Take an animated tour

Microsoft Office InfoPath 2003 is a new member of the Microsoft Office System. To help you get started, you can take an animated tour of the program. The tour provides an overview of InfoPath, including its key concepts and main product features. You can choose to view the entire tour or learn only about the tasks that interest you.

1. On the InfoPath **Help** menu, click **Microsoft Office InfoPath Tour**. The tour start page will be displayed in your Web browser.
2. On the tour start page, click **Start Tour**. The tour will be displayed in a new window.

Notes

For best results, viewing the tour on a computer with an 800 x 600 or higher screen resolution is recommended.

Macromedia Flash Player is required to view the tour. This software is included with all supported versions of Microsoft Windows.

About the InfoPath workspace

Regardless of whether you are filling out or designing a form, the Microsoft Office InfoPath 2003 workspace is divided into two main areas; the form area and the task pane area. The form area appears on the left side of the screen and the task pane area appears on the right.

▸ The form area

▸ The task pane area

Switch between modes

Microsoft Office InfoPath 2003 has two separate modes that are designed for two distinct tasks— filling out forms and designing forms.

To switch between modes, do one of the following:

To fill out a form, click **Fill Out a Form** on the **File** menu.

To design a form, click **Design a Form** on the **File** menu.

Note InfoPath doesn't actually switch modes until you use the commands on the corresponding task panes to create or open a form to work with. A (Design) prefix will appear in the application title bar when you have successfully switched to design mode.

About forms and form templates

In Microsoft Office InfoPath 2003, you can fill out a form or create your own form in design mode. When you create a form in design mode, you are actually creating a form template. Every InfoPath form is based on a form template.

▸ Forms

▸ Form templates

About form development

Microsoft Office InfoPath 2003 provides many features that allow developers to extend the functionality of their forms. Using these features, developers can write scripting code to perform a variety of tasks, including custom data validation, error handling, form submission, form merging, accessing external data sources, and responding to form and data validation events.

InfoPath uses the Microsoft Script Editor (MSE) integrated development environment for creating and debugging scripting code. When used with the InfoPath object model and form definition (.xsf) file, MSE provides developers with the tools they need to customize forms.

The following are some of the development features that are implemented in InfoPath:

Event handler creation in design mode

MSE

Support for the Microsoft JScript and Microsoft VBScript programming languages

Script debugging

InfoPath object model

InfoPath form definition file

Object model and form security measures

Roundtripping of custom view changes

Custom task panes

Custom toolbars

Custom dialog boxes

Custom menus

Developer Help in the InfoPath Developer's Reference, including conceptual topics, an InfoPath object model reference, an InfoPath XSF reference, documentation of the developer sample forms, and documentation of common programming tasks

Documentation for MSE, JScript, VBScript, and Microsoft XML Core Services (MSXML)

InfoPath Developer Center on the Microsoft Developer Network (MSDN)

About sample forms

Microsoft Office InfoPath 2003 provides many sample forms that are designed to meet various business needs. They can be filled out as is or easily customized to meet your specific needs. For example, you can add or remove controls from a sample form, change the data validation rules, or add a company logo.

Sample forms can also be used as examples for laying out or adding functionality to your forms. For example, you could open the Expense Report sample form in design mode to better understand how a repeating table works. You can view the table's properties, see what types of controls are used in it, as well as see which fields or groups the controls are bound to. This can help you think of ideas for creating and adding functionality in your own forms.

Navigate the InfoPath Tour by using the keyboard

The Microsoft Office InfoPath Tour can be navigated by using the mouse or the keyboard. To navigate the tour by using the keyboard, follow these steps.

1. In InfoPath, press ALT+H to open the **Help** menu, and then press T to start the InfoPath Tour.
2. On the tour start page, press TAB until the **Start Tour** button is outlined, and then press ENTER.
3. When the tour window opens and the navigation buttons appear, do one of the following:

- To move the yellow navigation box forward to the option you want to select, press the TAB key as needed.
- To move the yellow navigation box backward to the option you want to select, press SHIFT+TAB as needed.

4. When the button or option you want is selected with the yellow navigation box, press the SPACEBAR to carry out the selected command.

Troubleshoot viewing the InfoPath Tour

▸ How can I take the InfoPath Tour?

▸ I want to view the entire tour; what is the intended order of the tour contents?

▸ When I click Start Tour on the start page, nothing happens.

▸ When I start the InfoPath Tour, I get an error message that says my computer does not meet the system requirements.

Getting Help on form development

A variety of resources are available for getting Help on Microsoft Office InfoPath 2003 form development. Help is available both in the InfoPath Help system and online at various Microsoft Web sites.

The InfoPath Developer's Reference

The following resources are available from within the InfoPath Developer's Reference, which is available in the InfoPath Help system and in the Microsoft Script Editor (MSE) Help system:

Programming Concepts This set of topics includes conceptual information about programming InfoPath, including the InfoPath object model and form definition (.xsf) file.

Developer Sample Forms This set of topics includes documentation for the five developer sample forms installed with InfoPath that demonstrate some of the key features of InfoPath form development.

InfoPath Object Model Reference This set of topics provides the complete set of documentation for the InfoPath object model, including all of its collections, objects, properties, methods, events, and enumerations.

InfoPath XSF Reference This set of topics provides the complete set of documentation for the InfoPath .xsf file, including all of it namespaces, groups, types, elements, and attributes.

The Microsoft Script Editor Help system

The following resources are available in the MSE Help system, which is available only when working in MSE:

Microsoft Script Editor Help This set of topics provides information about using MSE, from customizing the script editor to debugging techniques.

Microsoft XML Core Services (MSXML) 5.0 SDK This set of topics provides information about using XML, the XML Document Object Model (DOM), XML Schemas, XSL Transformation (XSLT), and XPath. It includes various developer guides and language references that discuss each of these XML-based technologies.

JScript Reference This set of topics provides information about using the Microsoft JScript programming language, including a language reference and tutorial.

VBScript Reference This set of topics provides information about using the Microsoft VBScript programming language, including a language reference and tutorial.

Online resources

InfoPath Developer Center This Web site contains InfoPath form development information, including white papers, sample forms, tools, articles, and general information about developing InfoPath forms. The InfoPath Developer Center is accessed from InfoPath by clicking **Microsoft InfoPath Developer Resources** on the **Help** menu. The site is frequently updated with InfoPath form development information. Visit the InfoPath Developer Center.

Office Developer Center This Web site contains information about using all of the Microsoft Office applications as a development platform. The Office Developer Center is continually updated with information. Visit the Office Developer Center.

Microsoft Developer Network (MSDN) This Web site contains information about a variety of Microsoft development platforms and tools. The Microsoft Developer Network is continually updated with information. Visit the Microsoft Developer Network.

Microsoft XML Developer Center This Web site contains information about using and developing applications with Extensible Markup Language (XML). The Microsoft XML Developer Center is continually updated with information. Visit the Microsoft XML Developer Center.

Microsoft Scripting Developer Center This Web site contains information about using various Microsoft scripting languages, including JScript and VBScript. The Microsoft Scripting Developer Center is continually updated with information. Visit the Microsoft Scripting Developer Center.

Keyboard shortcuts

This Help topic describes keyboard shortcuts for common tasks in Microsoft Office InfoPath 2003, such as filling out or designing forms. The keyboard shortcuts refer to the U.S. keyboard layout. Keys on other keyboard layouts may not correspond exactly to the keys on a U.S. keyboard.

For keyboard shortcuts in which you press two or more keys simultaneously, the keys are separated by a plus sign (+). For keyboard shortcuts in which you press one key immediately followed by another key, the keys are separated by a comma (,).

To print this topic, press TAB to select **Show All**, press ENTER, and then press CTRL+P.

Filling out forms

Designing forms

Getting Help

- Using the **InfoPath Help** task pane
- Using the Help window
- Typing a search question

Using menus, toolbars, and task panes

- Accessing and using menu bars and toolbars
- Accessing and using task panes

Using dialog boxes

- Accessing and selecting options in dialog boxes
- Using edit boxes within dialog boxes

Add a button, menu, or command

You can rearrange and modify toolbar buttons, menus, and menu commands by using the mouse or you can use the **Rearrange Commands** dialog box which is accessible using a keyboard menu. These tasks are identified as (keyboard accessible).

Do one or more of the following:

- ▸ Add a button, menu, or command to a toolbar
- ▸ Add a button, menu, or command to a toolbar (keyboard accessible)
- ▸ Add a command to a menu
- ▸ Add a command to a menu (keyboard accessible)

Find and open a form

When filling out a form, you can open new, blank forms, as well as forms that you have saved. When designing a form, you can open form templates that are installed on your computer or an internet, as well as form templates associated with form libraries.

Open a form to fill it out

- Open a new form
- Open a saved form

Open a form in design mode

- Open a blank form
- Open a sample form
- Open a saved form template
- Open a form library's form template
- Tip

Select page orientation

1. On the **File** menu, click **Print**.
2. In the **Print** dialog box, click **Properties**.
3. Under **Orientation**, click **Portrait** or **Landscape**.

Print a form

To print the current form, click **Print** 🖨 on the **Standard** toolbar.

Preview a form before printing

1. On the **File** menu, click **Print**.
2. In the **Print** dialog box, click **Preview**.
3. Use the buttons on the toolbar to review the form or make adjustments before printing.

Check the progress of your print job

Double-click the **Printer** icon in the status area of the Microsoft Windows taskbar.

When a file is printed, the icon disappears.

Note If you're printing a simple form, the printer icon may not appear in the status area long enough for you to double-click it.

Cancel printing

1. Double-click the **Printer** icon in the status area of the Microsoft Windows taskbar.
2. Right-click the print job you want to cancel, and then click **Cancel** on the shortcut menu.

Note If you're printing a simple form, the printer icon may not appear in the status area long enough for you to double-click it.

Apply bold formatting

1. Select the text you want to change.

   Note When designing a form, to set the format of text users type into controls, select the control you want to format.

2. On the **Formatting** toolbar, click **Bold** B.

Note When filling out a form, you can only make text bold if it is typed into a field that allows text formatting. The **Bold** option is unavailable for fields that don't allow text formatting.

Apply italic formatting

1. Select the text you want to change.

   Note When designing a form, to set the format of text users type into controls, select the control you want to format.

2. On the **Formatting** toolbar, click **Italic** .

Note When filling out a form, you can only italicize text typed into fields that allow text formatting. The **Italic** option is unavailable for fields that don't allow text formatting.

Apply strikethrough formatting

1. Select the text that you want to change.

   Note When designing a form, to set the format of text users type into controls, select the control you want to format.

2. On the **Formatting** toolbar, click **Font**.

3. In the **Font** task pane, select the **Strikethrough** check box.

Note When filling out a form, you can only apply the strikethrough format to text typed into fields that allow text formatting. The **Strikethrough** option is unavailable for fields that don't allow text formatting.

Change the color of text

1. Select the text you want to change.

   Note When designing a form, to set the color of text users type into controls, select the control you want to format.

2. Do one of the following:

- To apply the color most recently used for text, click **Font Color** A on the **Formatting** toolbar.
- To apply a different color, click the arrow next to **Font Color** A, and then select the color you want.

Note When filling out a form, you can only change the color of text typed into fields that allow text formatting. The **Font Color** option is unavailable for fields that don't allow text formatting.

Change the font

1. Select the text you want to change.

   Note When designing a form, to set the font for text users type into controls, select the control you want to format.

2. On the **Formatting** toolbar, click a font name in the **Font** box .

Note When filling out a form, you can only change the font of text typed into fields that allow text formatting. The **Font** option is unavailable for fields that don't allow text formatting.

Apply or remove highlighting

Use the Highlight tool to mark and find important text or pictures in your form.

Do one of the following:

▸ Apply highlighting

▸ Remove highlighting

Note When filling out a form, you can only highlight text typed into fields that allow text formatting. The **Highlight** option is unavailable for fields that don't allow text formatting.

Change the size of text

1. Select the text you want to change.

   Note When designing a form, to set the size of text users type into controls, select the control you want to format.

2. On the **Formatting** toolbar, click a point size in the **Font Size** box .

Note When filling out a form, you can only change the size of text typed into fields that allow text formatting. The font size option is unavailable for fields that don't allow text formatting.

Remove formatting from text

1. Select the text whose formatting you want to remove.

   Note When designing a form, to remove formatting from text users type into controls, select the control you want to remove formatting from.

2. On the **Formatting** toolbar, click **Font** .
3. In the **Font** task pane, select **Clear Formatting** in the **Pick formatting to apply** box. Microsoft Office InfoPath 2003 restores the text to the default formatting for your form.

Note When filling out a form, you can only apply or remove formatting from text typed into fields that allow text formatting. The **Clear Formatting** option is unavailable for fields that don't allow text formatting.

Make text superscript or subscript

1. Select the text you want to format as superscript or subscript.

   Note When designing a form, to set the format of text users type into controls, select the control you want to format.

2. On the **Formatting** toolbar, click **Font**.

3. In the **Font** task pane, select either the **Superscript** or **Subscript** check box.

Note When filling out a form, you can only format text typed into fields that allow text formatting. Text formatting options are unavailable for fields that don't allow text formatting.

Underline text

1. Select the text you want to change.

   Note When designing a form, to set the format of text users type into controls, select the control you want to format.

2. On the **Formatting** toolbar, click **Underline** U.

Note When filling out a form, you can only underline text typed into fields that allow text formatting. The **Underline** option is unavailable for fields that don't allow text formatting.

Apply a heading format to text

▸ Tip

1. Select the text whose formatting you want to change.
2. On the **Formatting** toolbar, click **Font** .
3. In the **Pick formatting to apply** box in the **Font** task pane, select the heading style you want to apply. Microsoft Office InfoPath 2003 changes the text to the formatting specified for the style you applied.

Note When filling out a form, you can only apply a heading format to text typed into fields that allow text formatting. The heading options are unavailable for fields that don't allow text formatting.

Align text left or right

When filling out a form, you can align text typed into rich text fields. When designing a form, you can align text that appears on your form. You can also specify the text alignment for your users, when they type text into a date picker, expression box, text box, or rich text control.

▸ Align text

▸ Specify the text alignment for a control

Center text

When filling out a form, you can center text typed into rich text fields. When designing a form, you can center text that appears on your form. You can also specify that text should be centered when your users type text into a date picker, expression box, text box, or a rich text control.

- Center text
- Center text typed into controls

Justify text

When filling out a form, you can justify text typed into rich text fields. When designing a form, you can justify text that appears on your form. You can also specify that text should be justified when your users enter text into a date picker, expression box, text box, and a rich text control.

- Justify text
- Justify text typed into controls

Indent text

You can increase or decrease the left indent of text.

1. Select the text you want to indent.
2. On the **Formatting** toolbar, click **Increase Indent** or **Decrease Indent** .

Note When filling out a form, you can only indent text typed into fields that allow text formatting. The indentation options are unavailable for fields that don't allow text formatting.

About AutoComplete

Microsoft Office InfoPath 2003 uses the AutoComplete feature provided by Internet Explorer. AutoComplete saves previous entries you've typed into fields when filling out a form. If Autocomplete is turned on and enabled for a field on the form, when you type the first few characters into that field, AutoComplete lists possible matches from entries you've typed before. You can then insert the entry you want, or if none of the suggestions match what you are typing, you can continue typing into the field.

Change AutoComplete settings

When filling out a form, you can configure AutoComplete to save and suggest only the information you want. You can choose whether to use AutoComplete for Web addresses, fields on forms, and passwords, or not to use it at all. You can also delete AutoComplete entries you've added. When designing a form, you can specify whether or not AutoComplete is enabled for a text box control on a form.

Do one of the following:

- Turn AutoComplete on or off
- Delete previous AutoComplete entries
- Enable or disable Autocomplete for a text box control

Insert a hyperlink

Microsoft Office InfoPath 2003 automatically inserts a hyperlink for you when you type a Web site address (such as www.Microsoft.com), a file location, or an e-mail address when filling out or designing a form. However, when filling out a form you can only insert a hyperlink into fields that allow hyperlinks. You can also turn text or a picture into a hyperlink.

Note To automatically create a hyperlink for text that includes spaces, enclose the phrase in quotation marks ("). For example, you would need to type "www.hyperlink text" to have the entire phrase formatted as a hyperlink.

▸ Insert a hyperlink when filling out a form

▸ Insert a hyperlink when designing a form

Copy a hyperlink

1. Select the hyperlink you want to copy.
2. On the **Standard** toolbar, click **Copy** .
3. Select where you want to place a copy of the hyperlink in your form, and then click **Paste** on the **Standard** toolbar.

   Note When filling out a form, if the hyperlink loses its formatting when you paste it into the form, you may be trying to insert a hyperlink into a part of a form for which they are not allowed.

Edit a hyperlink

You can edit both a hyperlink's display text, as well as its destination.

- Edit a hyperlink when filling out a form
- Edit a hyperlink when designing a form

Remove a hyperlink

Do one of the following:

- Remove a hyperlink completely
- Remove a hyperlink but keep the text or picture

Note When filling out a form, you can only remove hyperlinks that are not a part of the form template design.

Insert a picture

When filling out a form, you can insert pictures into picture fields and rich text fields. When designing a form, you can add pictures to your form as embedded objects or as linked objects, which can help reduce the form's file size. Also, when designing a form, you can insert a picture control, which like rich text controls, lets users insert pictures into a form when they fill it out.

Note You can also insert pictures by using the Microsoft Clip Organizer.

Insert a picture when filling out a form

- Insert a picture into a picture field
- Insert pictures onto other parts of a form

Insert a picture when designing a form

- Embed a picture
- Link to a picture
- Insert a picture control

Position a picture relative to text

1. Select the picture you want to position.
2. On the **Format** menu, click **Picture**.
3. On the **Text Wrapping** tab in the **Format Picture** dialog box, click the text wrapping style you want.

Resize a picture

You can either resize pictures using a mouse or using specific measurements.

Note While filling out a form, you cannot resize pictures that were entered into a picture field.

▸ Resize a picture using the mouse

▸ Resize a picture using specific measurements

Provide alternative text for a picture

You can specify alternative text to appear instead of pictures in your form.

1. Click the picture.
2. On the **Format** menu, click **Picture**, and then click the **Text** tab.
3. In the **Alternative text** box, type the text you want. The text can be as long as you want, but some Web browsers might be able to display only a limited number of the characters.

Graphics file types InfoPath can use

You can insert many popular graphics file formats into your Microsoft Office InfoPath 2003 forms, either directly or with the use of separate graphics filters. You don't need a separate graphics filter installed to insert the following file formats:

Enhanced Metafile (.emf)

Graphics Interchange Format (.gif)

Joint Photographic Experts Group (.jpg)

Portable Network Graphics (.png)

Microsoft Windows Bitmap (.bmp, .rle, .dib)

Windows Metafile Graphics (.wmf)

Tagged Image File Format (.tiff)

Encapsulated PostScript (.eps)

File types that require filters

You need a graphics filter installed to insert all other graphics file formats. The separate graphics filters are not supplied with the stand-alone version of Microsoft Office InfoPath 2003. To use the filters, you need to install the Microsoft Office Edition or a stand-alone version of Microsoft Word along with your stand-alone version of Office InfoPath 2003. The following list contains common file types and the corresponding graphic filters that are supported.

- Computer Graphics Metafile (.cgm)
- CorelDRAW (.cdr, .cdt, .cmx, and .pat)
- Encapsulated PostScript (.eps)
- Hanako (.jsh, .jah, and .jbh)
- Macintosh PICT (.pct)
- WordPerfect Graphics (.wpg)
- Other filters

Find and replace text

Use Microsoft Office InfoPath 2003 to quickly find and replace text on your form.

Note When filling out a form, the Find and Replace features only search the text you have typed into a form and not text that is a part of the form template, such as field labels.

- Find text
- Replace text

Search options

- Search using wildcards
- Search using matching letter case

Undo mistakes

To undo the last action you took, click **Undo** ↶ on the **Standard** toolbar.

Note If you decide that you didn't want to undo an action, click **Redo** ↷ on the **Standard** toolbar.

Redo your last action

On the **Standard** toolbar, click **Redo** ↷.

Note If you can't redo the last action, the **Redo** button will appear dimmed.

Align contents of a table cell

You can change the horizontal (left, center, right, or justify) or vertical (top, center, or bottom) alignment of the contents in a table cell .

- Horizontally align the contents of a table cell
- Vertically align the contents of a table cell

Align a table

When filling out a form, you can align tables you have inserted into rich text fields. However, you cannot modify tables that are a part of the form template. When designing a form, you can align layout tables to help organize and position the content of your form.

1. Right-click the table you want to align, and then click **Table Properties** on the shortcut menu.

2. On the **Table** tab in the **Tables Properties** dialog box, click the horizontal alignment option you want.

   Note When filling out a form, the **Table Properties** option may be unavailable for tables that are a part of the form template.

Change the cell padding of table cells

You can adjust the space between the border of a cell and the contents within it by changing the cell padding. When filling out a form, you can change the cell padding of tables you have inserted into rich text fields. However, you cannot modify tables that are a part of the form template. When designing a form, you can change the cell padding of layout tables.

1. Right-click the cell or cells that contain the contents you want to position, and then click **Table Properties** on the shortcut menu.

   Note When filling out a form, the **Table Properties** option may be unavailable for tables that are a part of the form template.

2. Click the **Cell** tab on the **Table Properties** dialog box.

3. Under **Cell padding**, select the specific measurements you want for the cell padding.

Copy or move a table or its contents

When filling out a form, you can copy or move all or part of a table. However, you cannot move tables that are a part of the form template. When designing a form, you can copy and move all or part of a layout table or repeating table.

Note Copying and pasting repeating tables may cause binding issues with the corresponding fields or groups.

1. Select the table, cells, rows, or columns you want to copy or move.
2. Do one of the following:

- If you want to copy the selected items, click **Copy** on the **Standard** toolbar.
- If you want to move the selected items to another location, click **Cut** on the **Standard** toolbar.

3. Select the field where you want to insert the items, and then click **Paste** on the **Standard** toolbar.

Split or merge table cells

When filling out a form, you can split or merge the cells in tables you have inserted into rich text fields. However, you cannot modify tables that are a part of the form template. When designing a form, you can split or merge the cells in layout tables.

Do one of the following:

▸ Split table cells into multiple cells

▸ Merge multiple table cells into a single cell

Resize all or part of a table

When filling out a form, you can resize tables you have inserted into rich text fields. However, you cannot modify tables that are a part of the form template. When designing a form, you can resize layout tables and repeating tables.

Do one of the following:

- Resize an entire table
- Change column width
- Change row height

Delete rows or columns from a table

When filling out a form, you can delete rows or columns from tables you have inserted into rich text fields. However, you can not delete columns from tables that are a part of the form template. When designing a form, you can delete rows and columns from both repeating tables and layout tables.

Select the row or column you want to delete, and then press **DELETE**.

Note In design mode, if you delete a column from a repeating table in a form template that has existing forms (.xml files) based on it, data may be lost from those forms.

Delete a table

When filling out a form, you can delete tables that are inserted into rich text fields. However, you cannot delete tables that are a required part of the form. When designing a form, you can delete layout tables or repeating tables.

Note If you delete a repeating table (or any of its rows or columns) from a form template that has existing forms (.xml files) based on it, data may be lost from those forms.

1. Click in the table you want to delete.
2. On the **Table** menu, point to **Delete**, and then click **Table**.

Insert a table

When filling out a form, you can insert tables into rich text fields. When designing a form, you can insert layout tables to organize the form and repeating tables to allow users to enter information into the form.

Do one of the following:

- ▸ Insert a table when filling out a form
- ▸ Insert a layout table
- ▸ Insert a repeating table (automatically create the data source)
- ▸ Insert a repeating table (bind to an existing group)

Add rows and columns to a table

When filling out a form, you can add rows to tables that are part of the form template, or add rows and columns to tables you have inserted into rich text fields. When designing a form, you can add columns to repeating tables, or rows and columns to layout tables.

Add rows or columns when filling out a form

- Add rows to a table that is part of the form template
- Add rows or columns to a table inserted into a rich text field

Add rows or columns when designing a form

- Add rows or columns to a layout table
- Add columns to a repeating table (automatically create the data source)
- Add columns to a repeating table (bind to an existing field)

Modify a list

When filling out a form, you can modify lists you have inserted into rich text fields. However, you can not modify lists that are a part of the form template. When designing a form, you can format lists and list controls.

- Change the appearance of a list
- Change the starting number of a list

Remove a list

When filling out a form, you can remove bulleted or numbered lists inserted into rich text fields on your form. However, you cannot remove lists that are a part of the form template. When designing a form, you can remove both bulleted or numbered lists, as well as list controls.

▸ Remove bullets or numbering

▸ Remove a list control when designing a form

Insert a list

When filling out a form, you can insert bulleted or numbered lists into rich text fields on your form as you type, or you can add bullets or numbering to existing text. When designing a form, in addition to inserting bulleted or numbered lists, you can also insert list controls, which automatically format text into lists as users type into them.

Insert a list

- Create bulleted and numbered lists as you type
- Add bullets or numbering to existing text

Insert a list control when designing a form

- Insert a list control (automatically create the data source)
- Insert a list control (bind to an existing field)

Copy and paste a section

When filling out a form, because the form template specifies the structure and placement of sections within the form, copied sections can only be pasted into rich text fields or over similar sections.

Caution When designing a form, you should be cautious when you copy and paste sections (particularly when moving controls from a non-repeating section to a repeating section) because doing so may create binding issues with the corresponding fields or groups.

1. Point to the section of the form that you want to select until a dashed border appears around the section, and then click the section.

   ▸ Tips

2. Click the shortcut menu button ▼ that appears in the upper-left corner of the selected section, and then click **Copy** on the shortcut menu.
3. If you want to copy the section to a different form, open that form.
4. Click where you want to insert the section.
5. On the **Standard** toolbar, click **Paste** 📋.

Remove a section

When filling out a form, you can remove some sections from a form. However, if a section is a required part of a form, it cannot be removed. When designing a form, you can remove sections, optional sections, and repeating sections, but if there are existing forms (.xml files) created from the form template, removing sections may result in the loss of data from those forms.

Do one of the following:

- Remove a section when filling out a form
- Remove a section when designing a form

Insert a section

When filling out a form, you may be able to insert sections by using the **Insert** menu or placeholder text. In addition, when designing a form, you can insert a section to organize the controls in the form for your users.

Insert a section when filling out a form

▸ Insert a section using the **Insert** menu

▸ Insert an optional section using placeholder text

▸ Tip

Insert a section when designing a form

▸ Insert a section (automatically create the data source)

▸ Insert a section (bind to an existing group)

About views

In Microsoft Office InfoPath 2003, views are based on the Extensible Style Language (XSL) standard, which allows for the transformation and presentation of structured form data. Each Office InfoPath 2003 form has a default view, which is displayed to users when opening and filling out a form.

Form designers can also create custom views for their forms. Creating custom views can be useful in various scenarios. For example, if your form design is getting too long or too complex for everyone in your company, you can move parts of your form into different custom views and let users choose which view best meets their needs.

You can create custom views in the **Views** task pane in design mode. Users filling out the form can access the form's views on the **View** menu. When a user switches views, the form's data is not altered; only the presentation of the form and the amount of data displayed changes.

▸ Tip

Aside from custom views, you can also create print views for your forms. By default, when a user selects the **Print** command when filling out a form, the current view is printed. Form designers can also designate any existing view as a print view and specify custom printing options for it, such as headers, footers, and the page orientation of the form. When a user prints a form they are filling out that contains a designated print view, Office InfoPath 2003 will use the print view instead of the current view to print the form. This is useful when you want users to print out forms using a particular layout or to only print out certain parts of the form.

Switch between views

You can switch between views when filling out or designing a form.

- Switch between views when filling out a form
- Switch between views when designing a form

Check spelling

By default, Microsoft Office InfoPath 2003 checks spelling automatically as you type, using wavy red underlines to indicate possible spelling problems. You can also check all spelling errors at once.

- ▸ Check spelling automatically as you type
- ▸ Check spelling all at once

Notes

When filling out a form, the spelling checker will only check text typed into a field that allows spell checking, and will not check text that is a part of the form's design.

When designing a form, you can specify where in the form users are allowed to check their spelling.

Check spelling in other languages

When you use Microsoft Office InfoPath 2003 with multiple languages enabled, InfoPath checks the spelling in your form one language at a time. If you want to check text in another language, you must use the following procedure to switch to the dictionary for that language. You can repeat this procedure for each language for which you want to check spelling.

1. On the **Tools** menu, click **Set Language**.
2. In the **Language** dialog box, select the language whose spelling you want to check, and then do one of the following:

- ▸ Check spelling automatically as you type
- ▸ Check spelling all at once

Notes

If spelling options are not available for the language you have selected, InfoPath may require files that are available in the Microsoft Office 2003 Proofing Tools.

When you fill out a form, the spelling checker checks only text that is typed into a field that allows a spelling check and does not check text that is a part of the form's design.

When designing a form, you can specify where in the form users are allowed to check their spelling.

Turn on or off automatic spelling checker

1. On the **Tools** menu, click **Options**, and then click the **Spelling** tab.
2. Select or clear the **Check spelling as you type** check box.

Show or hide wavy underlines

You can show or hide the wavy red underlines that appear when you automatically check spelling.

1. On the **Tools** menu, click **Options**, and then click the **Spelling** tab.
2. Make sure the **Check spelling as you type** check box is selected.
3. Select or clear the **Hide spelling errors in this document** check box.

Set spelling preferences

1. On the **Tools** menu, click **Options**, and then click the **Spelling** tab.
2. Select the options you want.

Turn on or off German spelling rules

To check the spelling of German text, you must install the German spelling dictionary, which is available in the Microsoft Office 2003 Proofing Tools.

1. On the **Tools** menu, click **Options**, and then click the **Spelling** tab.
2. Select or clear the **Use German post-reform rules** check box.

About data validation

Data validation can be used to display error alerts when users enter incorrect values into a form. Rather than checking for errors after a form is completed, data validation verifies values as the form is being filled out.

Working with data validation when filling out a form

Microsoft Office InfoPath 2003 automatically checks the data you type into a form and displays an error alert if the value does not match the condition specified by the data validation rule. The most common types of data validation include required entries, where you must enter a value; range checking, where the value you enter must fall into a certain range; and data types, where the value you enter must be a certain type, such as a number or date. If a value you enter does not match the specified condition, an error alert provides you with immediate feedback.

Error alerts can be either dialog box alerts or inline alerts. You can correct errors immediately, or you can use the commands on the **Tools** menu to navigate through and correct all the errors in a form at one time. Uncorrected errors are marked with a dashed red border or, for required entries, a red asterisk. If you save a form without fixing the errors, you will be reminded about the errors with a dialog box alert. You can save forms with validation errors, but you cannot submit them to a database or Web service.

Working with data validation when designing a form

Data validation is always associated with a specific control and occurs when a user fills out a form. Note that Office InfoPath 2003 verifies a condition by checking the value of the field that is bound to the control, not the control itself.

▸ Types of data validation

▸ Types of error alerts

▸ Schema-based validation

To test a data validation constraint, including the conditions and error alerts, you can preview your form and enter valid and invalid data into the control you want to test. Error alerts will appear in the preview exactly as they appear to your users.

Note Conditional formatting is another effective way of responding to data entry. If you prefer to notify users with changes in color or text formatting, and you do not need to show an error alert, you can use conditional formatting for range checking and dynamic comparisons.

Check for data errors in a form

Microsoft Office InfoPath 2003 automatically checks the information you enter into a form for data errors as you type. If a problem with the information you type into a form is found, you will be presented with either a traditional dialog box alert or an inline alert.

1. Type into the form.
2. Do one of the following:
   - ▸ If a dialog box alert appears
   - ▸ If an inline alert appears

About saving and publishing forms

In Microsoft Office InfoPath 2003, you can save a form you are filling out to your computer or to a shared location. Every form (.xml file) you fill out is based on a form template (.xsn file), which is created in design mode. When designing form templates, you can save them to your computer as you work. When you finish designing a form template, you can publish it to make it available to other users.

- ▸ Saving forms you are filling out
- ▸ Saving form templates in design mode
- ▸ Publishing form templates in design mode

About sharing your forms with others

When you are finished filling out a form, there are several ways for you to share your form with others. In addition to making it accessible to others by saving it to a shared location, such as a Windows SharePoint Services site, you can send your form in an e-mail message, export it to the Web, or submit it directly to a database or Web service.

Note If you want to make form templates available for other users to fill out, you must publish them to a shared location by using the Microsoft Office InfoPath 2003 Publishing Wizard.

- Sending forms in an e-mail message
- Exporting forms to the Web
- Submitting form data to a database or Web service

Save a form

When you save a form you have filled out, it is saved with an .xml file name extension. When you save a form in design mode, you are actually saving a form template, which is saved with an .xsn file name extension.

- Tip
- Save when filling out a form
- Save a form template in design mode

Save a form as a Web page

When filling out a form, the Export to Web feature lets you save your form as a Single File Web Page (MHTML). This format allows other users to view your form in a Web browser, but it does not allow them to edit the data in the form.

1. Open the form you want to save as a Web page. If there are multiple views in the form, select the view you want to save.
2. On the **File** menu, point to **Export To**, and then click **Web**.
3. In the **Export to Web** dialog box, specify where you want to save the form.
4. In the **File name** box, type a name for the form, and then click **Export**.
5. Repeat this procedure for every view of the form you want to save as a Web page.

Submit a form

When filling out a form, you can submit the data in your form directly to a database or Web service, if the form you are filling out has been designed to work with databases or Web services and you are properly connected.

Note If this feature is not enabled for the form you are working with, the **Submit** command will not be available.

After you have finished filling out your form and have checked it for data errors, click **Submit** on the **File** menu.

Security Before submitting a form you have filled out to a database or Web service, make sure that the form you received is from a trusted source.

Send a form using e-mail

You can share your forms with others by sending them in the body of an e-mail message or as an attachment.

Note Because storing a form template (.xsn file) in a single location helps ensure that forms based on the form template open and function properly, it is recommended that you use the Microsoft Office InfoPath 2003 Publishing Wizard to share form templates with others.

▸ Send a form in the body of an e-mail message

▸ Send a form as an attachment

Security The recipient of an e-mail message may not receive an attachment depending on the attachment file type. For example, program file type (.exe) attachments are blocked by Microsoft Outlook. The security level of file types is, in some cases, set by the e-mail server administrator. Make sure to provide an alternative method for recipients to access the attachment.

Merge multiple forms

When filling out a form, if you have multiple forms that were either created from the same form template or designed to work together, you can merge them into a single form. The form template specifies whether the Merge Forms feature is enabled for a given form, what the merged form's appearance will be, and which data is included or merged into the new form.

1. Type or enter the data you want into your form, or open an existing form (.xml file).
2. On the **File** menu, click **Merge Forms**.
3. In the **Merge Forms** dialog box, select the files you want to merge with the form you are working on, and then click **Merge**.

Export a form to Excel

You can export the data in Microsoft Office InfoPath 2003 forms to Microsoft Office Excel 2003. This allows you to analyze and work with the data you have collected in the forms users have filled out. You can export data from the current view of an opened form using the Export to Excel Wizard.

Note The Export to Excel Wizard also allows you to export data from several forms at once.

1. Open the form you want to export to Office Excel 2003.

   Note If the form you open contains digital signatures, you must first remove the signatures before you can export the form to Office Excel 2003.

2. On the **File** menu, point to **Export To**, and then click **Microsoft Excel**.
3. Follow the steps in the Export to Excel Wizard.
4. To export data from additional views of the same form, repeat the previous steps for each view.

Understanding key design mode icons

Have you designed or customized a Microsoft Office InfoPath™ 2003 form and wondered what all those different symbols mean? This topic will help you understand the key icons used in InfoPath design mode.

Data source icons

Understanding the different icons in the **Data Source** task pane will help you to better understand how all the different pieces of data stored in fields and groups are related to one another.

Icon	Description
⊞ 📂	Group. An element in the data source that can contain fields and other groups.
⊞ 📂🔒	Locked group. This group cannot be renamed, moved, or deleted, and new fields or groups cannot be added to it.
⊞ 📂	Repeating group. A group in the data source that can occur more than once.
⊞ 📂🔒	Locked repeating group. This group cannot be renamed, moved, or deleted, and new fields or groups cannot be added to it.
📄	Element field. A field in the data source that stores the data entered into controls. This field can contain attribute fields.
📄🔒	Locked element field. This field cannot be renamed, moved, or deleted, and new attributes cannot be added to it.
📄	Repeating element field. A field in the data source that can occur more than once. Controls such as lists, repeating sections, and repeating tables can be bound to repeating fields.
📄🔒	Locked repeating element field. This field cannot be renamed, moved, or deleted, and new attributes cannot be added to it.
📄	Attribute field. A field in the data source that stores data entered into controls. Other fields cannot be added to it.
📄🔒	Locked attribute field. This field cannot be moved or deleted.
*	Required field. If a control on the form is bound to a required field, it cannot be left blank when a user fills out the form.

Binding icons

When you design a form, several icons may appear in the upper-right corner of controls. These are known as binding icons. They provide information about how a selected control relates to fields and groups in the data source. The following table describes the three types of binding icons you may see on your form.

Binding icons	Description
🟢	The binding is correct. This icon, along with the name of the field or group that the selected control is bound to in the data source, is displayed when a control is selected or when the mouse pointer rests on the control.
ℹ	The control may not function as expected because of a potential binding problem. This icon is displayed regardless of whether the control is selected. The name of the field or group that the control is bound to, along with a short message, is displayed when the control is selected, or when the mouse pointer rests on the control. To identify the binding problem, right-click the control, and then click **More Details** on the shortcut menu.
❗	The control will not function correctly because of a binding problem, and when a user saves the form, any data typed into the control will be lost. This icon is displayed regardless of whether the control is selected. To identify the binding problem, right-click the control, and then click **More Details** on the shortcut menu.

Other InfoPath icons

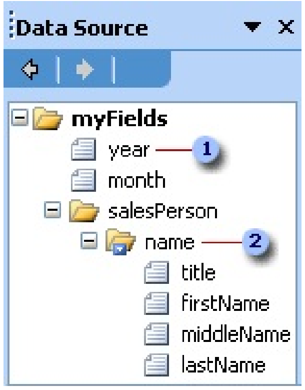
The following table lists a few other InfoPath icons that you should be familiar with.

Icon	Description
▼	Repeating control, field, or group. A control, field, or group that can appear more than once. This icon appears both in design mode and when users fill out a form.
□	Optional section. A control on a form that can contain other controls and does not usually appear by default. This icon appears both in design mode and when users fill out a form.
🗎	InfoPath form (.xml file). A file that contains the data that users enter into a form and is created after users save their form. This icon appears in task panes and dialog boxes, such as the **Open** dialog box.
🗎	InfoPath form template. A file or set of files that defines the data structure, appearance, and behavior of a form. This icon appears in task panes and dialog boxes, such as the **Open** dialog box.

About the data source

The data source, which stores all the data entered into a form, is made up of fields and groups. In the same way that folders on your hard drive contain and organize your files, form fields contain data, and groups contain and organize the fields. For example, the fields for first name, middle name, and last name can be contained in a group called "name."

You can work with your form’s fields and groups in the **Data Source** task pane.

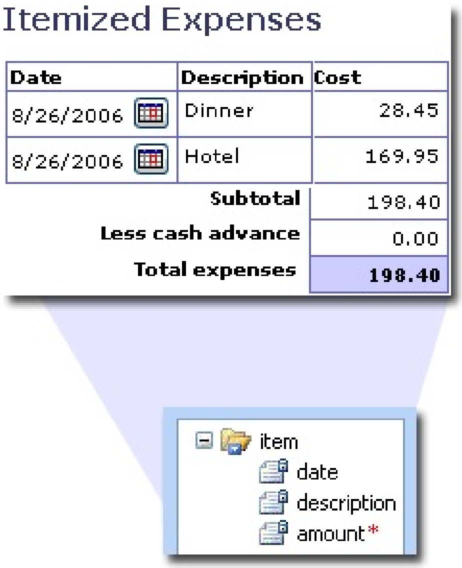


1 Field

2 Group

The controls on your form are bound to the fields and groups in the data source; this binding allows data entered into a control to be saved. Information entered into a bound control is saved in the field it is associated with.

Although the structure of the data source does not always exactly match the layout of your form, similarities are common, particularly for groups and fields associated with repeating tables, sections, repeating sections, and optional sections. In these cases, a table or section is bound to a group in the data source, and all the controls in the table or section are bound to fields that are part of that group. In the following illustration, the Itemized Expenses repeating table consists of three columns containing a Date date picker, a Description text box, and a Cost text box. These controls are bound to a date field, a description field, and an amount field. The table itself is bound to the item group.



Itemized Expenses

Date	Description	Cost
8/26/2006	Dinner	28.45
8/26/2006	Hotel	169.95
	Subtotal	198.40
	Less cash advance	0.00
	Total expenses	198.40

item
date
description
amount*

Using the Data Source task pane to insert controls

You can insert controls by dragging fields and groups from the **Data Source** task pane onto the form. Each time you drag a field or group, Microsoft Office InfoPath 2003 suggests the control that best matches the field or group. If the field has a date data type, for example, InfoPath suggests a date picker. If the group is repeating and contains three fields, InfoPath suggests a repeating table or a repeating section. Remember that you can drag the group that contains a set of fields to automatically create repeating tables, repeating sections, and sections. To see all the control recommendations for a field or group, right-click the field or group in the **Data Source** task pane.

Modifying fields and groups in the data source

When working directly with the data source in the **Data Source** task pane, you can add, move, reference, and delete fields and groups. In addition, you can change the following properties of fields and groups in the data source:

Name The name of the field or group. Names cannot contain spaces. They must begin with an alphabetic character or underscore ("_"), and they can contain only alphanumeric characters, underscores, hyphens ("-"), and periods (".").

Type The type of field or group. Options are element field (the standard type of field), attribute field, and group. Fields store data that is entered into controls; groups contain fields and cannot store data.

Data type Defines the kinds of data the field can store. Data types include Text, Rich Text, Whole Number, Decimal, True/False, Hyperlink, Date, Time, Date and Time, and Picture. Only fields can have data types.

Default value The initial value that a field has when the form is first opened by a user. Only fields can have default values.

Repeating Determines whether the field or group can occur more than once in a form. List controls, repeating sections, repeating tables, and controls that are part of a repeating section or repeating table are bound to repeating fields and repeating groups.

Cannot be blank Determines whether the field must contain a value. If the field cannot be blank, any control bound to the field that does not have a value entered into it will be marked with an error alert.

If your users have already filled out forms based on your form template, the following changes can cause data loss in those forms:

Moving, deleting, or renaming a field or group

Changing a repeating field or group to nonrepeating

Changing a rich text data type to a different data type

You can create matching fields and groups for when you want to store the same type of data in more than one place— such as a billing and shipping address in an invoice form. When you reference a field, InfoPath creates a new field whose properties are linked and matched to the properties of the original. Both fields become reference fields, and a change to one field updates the other field automatically. Reference groups, like reference fields, share the same properties; in addition, they contain the same fields and groups.

Relationship between the Data Source and an XML Schema

The forms that users fill out are stored as XML documents. When you create the data source for a form template, you are actually creating an XML Schema, which is an XML-based file that determines the structure of those XML documents. Each group in the data source is an XML element that can contain other elements and attributes, but not data, and each standard field in the data source is an XML element that can contain data. Each attribute field is an XML attribute. You can view additional schema details on the **Details** tab of the **Field or Group Properties** dialog box.

Existing XML Schemas, XML documents, databases, and Web services

When you design a form based on an existing XML Schema, InfoPath creates a data source based on the XML Schema. Because existing XML Schemas can be more restrictive than new blank forms, you cannot modify existing fields or groups in the data source. In addition, you might not be able to add fields or groups to part or all of the data source. When you design a form based on an existing XML document, InfoPath creates the data source based on the information in that XML document. The more detailed your XML document is, the more detailed the resulting data source will be.

When you design a new form that is connected to a database or Web service, InfoPath builds a data structure for your form based on the database or the operations of the Web service. The resulting form can be used to submit data to and query the database or Web service. Because the data source must match the database or Web service, you cannot modify existing fields or groups in the data source, and you can add fields or groups only to the root group in the data source.

Design a form based on an existing XML document

When designing a new form based on an existing XML document, you can work with the form in the same way as when you design a new blank form, including inserting and laying out controls, inserting and modifying fields and groups in the data source, adding data validation and conditional formatting, and publishing the form.

Note When you design a form based on an existing XML document, Microsoft Office InfoPath 2003 creates a data source based on the information in your .xml file. The more detailed your XML document is, the more detailed the resulting data source will be.

1. On the **File** menu, click **Design a Form**.
2. In the **Design a Form** task pane, click **New from Data Source**.
3. On the first page of the Data Source Setup Wizard, click **XML Schema or XML data file**, and then click **Next**.
4. On the second page of the Data Source Setup Wizard, click **Browse**, and then, in the **Open** dialog box, browse to and select the XML document you want to base the form on.
5. In the Data Source Setup Wizard, click **Finish**.
6. In the dialog box that appears, do one of the following:

- If you want to use the values in the XML document to create default data for your form, click **Yes**. The values stored in the XML document will become the default values for the fields in your form.
- If you want to use only the structure of the XML document and not its values, click **No**.

7. To begin adding controls to the form, right-click a field or group in the **Data Source** task pane, and then, on the shortcut menu, click the control you want to insert.

Design a form based on an existing XML Schema

When designing a new form based on an existing XML Schema (.xsd) file, you can work with the form in many of the same ways as when you design a new blank form, including inserting and laying out controls, adding data validation and conditional formatting, and publishing the form.

The new form will automatically re-use the data validation of the existing XML Schema.

1. On the **File** menu, click **Design a Form**.
2. In the **Design a Form** task pane, click **New from Data Source**.
3. On the first page of the Data Source Setup Wizard, click **XML Schema or XML data file**, and then click **Next**.
4. On the second page of the Data Source Setup Wizard, click **Browse**, and then, in the **Open** dialog box, browse to and select the XML Schema you want to base the form on.
5. In the Data Source Setup Wizard, click **Finish**.
6. To begin adding controls to the form, right-click a field or group in the **Data Source** task pane, and then, on the shortcut menu, click the control you want to insert.

Notes

Because existing XML Schemas can be more restrictive than new blank forms, the following limitations apply:

You cannot automatically create the data source when inserting controls.

You cannot modify existing fields or groups in the data source.

Depending on the schema, you might not be able to add fields or groups to part or all of the data source.

Customize a sample form

When creating a new form by customizing a sample form, you can work with the form in many of the same ways as when you design a new blank form, including inserting and laying out controls, inserting and modifying fields and groups in the data source, adding data validation and conditional formatting, and publishing the form.

1. On the **File** menu, click **Design a Form**.
2. In the **Design a Form** task pane, click **Customize a Sample**.
3. In the **Customize a Sample** dialog box, double-click the sample form you want to customize.

When you customize a sample form, the following limitations apply:

You cannot automatically create the data source when inserting controls.

You cannot modify existing fields or groups in the data source.

Depending on the sample form, you might not be able to add fields or groups to part of the data source.

Design a form connected to a Web service

When you design a new form that is connected to a Web service, Microsoft Office InfoPath 2003 builds a data structure for your form based on the operations of the Web service. The resulting form can then be used to submit data to and query the Web service. You can work with the form in many of the same ways as when you design a new blank form — you can insert and lay out controls, add or remove data validation and conditional formatting, and publish the form.

You can design forms that receive data from a Web service, submit data to a Web service, or do both.

Note InfoPath cannot connect to Web service operations that use remote procedure call (RPC) encoding. Only document literal encoding is supported.

▸ Design a form that receives data from a Web service

▸ Design a form that submits data to a Web service

▸ Design a form that receives data from and submits data to a Web service

Note Because the data structure for the form must match the Web service, the following limitations apply:

You cannot automatically create the data source when inserting controls.

You cannot modify existing fields or groups in the data source.

You can only add fields or groups to the root group in the data source.

Design a form connected to a database

When you design a new form that is connected to a database, Microsoft Office InfoPath 2003 builds a data structure for your form based on the tables and structure of the database. The resulting form can then be used to submit data to and query the database. You can work with the form in many of the same ways as when you design a new blank form— you can insert and lay out controls, add or remove data validation and conditional formatting, and publish the form.

Note InfoPath can connect only to Microsoft SQL Server and Microsoft Access databases.

1. On the **File** menu, click **Design a Form**.
2. In the **Design a Form** task pane, click **New from Data Source**.
3. On the first page of the Data Source Setup Wizard, click **Database (Microsoft SQL Server or Microsoft Access only)**, and then click **Next**.
4. On the second page of the Data Source Setup Wizard, click **Select Database**, and then, in the **Select Data Source** dialog box, browse to and select the SQL Server or Access database you want to base the form on, and click **Open**.

   Note If there is more than one table in your database, click the table that you want to have as the primary parent table for your form in the **Select Table** dialog box, and then click **OK**.

5. If you want to add more than one table, do the following:

- Click **Add Table**.
- In the **Add Table or Query** dialog box, click the table you want to add, and then click **Next**.
- In the **Edit Relationship** dialog box, make sure the connecting fields are correct. To add a set of connecting fields, click **Add Relationship**, select the two fields that are equal when a record in one table is associated with a record in another table, and then click **OK**.

- In the **Edit Relationship** dialog box, click **Finish**.

6. In the Data Source Setup Wizard, click **Next**.
7. In the **Summary** box, make sure that the information matches the options you specified.
8. At the end of the Wizard, InfoPath creates two views for your form: a query view, which already contains controls, and a data entry view, which is blank. To choose the view you want to design first, do one of the following:

- To start by designing the query view, in which users can enter values with which to query the database, click **Design query view first**, and then click **Finish**.
- To start by designing the data entry view, in which users can enter new records to submit to the database and edit existing records, click **Design data view first**, and then click **Finish**.

9. To begin adding controls to the form, right-click a field or group in the **Data Source** task pane, and then, on the shortcut menu, click the control you want to insert. Use only query fields in the query view and only data fields in the data entry view.

Note Because the data structure for the form must match the database, the following limitations apply:

You cannot automatically create the data source when inserting controls.

You cannot modify existing fields or groups in the data source.

You can only add fields or groups to the root group in the data source.

Add a field or group

The fields and groups that you add to the data source define the type and structure of the data that users can enter into your form. If you want to match the properties of an existing group or field, you can also add a reference field or reference group.

Note If your form template is based on a sample form, pre-existing XML Schema, database, or Web service, you might not be able to add to some of the existing fields and groups.

- Add a field
- Add a group
- Add a reference field
- Add a reference group

Change the name of a field or group

If users have already filled out forms based on your form template, changing the name of a field or group can cause data loss in those forms.

In addition, if your form template is based on a sample form, pre-existing XML Schema, database, or Web service, you can only change the names of fields and groups that you have added to the data source.

1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
2. In the **Data Source** task pane, right-click the field or group whose name you want to change, and then click **Properties** on the shortcut menu.
3. In the **Field or Group Properties** dialog box, type the new name in the **Name** box.

   Note Names cannot contain spaces. They must begin with an alphabetic character or underscore ("_"), and they can only contain alphanumeric characters, underscores, hyphens ("-"), and periods (".").

▸ Tip

Change the data type

The values that can be entered into a control are determined by the data type of the field that is bound to the control. If a user types a value with the wrong data type, a standard inline alert will appear.

You can change the data type of a field by using the control's properties for text boxes, list boxes, date pickers, check boxes, and option buttons. Other controls, such as sections, list controls, and picture controls, either do not have a data type or can store only one kind of data type. For fields that do not have a control bound to them, you can change the data type directly in the data source.

▸ Change the data type of a field using the control's properties

▸ Change the data type of a field directly in the data source

Note If your form template is based on a sample form, pre-existing XML Schema, database, or Web service, you can change the data type only of fields that you have added to the data source.

Move a field or group

Note If users have already filled out forms based on your form template, moving a field or group can cause data loss in those forms.

1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
2. In the **Data Source** task pane, right-click the field or group you want to move, and then click **Move** on the shortcut menu.

   Note If your form template is based on a sample form, pre-existing XML Schema, database, or Web service, you can move only the fields and groups that you have added to the data source.
3. In the **Move Field or Group** dialog box, select a new location for the field or group.
4. Verify the following:

- If the field or group is moved into a repeating group, the control bound to the field or group should be part of a repeating table or repeating section.
- If the field or group is moved out of a repeating group, the control bound to the field or group should not be part of a repeating table or repeating section.
- If the field or group is moved into a group that is bound to an optional section, make sure that the control bound to the field or group is part of the optional section.
- If the field or group is moved out of a group that is bound to an optional section, make sure that the control bound to the field or group is not part of the optional section.

View the details of a field or group

You can view the XML Schema properties of a field or group by using the **Field or Group Properties** dialog box.

1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
2. In the **Data Source** task pane, right-click the field or group whose details you want to view, and then click **Properties** on the shortcut menu.
3. In the **Field or Group Properties** dialog box, click the **Details** tab.

Delete a field or group

Note Deleting a field or group will cause an error in any controls on your form that are bound to that field or group. In addition, if users have already filled out forms based on your form template, deleting a field or group can cause data loss in those forms.

1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
2. In the **Data Source** task pane, right-click the field or group you want to delete, and then click **Delete** on the shortcut menu.

   Note If your form template is based on a sample form, pre-existing XML Schema, database, or Web service, you can delete only the fields and groups that you have added to the data source.

Set the default value for a control

A default value is the initial value a control contains when the form is first opened by a user. Default values are stored in fields and are displayed in the controls that are bound to those fields.

▸ Set the default value for a text box, rich text box, date picker, or list control

▸ Set the default value for a list box or drop-down list box

▸ Set the default value and default state for a check box

▸ Set the default value and default state for a group of option buttons

Note To set the default values for an optional section, repeating section, or repeating table, you can use the procedures above for the individual controls that appear in the section or table.

Set the default values for a form

A form's default values, which are the initial values a form contains when it is first opened by a user, can all be set at the same time. Default values are stored in fields and are displayed in the controls that are bound to those fields.

Note Fields that do not appear in the data source are not the same as fields with blank values. If a field does not appear in the data source, the control bound to that field should be an optional section or part of an optional section.

1. In design mode, click **Form Options** on the **Tools** menu, and then click **Edit Default Values**.
2. In the **Edit Default Values** dialog box, select the field whose default value you want to set.

   Note To change the default values for a field, the check box for the field must be selected, and the field name itself must be selected. A field with a cleared check box will not appear initially in the data source.
3. In the **Default value** box, enter the default value for the field. Only fields can have default values.
4. If the field is a repeating field and you want to specify the number of instances of the field that should initially appear on the form, enter the number in the **Number of occurrences of the field or group to insert** box.
5. Repeat steps 2 through 4 for each field whose default value you want to set.

About form design

When designing a form in Microsoft Office InfoPath 2003, starting with a blank form provides the most flexibility and allows you to create a design that best fits your needs. While you are free to experiment and work the way that suits you best, following the steps below may save you some valuable time and effort.

Whenever you enter design mode, Office InfoPath 2003 displays the **Design Tasks** task pane, which contains links to design tools in the recommended order of design tasks. These tasks are listed below, and provide useful information on how to design a form.

Note If you have task panes turned off, you can show them again by pressing CTRL+F1 on your keyboard. Or, to access the **Design Tasks** task pane at any time, click the **Design Tasks** button on the **Standard** toolbar.

1. Create a form layout

Before you begin adding items to a new, blank form, consider creating a form layout. Layout tables help to define the dimensions and order of your form's sections and areas into which you'll place text, images, and controls. From the **Layout** task pane, you can drag various layout tables, optional sections, and repeating sections onto your form. This task pane also provides easy ways to customize layout tables using familiar Microsoft Office 2003 table-editing commands, such as adding rows and columns, and splitting and merging table cells.

▸ Tip

2. Add controls to your form

When you have finished creating your form's layout, you can add functionality to your form by inserting controls. The **Controls** task pane provides access to all of the controls that Office InfoPath 2003 supports. Double-click or drag controls onto the form to design its functionality. Based on the choices you make in the form's design, users filling out the form will be able to type text into text fields, make selections from option buttons and check boxes, choose entries from lists, and click command buttons to carry out commands.

3. Bind controls to the data source

After you place and format controls in your form, you must bind them to the form's data source. If controls are left unbound, none of the data that form users type into the form will be saved. Office InfoPath 2003 automatically creates an XML data source when you create a new, blank form and add controls. If you're working from an existing XML Schema or data source, you will be prompted to bind your controls as you add them to your form's layout. In either case, you can use the **Data Source** task pane to view the structure of your form's data, show and hide details for each of the fields and groups, and modify their associations.

4. Create custom views

By default, every form that you open or create in Office InfoPath 2003 has a default view. While the default view is suitable for most forms, multiple views of the same data provide the ability to look at the same information in different ways. For example, a manager might want to see a summary view of an employee's detailed status report. In Office InfoPath 2003, you can modify the default view and create custom views, including print views that specify custom print settings for users filling out and printing the finished form.

5. Publish the form

While you can save the form you are designing to your local hard drive at any time, you must publish your finished form to a shared location in order to make it available to other users. Before you publish your form, you should test its layout and functionality to make sure it looks and works as expected. In design mode, you can use the **Preview** command to view your form exactly as it will appear to a user who is filling it out. When everything looks fine, use the Publishing Wizard to publish your finished form to a shared folder on your computer or your company's network, to your intranet or Internet Web server, or to a Windows SharePoint Services site.

About form layout

In Microsoft Office InfoPath 2003, a form's visual structure or layout is created using layout tables. Once you have defined your form's layout, you can type text into it, such as a title, insert fields and controls, and add sections and tables that your users can interact with.

In addition to layout tables, you can add optional sections and repeating sections to your form. These containers allow users to add or remove additional areas of information within the form when filling it out. Just like layout tables, you can place fields, controls, and even other layout tables, sections, or repeating tables into these containers.

Optional sections don't appear in, and aren't saved with, the form unless a user chooses to add them. For example, while all employees in a company might use a goal setting section in their Performance Review form, only managers might choose to add an optional section about leadership goals.

Repeating sections let users expand the contents of a form when it is filled out, displaying only the necessary number of entries in a series. For example, the itemized expenses in an Expense Report form can be formatted as a repeating table. This allows the user filling out the form to insert additional table rows for the appropriate number of items.

The **Layout** task pane provides a collection of drag-and-drop layout tables and sections that you can place onto a blank form. If none of the provided layout tables and sections meet your needs, you can insert a custom table with the exact number of rows and columns you want. Layout tables can be formatted and modified like tables in other Microsoft Office programs. Use the **Table** menu to format a selected table, or right-click a table to set its properties.

▸ Tip

About color schemes

Color schemes offer a quick and easy way to create professional-looking forms in Microsoft Office InfoPath 2003. In design mode, when you apply a color scheme to a form, Office InfoPath 2003 customizes certain elements of the form layout with complimentary colors based on your choice. Depending on the way the form was designed, this can include body and heading styles, table cells, and table borders. To change the color of these elements, click any of the available color schemes in the **Color Schemes** task pane.

Apply a color scheme

Note Color schemes affect only certain parts of a form. Depending on the way the form was designed, this can include body and heading styles, table cells, and table borders. If none of these exist in your form, applying a color scheme may not provide visual feedback about the change.

1. In design mode, click **Color Schemes** on the **Format** menu.
2. In the **Color Schemes** task pane, do one of the following:

- To apply a color scheme, click a color scheme in the **Apply a color scheme** list.
- To remove the currently applied color scheme, click **None** in the **Apply a color scheme** list.

Add or remove borders and shading

Borders and shading can add interest and emphasis to a form. You can add borders and shading to many parts of a form, including tables and table cells, controls, and objects such as pictures.

- Add a border
- Remove a border
- Add shading
- Remove shading

Add a border around a form

You can add a border to outline your entire form by placing your layout and controls into a single-cell layout table.

1. In design mode, press CTRL+A to select the contents of the entire form, and then press CTRL+X to cut the selection.
2. On the **Insert** menu, click **Layout Table**.
3. In the **Insert Table** dialog box, enter **1** for both the **Number of columns** and the **Number of rows**.
4. Right-click the layout table on the form, and then click **Borders and Shading** on the shortcut menu.
5. In the **Borders and Shading** dialog box, click the **Borders** tab.
6. Under **Presets**, click **Outline**.
7. With the insertion point inside the layout table, press CTRL+V to paste your form contents.

▸ Tip

Add or delete a view

Each Microsoft Office InfoPath 2003 form has a default view, called View 1, that users see when filling out a form. When designing a form, you can add new views to the form and delete those you no longer need.

1. In design mode, click **Manage Views** on the **View** menu.
2. In the **Views** task pane, do one of the following:

- To add a new view, click **Add a New View** under **Actions**, and then in the **Add View** dialog box, type a name for the new view.
- To delete a view, click the drop-down arrow next to the view name you want to delete, and then click **Delete**.

Note Deleted views cannot be recovered or restored. If you delete the default view, a new default view will be created automatically.

Rename a view

1. In design mode, click **Manage Views** on the **View** menu.
2. In the **Views** task pane, select the view you want to rename, and then click **View Properties**.
3. On the **General** tab in the **View Properties** dialog box, type a new name for the current view in the **View name** box.

Note You can not undo renaming a view. If you have associated a custom script with a particular view of your form, renaming the view will break the script association.

Modify view properties

In design mode, you can set the background color, layout width, and text formatting options for the form's default view. You can also show or hide a view, and create a print view for a form. Do one of the following:

- Set the background color
- Set the layout width
- Set text formatting options for controls
- Show or hide a view
- Create a print view

Set the default view

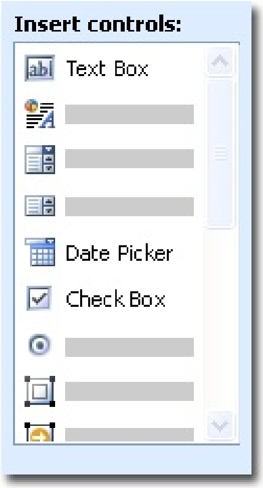
Each Microsoft Office InfoPath 2003 form has a default view, called View 1, that users see when filling out the form. If you create custom views for a form, you can choose which of the available views will be the default view for users.

1. In design mode, click **Manage Views** on the **View** menu.
2. In the **Views** task pane, click the drop-down arrow next to the view name you want to designate as the default view, and then click **Set as Default**.

Note If only one view is listed in the **Select a view** list it is automatically designated as the default view.

About controls

When filling out forms, users enter information by typing in text boxes, choosing items in a list, selecting check boxes, and so forth. The objects users interact with are called controls. A control is a graphical object, such as a text box, check box, or button, that appears on a form. Information entered into a control is stored in the form's data source. When you design a form, you use the **Controls** task pane to insert controls.

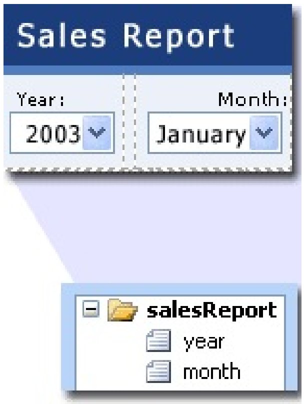


You can customize a control's appearance, including font, color, and alignment, its default value and data type, and other properties, such as the ScreenTip that appears when a user rests the mouse pointer on the control and a keyboard shortcut to manipulate the control without a mouse. To view the properties for a control, double-click the control on the form.

Controls and the data source

Almost all controls on an InfoPath form are associated with, or bound to, the data source. Most controls, including text boxes, check boxes, and list boxes, are bound to fields. A field is where the information entered into the control is saved. If a control is not bound, or if it is bound incorrectly, the information entered into the control cannot be saved. When you design forms in Microsoft Office InfoPath 2003, you can choose to have the fields in the data source created automatically each time you insert a control, or you can bind controls to existing fields.

The following example illustrates the relationship between two drop-down list boxes and the fields that store their data.



Together, a control and its field determine the type of information users can enter, as well as how they enter it. For example, a date picker control makes it easy for users to enter dates, and the field to which the date picker is bound saves only values that are dates (in other words, the field has a date data type).

Inserting controls in your form

When designing a form, you can use the **Controls** task pane to insert controls. You can choose to create the data source automatically when you insert a control, in which case a field or group will be automatically created for each control you add, or to bind the control to an existing field or group. If you choose to bind to an existing field or group, you are prompted to select a field or group from the data source when you insert the control. To switch between automatically creating the data source and binding controls to an existing field or group, use the **Automatically create the data source** check box in the **Controls** task pane.

You can also insert controls by dragging fields and groups from the **Data Source** task pane onto the form. Each time you drag a field or group, InfoPath suggests the control that best matches the field or group. If the field has a date data type, for example, InfoPath suggests a date picker.

Types of controls

- Standard controls
- Repeating and optional controls
- Hyperlink and picture controls
- Buttons and expression boxes

Testing controls

To test controls, you can preview the form and enter information into the controls, add and remove rows and sections from repeating controls, and add and delete optional sections. In addition, you can select any control to display an icon that indicates whether the control is correctly bound to its field. There are three possible icons:

🟢 The binding is correct.

ℹ The control may not function as expected because of a potential binding problem. To identify the potential problem, right-click the control, and then click **More Details** on the shortcut menu.

❗ The control will not function correctly because of a binding problem. To identify the problem, right-click the control, and then click **More Details** on the shortcut menu.

Insert a button

You can create a button on your form to make it easy for users to perform common tasks, such as submitting the form. You can also associate a button with script that runs when a user clicks the button.

1. In design mode, place the insertion point where you want to insert the button.
2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
3. In the **Controls** task pane, click **Button**.
4. Double-click the button icon and, in the **Button Properties** dialog box, click the action you want the button to perform in the **Action** list.

   ▸ Available actions

5. In the **Label** box, type the text that you want to appear on the button.
6. If you chose the Script action and you want to write script to associate with the button, click **Microsoft Script Editor**. In Microsoft Script Editor, type the script that you want to run when the button is clicked.

Insert a check box

You can create a check box in your form so users can easily and accurately enter true or false values. Check boxes are bound to fields.

- Insert a check box (automatically create the data source)
- Insert a check box (bind to an existing field)
- Tip

Insert a control into a section

When you insert a control into an existing section, repeating section, or optional section, the control's field must be part of the section's group. If you are automatically creating the data source while inserting the control, the field's location will be correct. If the control is bound to an existing field, however, you must make sure that the field is part of the section's group.

▸ Insert a control (automatically create the data source)

▸ Insert a control (bind to an existing field)

▸ Tip

Insert a date picker

You can create a date picker in your form so users can easily enter dates. Date pickers are bound to fields.

- Insert a date picker (automatically create the data source)
- Insert a date picker (bind to an existing field)
- Tip

Insert a list box

You can create a list box or drop-down list box on your form so users can easily and accurately enter predefined values. List boxes are bound to fields, and they can look up their values in a fixed list, an XML document, a Microsoft Access database or Microsoft SQL Server database, or a Web service.

- Insert a list box that looks up values in a fixed list
- Insert a list box that looks up values in an XML data file
- Insert a list box that looks up values in a database
- Insert a list box that looks up values from a Web service
- Tip

Insert an optional section

You can create an optional section in your form that users can insert and delete. By default, an optional section does not appear in the form until a user inserts it. Optional sections are bound to groups; any control in an optional section must be bound to a field that is part of the optional section's group. Unlike conditional formatting, which can make a section invisible but does not remove it from the data source, an optional section is not part of the data source of the form until a user inserts it.

- Insert an optional section (automatically create the data source)
- Insert an optional section (bind to an existing group)
- Tip

Insert a repeating section

You can create a repeating section in your form so users can add additional sections while filling it out. Repeating sections are bound to groups; any control in a repeating section must be bound to a field that is part of the repeating section's group.

▸ Insert a repeating section (automatically create the data source)

▸ Insert a repeating section (bind to an existing group)

▸ Tip

Insert a rich text box

You can create a rich text box in your form into which users can enter formatted text, paragraph breaks, tables, lists, and images. Rich text boxes are bound to fields.

- Insert a rich text box (automatically create the data source)
- Insert a rich text box (bind to an existing field)
- Tip

Insert a text box

You can create a text box in your form that users can enter data into. Text boxes are bound to fields.

- Insert a text box (automatically create the data source)
- Insert a text box (bind to an existing field)
- Tip

Insert option buttons

You can create a group of option buttons on your form so your users can easily and accurately enter specific values. A group of option buttons is bound to a single field in the data source. When a user clicks an option button, the value associated with that option button is stored in the field.

- Insert a group of option buttons (automatically create the data source)
- Insert a group of option buttons (bind to an existing field)
- Tip

Insert an ink picture control

You can create an ink picture control in your form in which people using Tablet PCs can draw images. Ink picture controls are bound to fields.

▸ Insert an ink picture control (automatically create the data source)

▸ Insert an ink picture control (bind to an existing field)

Security Ink Picture controls are not a secure method of collecting and storing signatures. If you want to collect and store secure signatures in your form, it is recommended that you use digital signatures.

▸ Tip

Add a label to a control

▸ Add a label to a single control

▸ Add and align a group of labels and controls

Note When you create a control by inserting a field or group into the form from the **Data Source** task pane, the label for the control is created automatically. If you do not want this to happen, click **Options** on the **Tools** menu, click the **Design** tab, and clear the **Create labels for controls automatically when adding from data source** check box.

Add placeholder text to a control

You can add placeholder text to controls in your form that instruct users on how to fill out your form. An example of placeholder text might be "Click here and type."

Note This option is only available for text box, rich text box, and date picker controls.

1. In design mode, right-click the control for which you want to add placeholder text, and then click ***Control*** **Properties** on the shortcut menu.
2. In the ***Control*** **Properties** dialog box, click the **Display** tab.
3. In the **Placeholder** box, type the text you want to display in the control.

Change a control to another control

In design mode, right-click the control you want to change, point to **Change To** on the shortcut menu, and click the type of control you want to change it to.

Notes

If you do not see the choice you want on the shortcut menu, check the properties of the field or group bound to the control. The properties of a field or group determine which type of control can be bound to it.

You cannot change a button, hyperlink control, expression box, or linked picture control to a different control.

Change text alignment in a control

When designing a form, you can specify the alignment of text users type into a selected control.

Note This option is only available for text box, rich text box, date picker, and expression box controls.

1. In design mode, right-click the control for which you want to set text alignment, and then click ***Control*** **Properties** on the shortcut menu.
2. In the ***Control*** **Properties** dialog box, click the **Display** tab.
3. In the **Alignment** box, select the text alignment option you want.

Change the padding in a control

After inserting a control in your form, you can change the amount of space, or "padding," between a control's border and its contents. For example, if you want to add extra white space at the bottom of a text box, you can change the bottom margin setting for the text box.

▸ Tip

1. In design mode, right-click the control whose padding you want to change, and then click ***Control*** **Properties** on the shortcut menu.

2. In the ***Control*** **Properties** dialog box, click the **Size** tab.

3. Under **Padding** in the **Top**, **Bottom**, **Left**, and **Right** boxes, type the amount of space that you want to appear between the control's border and the text inside it, and then select a unit of measure from the list for each amount.

▸ Tip

Change the margins for a control

After inserting a control in your form, you can specify the amount of blank space that appears between that control and the text, tables, and other controls that surround it. For example, to add extra space between label text and a text box, you can increase the margin setting for the text box.

1. In design mode, right-click the control whose margins you want to change, and then click ***Control* Properties** on the shortcut menu.
2. In the ***Control* Properties** dialog box, click the **Size** tab.
3. Under **Margins** in the **Top**, **Bottom**, **Left**, and **Right** boxes, type the amount of space that you want between the control and the surrounding text, tables, or other controls, and then select a unit of measurement from the list for each amount.

▸ Tip

Change the size of a control

1. In design mode, right-click the control you want to format, and then click ***Control*** **Properties** on the shortcut menu.

2. In the ***Control*** **Properties** dialog box, click the **Size** tab.

3. Under **Size**, do one of the following:

- To specify the control's height, type the height in the **Height** box, and then select a unit of measure from the list.

- To specify the control's width, type the width in the **Width** box, and then select a unit of measure from the list.

Notes

To have the control height adjust automatically, type **auto** in the **Height** box.

The minimum height of list box and drop-down list box controls depends on the number of list entries in the form.

Change the default state of a check box

1. In design mode, right-click the check box, and then click **Check Box Properties**.
2. On the **Data** tab in the **Check Box Properties** dialog box, do one of the following:

- To make the default state cleared, click **Cleared**.
- To make the default state checked, click **Checked**.

Note The default state of the check box will determine the value that is stored in the data source; if the default state is checked, for example, the value in the data source will match the value selected in the **Value when checked** list.

Change the binding of a control

In design mode, right-click the control whose binding you want to change, click **Change Binding** on the shortcut menu, and then do one of the following:

To create a new field or group to bind the control to, right-click the group to which you want to add the new field or group, click **Add** on the shortcut menu, enter the appropriate properties in the **Add Field or Group** dialog box, and then click **OK** twice.

To bind the control to an existing control, click the field or group you want to bind the control to, and then click **OK**.

Note Because data validation is based on fields and groups, not controls, changing the field or group to which a control is bound also changes any existing data validation for that control.

▸ Tip

Copy and paste a control

By using the **Copy** and **Paste** commands, you can create a new control that matches the formatting of an existing control. However, you must change the binding of the new control so it does not store its data in the same field or group as the original control.

1. In design mode, select the control you want to copy.

   Tip

2. On the **Edit** menu, click **Copy**.
3. Place the insertion point where you want to insert the new control, and then click **Paste** on the **Edit** menu.
4. Right-click the new control, click **Change Binding** on the shortcut menu, and then do one of the following:

   - To create a new field or group to bind the control to, right-click the group to which you want to add the field or group, click **Add** on the shortcut menu, enter the appropriate properties in the **Add Field or Group** dialog box, and then click **OK** twice.
   - To bind the control to an existing field or group, click the field or group you want to bind the control to, and then click **OK**.

Notes

When you copy and paste a control with conditional formatting, the original conditions will be copied to the new control.

Because data validation is based on fields and groups, not controls, changing the field or group to which a control is bound also changes any existing data validation for that control.

Tip

Delete a control

In design mode, select the control, and then press DELETE.

Note When a control is deleted from a form template, it is also deleted from all forms based on that form template that have already been filled out. Deleting the control will prevent users from seeing or working with the data that was displayed in the control. Deleting a control does not delete the field or group that the control was bound to.

Enable or disable character formatting in a control

Enabling character formatting for the selected control lets users format the information they enter when filling out a form. For example, enabling character formatting lets users apply a color to text or make text bold.

Note This option is only available for rich text box, bulleted list, numbered list, and plain list controls. In order to enable character formatting for list controls, the data type these controls are bound to must be rich text XHTML.

1. In design mode, right-click the control for which you want to enable or disable character formatting, and then click ***Control* Properties** on the shortcut menu.
2. In the ***Control* Properties** dialog box, click the **Display** tab.
3. To enable or disable character formatting for the selected control, select or clear the **Character formatting** check box.

Enable or disable full rich text formatting in a control

You can enable full, rich text formatting in rich text box controls. This lets users entering information into the control add more complex formatting, such as tables and images.

1. In design mode, right-click the rich text box control for which you want to enable or disable full rich text formatting, and then click **Rich Text Box Properties** on the shortcut menu.
2. In the **Rich Text Box Properties** dialog box, click the **Display** tab.
3. To enable or disable full rich text formatting for the selected control, select or clear the **Full Rich Text (images, tables, etc.)** check box.

Enable or disable paragraph breaks in a control

Enabling paragraph breaks for a rich text box control lets users create new paragraphs of text when entering information into the control.

1. In design mode, right-click the rich text box control for which you want to enable or disable paragraph breaks, and then click **Rich Text Box Properties** on the shortcut menu.
2. In the **Rich Text Box Properties** dialog box, click the **Display** tab.
3. To enable or disable paragraph breaks, select or clear the **Paragraph breaks** check box.

Enable or disable the spelling checker for a control

You can specify whether Microsoft Office InfoPath 2003 checks the spelling of the data that users enter into a text box, rich text box, or list. By default, the spelling checker is enabled.

1. In design mode, right-click the control for which you want to enable or disable the spelling checker, and then click ***Control* Properties**.
2. On the **Display** tab in the ***Control* Properties** dialog box, do one of the following:

- To enable the spelling checker for the control, select the **Enable spelling checker** check box.
- To disable the spelling checker for the control, clear the **Enable spelling checker** check box.

Enable or disable text wrapping for a control

You can enable text wrapping for a number of controls in Microsoft Office InfoPath 2003, including text boxes, rich text boxes, and expression boxes. When this option is enabled, text will automatically wrap inside controls when users type text into them. Without this option enabled, any text that exceeds the width of the control will be hidden. You can also choose a text wrapping style for picture controls.

▸ Enable or disable text wrapping for a text box, rich text box, or expression box control

▸ Enable or disable text wrapping for a picture control

Manage text scrolling in a control

If paragraph breaks are enabled for a control, or if you specify that the text should wrap inside the control, you can specify the scrolling options for that control. For example, you can make scroll bars appear on a text box when the user types more text than the text box is able to display by default.

Note Scrolling options are available for text box, rich text box, and expression box controls. By default, a text box scrolls automatically as the user types, and a rich text box and an expression box expand to show all of the text.

1. In design mode, right-click the control for which you want to set a text-scrolling option, and then click ***Control*** **Properties** on the shortcut menu.
2. In the ***Control*** **Properties** dialog box, click the **Display** tab.
3. In the **Scrolling** list, click the text-scrolling option that you want.

▸ Tip

Move a control

1. In design mode, move the control to its new location by dragging or by using the **Cut** and **Paste** commands on the **Edit** menu.
2. If you move a control into or out of a repeating section, optional section, or repeating table, make sure the data source for the control and the section or table is correctly structured. In particular, verify the following:

- If the control is moved into a repeating section, optional section, or repeating table, the control's field must be part of the section's or table's group.
- If the control is moved out of a repeating section, optional section, or repeating table, the control's field should not be a part of the section's or table's group.

▸ Tip

Rearrange columns in a repeating table

1. In design mode, double-click the **Repeating Table** label below the table.
2. In the **Repeating Table Properties** dialog box, on the **Data** tab, click **Change Binding.**
3. On the first page of the Repeating Table Binding Wizard, click **Next**.
4. In the **Columns in table** list on the second page of the Repeating Table Binding Wizard, select the column you want to move, and then do one of the following:

- To move a column to the left, click **Move Up**.
- To move a column to the right, click **Move Down**.

5. Repeat step 4 for each column you want to move, and then click **Finish**.

Set a control as read-only

Marking a selected control as read-only restricts users from entering information into it.

Note This option is only available for text box, bulleted list, numbered list, and plain list controls.

1. In design mode, right-click the control that you want to mark as read-only, and then click ***Control*** **Properties** on the shortcut menu.

2. In the ***Control*** **Properties** dialog box, click the **Display** tab, and then select or clear the **Read-only** check box.

Customize menu and toolbar commands for a repeating section

You can customize the commands your users use to add and remove a repeating section. To quickly update all the default menu commands associated with a repeating section, you can change the section name. If you want more control over specific commands, or if you do not want to change the section name, you can customize each command name separately.

Note Changing the section name modifies only those commands that currently contain the section name. For example, if you remove the section name from an individual command for inserting sections, that command will not be updated when you change the section name.

▸ Customize all the menu commands for a repeating section by changing the section name

▸ Add or customize a single menu or toolbar command name for a repeating section

Customize commands for an optional section

You can customize the commands your users use to add and remove an optional section. To quickly update all the default menu commands associated with an optional section, you can change the name of the group or field to which the optional section is bound. If you want more control over specific commands, or if you do not want to change the group or field name, you can customize each command name separately. You can also customize the instructional text that appears on the form, which users click to insert the section.

Note Changing the group or field name modifies only those commands that currently contain the group or field name. For example, if you remove the group name from an individual command for inserting a section, that command will not be updated when you change the group name.

▸ Customize all the menu commands for an optional section by changing the group or field name

▸ Add or customize a single menu or toolbar command name for an optional section

▸ Customize the instructional text for an optional section

Customize menu and toolbar commands for a repeating table

You can customize the commands your users use to add and remove rows in a repeating table. To quickly update all the default menu commands associated with a repeating table, you can change the name of the repeating group to which the table is bound. If you want more control over specific commands, or if you do not want to change the group or field name, you can customize each command name separately.

Note Changing the group or field name modifies only those commands that currently contain the group or field name. For example, if you remove the group name from an individual command for inserting rows, that command will not be updated when you change the group name.

▸ Customize all the menu commands for a repeating table by changing the group or field name

▸ Add or customize a single menu or toolbar command name for a repeating table

Set options for form submission

If your form template is set up so that users can submit their forms, you can customize the Submit command and the behavior of the form after submission.

▸ Customize the Submit command on the File menu

▸ Specify the form behavior after submit

▸ Customize the messages that appear when submit is successful and when submit fails

Add data validation

Data validation can be used to display error alerts when users enter incorrect values into a control. You can also write script to perform advanced validation. Note that Microsoft Office InfoPath 2003 verifies a condition by checking the value of the field that is bound to the control, not the control itself.

- Require a specific type of data
- Require a specific value or range of values
- Add script-based validation

Make a control required

If you want to require users to enter a value into a control, you must make the control's field required. Depending on how you want to display error alerts and the flexibility of your form template, you can make the field required by displaying either a standard error message or a custom error message.

Note If your form template prevents you from selecting the **Cannot be blank** check box in a control's **Properties** dialog box, you will need to display a custom error message.

▸ Display a standard error message when a control is left blank

▸ Display a custom error message when a control is left blank

Remove data validation

To prevent unwanted data validation alerts from appearing when users fill out a form, you can remove data validation conditions, change the data type, or change the control from required to optional.

- Remove data validation conditions
- Change the data type
- Change a control from required to optional

Note If a form template is based on a pre-existing XML Schema or a sample form, the schema might contain validation conditions that you cannot modify using Office InfoPath 2003.

About conditional formatting

Conditional formatting in Microsoft Office InfoPath 2003 allows you to change the appearance of a control based on values users enter into your form. For each control, you can set conditions that control formatting such as font style, text color, and background color. You can also hide a control entirely. Controls can have multiple conditions, which means that a control’s appearance can change in different ways based on different values. A control's appearance can also change based on the values entered into other controls on the form.

You can use conditions to determine whether the value of a control is blank, is within a specified range, equals the value of another control, or starts with or contains certain characters. Conditional formatting allows you to highlight important data and provide a more interactive experience for the person filling out the form. For example, conditional formatting can be set to:

Hide a control unless a specific check box is selected.

Change the font color to blue and the style to bold for all expenses that require a receipt.

Change row colors in a repeating table based on the value of a text box in the row.

Mark financial entries with red text when they are below $0 and green text when they are $0 or higher.

Note To display error alerts when users enter incorrect values into a form, you can also use data validation.

Add or change conditional formatting

You can use conditional formatting to change the appearance of a control based on values users enter into your form.

Note The following controls can have conditional formatting: text boxes, rich text boxes, sections, optional sections, repeating sections, repeating tables, and expression boxes.

▸ Add conditional formatting

▸ Change conditional formatting

Add conditional formatting based on another control

You can add conditional formatting to a control that is based on the value of another control on the form.

Note The following controls can have conditional formatting: text boxes, rich text boxes, sections, optional sections, repeating sections, repeating tables, and expression boxes.

1. In design mode, select the control to which you want to add conditional formatting, and then click **Conditional Formatting** on the **Format** menu.
2. In the **Conditional Formatting** dialog box, click **Add**.
3. In the first box under **If this condition is true**, click **Select a field or group**.
4. In the **Select a Field or Group** dialog box, select the field or group whose value you want to evaluate, and then click **OK**.

   Note Microsoft Office InfoPath 2003 verifies a condition by checking the value of the field that is bound to the control, not the control itself.
5. Enter the rest of the condition.
6. Select the formatting you want to apply when the condition is true.

Hide a control based on values in a form

You can use conditional formatting to hide a control based on the value of another control on the form. For example, a text box for a driver's license number can be hidden unless the user enters an appropriate age into another control on the form.

Note The following controls can have conditional formatting: text boxes, rich text boxes, sections, optional sections, repeating sections, repeating tables, and expression boxes.

1. In design mode, select the control you want to hide, and then click **Conditional Formatting** on the **Format** menu.
2. In the **Conditional Formatting** dialog box, click **Add**.
3. In the first box under **If this condition is true**, click **Select a field or group**.
4. In the **Select a Field or Group** dialog box, select the field whose value will determine whether the control is hidden, and then click **OK**.

   Note Microsoft Office InfoPath 2003 verifies a condition by checking the value of the field that is bound to the control, not the control itself.
5. Enter the rest of the condition.
6. Select the **Hide this control** check box.

Remove conditional formatting

To remove conditional formatting from a control, do the following:

1. In design mode, select the control you want, and then click **Conditional Formatting** on the **Format** menu.
2. In the **Conditional Formatting** dialog box, select the conditional format you want to remove, and then click **Remove**.

Add or remove a currency symbol

1. In design mode, right-click the text box or expression box whose data you want to format, and then click **Properties**.
2. On the **Data** tab or the **General** tab in the **Properties** dialog box, click **Format**.

   Note You can add currency symbols only for text boxes that have a decimal or whole number data type or for expression boxes that are formatted as a decimal or whole number.

3. In the **Format** dialog box, do one of the following:

- To add a currency symbol, click **Currency**, and then, in the **Currency** list, select the type of currency you want to display.
- To remove a currency symbol, click **Number**.

Note Adding or removing a currency symbol affects only the displayed value, not the value that is saved in the data source.

Add or remove a thousands separator in a number

1. In design mode, right-click the text box or expression box whose data you want to format, and then click **Properties**.

2. On the **Data** tab or the **General** tab in the **Properties** dialog box, click **Format**.

   Note You can add a thousands separator only for text boxes that have a decimal or whole number data type or for expression boxes that are formatted as a decimal or whole number.

3. In the **Format** dialog box, do one of the following:

- To add a thousands separator to the number, select the **Use a digit grouping symbol** check box.

- To remove the thousands separator from the number, clear the **Use a digit grouping symbol** check box.

Notes

Depending on locale, the separator might separate more or less than thousands.

Adding or removing a thousands separator affects only the displayed value, not the value that is saved in the data source.

Add or remove percent formatting for a number

To display a number as a percentage, you can format the text box or expression box so that numbers display their values as percentages. For example, a value of 0.2 would be displayed as 20. This formatting works both for display and data entry— if a user types **30** into a text box that has a percent data format, the saved value would be 0.30.

1. In design mode, right-click the text box or expression box whose data you want to format, and then click **Properties**.
2. On the **Data** tab or the **General** tab in the **Properties** dialog box, click **Format**.

   Note You can add percent formatting only for text boxes that have a decimal data type or for expression boxes that are formatted as a decimal.

3. In the **Decimal Format** dialog box, do one of the following:

- To format the number as a percentage, click **Percentage**.
- To remove percent formatting, click **Number**.

Note Microsoft Office InfoPath 2003 does not add percent symbols to numbers that are formatted as percentages. To add a percent symbol to a control's label, click where you want the percent symbol to appear and type **%.**

Change how a date or time is displayed

1. In design mode, right-click the date picker, text box, or expression box whose data you want to format, and then click **Properties**.

2. On the **Data** tab or the **General** tab in the **Properties** dialog box, click **Format**.

   Note You can specify how dates and times are displayed only for date pickers and text boxes that have a date, time, or date and time data type, or for expression boxes that are formatted as one of those data types.

3. In the **Format** dialog box, do one or both of the following, depending on what you have chosen to format:

- To format the date, click the display style you want in the **Display the date like this** list.

- To format the time, click the display style you want in the **Display the time like this** list.

  Note Display styles that have an asterisk will update based on the user's system settings.

4. To change the locale date and time settings, click the region you want in the **Locale** list.

Note Changing how a date or time is displayed affects only the displayed value, not the value that is saved in the data source.

Change how negative numbers are displayed

1. In design mode, right-click the text box or expression box whose data you want to format, and then click **Properties**.

2. On the **Data** tab or the **General** tab in the **Properties** dialog box, click **Format**.

   Note You can specify how negative numbers are displayed only for text boxes that have a decimal or whole number data type or for expression boxes that are formatted as a decimal or whole number.

3. In the **Format** dialog box, click the display style you want in the **Display negative numbers like this** list.

Specify the number of decimal places displayed

1. In design mode, right-click the text box or expression box whose data you want to format, and then click **Properties**.
2. On the **Data** tab or the **General** tab in the **Properties** dialog box, click **Format**.

   Note You can specify the number of decimal places only for text boxes that have a decimal data type or for expression boxes that are formatted as a decimal.

3. In the **Decimal places** list in the **Decimal Format** dialog box, do one of the following:

- To always display a fixed number of decimal places, click the number of decimal places you want.
- To show the number of decimal places that the user types into the control, click **Auto**.

Note Changing the number of decimal places affects only the displayed value, not the value that is saved in the data source.

Add, subtract, multiply, and divide values

You can use expression boxes to add, subtract, multiply, and divide values, as well as perform other calculations. When you create formulas in an expression box, you refer to the fields and groups to which controls on the form are bound, rather than the controls themselves. The formulas are created using XPath, but you can follow these steps without knowing XPath.

1. In design mode, place the insertion point where you want to insert the expression box.
2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
3. In the **Controls** task pane, click **Expression Box**.
4. To add, subtract, multiply, or divide values in two fields, enter an expression such as one of the following in the **XPath** box in the **Insert Expression Box** dialog box.

Expression type	Expression
Add two fields	my:field1 + my:field2
Subtract a field from another field	my:field1 - my:field2
Multiply two fields	my:field1 * my:field2
Divide a field by another field	my:field1 div my:field2

   Note In the preceding table, *my:field1* and *my:field2* are only examples. When you type your expression, you must use an XPath expression that identifies your specific fields.

   ▸ How?

5. To test the formula, click **Preview Form** on the **Standard** toolbar.

Note If an empty field is included in a formula that adds, subtracts, divides, or multiplies, the expression box will show the total as **NaN** or **1.#QNAN** (Not a Number). To avoid this, you can create a default value for the fields used in the equation.

Calculate the average of a column of numbers

You can use expression boxes to calculate the average of a column of numbers and perform other calculations. When you create formulas in an expression box, you refer to the fields and groups to which controls on the form are bound, rather than the controls themselves. The formulas are created using XPath, but you can follow these steps without knowing XPath.

1. In design mode, place the insertion point where you want to insert the expression box.
2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
3. In the **Controls** task pane, click **Expression Box**.
4. Enter an expression such as the following in the **XPath** box in the **Insert Expression Box** dialog box:

   (sum(my:group1/my:group2/my:field3)) div (count(my:group1/my:group2/my:field3))

   Note In this expression, *my:group1/my:group2/my:field3* is only an example. When you type your average expression, you must replace the example with an XPath expression that identifies your specific field.

   - How?
5. To test the formula, click **Preview Form** on the **Standard** toolbar.

Note If an empty field is included in an average formula, the expression box will show the total as **NaN** or **1. #QNAN** (Not a Number). To avoid this, you can create a default value for the fields used in the equation.

Count the instances of a field

You can use expression boxes to count the instances of a field and perform other calculations. When you create formulas in an expression box, you refer to the fields and groups to which controls on the form are bound, rather than the controls themselves. The formulas are created using XPath, but you can follow these steps without knowing XPath.

1. In design mode, place the insertion point where you want to insert the expression box.
2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
3. In the **Controls** task pane, click **Expression Box**.
4. Enter an expression such as the following in the **XPath** box in the **Insert Expression Box** dialog box:

   count(my:group1/my:group2/my:field3)

   Note In this expression, *my:group1/my:group2/my:field3* is only an example. When you type your count expression, you must replace the example with an XPath expression that identifies your specific field.

   ▸ How?
5. To test the formula, click **Preview Form** on the **Standard** toolbar.

Display the contents of two fields in one control

You can use expression boxes to display the contents of two fields in one control (or concatenate two fields), as well as perform other calculations. When you create formulas in an expression box, you refer to the fields and groups to which controls on the form are bound, rather than the controls themselves. The formulas are created using XPath, but you can follow these steps without knowing XPath.

1. In design mode, place the insertion point where you want to insert the expression box.
2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
3. In the **Controls** task pane, click **Expression Box**.
4. Enter an expression such as one of the following in the **XPath** box in the **Insert Expression Box** dialog box:

Expression type	Expression
Display two fields separated by a space	concat(my:field1, " ", my:field2)
Display two fields separated by a comma	concat(my:field1, ",", my:field2)
Display two fields run together	concat(my:field1, my:field2)

   In the preceding table, *my:field1* and *my:field2* are only examples. When you type your concatenation expression, you must replace the examples with XPath expressions that identify your specific fields. In addition, you can change the text that separates the two fields.

   ▸ How?

5. To test the formula, click **Preview Form** on the **Standard** toolbar.

Display the row number in a repeating table

You can use expression boxes to display the row number in a repeating table, as well as perform other calculations. When you create formulas in an expression box, you refer to the fields and groups to which controls on the form are bound, rather than the controls themselves. The formulas are created using XPath, but you can follow these steps without knowing XPath.

1. In design mode, place the insertion point where you want to insert the expression box.

   Note The expression box must be part of the repeating table.

2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
3. In the **Controls** task pane, click **Expression Box**.
4. In the **Insert Expression Box** dialog box, type **position()** in the **XPath** box.
5. To test the formula, click **Preview Form** on the **Standard** toolbar.

Sum a column of numbers

You can use expression boxes to sum a column of numbers and perform other calculations. When you create formulas in an expression box, you refer to the fields and groups to which controls on the form are bound, rather than the controls themselves. The formulas are created using XPath, but you can follow these steps without knowing XPath.

1. In design mode, place the insertion point where you want to insert the expression box.
2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
3. In the **Controls** task pane, click **Expression Box**.
4. Enter an expression such as the following in the **XPath** box in the **Insert Expression Box** dialog box:

   sum(my:group1/my:group2/my:field3)

   Note In this expression, *my:group1/my:group2/my:field3* is only an example. When you type your sum expression, you must replace the example with an XPath expression that identifies your specific field.

   ▸ How?
5. To test the formula, click **Preview Form** on the **Standard** toolbar.

Note If an empty field is included in a sum formula, the expression box will show the total as **NaN** or **1. #QNAN** (Not a Number). To avoid this, you can create a default value for the fields used in the equation, or you can filter out blank fields by adding the following text to the end of the field name in the XPath expression: **[.!=""]**. Example: **sum(my:group1/my:group2/my:field1[.!=""])**

Enable form submission

Microsoft Office InfoPath 2003 form submission can be enabled in several ways: forms can be submitted directly to a database (Microsoft Access or Microsoft SQL Server only) or Web service, through HTTP, or by using custom script. This topic covers submitting through HTTP and submitting using custom script. To learn about submitting to a database or Web service, see the topics about designing a form connected to a database and designing a form connected to a Web service.

▸ Enable submit through HTTP

▸ Enable submit using custom script

Enable a custom task pane

A custom task pane is an .html file whose content is displayed in the Microsoft Office InfoPath 2003 task pane when a user fills out a form. Custom task panes can provide form-specific commands and Help content. Before enabling a custom task pane, you must first create an .html file by using an HTML editor such as Microsoft FrontPage. You can add a custom task pane file directly to your form template, so users can work with the form and task pane while offline, or you can link the custom task pane to the form from another location.

▸ Enable a custom task pane that is part of the form template

▸ Enable a custom task pane that is linked to the form template

Enable or disable form merging for a form template

You can set up your form template so that your users can merge the data from multiple forms into one form. Usually, the data that is merged must be saved in a repeating field or in a field that is part of a repeating group; these repeating fields and groups are bound to list controls, repeating tables, or repeating sections. Data saved in a rich text box will be merged into one rich text box.

1. In design mode, click **Form Options** on the **Tools** menu.
2. On the **General** tab in the **Form Options** dialog box, do one of the following:

- If you want to enable merging, select the **Enable form merging** check box.
- If you want to disable merging, clear the **Enable form merging** check box.

Add a resource file to a form template

Resource files can include the HTML pages that make up a custom task pane or the XML document that provides the list items for a drop-down list box. To make sure that all your users have access to resource files, you can add them to your form.

1. In design mode, click **Resource Manager** on the **Tools** menu.
2. In the **Resource Manager** dialog box, click **Add**.
3. In the **Add File** dialog box, browse to and select the file you want to add.

About previewing and testing forms

When designing a form, you can use the Preview Form feature to test the functionality and appearance of your form before you make it available for users to fill out. Previewing and testing your form allows you to see and work with the form from your user's perspective.

Note Unlike the Print Preview feature, which only provides a static image of the form, Preview Form is interactive and provides a simple way for you to test your form without leaving design mode.

When you click **Preview Form** on the **Standard** toolbar, your form opens in a new window. In this new window you can test your form by entering data into controls to check various features, such as text formatting, conditional formatting, and data validation. This allows you to identify mistakes in the previewing window, and then quickly switch to the designing window where you can correct them.

Note To help you identify which window you are in when you are previewing a form, **Preview** will appear in the window's title bar instead of **Design**, which appears when you are in design mode.

In addition to previewing a form, you can also use sample data to quickly test the appearance of your form. Sample data is placeholder text that appears in controls on your form, and it provides an example of how text will appear in the control when a user fills it out. Sample data can only be seen in design mode and is not visible when you preview or fill out a form.

Preview a form

When designing a form, you can preview it to test its appearance and functionality and ensure that it will work as you want it to when users fill it out.

Note Before you publish your form template and make it available for others to use, it is recommended that you preview and test your form's appearance and functionality.

On the **Standard** toolbar, click **Preview Form**. The form will open in a new window.

Show or hide sample data

When designing a form, you can use sample data to quickly check the text formatting of controls. Sample data illustrates how text will appear on your form when users fill it out. Sample data is not visible when you preview a form or when a user fills it out.

In design mode, click **Sample Data** on the **View** menu.

Publish a form template

To make your form template available for others to use, you must publish it by using the Microsoft Office InfoPath 2003 Publishing Wizard.

1. On the **File** menu, click **Publish**.
2. Follow the instructions in the wizard.

▸ Tip

Save a form template's form files to a folder

When designing a form, you can view and work with a form template's form files by saving them to a folder. The Extract Form Files feature saves the form files to a folder, which allows you to manually make changes to individual files. For example, by extracting form files, you can manually make changes to the form definition (.xsf) file.

1. In design mode, open the form template (.xsn file) you want to save to a project folder.
2. On the **File** menu, click **Extract Form Files**.
3. In the **Browse For Folder** dialog box, select a location in which to save the folder.

About modifying a published form template

After you have published your form template, you may need to make changes to it. For example, you may need to add more controls, change the data validation rules, or change the text that appears on the form. However, before modifying a form template, you need to be aware of how your changes will affect existing forms.

When you modify a form template in design mode, the Microsoft Office InfoPath 2003 **Automatically upgrade existing forms** option, which is enabled by default, helps to ensure that changes you make to a published form template don't prevent users from opening and working with any existing forms (.xml files) that are based on it. This feature allows users to open the existing forms even if the XML Schema of the form template has been changed. This is because InfoPath is able to schematically reconstruct the .xml file so that it is compatible with the new XML Schema of the modified form template.

Although users will still be able to open and work with the existing forms, some changes you make to the form template can cause data to be permanently removed from those forms. For example, if you delete a field from the data source of a form template, any data users have typed into that field will be deleted from all of the existing forms based on that form template. This is because the form template's schema now specifies that the field does not exist. The following list contains the changes that can cause similar deletions of data from existing forms:

Renaming, moving, or deleting a field or group from the data source

Changing a repeating field or repeating group to a non-repeating field or group

Changing the data type of a rich text field

Manually changing the version number of the form template

▸ Tip

Notes

If the **Automatically upgrade existing forms** option is not enabled when you modify a form template, existing forms based on that form template will not be automatically upgraded to be compatible with the changes, and they may not open or work properly.

Renaming or moving a form template that has existing forms based on it can prevent those forms from opening or working properly because they will not reference the new name or location of the form template.

About upgrading forms

When you modify a form template in design mode, the existing forms (.xml files) that are based on it may need to be upgraded so that they are compatible with the modified form template. This ensures users can continue to work with the existing forms. If forms are not upgraded, they may not open or function properly, and data from existing forms could be lost. When upgrading forms using Microsoft Office InfoPath 2003, you can either automatically upgrade existing forms or create a custom version upgrade using script.

Choosing to automatically upgrade existing forms when you modify their form template helps to ensure that changes you make to a published form template don't prevent users from opening and working with existing forms (.xml files) that are based on it. Choosing to automatically upgrade existing forms also allows users to open the existing forms, even if the form template's XML Schema has been changed. This is because InfoPath is able to schematically reconstruct the .xml file, so that it is compatible against the modified form template's new XML Scheme.

Choosing to create a custom version upgrade of your forms using script can be used when you need to handle more complicated upgrade tasks automatically. When you create a custom version upgrade using script, you can immediately edit the script file in Microsoft Script Editor (MSE), and then update the form template with your custom code for handling version upgrades.

Modify a form library's form template

If you need to make changes or updates to an existing Microsoft Windows SharePoint Services form library's form template, you can modify it directly from the form library.

Note Some changes made to a form template can result in data loss from existing forms that are based on that form template. For example, if you remove a control in a form template that collected specific data, the control will no longer appear in the existing forms that are based on that form template, and the data will be lost. To prevent compatibility issues with existing forms, republish the new form template to a different location.

▸ Tip

1. Navigate to the site where the form library is located.
2. Under **Actions** on the page that displays the form library, click **Modify settings and columns**.
3. In the **General Settings** section on the **Customize** page, to the right of the **Template** location, click **Edit Template**. The form template will open in the Microsoft Office InfoPath 2003 design mode.
4. Make the changes you want, and then click **Save** 💾 on the **Standard Toolbar**.

Specify a form upgrade method

When you modify a form template in design mode, the existing forms (.xml files) that are based on it may need to be upgraded so that they are compatible with the newer version of the form template. This ensures users can continue to work with the existing forms. Microsoft Office InfoPath 2003 allows you to select how the existing forms' versions will be upgraded.

1. In design mode, click **Form Options** on the **Tools** menu.
2. In the **Forms Options** dialog box, click the **Advanced** tab.
3. Under **Form template version** in the **On version upgrade** drop-down list, select the version upgrade method you want to use.

About form libraries

Form libraries provide a simple way for you to use and share your Microsoft Office InfoPath 2003 forms. Form libraries are located on Windows SharePoint Services sites and act as a file server, where users can fill out and store forms based on the same form template. One form template is associated with each form library, so that every form created from the library will have the same structure and appearance. For example, you could create a form library that provides a single location for your team members to quickly fill out, save, and view their status report forms.

Other advantages of using a form library to store your forms are the ability to use form library columns and merge forms. Form library columns are directly populated with data from the forms stored in the form library. This not only allows users to view important information without having to open and view each form, but also provides a simple way to sort and group forms according to their content. For example, if you stored your team's expense reports in a form library, the form library columns could display the employee's name, the business purpose, and the total expenses claimed. This would be displayed for each form in the form library. Since form library columns allow you to easily view and sort forms based on their content, you can easily create a summary report by merging several forms into a single form.

Create a form library

1. In design mode, open the form template you want to set as the default for all forms created in the form library.
2. On the **File** menu, click **Publish**.
3. On the second page of the Publishing Wizard, select **To a SharePoint form library**, and then click **Next**.
4. On the next page of the wizard, select **Create a new form library (recommended)**, and then click **Next**.
5. Follow the remaining steps in the Publishing Wizard.

Add, remove, or modify form library columns

When designing a form, you can add, remove, or modify form library columns to be displayed in the default view of a Windows SharePoint Services form library.

Note You can also add form library columns to your form when you publish it to a Windows SharePoint Services form library.

Do one of the following:

- ▸ Add form library columns using the Publish Wizard
- ▸ Add, remove, or modify form library columns for a form

How InfoPath uses XML technologies

Microsoft Office InfoPath 2003 makes use of a variety of XML technologies, including XSLT, XML Schemas, XHTML, XPath, the DOM, XML Signatures, and an XML processor that works with all of these technologies. These technologies are governed by the World Wide Web Consortium (W3C), which recommends a specific set of rules, guidelines, and conventions for producing files that are easy to generate and process by different computers and applications. In addition to the XML technologies that Office InfoPath 2003 supports, Office InfoPath 2003 also makes use of special processing instructions and namespaces and has a number of methods for accessing the DOM through the Office InfoPath 2003 object model.

Office InfoPath 2003 is a Microsoft Windows–based application that is used to create forms based on XML and its associated technologies. While it is not necessary to understand all of the details of these XML technologies, the following table will help you understand how they are used within Office InfoPath 2003.

Name	Description
XML	XML is the output format that is produced by an Office InfoPath 2003 form. XML is also used to contain the sample data that is displayed in the fields of a form before a user fills it out.
XSLT	XSLT is the format of the view files that are produced when a form is designed. Office InfoPath 2003 does not support the use of XSL Formatting Objects (XSL-FO) for the presentation of XML data, nor does it support importing of arbitrary .xsl files.
XML Schema	XML Schemas are used as the primary means of data validation within a form, and they provide the underlying structure of the XML document that is to be produced by the form. In addition, an XML Schema is used to define the structure of the form definition (.xsf) file that is generated by Office InfoPath 2003 when a form is designed. Office InfoPath 2003 does not support the use of XML-Data Reduced (XDR) or Document Type Definition (DTD) for defining the schema of

	XML documents.
XHTML	XHTML is well-formed HTML, and is primarily used when developing rich text areas.
XPath	XPath expressions are used for binding controls to a form, as well as for data validation and conditional formatting expressions.
DOM	The DOM is used to programmatically access the contents of the source XML document that a form produces. It is also used to provide read-only access to the contents of the .xsf file.
XML Signature	XML Signatures are used to digitally sign forms created by Office InfoPath 2003. Each Office InfoPath 2003 form can contain one or more digital signatures. Office InfoPath 2003 does not support digital signing of portions of a form, only the form as a whole.
XML processor	The XML processor is used to load the source XML of a document into memory, validate it using the XML Schema, and produce the view of the document using XSLT. The XML processor supported by Office InfoPath 2003 is Microsoft XML Core Services (MSXML) 5.0. Earlier versions of this processor are not supported.

About XML

Extensible Markup Language, or XML, is a markup language defined by the XML Working Group of the World Wide Web Consortium (W3C). XML is similar to Hypertext Markup Language (HTML) in that it is a tag-based language specifically designed for delivering information on the Web. However, XML is different from HTML in that the tags that it uses are not predefined. Instead, the W3C XML recommendation specifies a set of rules that must be followed so that you can create your own meaningful set of tags.

- XML rules
- Structure of XML documents

About XSLT

Extensible Stylesheet Language Transformation (XSLT) is a declarative, XML-based language that is used to present or transform XML data. The transformation of XML data involves taking the source document, considered to be a tree of nodes, and creating a second XML document from it in another format. The XSLT style sheet provides the rules and format that specify how the output document is built. If the output is in an appropriate format such as HTML, it can be used for presentation of the source document.

XSLT matches components (or nodes) of an XML document— by using XPath expressions— with XSLT templates, and then applies transformation and formatting rules to the matched components. XSLT templates are structures that specify the components of the XML document to be transformed, as well as the final format into which the matching components should be transformed. Since XSLT is implemented in XML, it defines a special set of elements and attributes that can be used to create the transformations. XSLT can be used to transform any type of XML document, including XML data, XML Schemas, or other XSLT style sheets.

From a programming perspective, XSLT supports a number of general methods for processing XML data:

Data types A small set of data types, including Boolean, number, string, and node-set.

Operations Various operations that can be performed on the XML data, including **template**, **apply-template**, **sort**, and **output**.

Flow control Programming statements used to control the flow of the transformation, including **if**, **for-each**, and **choose**.

About XML Schemas

An XML Schema is an XML file with an .xsd file name extension. An XML Schema is used to define the structure and type of data that an XML document can contain. It specifies the elements, attributes, and data types that can be used in an XML document, along with the structure that must be followed for the XML document to be valid for that particular XML Schema.

Schema files have a root element named **schema**, and all of the element, attribute, and data type definitions are nested within it.

The XML Schema can also be used to validate the data contained in the elements and attributes. The following is a list of some of the types of data validation that can be performed by an XML Schema:

Data types Control the types of data that an element or attribute can contain. Examples include string, Boolean, float, integer, date, and time.

Constraining facets Place limits on the value space of data types. Examples include length, pattern, enumeration, minimum and maximum ranges, and precision.

Cardinality Controls the number of permitted occurrences. Examples include one, zero or one, zero or more, and one or more.

Choice Limits values to those contained in a list of values.

Sequence Defines the order in which elements may be used.

Defaults Provide values that are used when no other value is specified.

The namespace for XML Schemas is:
xmlns:xsd="http://www.w3.org/2001/XMLSchema"

About XPath

XML Path Language (XPath) is a query language that is used to search for and retrieve information contained within the nodes of an XML document, which in Microsoft Office InfoPath 2003 equate to fields or controls on a form. XPath queries are formed as expressions. These expressions are used to address the various parts of an XML document; to manipulate strings, numbers, and Boolean values; and to match a set of nodes within the document.

XPath views an XML document as a tree of different node types. XPath expressions identify these nodes in the XML document based on their type, name, and values, as well as the relationship of a node to other nodes in the document. An XPath expression yields one of the following basic objects:

Node set   A collection of nodes that match the expression. Node types include root (document), element, attribute, namespace, processing instruction, comment, and text.

Boolean   A **true** or **false** value.

Number   A numerical value.

String   A text value.

XPath expressions are formed using patterns that specify the location path of a node or set of nodes, and that are relative to the context node. The context node is the section of the XML document from which you are currently positioned, and it is represented in XPath as a period (.). XPath also supports using patterns that have absolute locations based on the document root. The document root is represented in XPath as a forward slash (/).

In addition to location paths, XPath has a number of functions that can be used to retrieve data from an XML document. The following is a list of the types of functions that XPath supports:

Node   Used for working with the various node types. Examples of node functions include **name**, **node**, **processing-instruction**, **comment**, and **text**.

Positional Used for nodes that belong to a node set. These functions determine or match the position of the nodes. Examples of positional functions include **name**, **id**, **last**, and **count**.

Numeric Used to return numeric values. Examples of numeric functions include **ceiling**, **floor**, **number**, **round**, and **sum**.

Boolean Used to evaluate XPath expressions to **true** or **false**. Examples of Boolean functions include **boolean**, **false**, **not**, and **true**.

String Used to manipulate and parse string text. Examples of string functions include **concat**, **contains**, **starts-with**, **string**, **substring**, and **translate**.

About namespaces

XML namespaces are used to uniquely identify elements and attributes within an XML document to avoid naming conflicts. Namespaces are declared as a Uniform Resource Identifier (URI) and are typically located in the start tag of the root element of an XML document. However, they can also be defined at the node level, where they are valid for that node and all of its descendants. Associating a URI with a namespace ensures that elements with the same name remain distinct.

The World Wide Web Consortium (W3C) Namespaces in XML Recommendation provides the **xmlns** attribute to uniquely define a namespace for an XML document. The following is an example of a namespace that is used to define the vocabulary for Extensible Stylesheet Language Transformation (XSLT):

```
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
```

A namespace is divided into three sections:

xmlns keyword The keyword defined by the W3C for identifying XML namespaces. This section is defined first and is separated from the namespace prefix by a colon (:).

Prefix The abbreviated name of the namespace that is used in declaring all elements and attributes that belong to it. The prefix can be any name other than xmlns or xml, which are reserved words.

Definition The URI that uniquely defines the namespace. Can also be defined as a URN or a URL.

Once a namespace is declared for an XML document, any element or attribute that belongs to that namespace uses the namespace prefix in its declaration. For example, .xslt files use the xsl namespace prefix as shown in the following example:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XS
<xsl:output method="html" indent="no"/>
   <xsl:template match="myNode:myNodes">
      ...
   </xsl:template>
</xsl:stylesheet>
```

About processing instructions

Processing instructions (PIs) are used to provide information in an XML document that the XML parser passes on to the calling application. PIs are primarily used to tell an application how to handle the data contained within an XML document. PIs must begin with an identifier called a *target*, which follows rules similar to those for element and attribute names. Targets are case-sensitive and must start with a letter or underscore. The rest of the target can contain letters, numbers, hyphens, underscores, periods, and colons. Any valid XML textual characters can appear after the target. The following is an example of a processing instruction that is used to indicate that the XML document should be displayed using an XSL style sheet:

```
<?xml-stylesheet type="text/xsl" href="MyStylesheet.xsl"?>
```

PIs can be placed anywhere in an XML document outside of other markup. They can be placed in the prolog, following the root element, or within the value of an element. They are usually placed in an XML document's prolog.

About the XML DOM

The XML Document Object Model (DOM) is used to access and update the content, style, and structure of XML documents through a programmatic interface. This interface takes the form of an object model made up of objects, properties, methods, and events that can represent and manipulate the components of an XML document. The DOM stores the data of an XML document in a hierarchical, tree-like structure that mimics the structure of the document itself.

The DOM can be used to access any component of an XML document, including its elements, attributes, processing instructions, comments, and entity declarations. Any XML document can be loaded into the DOM. When an XML document is loaded into the DOM, it is read from start to finish and stored in the DOM as a logical model of nodes.

The DOM programming interface enables applications to traverse the tree and manipulate its nodes. Each node is defined as a specific node type according to the XML DOM enumerated constants, which also define valid parent and child nodes for each node type. For most XML documents, the most common node types are element, attribute, and text. Attributes occupy a special place in the object model because they are not considered child nodes of a parent; instead, they are treated as properties of elements.

The following table lists some of the DOM objects that can be used to work with an XML document, along with the XML node types that they represent.

DOM object	XML node type
XMLDOMDocument	Represents the XML document as a whole. This object exposes properties and methods that allow you to navigate, query, and modify the content and structure of an XML document.
XMLDOMNode	Represents a single node in the document tree. This object is the base object for accessing data in the XML DOM, and it includes support for data types, namespaces, and XML Schemas.
XMLDOMNodeList	Represents a collection of nodes. This object

	enables iteration and indexed access operations on the live collection of IXMLDOMNode.
XMLDOMElement	Represents an element of the XML document.
XMLDOMAttribute	Represents an attribute of the XML document.

About XML processors

XML processors are software programs that are used to parse XML documents and provide access to their content and structure. XML processors can also validate the structure of XML documents and transform them into a variety of formats.

XML documents are text files that contain data that is formatted as a series of elements and attributes. Elements within an XML document can also be thought of as nodes of the document. The root element of the document is the root node, and all of the other elements are child nodes (or descendants) of the root node. Attributes are additional data associated with an element, and are considered to be properties of the element. When an XML processor first loads an XML document, it parses the nodes of the document and places them in random access memory (RAM) in the form of a Document Object Model (DOM).

Once an XML document has been parsed and loaded into the DOM, you can use the programmatic interfaces of the DOM to access and manipulate the XML document that it contains. If an XSL Transformation (XSLT) is associated with an XML document, the XML processor can be used to apply it to the document to produce the specified output format.

About XML Signatures

XML Signatures are digital signatures designed for use in XML-based transactions. They are defined through a joint effort of the World Wide Web Consortium (W3C) and the Internet Engineering Task Force (IETF). XML Signatures contain a series of XML elements and attributes that define various components needed to securely sign an XML document, including information about the location of the XML document to be signed, the algorithm used for signing, and information about the key used for signing.

An important feature of XML Signatures is the ability to have multiple digital signatures contained within one XML document. This allows multiple users to work on the same XML document, with each user providing a new digital signature on top of signatures added by other users.

Adding a digital signature to an XML document helps ensure that the data contained in the document is more secure, and that the identity of the person who signed the XML document can be verified.

About InfoPath security

The Microsoft Office InfoPath 2003 security model is based on the security model implemented by Microsoft Internet Explorer. The Internet Explorer model involves protecting your computer from unsafe operations by using security zones and levels. InfoPath also allows for other form security measures, including protecting form design, using digital signatures, managing certain form operations such as form merging and submission, and trusting installed forms.

▸ Using Internet Explorer security zones and levels

▸ Using form security measures

Protect your form's design

You can protect your form's design so that users filling out the form can not open or modify its form template in design mode.

1. In design mode, click **Form Options** on the **Tools** menu.
2. On the **General** tab of the **Form Options** dialog box, under **Protection**, select the **Enable protection** check box.

Set security options for installed forms

You can specify whether or not you want to trust custom installed forms and allow them to have access to files and settings on your computer.

Security note Because trusting installed forms gives them greater access to files, settings, and resources on your computer, make sure you have received the forms from a reliable source before installing them.

1. On the **Tools** menu, click **Options**.
2. Under **Security options** in the **Options** dialog box, select or clear the **Allow forms that I install with a custom setup program to have access to files and settings on my computer** check box.

About security zones

Because the Microsoft Office InfoPath 2003 security model is based on the security model implemented by Microsoft Internet Explorer, you can use security zones to control the level of access that forms can have to the resources on your computer. Internet, local intranet, trusted sites, and restricted sites zones apply to forms whose form templates are located on a networked location, and the local machine zone applies to form templates installed on your local computer.

- Internet zone
- Local intranet zone
- Trusted sites zone
- Restricted sites zone
- Local machine zone

Set a security level for a zone

You can assign a security level to each security zone to help control the level of access that forms can have to the resources on your computer. The settings you specify for security levels here will also be the settings for security levels used by Microsoft Internet Explorer.

1. On the **Tools** menu, click **Options**.
2. Under **System options** in the **Options** dialog box, click **Internet Options**, and then click the **Security** tab.
3. Click the zone that you want to set the security level for.
4. Under **Security level for this zone**, do one of the following:

- To use the default security level for the zone, click **Default Level**.
- To use a custom security level, click **Custom Level**, and then in the **Security Settings** dialog box, select the security level you want in the **Reset to** box.
- To set additional, custom security settings, in the **Settings** list in the **Security Settings** dialog box, select the appropriate settings.

Note To set a security level for the Restricted sites security zone, click **Custom Level**, and then click the level you want in the **Reset to** list.

About digital signatures

Microsoft Office InfoPath 2003 uses XML Signatures to enable you to digitally sign a form using a digital certificate. The certificate used to create this signature confirms that the form originated from the signer, and the signature confirms that it has not been altered.

Obtaining digital certificates

You can obtain a digital certificate from a commercial certification authority, such as VeriSign, Inc., or from your internal security administrator or Information Technology (IT) professional. Or you can create a digital signature yourself using the Selfcert.exe tool.

InfoPath will use only certificates that have a private key and that have a **Digital Signature** value for the **Key Usage** attribute. In addition, the purpose of the certificate should be **Client Authentication**. These restrictions are a result of InfoPath's use of XML Signatures for digitally signing forms. Other types of certificates that are valid in other Microsoft Office applications may not be valid for digital signing in InfoPath.

Note Because a digital certificate you create yourself is not issued by a formal certification authority, forms signed by using such a certificate are referred to as self-signed forms. Certificates you create yourself are considered unauthenticated and will generate a security warning if the security level is set to High or Medium. InfoPath trusts self-signed certificates only on computers that have the private key for that certificate available (usually, this is only the computer that created the certificate, unless the private key is shared with other computers).

Commercial certification authorities

To obtain a digital certificate from a commercial certification authority, such as VeriSign, Inc., you or your organization must submit an application to that authority.

To learn more about certification authorities that offer services for Microsoft products, see the Microsoft TechNet Security Web site.

Depending on your status as a developer, you should apply for a Class 2 or Class 3 digital certificate for software publishers:

A Class 2 digital certificate is designed for people who publish software as individuals. This class of digital certificate provides assurance about the identity of the individual publisher.

A Class 3 digital certificate is designed for companies and other organizations that publish software. This class of digital certificate provides greater assurance about the identity of the publishing organization. Class 3 digital certificates are designed to represent the level of assurance provided today by retail channels for software. An applicant for a Class 3 digital certificate must also meet a minimum financial stability level based on ratings from Dun & Bradstreet Financial Services.

When you receive your digital certificate, you are given instructions on how to install it on the computer you use to sign your InfoPath forms.

Internal certification authorities

Some organizations and corporations might have a security administrator or group act as their own certification authority and produce or distribute digital certificates by using tools such as Microsoft Certificate Server. Microsoft Certificate Server can function as a stand-alone certification authority or as part of an existing certification authority hierarchy. Depending on how Microsoft Office digital-signature features are used in your organization, you might be able to sign forms by using a digital certificate from your organization's internal certification authority. Or you might need to have an administrator sign forms for you by using an approved certificate. For information about your organization's policy, contact your network administrator or IT department.

Add, remove, or review digital signatures

After filling out a form, you can add one or more digital signatures to it by using a digital certificate. Once a digital signature is added, the content of the form cannot be modified. You can also remove or review all digital signatures that users have added to a form.

Note You must have a digital certificate installed on your computer before you can add a digital signature. To obtain a digital certificate, see your system administrator, a certification authority (such as VeriSign, Inc.), or request one from your network's default certification authority.

- Add a digital signature
- Remove a digital signature
- Review digital signatures
- Tip

Enable or disable digital signatures

When designing a form, you can specify whether or not users are able to use digital signatures when they fill out the form. However, you cannot enable digital signatures on a form template that was created from a database, Web service, or an XML Schema that does not have a digital signature namespace.

Note Enabling digital signatures allows users to sign and certify the content of the form, not the underlying code.

- Enable digital signatures
- Disable digital signatures

Request a digital certificate

In order to digitally sign a form, you must have a digital certificate installed on your computer. You can request a digital certificate from the default certification authority on your company's intranet.

1. In Microsoft Windows XP or Microsoft Windows 2000, on the **Start** menu, click **Run**.
2. In the **Open** box in the **Run** dialog box, type **MMC**, and then click **Enter**.
3. On the **File** menu in the **Console** window, click **Add/Remove Snap-in**.
4. In the **Add/Remove Snap-in** dialog box, click **Add**.
5. In the **Add Standalone Snap-in** dialog box, select **Certificates**, and then click **Add**.
6. In the **Certificates snap-in** dialog box, select **My user account**, and then click **Finish**.

Note You can also obtain a digital certificate from a certification authority such as Verisign, Inc. or from your system administrator.

Adjust line breaks for East Asian text

The feature or some of the options described in this Help topic are only available if support for Japanese, Simplified Chinese, Traditional Chinese, or Korean is enabled through Microsoft Office Language Settings.

1. Place the insertion point in the text where you want to apply a line break.
2. On the **Format** menu, click **Asian Typography**.
3. Under **Line break** in the **Asian Typography** task pane, click **Normal** or **Strict**.
4. If you are designing a form and you want to apply a default line break setting to all new forms that you design, click **Options** in the **Asian Typography** task pane. Under **Asian typography line break** in the **Options** dialog box, click the line break option you want, and then click **Set as default for new forms**.

Note When filling out a form, you can adjust line breaks for East Asian text in rich text boxes, but you cannot apply a default line break setting for new forms.

Automatically adjust character spacing for mixed East Asian and Latin text

The feature or some of the options described in this Help topic are only available if support for Japanese, Simplified Chinese, Traditional Chinese, or Korean is enabled through Microsoft Office Language Settings.

When you type East Asian characters and numerals together, or East Asian text in the same paragraph with Latin alphabet characters or numerals, the spacing between certain characters or punctuation may become irregular and make the text difficult to read. In such cases, Microsoft Office InfoPath 2003 allows you to adjust the character spacing automatically.

1. On the **Format** menu, click **Asian Typography**.
2. In the **Asian Typography** task pane, do one of the following:

- To adjust the character spacing between East Asian and Latin alphabet characters, select the **Automatically adjust space between Asian and Latin text** check box.
- To adjust the character spacing between East Asian characters and numerals, select the **Automatically adjust space between Asian text and numbers** check box.

InfoPath Glossary

A B C D E F G H I J L M N
O P R S T U V W X

A

Back to top

absolute URL
A Universal Resource Locator (URL) that contains a full address, including the protocol, the Web server, and the path and file name.

active field
The field on a form in which the cursor is placed.

attribute field
A field in the data source that can contain data and that is an attribute (instead of an element). Attribute fields cannot contain other fields.

B

Back to top

bind
To connect a control to a field or group in the data source so that data entered into the control is saved. When a control is unbound, it is not connected to a field or group, so data entered into the control will not be saved.

C

Back to top

cascading style sheet (CSS)
Data formatting declarations that determine the appearance of specified elements in an .html or .xml file. CSS can be embedded in a Web page or .xsl file or stored in a separate .css file linked to a

Web page or .xsl file.

child element

In an Extensible Markup Language (XML) tree structure, the element that is contained by the parent element.

class

The definition for an object, including its name, its properties and methods, and any events that it has. When you create an instance of a class, you create a new object with all of the characteristics defined by that class.

closed content model

A set of Extensible Markup Language (XML) document rules that states that an element in an XML document cannot include any information that is not declared in the schema that the document references.

COM

See Component Object Model.

Component Object Model (COM)

A specification developed by Microsoft for building software components that can be assembled into programs or can add functionality to existing programs running on Microsoft Windows operating systems.

conditional formatting

Changing the appearance of a control, including its visibility, based on values entered into the form.

constant

A value that is not calculated and, therefore, does not change. For example, the number 210 and the text "Quarterly Earnings" are constants. An expression, or a value resulting from an expression, is not a constant.

control

A graphical user interface object, such as a text box, check box, scroll bar, or command button, that lets users control the program. You use controls to display data or choices, perform an action, or make the user interface easier to read.

CSS

See cascading style sheet.

CSS selector

The name of the element type to which the display information applies.

custom installed form

A form that is installed with a custom setup program and that usually has access to files and settings on the computer.

custom task pane

An .html file whose content is displayed in a window next to a form. Custom task panes can provide form-specific commands and Help content.

D

Back to top

---

data adapter

An object used to submit data to and retrieve data from databases, Web services, and Extensible Markup Language (XML) files.

data source

The collection of fields and groups that define and store the data for an InfoPath form. Controls on the form are bound to the fields and groups in the data source.

data type

Property of a field that defines the kinds of data the field can store. Data types include Text, Rich Text, Whole Number, Decimal, True/False, Hyperlink, Date, Time, Date and Time, and Picture.

data validation

The process of testing the accuracy of data; a set of rules you can apply to a control to specify the type and range of data that users can enter.

declarative programming

Rule-based programming in which each rule is independent and the order in which rules are declared or executed is not significant.

design mode

The InfoPath design environment in which you can create or modify a form template.

DHTML

See Dynamic HTML.

dialog box alert

A data validation alert that opens a dialog box with a custom error message when invalid data is entered into a control.

digital signature

An electronic, encryption-based, stamp of authentication on a macro or document. This signature confirms that the macro or document originated from the signer and has not been altered.

Document Object Model (DOM)

A World Wide Web Consortium (W3C) specification that describes the structure of dynamic Hypertext Markup Language (HTML) and Extensible Markup Language (XML) documents in a way that allows them to be manipulated through a Web browser.

Document Type Definition (DTD)

The predecessor of a schema definition file that specifies rules for Extensible Markup Language (XML) document elements, attributes, and containment relationships.

DOM

See Document Object Model.

DTD

See Document Type Definition.

Dynamic HTML (DHTML)

An extension of Hypertext Markup Language (HTML) that adds multimedia, database access, and an object model that programs can use to change styles and attributes of page elements (objects) and to replace existing elements (objects) with new ones.

E

Back to top

embedded object

Information (object) contained in a source file and inserted into a destination file. Once embedded, the object becomes part of the destination file. Changes you make to the embedded object are reflected in the destination file.

enumerated data type

A data type consisting of a sequence of named values given in a particular order.

event

An action recognized by an object, such as a mouse click or key press, for which you can define a response. An event can be caused by a user action or a Microsoft Visual Basic statement, or it can be triggered by the system.

event handler

The script function code in an InfoPath form that responds to a form event or data validation event.

expression box

A control on a form that displays read-only data as the result of a lookup or calculation against the data using an XPath expression.

Extensible Hypertext Markup Language (XHTML)

A markup language that extends Hypertext Markup Language (HTML) and reformulates it as Extensible Markup Language (XML).

Extensible Markup Language (XML)

A condensed form of Standard Generalized Markup Language (SGML) that enables developers to create customized tags that offer flexibility in organizing and presenting information.

Extensible Stylesheet Language (XSL)

An Extensible Markup Language (XML) vocabulary that is used to transform XML data to another form, such as Hypertext Markup Language (HTML), by means of a style sheet that defines presentation rules.

F

Back to top

field

An element or attribute in the data source that can contain data. If the field is an element, it can contain attribute fields. Fields store the data that is entered into controls.

form

In InfoPath, a document with a set of controls into which users can enter information. InfoPath forms can contain features such as rich text boxes, date pickers, optional and repeating sections, data validation, and conditional formatting.

form area

The area of the InfoPath workspace that displays the form you are working with.

form definition file

An Extensible Markup Language (XML) file with an .xsf extension that contains information about all other files and components used within a form, including user interface customizations, XML Schemas, views, business logic, events, and deployment settings.

form files

A collection of files that are used to implement an InfoPath form. File types can include .html, .xml, .xsd, .xslt, script, and other file types that are necessary to support the functionality of the form.

form library

A folder in which a collection of forms based on the same template is stored and shared. Each form in a form library is associated with user-defined information that is displayed in the content listing for that library.

form library columns

Information from forms displayed in columns in a Windows SharePoint Services form library. For every form in the library, columns display information entered into controls that the form designer designated as form library columns.

form template

In InfoPath, a file or set of files that defines the data structure, appearance, and behavior of a form.

G

Back to top

GIF

A graphics file format (.gif extension in Windows) used to display indexed-color graphics on the World Wide Web. It supports up to 256 colors and uses lossless compression, meaning that no image data is lost when the file is compressed.

global

Pertaining to an entire document, file, or program rather than to a restricted segment of it.

group

An element in the data source that can contain fields and other groups. Controls that contain other controls, such as repeating tables and sections, are bound to groups.

H

Back to top

HREF

See hypertext reference.

HTML

See Hypertext Markup Language.

hyperlink

Colored and underlined text or a graphic that you click to go to a file, a location in a file, a Web page on the World Wide Web, or a Web page on an intranet. Hyperlinks can also go to newsgroups and to Gopher, Telnet, and FTP sites.

Hypertext Markup Language (HTML)

The standard markup language used for documents on the World Wide Web. HTML uses tags to indicate how Web browsers should display page elements such as text and graphics and how to respond to user actions.

hypertext reference (HREF)

An attribute in a Hypertext Markup Language (HTML) document that defines a link to another location within a document, a document on the Web, or an e-mail address.

I

Back to top

IDE

See integrated development environment.

inline alert

A data validation alert that marks controls that contain invalid data with a dashed red border. Users can right-click controls to display custom error messages.

integrated development environment (IDE)

A set of integrated tools for developing software. The tools are generally run from one user interface and consist of a compiler, an editor, and a debugger, among others.

Internet

A worldwide network of thousands of smaller computer networks and millions of commercial, educational, government, and personal computers. The Internet is like an electronic city with virtual libraries, stores, art galleries, and so on.

intranet

A network within an organization that uses Internet technologies (such as the HTTP or FTP protocol). By using hyperlinks, you can explore objects, documents, pages, and other destinations on the intranet.

J

Back to top

JavaScript

A scripting language that can be used to add functionality to or enhance the look of a Web page or a Web site. JavaScript scripts can run on any type of client or server computer.

JPEG

A graphics file format (.jpg extension in Microsoft Windows) supported by many Web browsers that was developed for compressing and storing photographic images. It's best used for graphics with many colors, such as scanned photos.

JScript

An interpreted, object-based scripting language that borrows from C, C++, and Java. It is the Microsoft implementation of the ECMA 262 language specification.

L

Back to top

---

layout table

A collection of cells used to arrange form content such as text or controls.

linked object

An object that is created in a source file and inserted into a destination file, while maintaining a connection between the two files. The linked object in the destination file can be updated when the source file is updated.

list control

A control on a form that can be formatted as a bulleted, numbered, or plain list. A user can enter text into a list control, which repeats as needed.

M

Back to top

---

metadata

Data that describes other data. For example, the words in a document are data; the word count is an example of metadata.

method

A procedure similar to a statement or function that operates on specific objects. For example, you can apply the **Print** method to the

Debug object to transfer printed output to the Immediate window in the Visual Basic Editor.

Microsoft Script Editor (MSE)

A programming environment used to create, edit, and debug Microsoft JScript or Microsoft VBScript code in an InfoPath form.

Microsoft Windows Installer (MSI)

A tool that is used to create installation packages for software that is to be deployed on Windows operating systems.

MSDN

A Microsoft Web site (and subscription program) that provides developers with the latest tools as well as information, code samples, and training materials for Microsoft development tools (http://msdn.microsoft.com).

MSE

See Microsoft Script Editor.

MSI

See Microsoft Windows Installer.

N

Back to top

namespace

A mechanism that uniquely qualifies element names and relationships to avoid naming conflicts on elements that have the same name but are coming from multiple sources.

O

Back to top

object

A table, chart, graphic, equation, or other form of information. Objects created in one application, such as spreadsheets, and linked or embedded in another application are OLE objects.

object browser

A dialog box that displays information about objects, properties, methods, and constants in the current project and in referenced object libraries, and that you use to search for an element, get help on it, or paste it into a module.

object model

A hierarchical library of objects and collections that represents the content and functionality of an application. Objects and collections have properties, methods, and events that can be used to manipulate and interact with an application.

object model diagram

A graphical representation of the objects and collections that are contained within a specified object model.

offline

In reference to one or more computers, not being connected to a network.

open content model

A set of Extensible Markup Language (XML) document rules that states that an element in an XML document can have additional child elements and attributes that are not declared in the schema that the document references.

optional section

A control on a form that contains other controls and that usually does not appear by default. Users can insert and remove optional sections when filling out the form.

P

Back to top

---

parameter

A value that is assigned to a variable at the beginning of an operation or before an expression is evaluated by a program. A parameter can be text, a number, or an argument name assigned to another value.

parent element

In Extensible Markup Language (XML), the element of which a given element is a child.

processing instructions

Information stored in the prologue of an Extensible Markup Language (XML) document. This information is passed through the XML parser to any application that uses the XML document.

property

A named attribute of a control, a field, or an object that you set to define one of the object's characteristics (such as size, color, or screen location) or an aspect of its behavior (such as whether the object is hidden).

publish

To make a form available for others to fill out. A published form template is saved to a shared location such as a Web server or a company intranet.

R

Back to top

---

reference field

A field that is associated with another field so that their properties always match. If the properties in one field are changed, the properties in the other field are updated automatically.

reference group

A group that is associated with another group so that their properties and the properties of the groups and fields they contain always match. If the properties of one group are changed, the other group is updated automatically.

relative URL

A form of Uniform Resource Locator (URL) in which the domain and some or all directory names are omitted, leaving only the document name and extension (and possibly a partial list of directory names).

repeating field

A field in the data source that can occur more than once. Controls such as bulleted, numbered, and plain lists; repeating sections; and repeating tables can be bound to repeating fields.

repeating group

A group in the data source that can occur more than once. Controls such as repeating sections and repeating tables are bound to repeating groups.

repeating section

A control on a form that contains other controls and that repeats as needed. Users can insert multiple sections when filling out the form.

repeating table

A control on a form that contains other controls in a table format and that repeats as needed. Users can insert multiple rows when filling out the form.

Resource Manager

The interface for adding and removing resource files that support the functionality of a form.

rich text field

A field that can show formatting and graphics, such as an embedded object, in addition to text.

root element

The element in an Extensible Markup Language (XML) document that contains all other elements. It is the top-level element of an XML document and must be the first element in the document.

S

Back to top

scripting language

A simple programming language designed to perform special or limited tasks, sometimes associated with a particular application or function.

secondary data source

An Extensible Markup Language (XML) data file, database, or Web service that is used by a form for the entries in a list box or for script actions.

section

A control on a form that contains other controls.

server

On a local area network (LAN), a computer that controls access to all or part of the network and its resources, such as printers. On the World Wide Web, a computer that runs Web server software that responds to Hypertext Transfer Protocol (HTTP) requests. Also called a host.

shortcut key

A function key or key combination, such as F5 or CTRL+A, that you use to carry out a menu command. In contrast, an access key is a key combination, such as ALT+F, that moves the focus to a menu, command, or control.

shortcut menu

A menu that shows a list of commands relevant to a particular item. To display a shortcut menu, right-click an item or press SHIFT+F10.

sizing handle

One of the small circles or squares that appears at the corners and sides of a selected object. You drag these handles to change the size of the object.

structural editing

Editing that occurs within a structured document, such as a well-formed Extensible Markup Language (XML) document.

subscript

Describes text that is slightly lower than other text on a line. Subscripts are often used in scientific formulas.

superscript

Describes text that is slightly higher than other text on a line, such as a footnote reference mark.

T

Back to top

---

table

One or more rows of cells commonly used to display numbers and other items for quick reference and analysis. Items in a table are organized into rows and columns.

task pane

A window within an Office application that provides commonly used commands. Its location and small size allow you to use these commands while still working on your files.

text box

A control into which a user can enter text. Rich text boxes allow data entry with text formatting choices.

U

Back to top

Unicode

A character encoding standard developed by the Unicode Consortium. By using more than one byte to represent each character, Unicode enables almost all of the written languages in the world to be represented by using a single character set.

Uniform Resource Identifier (URI)

A character string used to identify a resource on the Internet by type and location.

Uniform Resource Locator (URL)

An address that specifies a protocol (such as HTTP or FTP) and a location of an object, document, Web page, or another destination on the Internet or an intranet— for example, http://www.example.com.

Uniform Resource Name (URN)

A scheme for uniquely identifying resources that may be available on the Internet by name, without regard to where they are located.

URI

See Uniform Resource Identifier.

URL

See Uniform Resource Locator.

URN

See Uniform Resource Name.

V

Back to top

valid

Error free or acceptable according to an established set of rules or standards. For example, data entered into a form is valid if it meets the criteria set in the form's data validation rules.

valid XML

A well-formed Extensible Markup Language (XML) document that conforms to a specific set of constraints, usually defined in an XML Schema.

VBScript

See Visual Basic Scripting Edition.

view

A form-specific display setting that can be saved with a form template and applied to form data when the form is being filled out. Users can switch between views to choose the amount of data shown in the form.

Visual Basic Scripting Edition (VBScript)

An interpreted, object-based scripting language that is a subset of the Microsoft Visual Basic programming language.

W

Back to top

Web browser

Software that interprets .html files, formats them into Web pages, and displays them. A Web browser, such as Microsoft Internet Explorer, can follow hyperlinks, transfer files, and play sound or video files that are embedded in Web pages.

well-formed XML

An XML document that meets the minimal criteria for a conforming

Extensible Markup Language (XML) document.

wizard

A feature that asks questions and then creates an item, such as a form or Web page, according to your answers.

X

Back to top

XHTML

See Extensible Hypertext Markup Language.

XML

See Extensible Markup Language.

XML attribute

An Extensible Markup Language (XML) structural construct. A name-value pair, separated by an equal sign and included in a tagged element, that modifies features of an element. All attribute values are text strings and must be enclosed in quotation marks.

XML data

Data that is saved in an Extensible Markup Language (XML) format. InfoPath forms are saved as XML data files.

XML element

An Extensible Markup Language (XML) structure that consists of a start tag, an end tag, and the information between the tags. Elements can have attributes and can contain other elements.

XML parser

A software component used to process Extensible Markup Language (XML) documents by validating their structure and producing views or transformations of their content.

XML Path Language (XPath)

A language used to address parts of an Extensible Markup Language (XML) document. XPath also provides basic facilities for manipulation of strings, numbers, and Booleans.

XML Schema

A formal specification, written in Extensible Markup Language (XML), that defines the structure of an XML document, including element names and rich data types, which elements can appear in combination, and which attributes are available for each element.

XML Signature

An Extensible Markup Language (XML)–based digital signature that can be used to help secure the data contained in XML documents. XML Signatures are a standard governed by the World Wide Web Consortium (W3C).

XML template file

An XML file that contains the sample data that is displayed in the fields of a form before a user fills it out.

XPath

See XML Path Language.

XSL

See Extensible Stylesheet Language.

XSL Transformation (XSLT)

A language that is used to transform Extensible Markup Language (XML) documents into other types of documents, such as Hypertext Markup Language (HTML) or XML. It is designed for use as part of Extensible Stylesheet Language (XSL).

XSLT

See XSL Transformation.