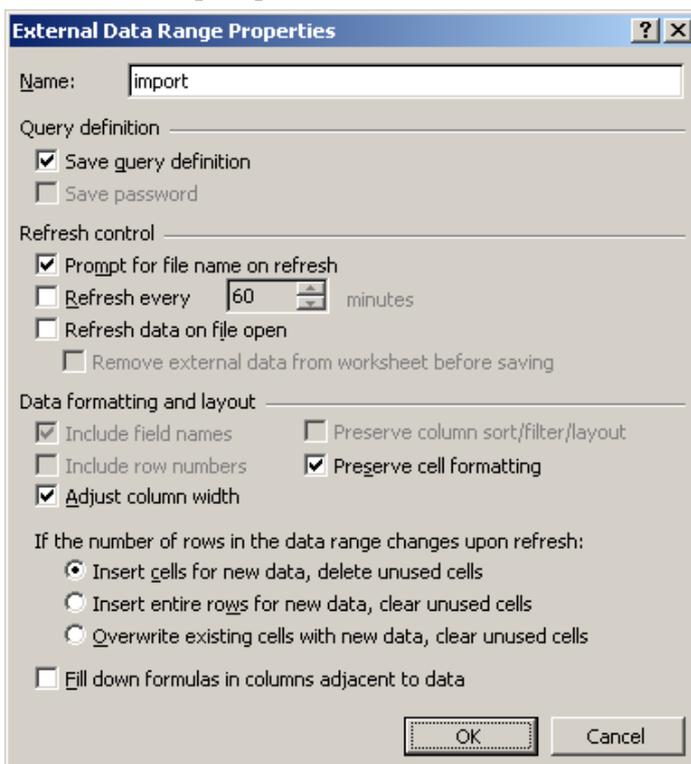
##### B) Create a new data source connection

- In the **Select Data Source** dialog box, click [**New Source**]. Choose a data source to connect to in the **Data Connection Wizard**, and then click [**Next**].

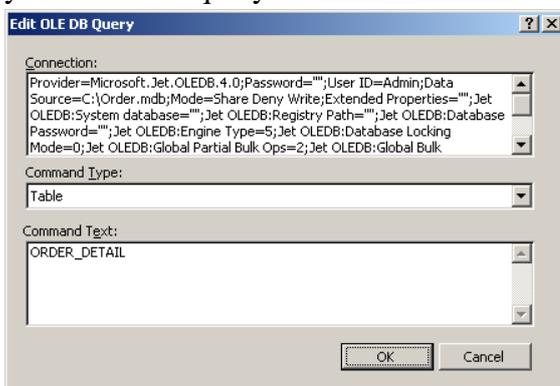


#### 3. You can customize the properties in the **Import Data** dialog box:

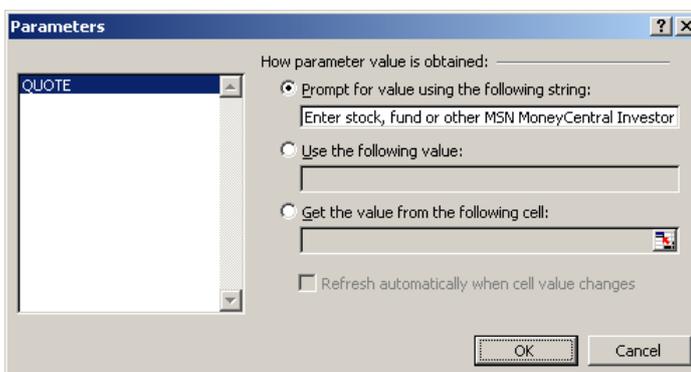
- Click [**Properties**] to choose query definition, refresh, and formatting and layout options, and then click [**OK**].



- Click [**Edit Query**] to edit the **Connection**, **Command Type**, and **Command Text** of your OLE DB query.



- For a query created with Microsoft Query or for a Web query, click [**Parameters**] to specify how a parameter value is obtained for the external data range.

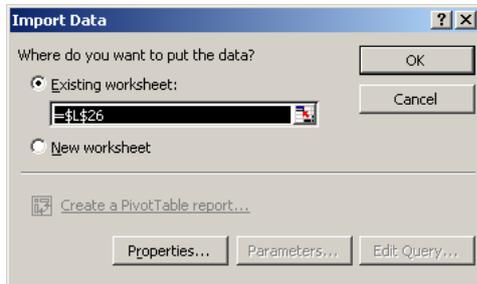


4. Locate the data to be placed

- To import the external data to the selected worksheet, click **Existing worksheet** and

select the area you want to place the imported data, and then click **[OK]**.

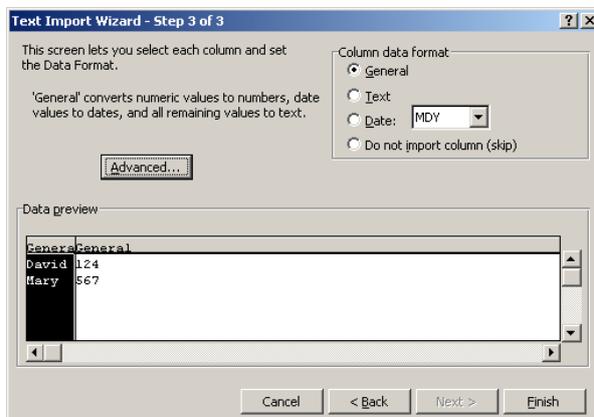
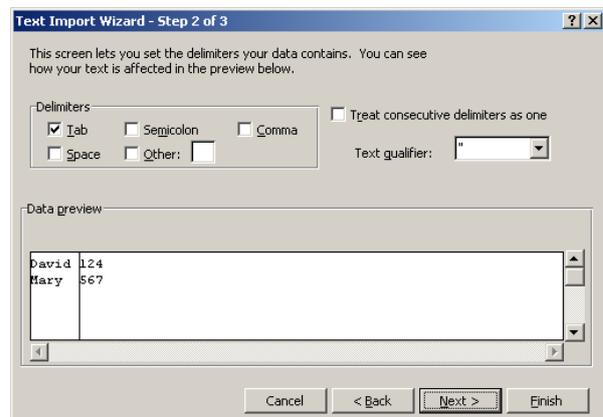
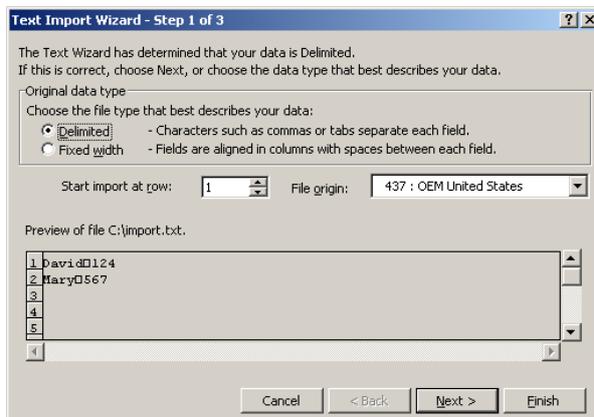
- To import the external data to a new worksheet, click **New worksheet**, and then click **[OK]**. Excel adds a new worksheet to your workbook and automatically puts the external data range in the upper-left corner of the new worksheet.



### 2.1.1 Import a Text File

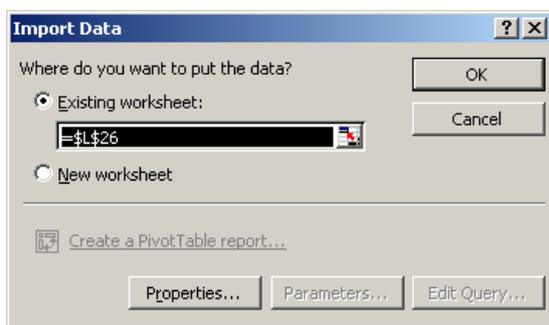
In order to import a text file in fixed-length or tab-delimiter format, follow these steps:

- Click the cell you want to put the data and then select **Data → Import External Data → Import Data**.
- In the **Select Data Source** dialog box, select **Text Files** in the **Files of type** box, and locate and double-click the text file you want to import as an external data range in the **Look in** list.
- To specify how you want to divide the text into columns, follow the instructions in the **Text Import Wizard**, and then click **[Finish]**.



- In the **Import Data** dialog box, click **Properties** to set formatting and layout options for the imported data.
- In the **Import Data** dialog box, do one of the following:

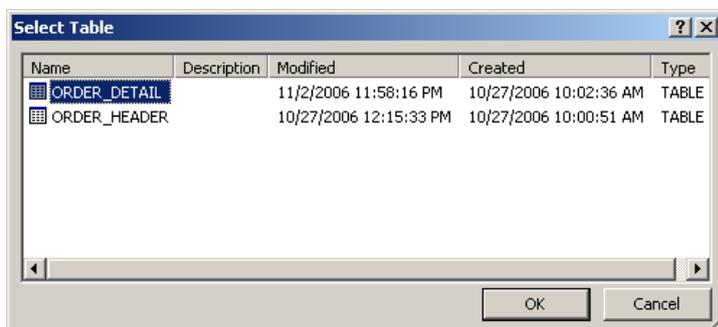
- To return the data to the location you selected, select **Existing worksheet**, and then click **[OK]**.
- To return the data to a new worksheet, select **New worksheet**, and then click **[OK]**. Excel adds a new worksheet to your workbook and automatically puts the external data range in the upper-left corner of the new worksheet.



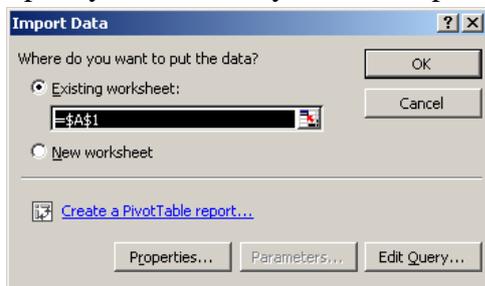
## 2.1.2 Import Microsoft Access Tables

In order to import a Microsoft Access table, follow these steps:

1. Click the cell you want to put the data and then select **Data → Import External Data → Import Data**.
2. In the **Select Data Source** dialog box, select **Access Database** in the **Files of type** box, and locate and double-click the text file you want to import as an external data range in the **Look in** list.
3. Specify which table you wish to import in the **Select Table** dialog, and then click **[OK]**.



4. Specify the location you want to place the import data and click **[OK]** to finish.



## 2.1.3 Importing an XML List

In order to import a XML file, follow these steps:

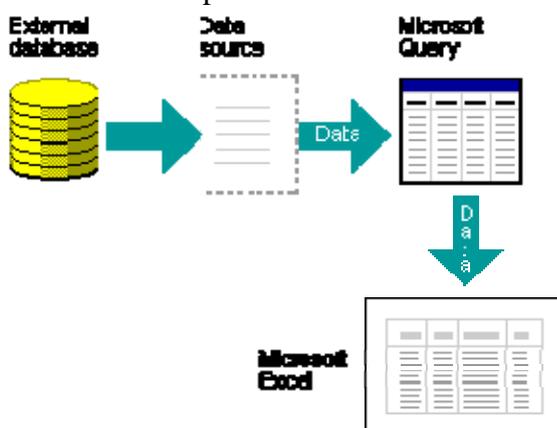
1. Click the cell you want to put the data and then select **Data → Import External Data → Import Data**.

2. In the **Select Data Source** dialog box, select **XML File** in the **Files of type** box, and locate and double-click the text file you want to import as an external data range in the **Look in** list.
3. Specify the location you want to place the import data and click **[OK]** to finish.

## 2.2 Using a Query to Retrieve External Data

In most cases, you can import data by using the Import Data command as described in the preceding section. Use Query or another program only if you need to perform specialized query tasks such as the following:

- Filter rows or columns of data before they are brought into Excel.
- Create a parameter query.
- Sort data before it is brought into Excel.
- Join multiple tables.



### 2.2.1 Prerequisites of Using Query

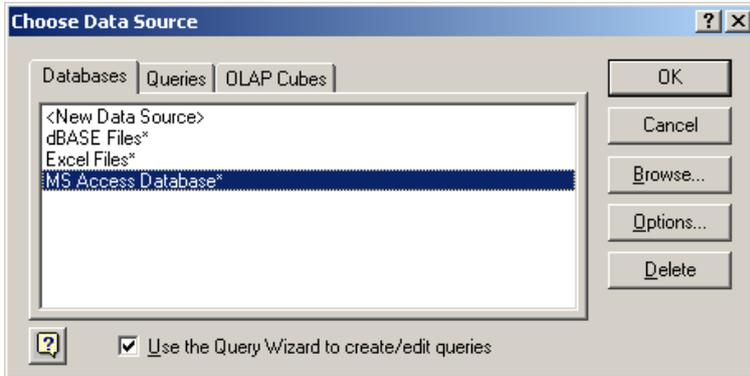
To import data using Query, you must first:

1. **Install Query.** Query including the Query Wizard, is an optional feature for Excel. Under most circumstances, you are prompted to install Query when you point to Import External Data on the Data menu and then click New Database Query.
2. **Install ODBC drivers.** An ODBC driver is required to retrieve data in relational databases, text files, or Excel using Query. When you install Query, you automatically install a set of ODBC drivers. If you use a driver other than one installed with Query, you must install the driver separately.
3. **Install data source drivers.** A data source driver is required to retrieve OLAP source data. Query supports connecting to databases that are created by using SQL Server OLAP Services; when you installed Query, you automatically installed support for this type of OLAP database. To connect to other OLAP databases, you must install a data source driver and client software.

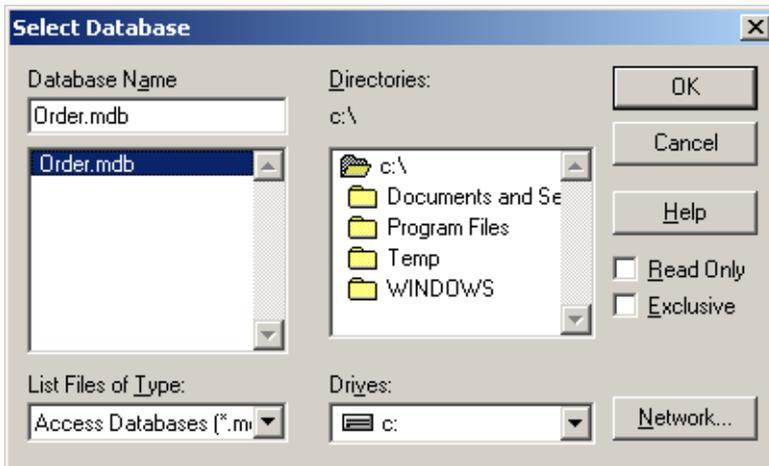
### 2.2.2 Creating a New Database Query

In order to import a Microsoft Access table, follow these steps:

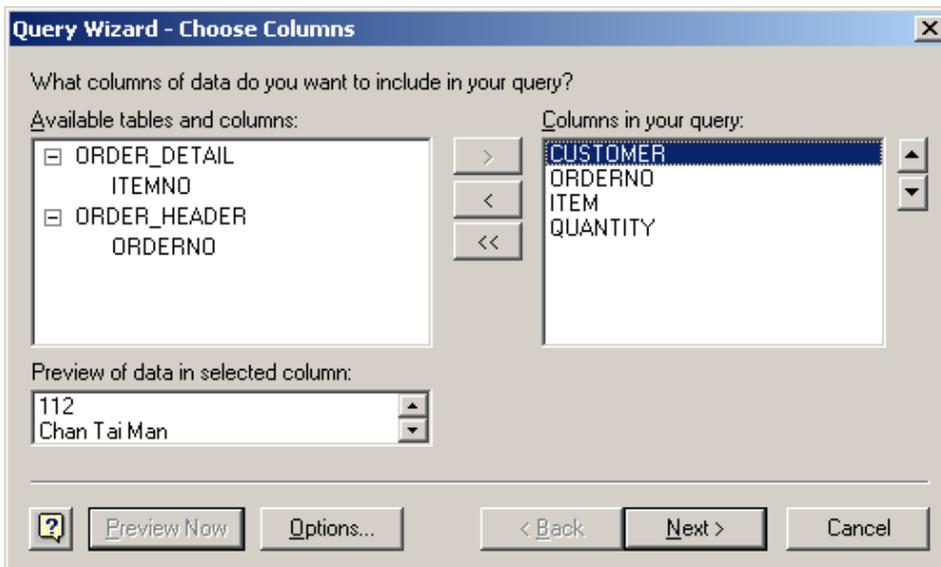
1. Click the cell you want to put the data and then select **Data → Import External Data → New Database Query**.
2. Select the databases in the **Choose Data Source** dialog. For example, we use the Access database in this demo.



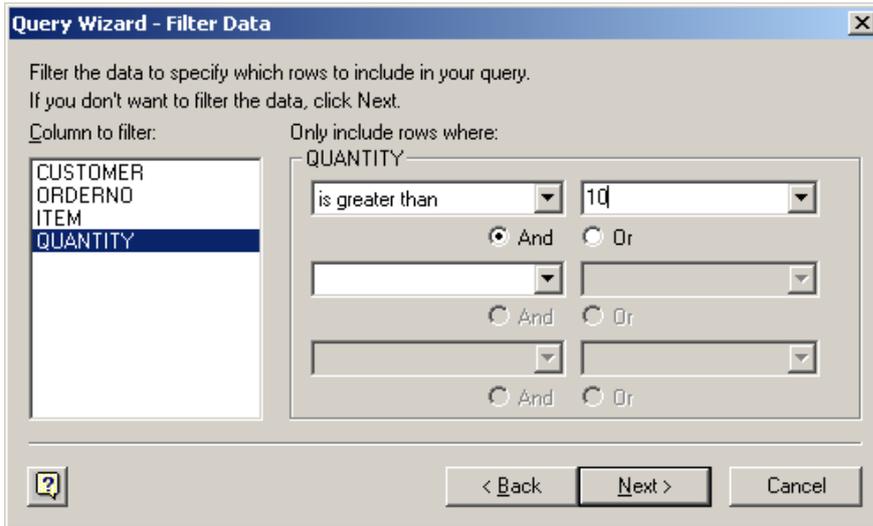
3. Specify the location of the database file in the **Select Database** dialog.



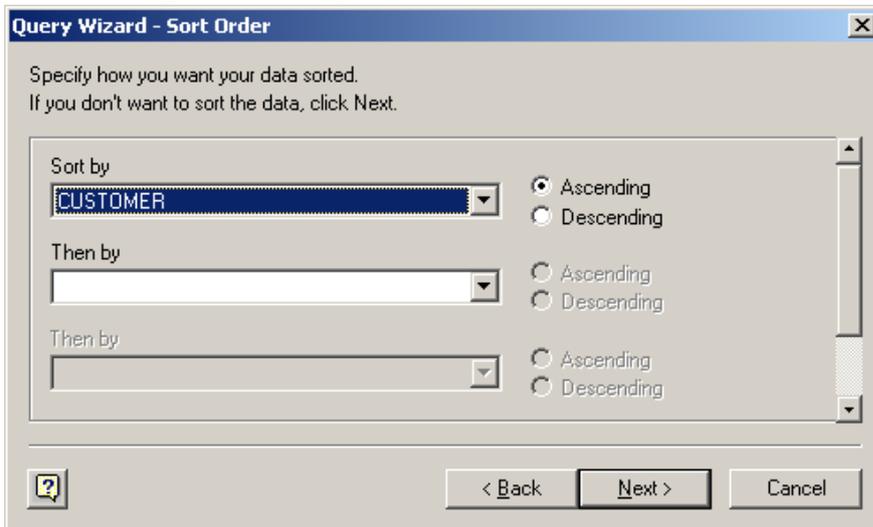
4. Select the columns you wished to import in the **Choose Columns** dialog of the **Query Wizard**. You can use the **[Preview Now]** button to preview the import data. Press **[Next]** to continue.



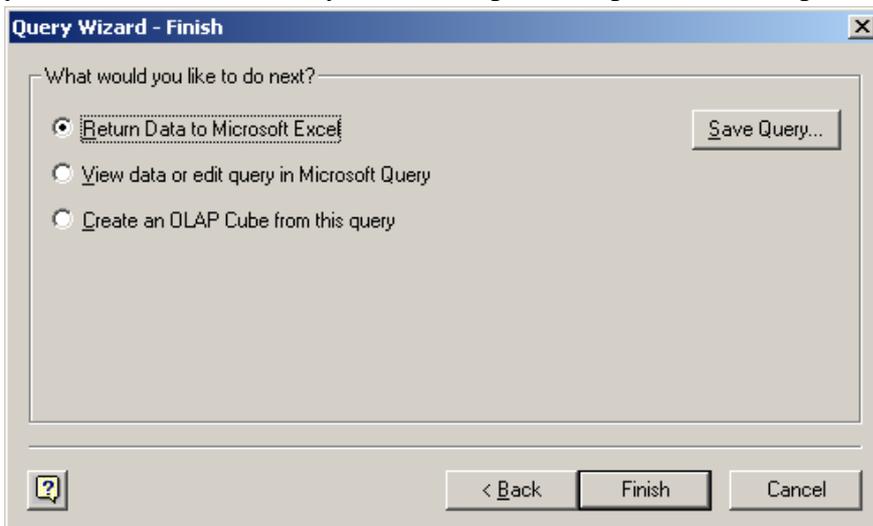
- 5. Define the selection criteria for the import data in the **Filter Data** dialog of the **Query Wizard**. You can define up to three conditions here, and then press **[Next]** to continue.



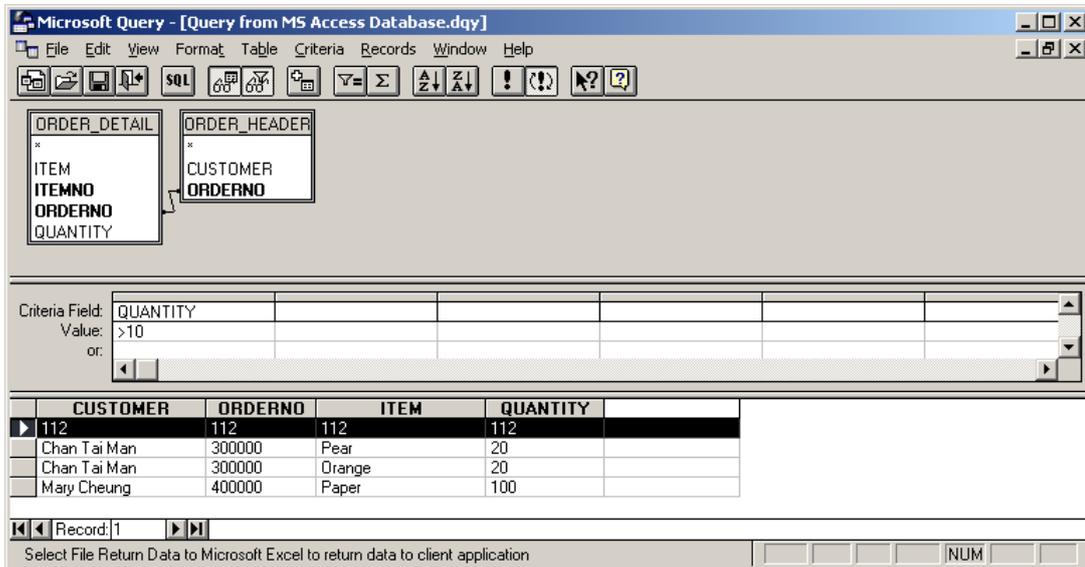
- 6. If you need to sort the data, you can define up to 3 sorting criteria for the data in the **Sort Order** dialog of the **Query Wizard**.



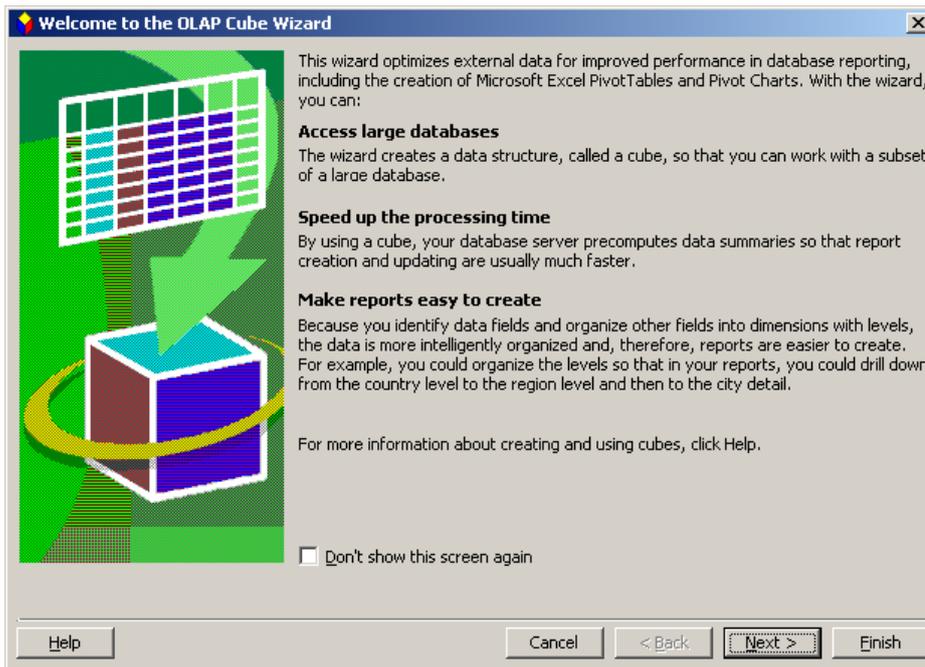
- 7. In the Finish dialog of the **Query Wizard**, you can use the **[Save Query]** to store the query you created. In addition, you have 3 options to precede the import data.



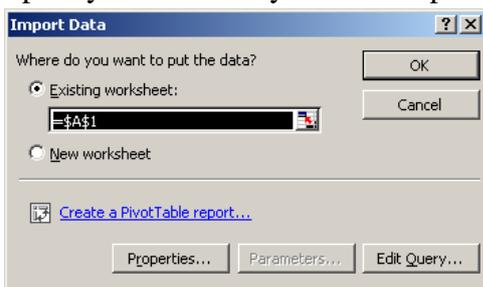
- If you want to import the data to Excel immediately, select **Return Data to Microsoft Excel** and press **[Finish]**.
- If you want to edit the query before import, select **View data or edit query in Microsoft Query** and press **[Finish]**.



- If you want to create a data club for data mining, select **Create an OLAP Cube from this query** and press **[Finish]** to popup the **OLAP Cube Wizard**.



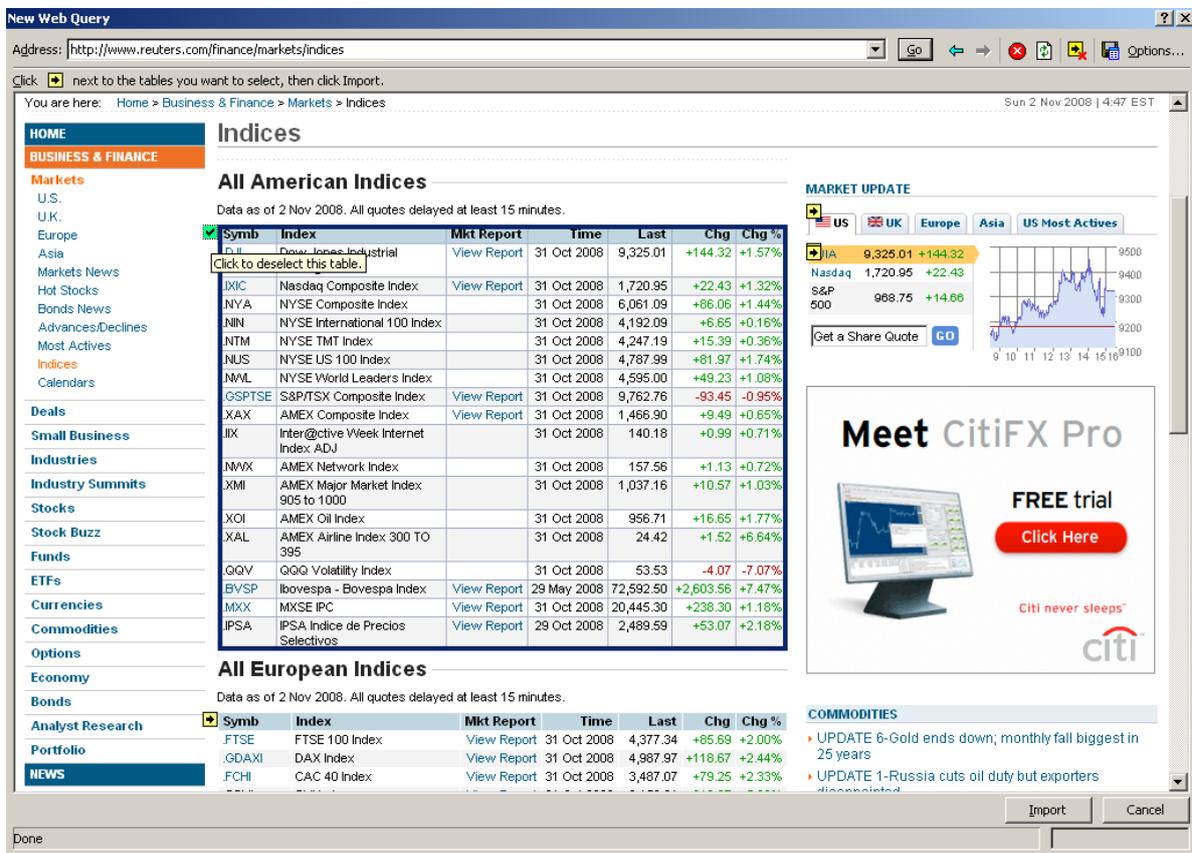
8. Specify the location you want to place the import data and click **[OK]** to finish.



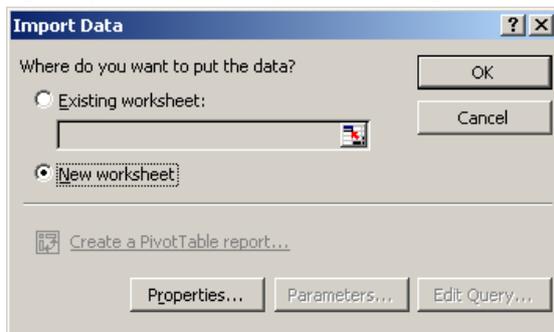
## 2.3 Using a Web Query to Return Internet Data

Web queries can assist you in inserting refreshable data from the Web into your worksheets. Web queries were available in earlier versions of Excel, but have been greatly improved. You can now navigate to any Web page while setting up a Web query, the same way as you would navigate to that page in your browser. Once you are at the page, you can select one or more tables by clicking the icon images automatically added to the page by Excel

1. Click the cell you want to put the data and then select **Data → Import External Data → New Web Query**.
2. Enter the URL in the **Address** box of the **New Web Query** windows and press the **[Go]** button to retrieve the target web page.
3. Select the data area you wish to import from the **New Web Query** windows, and then press **[Import]** button to continue..



4. Specify the location you want to place the import data and click **[OK]** to finish.



## 2.4 Refresh Imported Data

Excel provides many options for refreshing imported data, including refreshing the data whenever you open the workbook and automatically refreshing data at timed intervals. You can continue to work in Excel while data is being refreshed, and you can also check the status of the refresh while it's being refreshed.

If your external data source requires a password to gain access to the data, you can require that the password be entered each time the external data range is refreshed.

When an external data range expands and additional records are returned, Excel can fill formulas in adjacent columns or within the data range so that they remain next to the appropriate data.

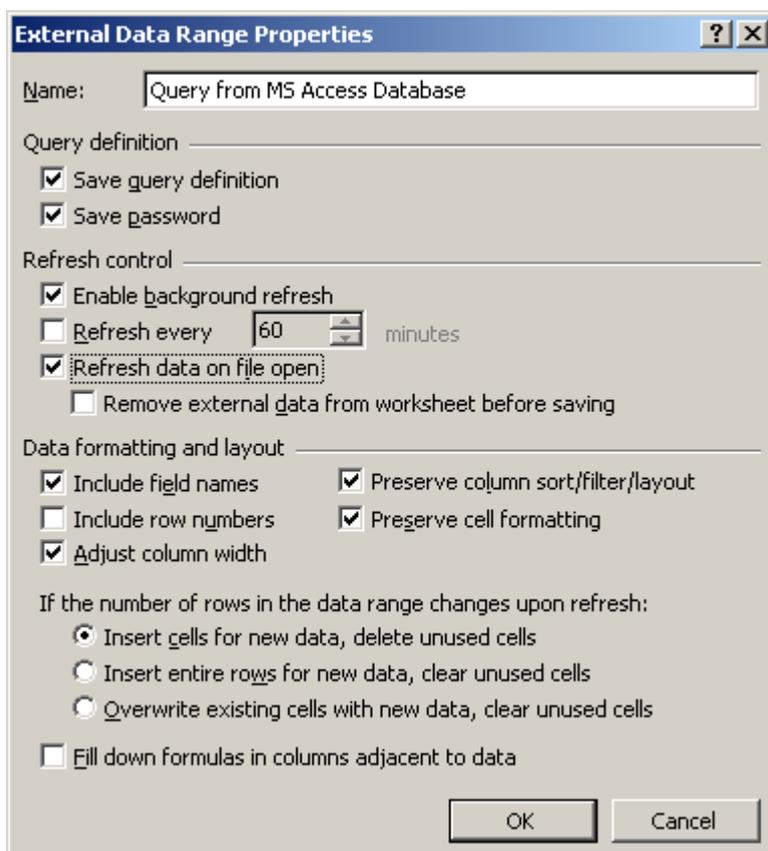
### 2.4.1 Automatically Refresh Data when a Workbook is Opened

You can refresh an external data range automatically when you open the workbook, and optionally save the workbook without saving the external data, so that the workbook file size is reduced.

1. Click a cell in the external data range you want to refresh, and then click **Data Range Properties**  on the **External Data** toolbar.



2. Select the **Refresh data on file open** check box.



3. If you want to save the workbook with the query definition but without the external data, select the **Remove external data from worksheet before saving** check box.

## 2.4.2 Refresh Multiple External Data Ranges

1. If your worksheet contains more than one external data range that you want to refresh, click **Refresh All**  on the **External Data** toolbar.
2. If you have more than one workbook open, you must click **Refresh All** in each workbook to refresh external data.

## 2.4.3 Refresh Data in an Imported Text File

1. Select the worksheet with your imported text file. Select **Data** → **Refresh Data**.
2. In the **Import Text File** dialog box, select your text file, and then click [**Import**].
3. If you want to import only part of the data in a text file as a refreshable data range, you can create a query to retrieve the data..

## 2.4.4 Automatically Refresh Data at Intervals

1. Click a cell in the external data range you want to refresh, and then click **Data Range Properties**  on the **External Data** toolbar.
2. Select the **Refresh every** box, and then enter the number of minutes between refreshes.

## 2.4.5 Require a Password to Refresh an External Data Range

If your data source requires a password to connect to it, you can require that the password is entered before the external data range can be refreshed. This procedure does not apply to data retrieved from a text file (\*.txt) or a Web query (\*.iqy). Please note that the stored passwords are not encrypted. Moreover, Excel prompts for the password only the first time that the external data range is refreshed in each Excel session. The next time you start Excel, you will be prompted for the password again if you open the workbook that contains the query, and then click **Refresh Data**

1. Click a cell in the external data range you want to refresh, and then click **Data Range Properties**  on the **External Data** toolbar.
2. Clear the **Save password** check box under **Query definition** section.

## 2.4.6 Preserve Cell Formatting when you Refresh an External Data Range

1. Click a cell in the external data range you want to refresh, and then click **Data Range Properties**  on the **External Data** toolbar.
2. Do one or both of the following under **Data formatting and layout** section:
  - If you want to preserve cell formatting that you apply, select the **Preserve cell formatting** check box.
  - If you want to preserve column widths that you set, clear the **Adjust column width** check box.
3. Click [**OK**].
4. To refresh the external data range, select **Data** → **Refresh Data**..

## 2.4.7 Run a Query in the Background or While you Wait

Running a query in the background allows you to use Microsoft Excel while the query runs.

1. Click a cell in the external data range you want to refresh, and then click **Data Range Properties**  on the **External Data** toolbar
2. Select the **Enable background refresh** check box to run the query in the background or clear it to run the query while you wait.

### 2.4.7.1 Stop a Query

To stop a query from running when Enable background refresh is turned off, press [ESC] button.

### 2.4.7.2 Stop a Background Refresh

To stop a query that is running in the background, double-click the **Background Refresh**  icon on the status bar to display the **External Data Refresh Status** dialog box, and then click [Stop Refresh].

### 2.4.7.3 Recording a Macro

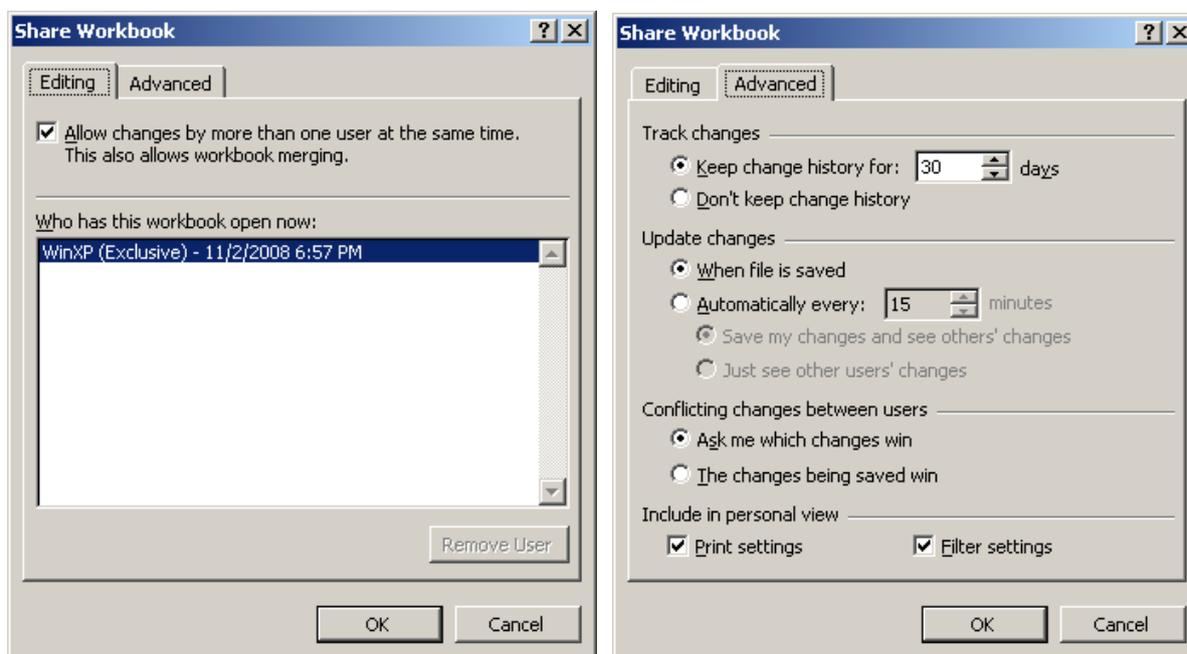
While you are recording a macro that includes a query, Excel will not run the query in the background. To change the recorded macro so that it runs in the background, edit the macro in the Visual Basic Editor and change the refresh method for the **QueryTable** object from "**BackgroundQuery := False**" to "**BackgroundQuery := True**".

## 3. Collaboration and Security

### 3.1 Sharing Workbook

It has always been possible to share Excel files on a network. You just had to make sure that you coordinated your efforts to avoid having more than one person open a file at the same time. Recent versions of Excel, however, allow two or more people to work on the same workbook simultaneously,

1. Select **Tools** → **Share Workbook** to popup the **Share Workbook** dialog.
2. Only one user is allowed to change the workbook by default. You can change some aspects of the default behavior of shared workbooks. Each shared workbook user can set these options individually. Use the first section on the **Advanced** tab to specify the length of time you want to keep track of changes, or whether you want to track them at all.
3. Save your workbook.



### 3.2 Tracking

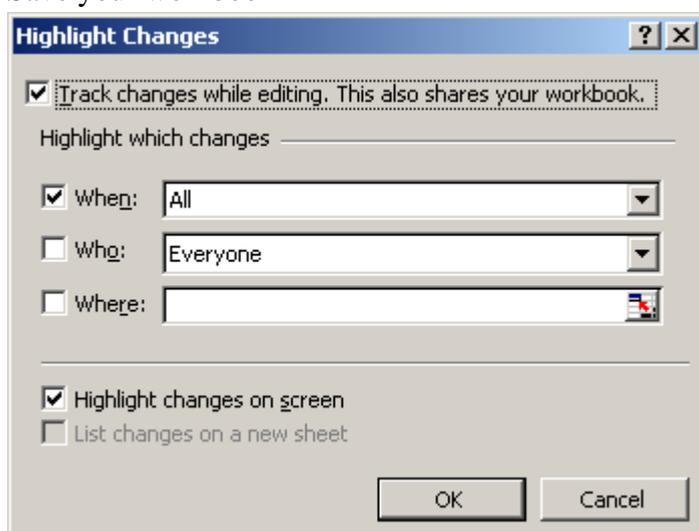
Excel can maintain and display information about how a worksheet was changed. Change tracking logs details about workbook changes each time you save a workbook. You can use this history to understand what changes were made, and to accept or reject revisions.

This capability is particularly useful when several users edit a workbook. It's also useful when you submit a workbook to reviewers for comments, and then want to merge input into one copy, selecting which changes and comments to keep

### 3.2.1 Tracking Changes

Change tracking in Excel is closely linked with shared workbooks.

1. Select **Tools** → **Track Changes, Highlight Changes**.
2. Select **Track Changes While Editing. This also shares your workbook** check box
3. Save your workbook



### 3.2.2 Protecting the Change History

If you want to ensure that every change made during a sharing session is documented, follow these procedures:

1. Choose **Tools** → **Protection** → **Protect and Share Workbook**.
2. When you do, the **Protect Shared Workbook** dialog box appears.
3. If you click **Sharing with track changes** and then click **[OK]**, change tracking for the shared workbook is protected so no one in your workgroup can turn it off directly. However, anyone can turn off the protection. To eliminate this possibility, you can enter a password in the **Protect Shared Workbook** dialog box. But you must do this when the workbook is not in shared mode. Then anyone who tries to turn off protection must enter the identical, case-sensitive password.



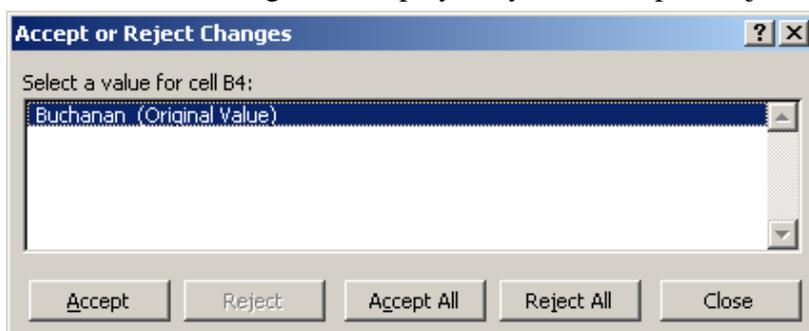
### 3.2.3 Reviewing Changes

You can decide at any time to go through each change that has been made to the shared workbook, provided the Track Changes While Editing check box was selected in the Highlight Changes dialog box when the worksheet was first saved for sharing.

1. Select the cell you wish to review, and then choose **Tools** → **Track Changes** → **Accept Or Reject Changes**
2. Select the changes you wish to review in the **Select Changes to Accept or Reject dialog**.



3. The list for the change will display for you to accept or reject.



### 3.3 Protecting Files

Excel features related to hiding data or locking data with passwords are not intended to secure or protect confidential information in Excel. These features are merely meant to obscure data or formulas that might confuse some users or to prevent others from viewing or making changes to that data. Excel does not encrypt data that is hidden or locked in a workbook. To help prevent modification of confidential data and to help protect it from being viewed, you may want to limit access to any workbook files that contain such information by storing them in locations that are available only to authorized users.

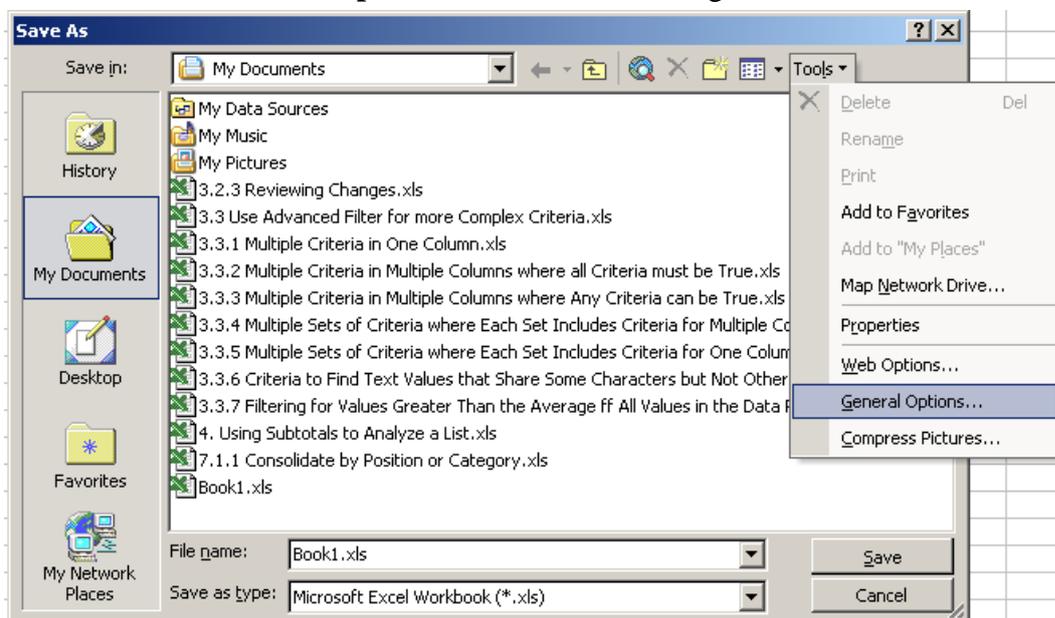
Excel provides several layers of security and protection to control who can access and change your data:

- For optimal security, you should protect your entire workbook file with a password, allowing only authorized users to view or modify your data.
- For additional protection of specific data, you can protect certain worksheet or workbook elements, with or without a password. Use element protection to help prevent anyone from accidentally or deliberately changing, moving, or deleting important data.

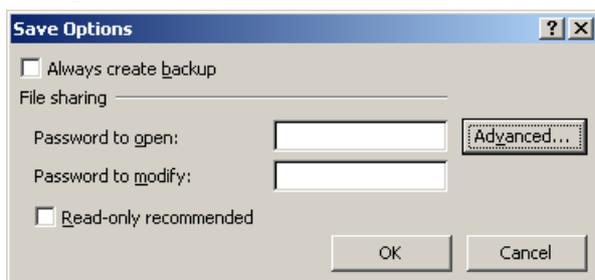
### 3.3.1 Securing a Workbook file

Password security at the workbook file level uses advanced encryption to help protect your workbook from unauthorized access.

1. Select **File** → **Save As**.
2. Select **Tools** → **General Options** in the **Save As** dialog.



3. You can specify two separate passwords that users must type to and then save the workbook.
  - **Open and view the file** – This password is encrypted to help protect your data from unauthorized access.
  - **Modify the file** – This password is not encrypted and is only meant to give specific users permission to edit workbook data and save changes to the file.



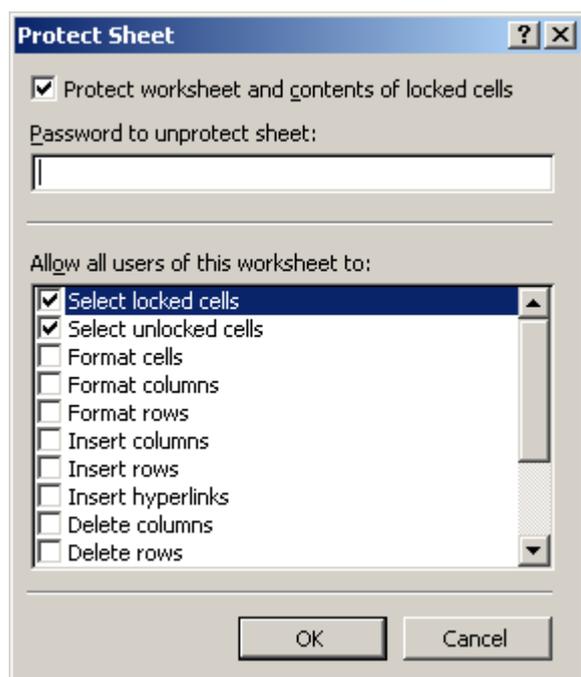
### 3.3.2 Protecting Specific Worksheet or Workbook Elements

When you share an Excel file so that others can collaborate on the data, you can prevent any user from making changes to specific worksheet or workbook elements by protecting (or locking down) certain parts of the file. You can also specify a password to allow individual users to modify specific elements.

When you protect a worksheet, all cells on the worksheet are locked by default, and users cannot make any changes to a locked cell. For example, they cannot insert, modify, delete, or format data in a locked cell. You can, however, specify which elements users will be allowed to change when you protect the worksheet.

To protect a worksheet, select **Tools** → **Protection** → **Protect Sheet**. You can control access to

individual worksheet or chart sheet elements by selecting or clearing the following check boxes



### 3.3.3 Permission to Access Specific Areas of a Protected Worksheet

Before you protect a worksheet, you can unlock the ranges that you want users to be able to change or enter data in. You can:

- Unlock cells for all users on the Protection tab of the Format Cells dialog box by choosing **Format → Cells**.
- Unlock cells for specific users in the Allow Users to Edit Ranges dialog box by choosing **Tools → Protection → Allow Users to Edit Ranges**.

Users whom you specify in the Permissions for range dialog box (Permissions button) can automatically edit the range without entering the password. All other users are prompted for the password when they want to edit the range.



## 4. Macro and VBA

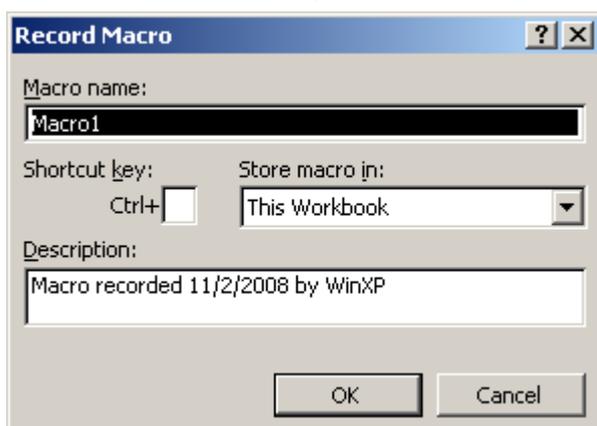
If you perform a task repeatedly in Microsoft Excel, you can automate the task with a macro. A macro is a series of commands and functions that are stored in a Microsoft Visual Basic module and can be run whenever you need to perform the task. For example, if you often enter long text strings in cells, you can create a macro to format those cells so that the text wraps.

### 4.1 Record a Macro

#### 4.1.1 Record a Macro using Macro Recorder

When you record a macro, Excel stores information about each step you take as you perform a series of commands. You then run the macro to repeat, or "play back," the commands. If you make a mistake when you record the macro, corrections you make are also recorded. Visual Basic stores each macro in a new module attached to a workbook.

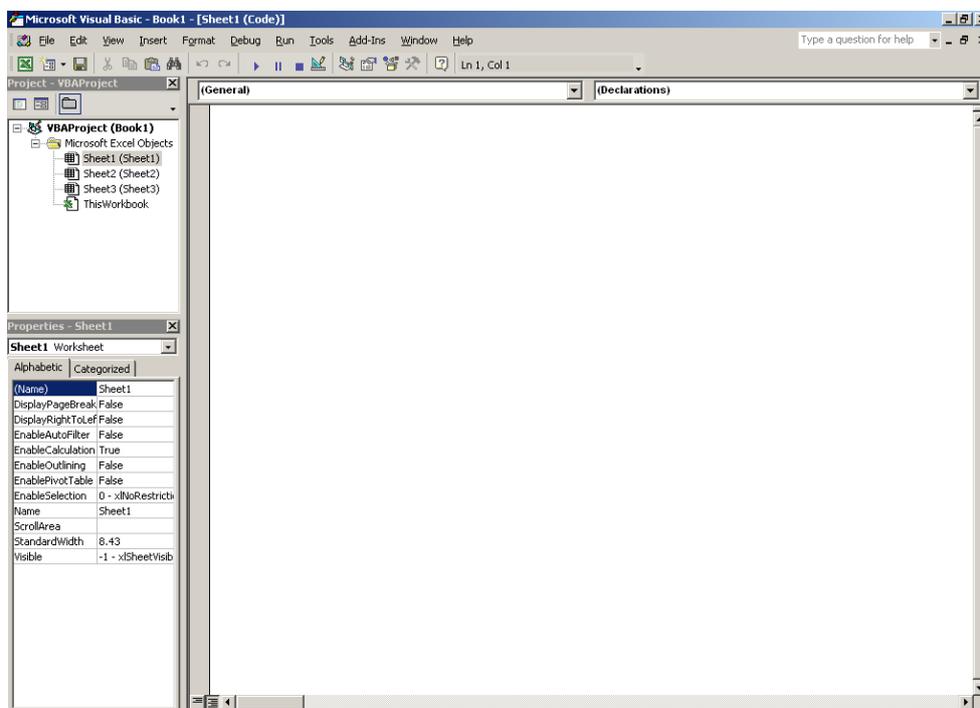
1. Select **Tools** → **Macro** → **Record New Macro**.
2. In the **Macro** name box, enter a name for the macro.



3. If you want to run the macro by pressing a keyboard shortcut key, enter a letter in the **Shortcut** key box. The shortcut key will override any equivalent default Excel shortcut keys while the workbook that contains the macro is open
4. In the **Store** macro in box, click the location where you want to store the macro. If you want a macro to be available whenever you use Excel, select **Personal Macro Workbook**.
5. If you want to include a description of the macro, type it in the **Description** box.
6. Click **[OK]**. If you want the macro to run relative to the position of the active cell, record it using relative cell references. On the **Stop Recording** toolbar, click **Relative Reference** so that it is selected. Excel will continue to record macros with relative references until you quit Excel or until you click **Relative Reference** again, so that it is not selected.
7. Carry out the actions you want to record.
8. Select **Tools** → **Macro** → **Stop Recording** when finish the record.

## 4.1.2 Create a macro using Visual Basic Editor

1. Select **Tools** → **Macro** → **Visual Basic Editor**.
2. On the **Insert** → **Module** in the **Microsoft Visual Basic Editor**.
3. Type or copy your code into the code window of the module.
4. If you want to run the macro from the module window, press **[F5]**.
5. When you're finished writing your macro, click **File** → **Close and Return to Microsoft Excel**.

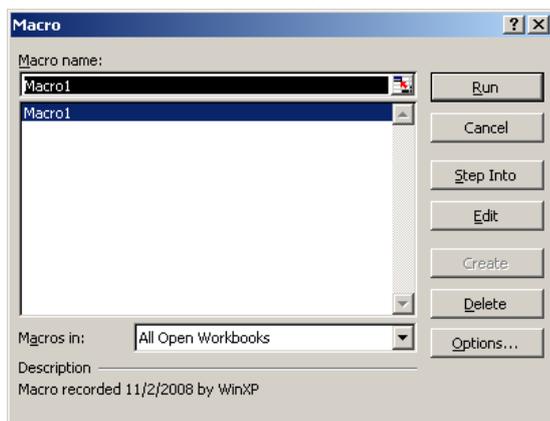


## 4.2 Execute the Macro

The next time you need to flag a cell, you can run the macro. If you're going to use the macro frequently, you can create a toolbar button for it, or assign a keystroke for it, or both.

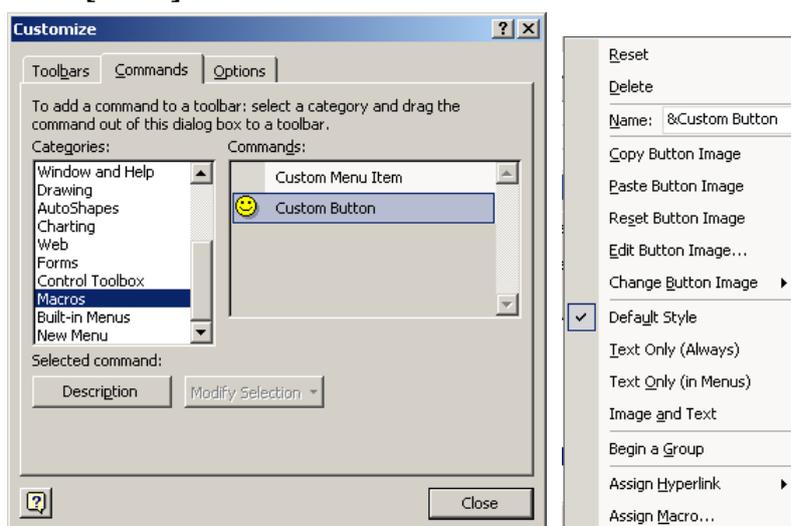
### 4.2.1 Run your Macro Using the Tools Menu

1. Click the worksheet, and then select **Tools** → **Macro** → **Macros**.
2. Click the name of your macro, and then click **Run**.



## 4.2.2 Create a Toolbar Button to Run your Macro

1. Click the worksheet, and then select **Tools** → **Customize**.
2. In the **Customize** dialog, click the **Commands** tab and select **Macros** under **Categories** session. Then drag the custom button to the toolbar where you want it.
3. On the Customize dialog box, click [**Modify Selection**], and then click **Assign Macro**.
4. In the **Assign Macro** dialog box, click the name of your macro, and then click OK.
5. To change the appearance of the button, click [**Modify Selection**] again, point to **Change Button Image**, and click one of the available images; or click [**Edit Button Image**] and use the Button Editor to create your own image.
6. Click [**Close**].



## 4.2.3 Assign a Keystroke to Run your Macro

1. Click the worksheet, and then select **Tools** → **Macro** → **Macros**.
2. Select the name of your macro, and then click [**Options**].
3. In the Shortcut key box, type the key to use along with [**Ctrl**] button to run your macro.

## 4.3 Managing your Macros

After you record a macro, you can view the macro code with the Visual Basic Editor to correct errors or change what the macro does. For example, if you wanted the text-wrapping macro to also make the text bold, you could record another macro to make a cell bold and then copy the instructions from that macro to the text-wrapping macro.

The Visual Basic Editor is a program designed to make writing and editing macro code easy for beginners, and provides plenty of online Help. You don't have to learn how to program or use the Visual Basic language to make simple changes to your macros. With the Visual Basic Editor, you can edit macros, copy macros from one module to another, copy macros between different workbooks, rename the modules that store the macros, or rename the macros.

## 4.4 Introduction to Visual Basic Editor

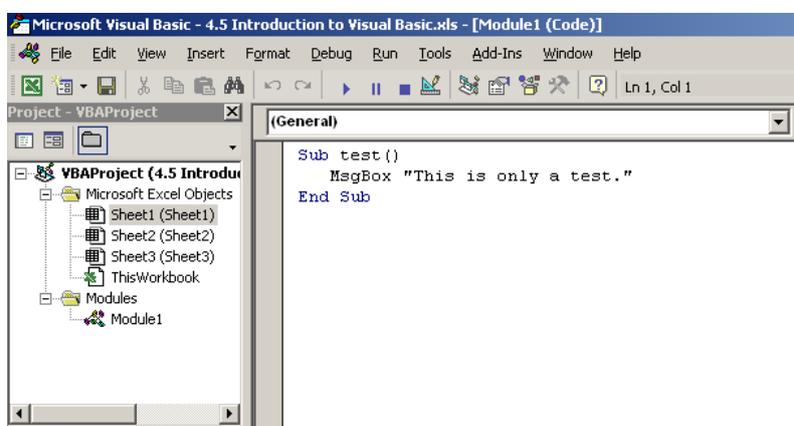
### 4.4.1 Using the Visual Basic Editor

The Visual Basic Editor is a powerful tool that lets you extend the power and versatility of macros beyond anything that can be done through recording alone. To open it, select **Tools** → **Macro** → **Visual Basic Editor**.

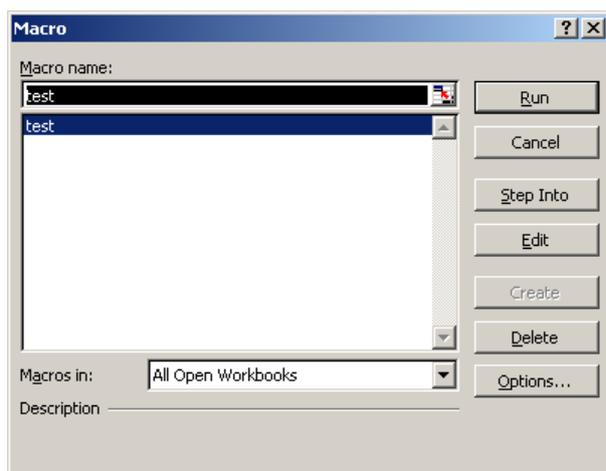
1. Start Excel and open a new, blank workbook.
2. Select **Tools** → **Macro** → **Visual Basic Editor**.
3. In the **Project** window, double-click **ThisWorkbook**.
4. Enter this code into the code window:

```
sub test()
    MsgBox "This is only a test."
end sub
```

5. Save the file and then select **File** → **Visual Basic Editor** and close the workbook.



6. Select **Tool** → **Macro** → **Macros** in the Excel worksheet. Then select the previous macro “text” we created in the Visual Basic Editor.

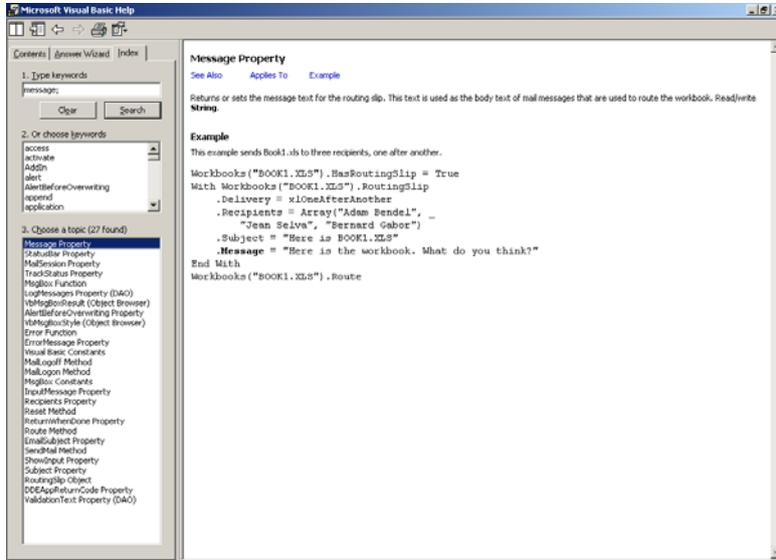


7. Press **[Run]** to execute the macro and a message box will be displayed..



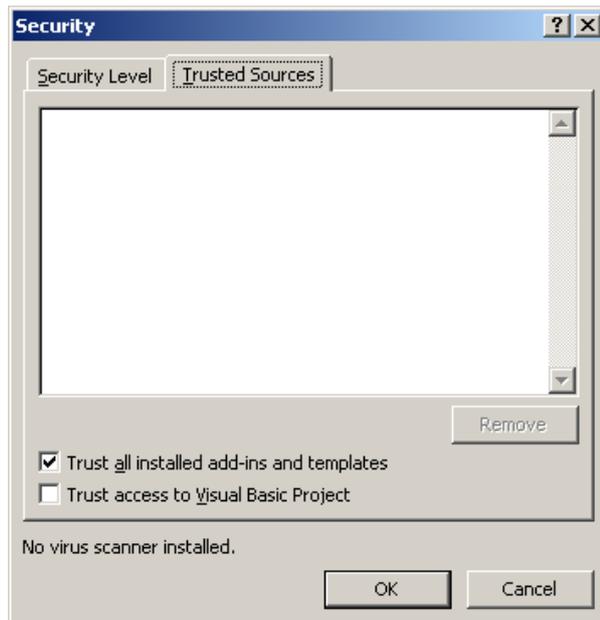
### 4.4.2 Use Visual Basic for Applications Help

1. In Excel workbook, select **Tools** → **Macro** → **Visual Basic Editor**.
2. Select **Help** → **Microsoft Visual Basic Help** in the **Visual Basic Editor**.
3. In the **Search** box, type the method, property, function, statement, or object you want Help on, or type a query.



### 4.5 Macro Security

In Excel, you can set a macro security level to control what happens when you open a workbook that contains a macro. For example, you can choose to run macros based on whether they are digitally signed by a developer on your list of trusted sources. There are several security levels in Excel: High, Medium, and Low.



## 4.5.1 Macro Security Level – High

### 4.5.1.1 Unsigned Macros

Macros are automatically disabled, and the file is opened.

### 4.5.1.2 Signed Macros

The source of the macro and the status of the signature determine how signed macros are handled.

- **A trusted source. Signature is valid** – Macros are automatically enabled, and the file is opened.
- **An unknown author. Signature is valid** – A dialog box is displayed with information about the certificate. Macros can be enabled only if the user chooses to trust the author and certification authority. A network administrator can lock the list of trusted sources and prevent the user from adding the developer to the list and enabling the macros.
- **Any author. Signature is invalid, possibly because of a virus** – User is warned of a possible virus. Macros are automatically disabled.
- **Any author. Signature validation is not possible because public key is missing or incompatible encryption methods were used** – User is warned that signature validation is not possible. Macros are automatically disabled.
- **Any author. The signature was made after the certificate had expired or had been revoked** – User is warned that the signature has expired or been revoked. Macros are automatically disabled.

## 4.5.2 Macro Security Level – Medium

### 4.5.2.1 Unsigned Macros

User is prompted to enable or disable macros.

### 4.5.2.2 Signed Macros

The source of the macro and the status of the signature determine how signed macros are handled.

- **A trusted source. Signature is valid** – Macros are automatically enabled, and the file is opened.
- **An unknown author. Signature is valid** – A dialog box is displayed with information about the certificate. The user is prompted to enable or disable macros. The user can choose to trust the developer and certification authority.
- **Any author. Signature is invalid, possibly because of a virus** – User is warned of a possible virus. Macros are automatically disabled.
- **Any author. Signature validation is not possible because public key is missing or incompatible encryption methods were used** – User is warned that signature validation is not possible. User is prompted to enable or disable macros.
- **Any author. The signature was made after the certificate had expired or had been revoked** – User is warned that the signature has expired or been revoked. User is prompted to enable or disable macros.

### 4.5.3 Macro Security Level – Low

When security is set to low, all macros are treated equally regardless of origin or certificate status. With low security, you receive no prompt or signature validation and macros are automatically enabled. Use this setting only if you are certain that all macros in your files are from trusted sources.

## 4.6 Deploy your Excel Macros from a Central File

By default, when you create a macro for a Microsoft Excel worksheet, that macro works only on that worksheet. This behavior is okay as long as the macro automates tasks that are unique to that worksheet. But have you ever found yourself recreating the same macro on more than one worksheet? If you have (and even if you haven't), you can copy your macros to a worksheet called `personal.xls`. From there, your macros become available to you whenever you start Excel, and you can run them against the data on any open worksheet.

Of course, you need to be careful here. You don't want to run macros at random, because doing so might corrupt your data and cost you time and money. But if you have macros that you can use in a variety of situations, using a `personal.xls` file can also save you time and money.

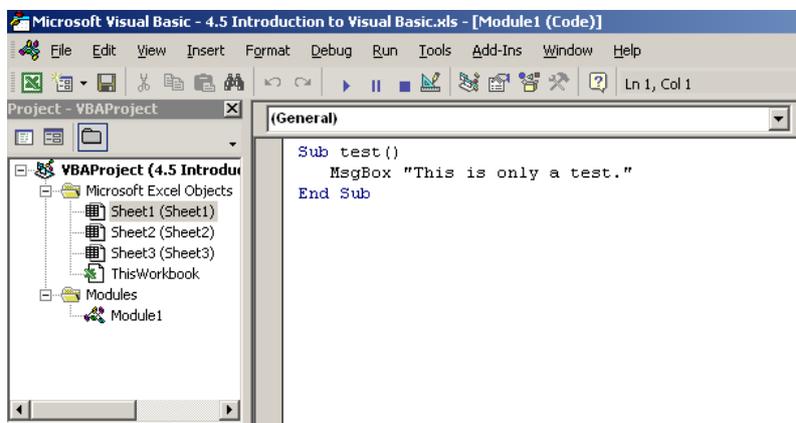
### 4.6.1 Getting started

To follow the steps in this column, you need an Excel worksheet that contains at least one macro. If you need a sample macro, follow these steps:

4. Start Excel and open a new, blank workbook.
5. Select **Tools** → **Macro** → **Visual Basic Editor**.
6. In the **Project** window, double-click **ThisWorkbook**.
7. Enter this code into the code window:

```
sub test( )  
    MsgBox "This is only a test."  
end sub
```

8. Save the file and then select **File** → **Visual Basic Editor** and close the workbook.



## 4.6.2 Create a personal.xls file

To create and use a personal.xls file, you follow this basic process:

1. Create a blank personal.xls file in the folder called **XLStart**. If you installed Microsoft Office in the default location, you'll find **XLStart** under C:\Program Files\Microsoft Office\Office 10. If you installed Office in a different location, find **XLStart** there.
2. Hide the file in Excel by choosing **Window → Hide**.
3. Unhide the file by choosing **Windows → Unhide**, copy your macros from their parent worksheets to the file, and then hide personal.xls again.
4. You keep personal.xls hidden unless you want to add macros to it. Whenever you start Excel, personal.xls opens automatically and runs in the background. If you don't hide the file, Excel opens it and allows you to add data to it.

## 4.6.3 Add Macros to Personal.xls

When you create a new macro, you can choose to save it to your personal.xls file. In addition, you can add your existing macros to personal.xls. The following steps explain how to perform both actions.

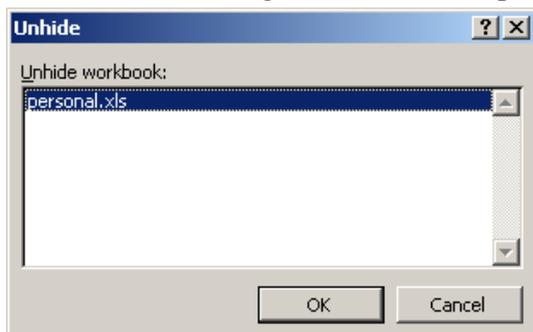
### 4.6.3.1 To Add New Macros

1. Select **Tools → Macro → Record new macro**.
2. In the **Store macro in** list, click **Personal Macro Workbook**.
3. Click **[OK]** and record your macro normally.



### 4.6.3.2 To Add Existing Macros

1. Open personal.xls, and select **Window → Unhide**.
2. In the **Unhide** dialog box, ensure that personal.xls is selected, and then click **[OK]**.



3. Open the worksheet that contains the macro you want to copy, select **Tools → Macro →**

**Macros.**

4. Select the macro you want to add to personal.xls, and then click **[Edit]**. This step starts the Visual Basic Editor and opens the code module for the macro.
5. In the Visual Basic Editor, select **File → Export File**.
6. Locate the **\XLStart** folder and click Save to save the code module in that folder.
7. Go back to personal.xls, press **[Alt] + [F11]** to start the Visual Basic Editor there, and select **File → Import File**.
8. Navigate to the **\XLStart** folder, click the module that you just exported, and then click **[Open]**.
9. Save the module, quit the Visual Basic Editor, and then hide personal.xls by choosing **Window → Hide**.
10. Quit Excel, clicking **[Yes]** to save your changes.
11. Go back to the **\XLStart** folder and delete any code modules. If you leave the module files in the folder, they appear automatically the next time you start Excel.

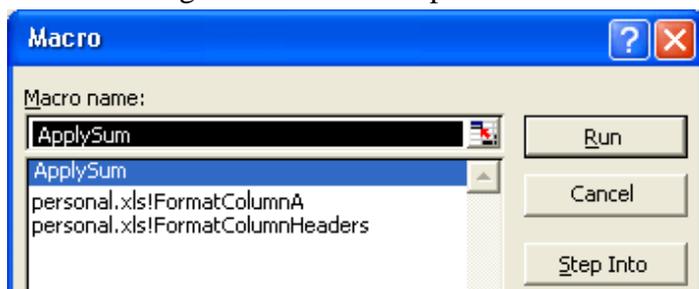
#### 4.6.4 Use personal.xls

You use personal.xls by starting Excel normally and running macros as needed.

1. Open a new, blank worksheet, and use your favorite method to enter some data.
2. Select **Tools → Macro → Macros**.
3. Select the macro you want to run, and then click **[Run]**.

Okay, that's pretty basic, but keep these facts in mind as you go:

- Excel distinguishes between macros that reside in personal.xls and those that reside on the open worksheet. In the following figure, the Apply Sum macro resides on the local worksheet. The remaining macros reside on personal.xls:



- You can't edit macros that reside in personal.xls unless you open that file. For example, repeat steps 1 and 2 at the beginning of this section, select a macro that resides in personal.xls, and then click Edit. Excel displays the following message: Cannot edit a macro on a hidden workbook. Unhide the workbook using the Unhide command.
- In addition to containing macros, personal.xls can store information that you use frequently. For example, you can store an address or some boilerplate text in a cell and use a macro to copy that data and paste it onto another worksheet. Go ahead and use your creativity!