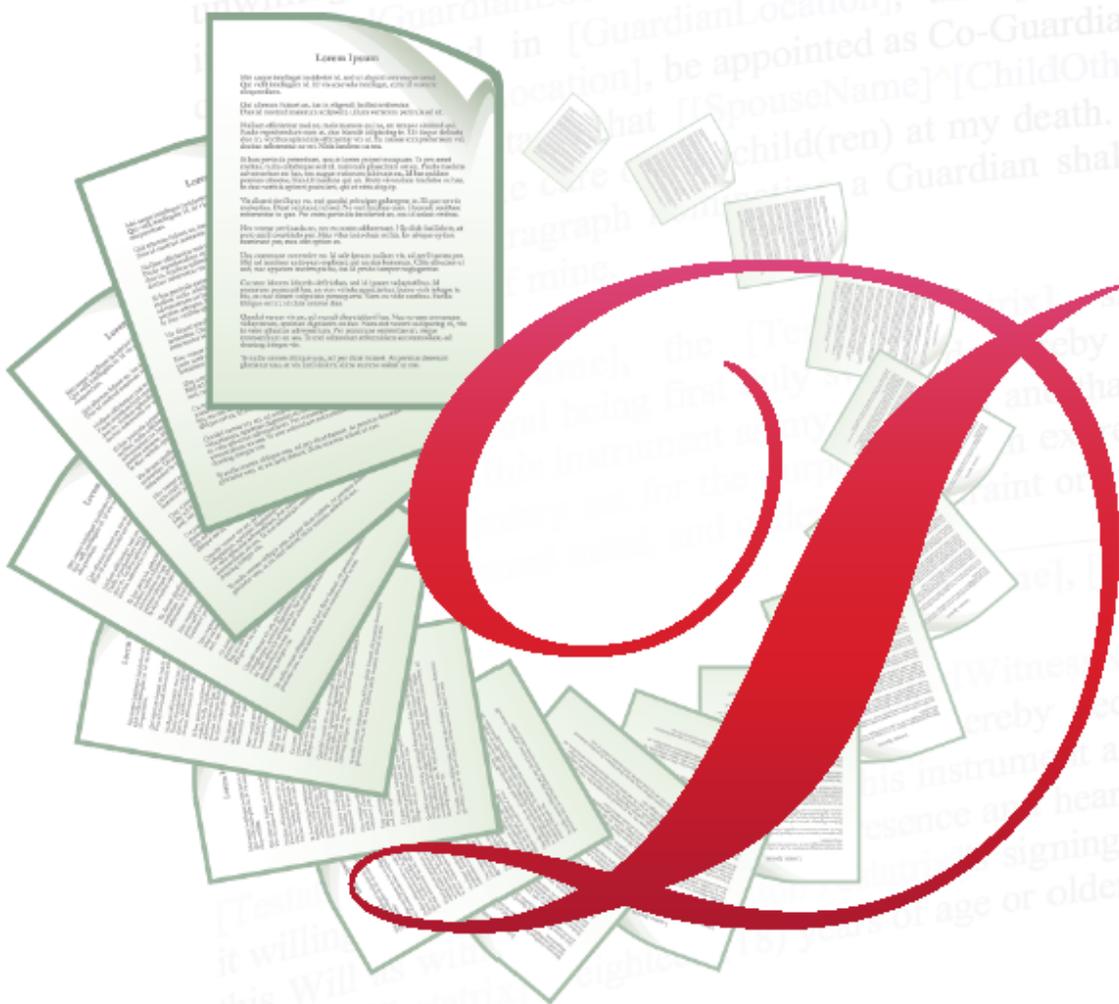


HotDocs®

2008 Professional Edition

Installation Guide and Tutorial



Copyright

© 2010 HotDocs Limited.

All rights reserved. No part of this product may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language in any form by any means without the express written permission of HotDocs Limited. (“HotDocs”)

Warranty Information

HotDocs makes no representations or warranties with respect to the contents or use of this product and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Information in this document is subject to change without notice and does not represent a commitment on the part of HotDocs.

Government Use

Use, duplication, or disclosure by the Federal Government is subject to restrictions as set forth in FAR clauses 52.227--14, “Rights in Data--General”; 52.227--19, “Commercial Computer Software--Restricted Rights”; and subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software clause DFAR 252.227--7013; and the limitations set forth in the standard commercial license agreement for this software. Unpublished rights are reserved under the copyright laws of the United States.

Trademark Information

HotDocs is a registered trademark of HotDocs Limited.
Other product names may be trademarks or registered trademarks of their respective companies.

Sample Templates and Forms

This product includes sample templates and forms only. Using them may have significant legal implications in some situations, and these implications vary by state and depending on the subject matter. Before using these templates and forms or adapting them for your business, you should consult with a lawyer and financial advisor.

Contact Information

For customer inquiries, contact:
HotDocs Corporation
387 S 520 W
Suite 210
Lindon, UT 84042
U.S.A.
publications@hotdocs.com

For technical support inquiries, call:
(800) 828-8328

November 2007
1128420074

Table of Contents

Chapter 1: Getting Started	1
Default File Locations in HotDocs 2008	2
Compatibility of HotDocs 2008 Files with Earlier Versions of HotDocs	3
HotDocs System Requirements	4
Install HotDocs 2008	5
Uninstall HotDocs	6
Understand HotDocs Installation	7
Start and Exit HotDocs	9
Technical Support and Customer Service	10
Get Help While Using HotDocs	11
Documentation Feedback	12
Chapter 2: HotDocs Professional Edition Tutorial	13
Lesson 1: Assemble a Document	14
Lesson 2: Create a Template File	21
Lesson 3: Replace Text and Dates in the Document with Variables	24
Lesson 4: Replace Numbers	30
Lesson 5: Create Conditional Text and Replace Multiple-Choice Text	37
Lesson 6: Test Assemble a Template	46
Lesson 7: Create Custom Dialogs	48
Lesson 8: Add a Clause Library to a Template Library	56
Lesson 9: Use Instruction and Expression Models	61
Lesson 10: Create Lists of Answers	69
Lesson 11: Use a Table to Display a List of Answers	78
Lesson 12: Use Complex IF Instructions	84
Lesson 13: Incorporate Clause Selection in a Template	91
Lesson 14: Change the Layout of Dialogs	96
Lesson 15: Create Scripts for Your Dialogs	100
Lesson 16: Create Lists Within a List	108
Chapter 3: HotDocs Automator Tutorial	115
Lesson 1: Create a Form Template File	116
Lesson 1a: Use an Already-Created Form Template	118
Lesson 2: Create Fields and Attach Text Variables	119
Lesson 3: Attach Date and Computation Variables	128
Lesson 4: Attach Multiple Choice Variables	133
Lesson 5: Attach Number Variables	137

Lesson 6: Attach True/False Variables and Make Fields Conditional	141
Lesson 7: Create a List Using a Table	146
Lesson 8: Type Your Answers Directly on the Form	150
Lesson 9: Create an Answer Wizard	152
Lesson 10: Create Answer-Gathering Dialogs	156
Lesson 11: Instruct HotDocs to Create a Default Interview	167
Lesson 12: Assemble a Document Using an Interview	169
Lesson 13: Add Templates to a Library	172

Chapter 4: HotDocs Database Connectivity Tutorial 175

Lesson 1: Create an ADO Connection String	176
Lesson 2: Link Variables to Fields in a Database Table	182
Lesson 3: Select Fields and Records for Display	188
Lesson 4: Filter the List of Records	191
Lesson 5: Select Related Database Records	194
Lesson 6: Test the Template	197

Chapter 1: Getting Started

Overview

This manual is divided into four chapters:

Chapter 1: Getting Started

This chapter contains information about installing and uninstalling HotDocs® 2008 Professional Edition. It also contains information on how you can get help while using HotDocs.

Chapter 2: HotDocs Professional Edition Tutorial Lessons

This chapter contains lessons for creating text-based templates, or templates you automate in your word processor. Each lesson is self-contained and can be completed in short blocks of time.

Chapter 3: HotDocs Automator Tutorial Lessons

This chapter contains lessons for creating form-based templates, or templates you automate using HotDocs Automator. Like the tutorials in Chapter 2, these lessons are self-contained and can be completed in short blocks of time.

Chapter 4: HotDocs Database Connection Tutorial Lessons

This chapter contains seven short lessons that teach you how to integrate your templates with a database.

All other documentation about HotDocs is included in the HotDocs Help, which you can access from any HotDocs **Help** menu. (See “Get Help While Using HotDocs” on page 11.)

Default File Locations in HotDocs 2008

When you install and use HotDocs, the program makes several entries in the System Registry that indicate to HotDocs where to look for files, as well as where to suggest you save files. In most situations, HotDocs uses this information each time you perform a task in HotDocs. For example, each time you create a new template, HotDocs suggests a default location for the new template, based on the information it finds in the System Registry.

The following is a list of these default locations:

 On Windows Vista, the **My Documents** folder has been named **Documents**. The **All Users Documents** folder has been named **Public Documents**.

File Type	Default Location
Program Files	C:\Program Files\HotDocs 6
JavaScript Files	<Program Files>\HotDocs 6\JavaScript
Help Files	<Program Files>\HotDocs 6
Spelling Dictionary	<My Documents>\HotDocs\Spelling
Library Files	<My Documents>\HotDocs\Libraries
Template Files	<My Documents>\HotDocs\Templates
<small>Includes text templates, form templates, interview templates, and auto-assemble files</small>	
WordPerfect Macro Files	Word Processor Macro Folder
MS Word Startup Files	Word Processor Startup Folder
Template Sets	<All Users Documents>\HotDocs\Templates
Text Document Files	Word Processor Documents Folder
Form Document Files	<My Documents>
Answer Files	<My Documents>\HotDocs\Answers
Publish Settings Files	<My Documents>\HotDocs\Publish

Once HotDocs is installed, you can change most of these locations. To do this, click the  **HotDocs Options** button in the library toolbar and click either the **File Locations** folder or the **Word Processors** folder. (See the HotDocs Help for details.)

 If you are upgrading from HotDocs 6.1 or earlier and your libraries are saved in the HotDocs 6 **Program Files** folder, HotDocs will set this folder as the default location for libraries. If you do not have library files saved here, HotDocs will use **<My Documents>\HotDocs\Libraries**.

 If you perform a custom installation, HotDocs will install files and set registry entries using the information you specify during installation. Whatever settings you specify during installation will be used for all users of the workstation.

Compatibility of HotDocs 2008 Files with Earlier Versions of HotDocs

Installation of Program Files

When you install HotDocs 2008 over HotDocs 6, HotDocs 2005, HotDocs 2006, or HotDocs 2007, it overwrites that installation.

Templates and Component Files

If you are upgrading from HotDocs 6 or HotDocs 2005, you will be prompted to convert your component files to HotDocs 2006-2008 format. (You will not be prompted to do this if you are upgrading from HotDocs 2006 or HotDocs 2007.)

After you convert the component file to the latest version, if you need to use it with an earlier version of HotDocs, you can specify which version at the **Component File Properties** dialog box of Component Manager. (Please see the HotDocs Help for a complete description on how to do this.)

 If you are using HotDocs 2008 to automate templates for use with an earlier version of HotDocs, do not use any new features that aren't supported in the earlier version or your templates may not work correctly. See the HotDocs Help for a list of these features.

Answer Files

HotDocs answer files are not version-specific. They can be created and used with any version of HotDocs.

HotDocs System Requirements

Before installing HotDocs 2008, make sure your computer meets the minimum requirements necessary for using HotDocs:

Hardware

- 900 MHz processor (2GHz, recommended)
- 256 MB RAM
- 50 MB hard disk storage
- 1024 x 768 screen resolution with at least 16-bit color

Software

- Microsoft Windows XP or Windows Vista
- Microsoft Internet Explorer 6.0 or later
- Microsoft Word 97, 2000, XP, 2003, or 2007; or Corel WordPerfect 8, 9, 10, 11, 12, of X3

 Microsoft Word97 is not supported for use with HotDocs on Windows Vista. Additionally, WordPerfect X3 SP2 is the only version of WordPerfect supported for use on Windows Vista.

Databases

If you plan to integrate HotDocs with a database, you must have one of the following applications or server:

- Microsoft Access
- Microsoft SQL Server
- Oracle 8i or 9i with the OraOLEDB OLEDB provider
- Most other ODBC-compliant databases

Install HotDocs 2008

When you insert the HotDocs installation disk in your CD-ROM drive, a CD browser window should automatically appear. From this window, you can view the Readme, browse the contents of the CD, and ultimately install HotDocs. Once installation has started, HotDocs prompts you for all the information needed to complete the process.

Before installing HotDocs 2008, make sure your computer meets the minimum requirements necessary for using HotDocs. (See “HotDocs System Requirements” on page 4.)

To install HotDocs 2008

- 1 Close all open applications, especially any word processors or existing versions of HotDocs you may have open.
- 2 Insert the HotDocs installation CD into your CD-ROM drive. The CD browser window appears.
- 3 Click **Install HotDocs 2008 Professional Edition**. The installation program begins installing HotDocs.
- 4 Follow the on-screen prompts to complete the installation.

 To manually install HotDocs, select **Run** from the **Start** menu and type **D:\setup**. (Substitute the correct drive letter if your CD-ROM drive is not mapped to **D:**.) Follow the on-screen prompts to complete the installation.

Uninstall HotDocs

You can remove existing versions of HotDocs from your computer by using the HotDocs uninstall program. When you uninstall, all of the program files are removed from your computer. However, the program does not remove any templates, libraries, or answer files you have created.

To uninstall HotDocs

- 1 At the **Start** menu, select **Settings > Control Panel > Add or Remove Programs**. The **Add or Remove Programs** dialog box appears.
- 2 Select the version of HotDocs that needs to be removed, then click **Remove**. The HotDocs uninstall process starts.
- 3 Follow the on-screen prompts to remove the application.



If you plan to reinstall HotDocs to restore default settings, you must delete the user-specific settings HotDocs entered in the System Registry. For details on doing this, see “Understand HotDocs Installation” on page 7.

Understand HotDocs Installation

When you install HotDocs, two different installations actually happen—the *HotDocs Setup* and the *Current User Setup*. This two-part installation process allows the application to be installed just once for multiple users on a single workstation. It also allows each user who logs on to the workstation to have his or her own custom files and settings.

HotDocs Setup

When you first install HotDocs from *Setup.exe* (see “Install HotDocs 2008” on page 5), HotDocs installs the program files needed to use HotDocs, including any executable files used to run HotDocs. Among others, it also installs the HotDocs Help files and JavaScript files (which are needed for developing HotDocs Server templates). These files are installed, by default, to *C:\Program Files\HotDocs 6*. This location is not user-specific, which allows all users of a single workstation access to the files.

As HotDocs installs these files, it registers information about them in the *Local Machine* key of the Windows System Registry. For example, the Registry stores information about where the executable file for HotDocs is located on disk, as well as which word processors HotDocs supports. It also sets some preliminary file location information for the various files HotDocs uses, including information about library files and template set files. All of this information must be registered in order for HotDocs to work correctly.

Current User Setup

When you install HotDocs, the installation creates a folder named *Source* in the HotDocs program folder. This folder contains several files, including word processor macro or startup files, the user spell-checking dictionary, and the HotDocs tutorial files. These files must be in user-specific folders on the disk. However, because multiple users may use a single workstation, HotDocs Setup doesn't install these files. Instead, when a user logs on to the workstation and runs HotDocs, the Current User Setup copies these files to the correct locations for the user.

The Current User Setup also registers information about each user in the *Current User* key of the System Registry. This information includes default file locations for such things as templates, answer files, and library files. As each user specifies preferences for working with HotDocs, these settings are likewise written to the *Current User* key. This allows each workstation user to have his or her own HotDocs settings without other users overriding those settings with their own preferences.

The Current User Setup happens each time a user starts HotDocs. This means that each time you run HotDocs, HotDocs checks to make sure all of the required files (such as word processor startup files) are located where it expects to find them. If the file isn't found, HotDocs replaces it using a copy it finds in the *Source* folder.

 Even if you delete the tutorial files that are installed to your default **Templates** folder, because of the Current User Setup, the next time you launch HotDocs, HotDocs will recopy the files to that folder. To keep this from happening, open the **Source** folder, select the **Tutorials** subfolder and press **Delete**.

Restore Installation Defaults

At times, you may want to restore default settings in HotDocs. You can accomplish this by clearing the *Current User* key of the Registry.

 You must be extremely careful when working in the System Registry. Failure to follow the instructions below exactly could result in your making changes that negatively affect all the programs on your computer. You may want to ask your system administrator for help if you are unsure of what you are doing.

To restore default settings

- 1 At the **Start** menu, select **Run**. The **Run** dialog box appears.
- 2 In the **Open** box, type **regedit** and click **OK**. The **Registry Editor** appears.
- 3 Navigate to **HKEY_CURRENT_USER > Software > LexisNexis > HotDocs 6**.
- 4 Select the **HotDocs 6** subfolder and press **Delete**. Click **Yes** to confirm the deletion.
- 5 Close the **Registry Editor**.

Start and Exit HotDocs

Before you edit templates or assemble documents, you must first open the HotDocs library, which provides the functionality for working with your template files.

To start HotDocs

- Choose **HotDocs 2008 > HotDocs Professional** at the **Start** menu. The HotDocs library window appears.

To exit HotDocs

- Click **Exit** (**File** menu).

 To start HotDocs, you can also click the  **HotDocs 2008** button in the word processor toolbar. This will open and display the HotDocs library window.

Technical Support and Customer Service

Contact Technical Support

Technical Support specialists are available 24 hours a day, seven days a week to help you use HotDocs. If you are having difficulty installing or using the software, or if you encounter an error message, please contact a technical support representative at (800) 223-5297.

Additionally, you may find answers or solutions to questions you have in the HotDocs Knowledge Base. To view the Knowledge Base (as well as access other support options), please visit <http://www.hotdocs.com/support/>.

Contact Customer Service

Experienced HotDocs consultants are available to help you with a variety of services, including integrating HotDocs with other products, building a template library, or providing training. Please contact us using any of the following methods:

HotDocs Corporation
387 S 520 W, Suite 210
Lindon, UT 84042,
U.S.A.

(800) 500-3627 (Sales)
(801) 615-2200 (Business)
(877) 356-3627 (Fax)

sales@hotdocs.com
<http://www.hotdocs.com/products/>
<http://www.hotdocs.com/services/>

HotDocs Information on the Web

You can also access other information about HotDocs by visiting the HotDocs Web site (<http://www.hotdocs.com>). At this site, you can

- Subscribe to the HotDocs discussion e-mail list.
- Download HotDocs templates.
- Find other information about HotDocs products.
- View a list of other HotDocs products and services that are available.

Get Help While Using HotDocs

You can use the HotDocs Help system to find information about HotDocs—including explanations of HotDocs features as well as step-by-step instructions for completing most tasks in HotDocs. The HotDocs Help system also provides dialog-level help and **What's This** help. Dialog-level help explains the overall purpose for the dialog box you are viewing, while  **What's This** help explains specific options or properties of the dialog box.

To access task and reference information

- 1 Select **HotDocs Help** at any **Help** menu. (If you are working in a text template, click the  **Help** button in the HotDocs toolbar.) The HotDocs Help window appears.
- 2 Perform any of the following tasks:
 - Click the **Contents** tab, double-click the book icons to expand (and collapse) them, and then click on a topic to display its contents. (The **Contents** tab is arranged by subject matter.)
 - Click the **Index** tab, and type the keyword or phrase for which you are searching in the box. As you type, HotDocs displays relevant topics. At any time, select the topic that fits your search criteria and either double-click the entry or click **Display**.
 - Click the **Search** tab, and type the word or phrase for which you are searching in the box. Click **List Topics** to display the search results in the **Select Topic to display** list. Once you have located the topic you want to view in this list, double-click it or click **Display**.
 - Click the **Favorites** tab and then click **Add** to add a topic you reference frequently. This bookmarks the topic so you can find it quickly. To view it later, select it from the **Favorites** list and click **Display**.

To access dialog-level help

- At any dialog box that contains a **Help** button, click the button. HotDocs displays the information about the contents of that specific dialog box.

To access What's This help

- At any dialog box where What's This help is available, click the **What's This** button, either in the title bar () or the toolbar (), and then click the item for which you want help.

 While at the **Search** tab, you can search for a literal phrase by including quotation marks around the phrase (for example, "add text to"). Likewise, you can use a wildcard expression by including an asterisk or question mark in the word or phrase for which you are searching (for example, *explore** or *HotDocs 6.?*). Finally, search using the Boolean operators **AND**, **OR**, **NOT**, or **NEAR**.

 To access What's This-level help using your keyboard, press **Shift+F1**.

Documentation Feedback

To improve the quality of the tutorials and the help files, we invite you to make comments or suggestions. When doing so, please include as much information about your experience using the documentation as possible. For example, include which version of the product you are using, as well as whether your suggestion is in regards to the printed *Installation Guide and Tutorial* or the electronic help file. If commenting about a specific topic, include that information as well.

 Please keep in mind that we cannot respond to technical support or project consulting issues. We are mainly interested in problems with the documentation itself—such as erroneous information, grammatical and spelling errors, or suggestions for topics to include in the next release of the software.

E-mail your comments and suggestions to publications@hotdocs.com.

Chapter 2: HotDocs Professional Edition Tutorial

Overview

The lessons in this chapter introduce you to important concepts and features available in HotDocs Professional Edition. Because of the work you do in these lessons, when you begin automating your own documents, the procedure should be familiar to you, allowing you to focus on your documents, not HotDocs.

These lessons are short and are structured so you can complete each one in separate sessions. This allows you to learn HotDocs in the spare minutes you have during the day. You do not need to dedicate a large block of time to using the tutorial.

You should complete all of these lessons in order.

Lesson 1 walks you through the process of using HotDocs templates and clause libraries to assemble documents.

Lessons 2 through 8 cover the basics of template automation by showing you, step by step, how to create the text template and clause library you assembled in Lesson 1.

Lessons 9 through 16 show you how to perform more advanced HotDocs tasks.

In Chapter 3, you will complete the tutorial for HotDocs Automator. Then, in Chapter 4, you will complete the tutorial for HotDocs database connectivity

 The documents and clauses used in the tutorial were drafted to suit the purposes of the tutorial. They are not provided as valid legal documents or clauses.

Lesson 1: Assemble a Document

Overview

HotDocs Professional Edition provides two ways to create a finished document: you can use a template to assemble a customized document, or you can select clause text from a clause library and insert it into a document. This lesson shows you how to create a customized document using each method. Later lessons show you how to create the template and clause library you will use in this lesson.

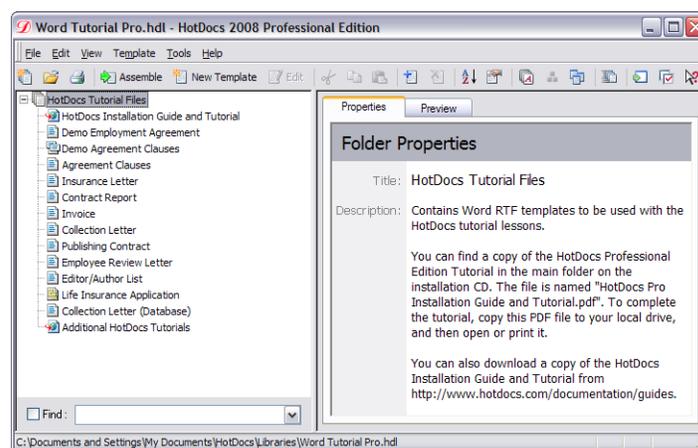
Assemble a Complete Document

Each time you use a template to assemble a document, HotDocs displays a series of dialogs that prompt you to provide variable information, or information that is different for each document (such as names, dates, and instructions for conditional paragraphs). Once you have entered the information the document requires, HotDocs merges your answers and carries out your instructions to produce a finished document. The document can then be printed, saved, or edited further.

To see how a HotDocs template produces a complete, customized document, complete the following steps. You'll assemble an *Employment Agreement* using a template that has already been automated.

To assemble the Employment Agreement document

- 1 Choose **HotDocs 2008 > HotDocs Professional** from **Programs** on the **Start** menu. The HotDocs template library appears. You see a list of folders and templates in the left pane of the window and information about the selected library item in the right pane.



✓ Click the **Preview** tab to see what the template looks like before you assemble a document from it.

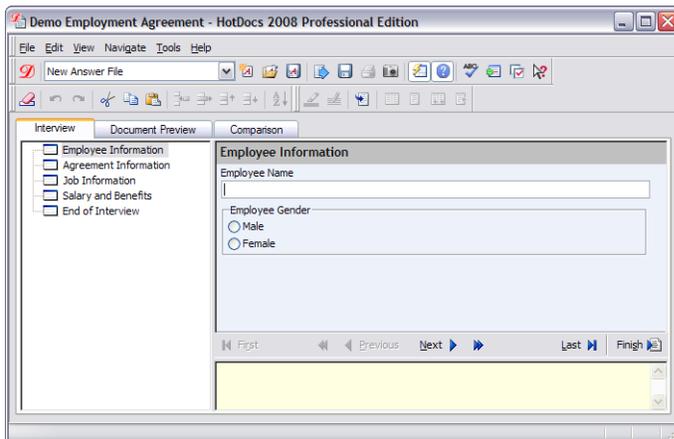
⚠ If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library (File menu)**, select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete step 2.

- 2 From the template list, select **Demo Employment Agreement** and click  **Assemble**. The **Answer File** dialog box appears.



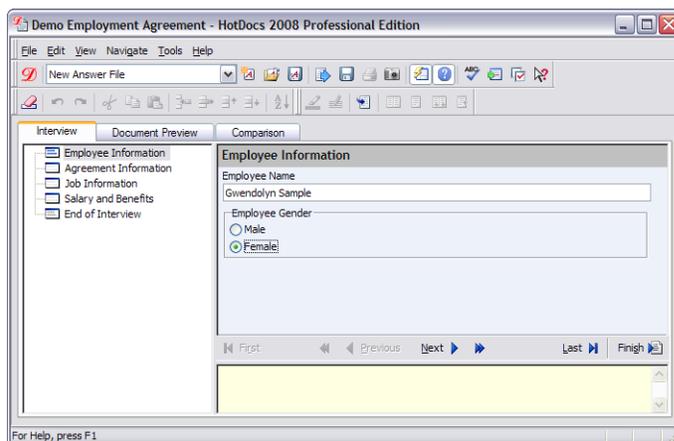
The **Answer File** dialog box shows a new, untitled answer file as the answer file that will be used with this assembly. This file does not yet contain any answers.

- 3 Click **OK** to continue. HotDocs displays the interview.



You see a list of dialogs (or groups of questions) in the left pane of the window. These dialogs comprise the *interview outline*. When you click one of these dialogs, HotDocs displays the associated questions and answer fields in the right pane, or *dialog pane*.

- 4 At the first dialog, **Employee Information**, type a name in the **Employee Name** answer field and select a gender.



- 5 Click  **Next** (or press **Enter**) to advance to the next dialog, **Agreement Information**.

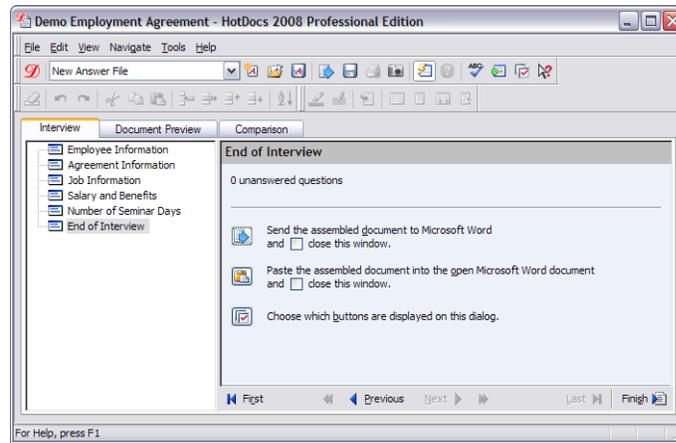
 As you assemble documents, you can save your answers in an answer file, which can then be reused if you reassemble this document or if you assemble other documents that require similar information.

 As you proceed through the interview, dialog icons in the interview outline change appearance, indicating whether the questions in the dialog are unanswered, partially answered, or completely answered.

 To enter dates, you can type the date directly in the answer field, or you can click the  **Calendar** icon to the right of the answer field and choose a date there.

- 6 Continue answering questions as dialogs appear. After the last dialog, the **End of Interview** dialog appears.

The *End of Interview* dialog gives you options for working with the assembled document, such as sending a copy of the assembled document to the word processor.



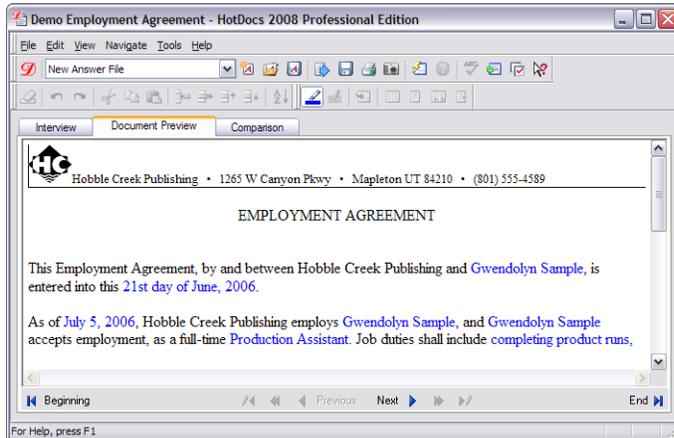
Preview the Assembled Document

Once an assembled document is sent to the word processor, the document is no longer associated with HotDocs. This means that if you change any text in the document, including any text that was merged because of answers you entered during the interview, the changes are only reflected in that document—not the template or the answer file. Because of this, you should always review your assembled documents for accuracy before sending them to the word processor. You can review a copy of the assembled employment agreement while at the HotDocs assembly window.

 The quality of the reviewed document depends on your word processor and the amount of text formatting that is used in the underlying document. Specifically, WordPerfect documents present only a rough view of the document, while Word documents present a more accurate preview of the document.

To preview the assembled document

- At the assembly window, click the **Document Preview** tab. The assembled document appears. It shows all of the answers you entered during the interview merged into the document.



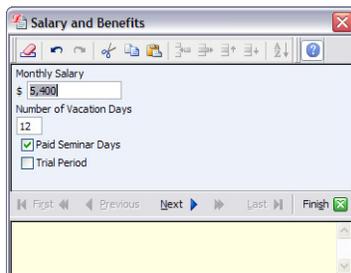
Edit Answers at the Document Preview Tab (Word Users)

If you are using Microsoft Word, you can change answers while viewing the **Document Preview** tab. When you edit an answer, any other questions affected by the answer are likewise updated.

If you are using WordPerfect, skip to “Send the Assembled Document to the Word Processor” on page 18.

To edit an answer in the document

- If answers aren't already highlighted, click the  **Highlight Answers** button in the assembly window toolbar.
- In the second paragraph, double-click the answer for the monthly salary. The **Salary and Benefits** dialog appears in a pop-up interview.



- Change the answer in the **Monthly Salary** field and click  **Finish**. The pop-up interview closes and the document is updated with your new answer. (Note that the annual salary (the next answer in the document) is likewise updated.)

Send the Assembled Document to the Word Processor

Now that you have completed the interview and previewed the assembled document, you can send a copy of it to your word processor.

To send the document to the word processor

- 1 At the assembly window toolbar, click the  **Send Document to Word Processor** button. HotDocs opens the word processor and displays the complete document. You can print, save, or edit the document.

In the next part of the lesson, you will insert additional information about the agreement into the document you just sent to the word processor. This text will be retrieved from a clause library.

- 2 In the document, insert your cursor two lines below the last paragraph (but before the signature lines). This is where the clauses will be inserted.
- 3 Click the **Demo Employment Agreement** icon on the Windows taskbar to bring the assembly window to the front.
- 4 Choose **Close** (File menu). When HotDocs prompts you, don't save your answers or a copy of the assembled document.

Insert Clauses into a Document

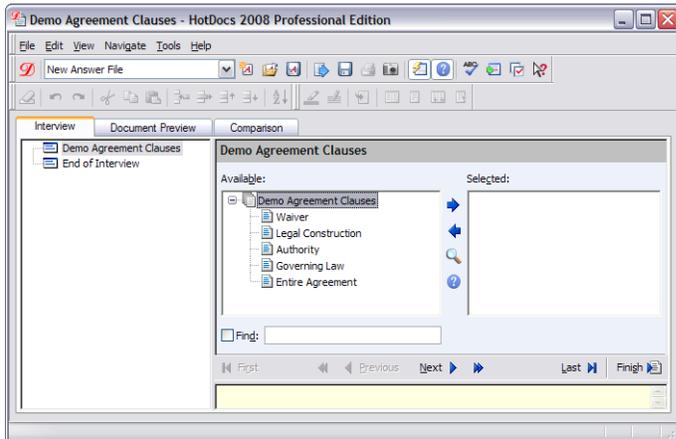
A clause is a section of text that users can insert into either a new, empty document or a document that already contains text. Usually, clauses are grouped together in a clause library, which you open from the HotDocs template library. After you select the clauses you want and place them in the order you want them inserted in the document, HotDocs prompts you for any variable information. You can then merge the assembled clauses in the word processor document.

In the following steps, you will select clauses, answer any required questions, and then paste the assembled clause text into the *Employment Agreement* you just assembled.

To assemble a document of clauses

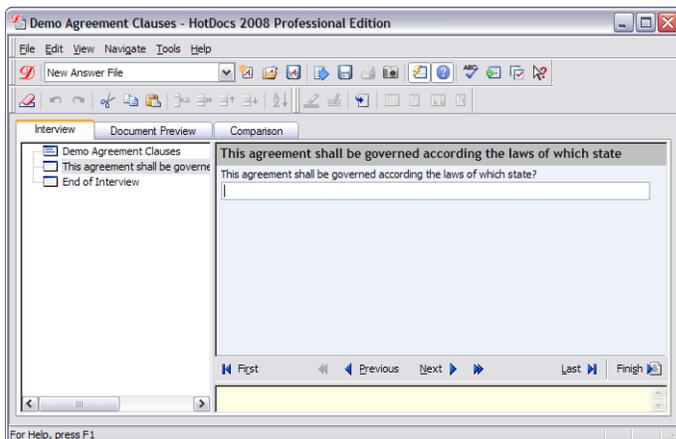
- 1 At the template library, select **Demo Agreement Clauses**.
- 2 Click  **Assemble**. The **Answer File** dialog box appears, with **New Answer File** selected.

- 3 Click OK. An assembly window appears with the **Demo Agreement Clauses** library displayed in the dialog pane.



- 4 In the **Available** list, select **Waiver** and click the **Select** button. The clause is added to the **Selected** box.
- 5 Add **Legal Construction**, **Authority**, **Governing Law**, and **Entire Agreement** to the **Selected** box.
- 6 Click **Next**. A dialog appears to gather information about the governing state.

✓ To change the order of clauses in the **Selected** list, select a clause and drag it up or down. (The order of the clauses in the list determines the order they are merged into the assembled document.)



- 7 Type an answer and click **Next** to advance. The **End of Interview** dialog appears.

Just as you used the **Send Document to Word Processor** button in the HotDocs toolbar to send the *Employment Agreement* to the word processor, you can also use shortcut options in the *End of Interview* dialog.

- 8 At the **Paste the assembled document into the open word processor document** shortcut option, select **close this window** and then click the corresponding button ().
- 9 As HotDocs is closing the assembly window, it prompts you to save your answers and a copy of the assembled document. Click **Don't Save** at each message box.

HotDocs then brings the open **Employment Agreement** document to the front and displays the assembled clauses in it.

10 Once you have viewed the *Employment Agreement* with the new clauses added, close the word processor window. (Do not save the document.)

In Lessons 2 through 7, you will recreate the *Employment Agreement* you assembled earlier in this lesson. Lesson 8 will teach you how to create the *Agreement Clauses* clause library.

You are now finished with this lesson.

Lesson 2: Create a Template File

Overview

The next six lessons teach you how to create and automate an *Employment Agreement* template like the one you assembled in Lesson 1.

The first step in the automation process is to create a HotDocs template file. When you do this, you must specify what text will be used in the template. For example, either you can use text from an existing document as the basis for the template, or you can enter your text after you create a new, empty template. In this lesson, you will use text from an existing agreement.

Once you have created a template, you can use HotDocs template-building features.

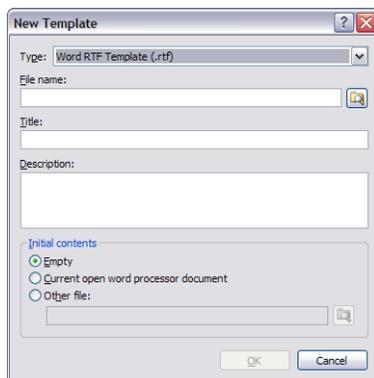
To open HotDocs

1 If HotDocs isn't already running, choose **HotDocs 2008 > HotDocs Professional** from **Programs** on the **Start** menu. The HotDocs library window appears.

2 Click on the **HotDocs Tutorial Files** folder.

 If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library (File menu)**, select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete step 2.

3 Click  **New Template**. The **New Template** dialog box appears.



4 Select the type of template you want to create from the **Type** drop-down list, based on which word processor you have installed. (For example, to create a Microsoft Word template, select **Word RTF Template**.)

5 Type **Employment Agreement** in the **File name** box.

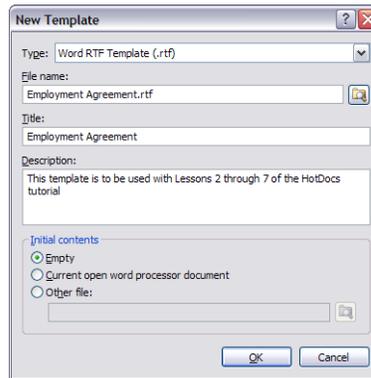
6 Click in the **Title** box and accept the suggested title, **Employment Agreement**.

 If you are using Word to develop templates, develop them using RTF (or Rich Text Format). RTF templates assemble faster than DOT templates.

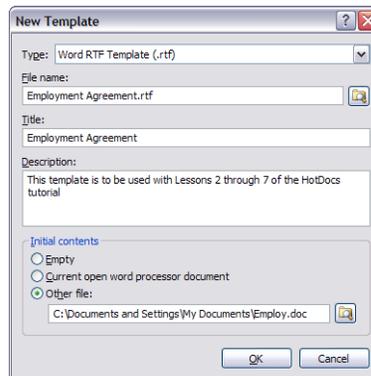
✓ Template titles and descriptions help the user identify the template at the HotDocs library.

✓ If you performed a custom installation, the document will be located in the default *Documents* folder you specified during installation. Use your computer's **Find** command (**Start > Search > For Files or Folders**) if you are having a difficult time finding the file.

- 7 Type **This template is to be used with Lessons 2 through 7 of the HotDocs tutorial** in the **Description** box.



- 8 In the **Initial contents** group, select **Other file** and click the **Browse** button. The **New Template Initial Contents** dialog box appears.
- 9 Select either **employ.doc** or **employ.wpd**, located in the default word processor *Documents* folder. The text in this document will be used as the text for the template you are creating.
- 10 Click **OK** to return to the **New Template** dialog box.



The folder path and file name of the *Employment Agreement* document appear in the **Other file** box.

- 11 Click **OK**. The new template file is created, and the text from the *Employment Agreement* document is copied into it.

When you create a template, HotDocs automatically creates a companion file for the template called a *component file*. This component file is necessary for a template to work, since it stores all of the information about the template's automation (such as component names, template properties, and so forth).

The component file works automatically in the background—as you create various components in the template, they are automatically stored in the component file.

HotDocs gives the component file the same name as the template file. In this instance, HotDocs has created a component file named *Employment Agreement.cmp* (since your template file is named *Employment Agreement.rtf*).

Familiarize Yourself with the HotDocs Toolbar

When you create a new template, the template opens in the word processor. The HotDocs editing toolbar automatically appears below the word processor toolbars. You use this toolbar to edit the HotDocs template.

The button designs on the toolbar vary slightly, depending on the word processor you are using. The buttons are, from left to right:

- «» Variable Field
-  IF Field
-  ASK Field
-  INSERT Field
-  REPEAT Field
-  SPAN Field (Word users only)
-  Assemble
-  Edit Component
-  Component Manager
-  Clause Library
-  Save
-  Save and Close
-  HotDocs Help
-  Apply Colors
-  Label Fields
-  Match Fields
-  Go to Field
-  Markup View (Word users only)
-  Developer View (Word users only)
-  Previous Field
-  Next Field
-  HotDocs Outliner (Word users only)

 In Word 2007, all HotDocs editing and navigation buttons can be found on the **HotDocs** tab of the Word ribbon.

In addition to these buttons, the  **HotDocs 2008** button appears in the main word processor toolbar. Clicking it displays the HotDocs library.

 You should always use the  **Save** button to save your work, or the  **Save and Close** button to save and close a template, rather than your word processor's **Save** and **Close** commands. Using these buttons ensures the template and component file are both properly saved and closed.

You are now finished with this lesson. If you do not want to go on to Lesson 3 at this time, click the HotDocs  **Save and Close** button to close the template.

 At any time during these lessons, you can click the  **HotDocs Help** button to display the HotDocs Help file. You can then search for more information about anything in the tutorial you don't understand or about which you would like to learn more.

Lesson 3: Replace Text and Dates in the Document with Variables

Overview

After you have created a template file, the next step is replacing the text in the template that will be different for each document (such as names, dates, numbers, and calculations) with HotDocs variables.

This lesson teaches you how to replace text and dates with variables. It also teaches you how to create prompts for variables. Prompts are the questions the user sees when providing the answers that will be merged into the document.

If you are continuing immediately from Lesson 2, skip the instructions for opening the template and proceed to “Replace Text.”

If you closed the template at the end of Lesson 2, complete the following steps.

To open the Employment Agreement template you created in Lesson 2

- 1 Open your word processor and click the  **HotDocs 2008** button, located on your word processor’s toolbar. The HotDocs library window appears.
- 2 Select **Employment Agreement** and click  **Edit**. The template appears, ready for you to edit.

Replace Text

The first instance of text that needs to be replaced in the template is the name of the employee, *Aaron Jameson*, in the first paragraph.

To replace the employee name with a HotDocs variable

- 1 Select (highlight) **Aaron Jameson**.

EMPLOYMENT AGREEMENT

This Employment Agreement, by and between Hobble Creek Publishing and **Aaron Jameson**, is entered into this 12th day of October, 2007.

- 2 Click the  **Variable Field** button in the HotDocs toolbar. The **Variable Field** dialog box appears.

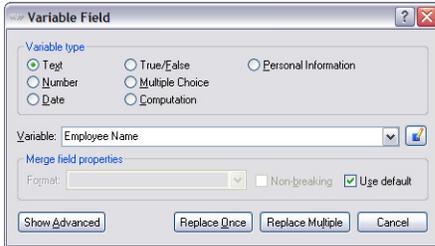


 In HotDocs, there are two aspects of a variable with which you work—the variable *field* and the variable *component*. The variable field controls how the answer is merged into the document. When editing the variable field, for example, you can specify the way you want that specific answer formatted. When editing the variable component, you control specific properties of the variable. For example, you can assign a prompt to help the user know what information is required during the interview.

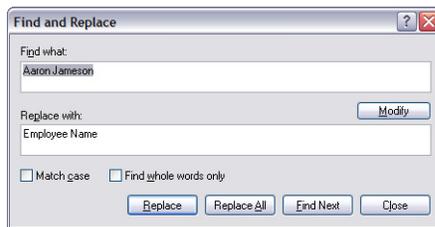
The dialog box lists several variable types. The text you selected, *Aaron Jameson*, should be replaced with a Text variable, which is selected by default.

Now you must name the variable.

- 3 Type **Employee Name** in the **Variable** box.

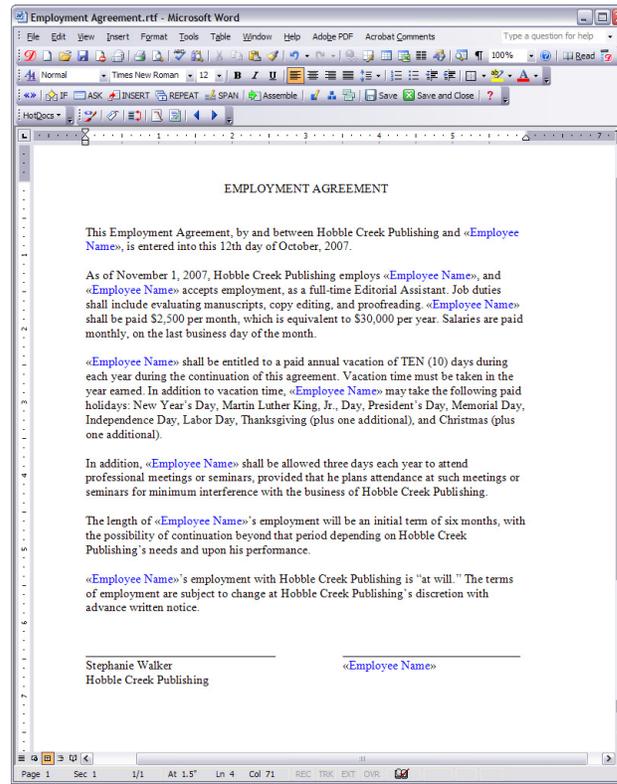


- 4 Because there are multiple instances of *Aaron Jameson* in the template, click **Replace Multiple**. A variable component named *Employee Name* is created and the **Find and Replace** dialog box appears.



- 5 Make sure the check box labeled **Find whole words only** is cleared. (Clearing this option allows HotDocs to replace all instances of *Aaron Jameson* with the variable *Employee Name*, even if a specific instance ends with *'s*.)

- 6 Click **Replace All**. HotDocs replaces all instances of *Aaron Jameson* with «Employee Name».



Replace Dates

The first date to be replaced in the template is the date of the agreement, *12th day of October, 2007*. It is located in the first paragraph.

To replace the date with a Date variable

- 1 Select **12th day of October, 2007** and click the «» Variable Field button. The Variable Field dialog box appears.
- 2 Select **Date** from the Variable type group.



Notice that when you select **Date**, the **Format** box in the **Merge field properties** group shows an example format. This controls the format of an answer when it is merged into a document. It is suggested based on the format of the template text you selected.

- 3 In the **Variable** box, type **Agreement Date** and click **Replace Once**, since the agreement date appears only once in the document.

In addition to the variable name, the format, *3rd day of June, 1990*, is merged into the field as well.

✓ The format you assign here will be used for this instance of the variable only. In other words, if you insert this same variable later in the template, it will not have this format assigned. If you want to assign a format that will be used every place the variable is merged, click the  **Edit Component** button at the **Variable Field** dialog box and select an example format from the **Default format** drop-down list.

On Your Own

The second date to be replaced is the hire date, *November 1, 2007*, located in the second paragraph. Replace *November 1, 2007* with a Date variable named *Hire Date*. (Follow the instructions for creating the variable *Agreement Date*, earlier.)

Later in the same paragraph, replace *Editorial Assistant* with a Text variable named *Job Title*. (Follow the instructions for creating the variable *Employee Name*, earlier.)

Click the  **Save** button to save your work.

Create Prompts and Adjust the Height of the Answer Field

The next item of text that needs to be replaced is the list of job duties in the second paragraph.

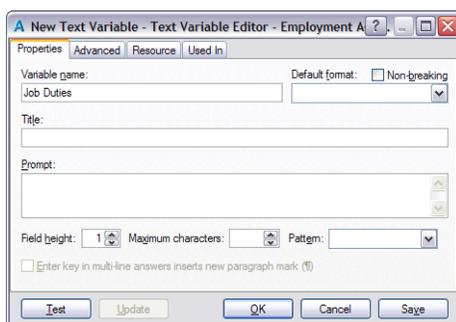
To replace the list of job duties with a HotDocs variable

- 1 Select **evaluating manuscripts, copy editing, and proofreading**. (Do not select the period.)
- 2 Click the **«» Variable Field** button. The **Variable Field** dialog box appears.
- 3 Select **Text** and type **Job Duties** in the **Variable** box. (Do not click **Replace** yet.)

Before you insert the variable into the template, you can test the variable to see how it will look to the user during the interview.

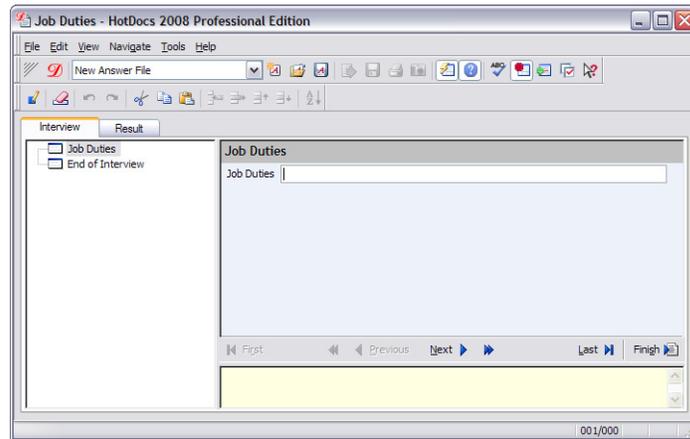
To test the variable

- 1 With the **Variable Field** dialog box still open, click the  **Edit Component** button (next to the variable name). The **Text Variable Editor** appears.



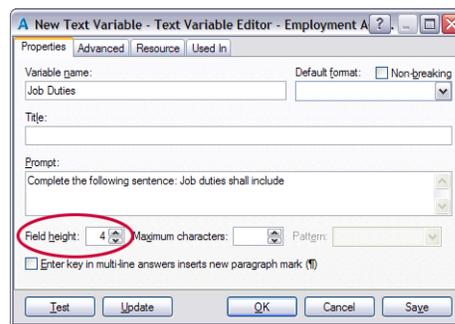
✓ You can have HotDocs arrange the template and the test assembly window so that you can view both simultaneously. To do this, resize the test assembly window to the height you want and click the **Arrange** button in the test assembly window toolbar.

- 2 Click **Test**. A test assembly window appears, with the question and answer field showing in the dialog pane.

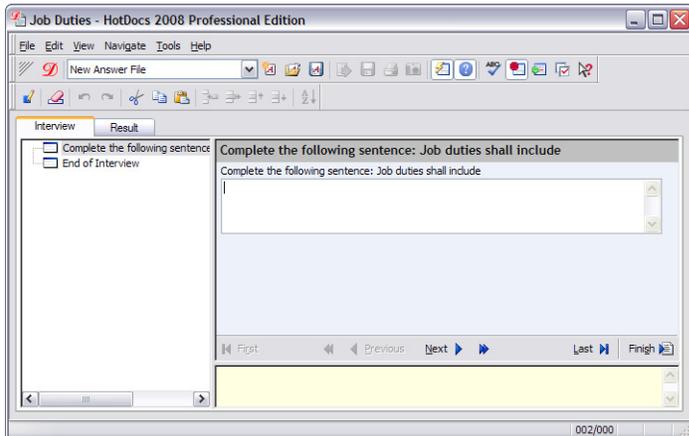


Note that the variable name is used to ask the question the user must answer. In this case, the name does not make a good prompt because it doesn't help the user's answer fit into the rest of the sentence. Also, the answer field shows only a single line, which could be a problem for an employee who must list many different job responsibilities.

- 3 With the test assembly window still displayed, place your cursor in the answer field and click the **Edit Component** button in the assembly window toolbar. The **Text Variable Editor** comes to the front.
- 4 In the **Prompt** box, type **Complete the following sentence: Job duties shall include.**
- 5 At the **Field height** box, click the up arrow until the number **4** appears. This will make the answer field four lines high.



- 6 Click **Update** to view the test assembly window again.



The new prompt appears and the answer field is large enough to hold a longer answer.

- 7 Choose **Close** (File menu) to close the test assembly window.
- 8 Click **OK** at the **Text Variable Editor**, and click **Replace Once** at the **Variable Field** dialog box.

You are now finished with this lesson. If you do not want to go on to Lesson 4 at this time, click the  **Save and Close** button to close the template.

If you are continuing on to Lesson 4, click the  **Save** button to save your work.

 Clicking **Update** allows HotDocs to redisplay the test assembly window with your changes. However, your changes aren't saved to the component file until you click **OK** or **Save** at the **Variable Editor**.

Lesson 4: Replace Numbers

Overview

This lesson teaches you how to replace numbers—both simple and computed. It also teaches you how to insert the same variable twice, but change how the answer will look each place it's merged in the assembled document.

Number variables are typically used to merge digits, figures, or numerals you frequently use in calculations, like monetary amounts or quantities. (You use Text variables to replace other number-based answers, such as telephone numbers.)

If you are continuing immediately from Lesson 3, skip the instructions for opening the template and proceed to “Replace Simple Numbers.”

If you closed the template at the end of Lesson 3, complete the following steps.

To open the Employment Agreement template you used in Lesson 3

- 1 Open your word processor and click the  HotDocs 2008 button. The HotDocs library appears.
- 2 Select **Employment Agreement** and click  **Edit**. The template appears, ready for you to edit.

Replace Simple Numbers

The first number to be replaced in the template is the monthly salary, *2,500*, located in the second paragraph.

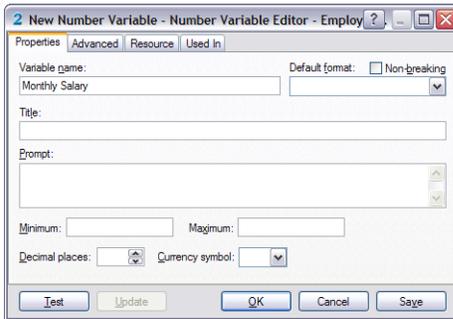
To replace the monthly salary with a Number variable

- 1 Select *2,500* (do not select the \$ character) and click the  **Variable Field** button. The **Variable Field** dialog box appears.
- 2 Select **Number** and, in the **Variable** box, type **Monthly Salary**.
- 3 Clear **Use default** and select **9,999.00** from the **Format** drop-down list. This forces HotDocs to include the cents in an answer. (If a user doesn't enter cents, zeros will be merged.)

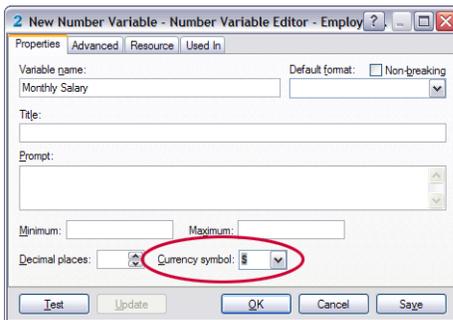


To help the user know what type of information to enter during the interview, you can specify a currency symbol as well as decimal places for cents.

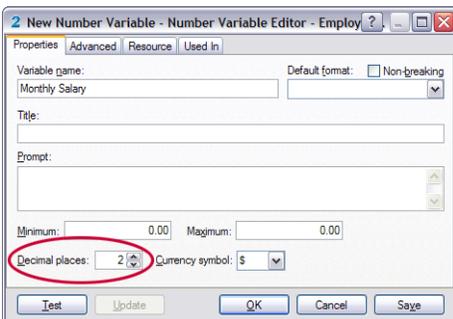
- 4 Click the  **Edit Component** button. The **Number Variable Editor** appears.



- 5 Click the **Currency symbol** drop-down button and select \$.



- 6 At the **Decimal places** box, click the up arrow until the number **2** appears.



- 7 Click **OK** at the **Number Variable Editor**, and click **Replace Once** at the **Variable Field** dialog box. The variable is inserted into the template.

 Currency symbols appear next to the prompt when the variable appears during the interview—they are not actually merged into the assembled document.

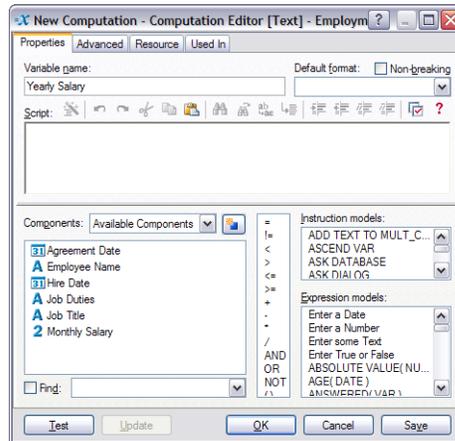
Replace Computed Numbers

The second number to be replaced is the yearly salary, also located in the second paragraph. You could replace the yearly salary with a Number variable; however, this would require the user to multiply the monthly salary by 12. To save the user time and reduce the chance of a mistake, you will replace the yearly salary with a Computation variable, which will do the multiplication for the user.

To replace the yearly salary with a Computation variable

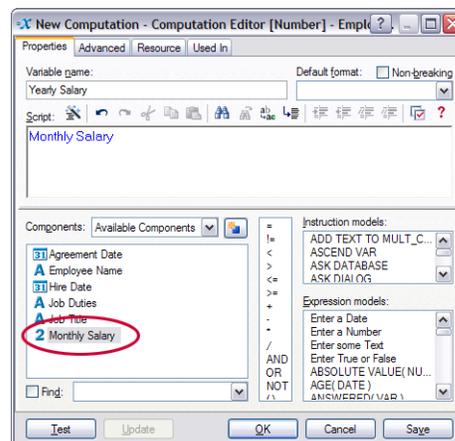
- 1 Select **30,000** (do not select the \$ character) and click the  **Variable Field** button. The **Variable Field** dialog box appears.

- 2 Select **Computation** and type **Yearly Salary** in the **Variable** box.
- 3 Clear **Use default** and select **9,999.00** from the **Format** drop-down list.
- 4 Click the  **Edit Component** button. The **Computation Editor** appears.

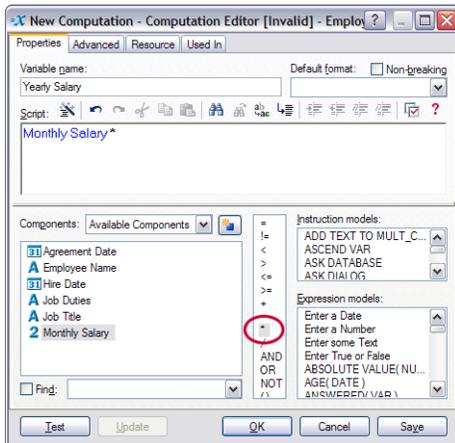


The instructions for HotDocs to multiply the monthly salary by 12 must be entered in the **Script** box. Since you don't know what the monthly salary will be at this point, you can use the variable in the equation, rather than a specific number value. Then, when the user completes the interview, whatever amount the user enters will be substituted and the yearly salary can be calculated.

- 5 In the **Components** list, select **Monthly Salary** and drag it to the **Script** box. (You may need to scroll through the list of components until you see it.)

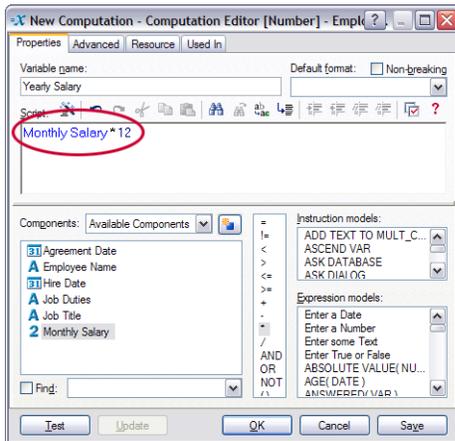


- 6 Press the spacebar to add a space and then drag the multiplication sign (the asterisk *) from the Operators list into the Script box following *Monthly Salary*.



✓ Items in a script must be separated by spaces.

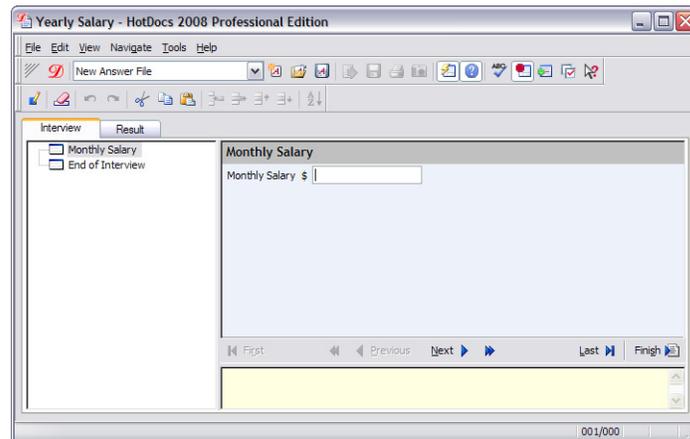
- 7 Press the spacebar again and type 12. (Make sure there is a space on both sides of the multiplication sign.)



Now that you have created the calculation, you can test it to make sure it produces the correct result.

To test the calculation

- 1 Click **Test**. A test assembly window appears.

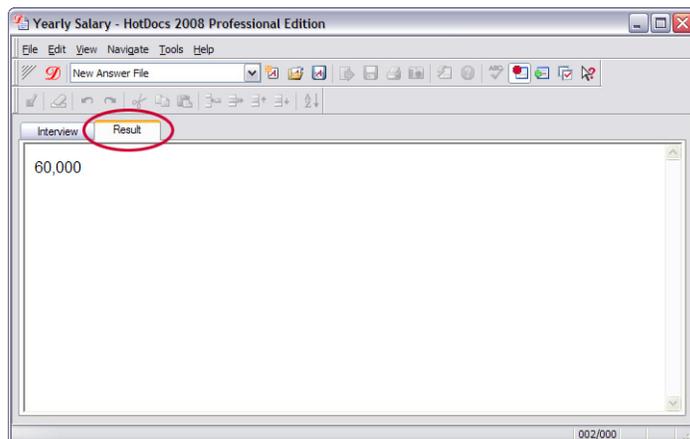


HotDocs prompts you to enter the amount of the monthly salary.

- 2 Type **5000** for the monthly salary and click the **Result** tab.

When you click the **Result** tab, HotDocs reformats the answer you entered so that it appears as *5,000.00*. It keeps you in the dialog so you can make sure it correctly interpreted the answer you entered.

- 3 Click the **Result** tab again. The computation result, *60,000*, is displayed.



- 4 Choose **Close** (**File** menu) to close the test assembly window. HotDocs asks if you want to save your answers.

- 5 Choose whether to save your answers, based on the following information:

- If you click **Save** or **Save As**, you will be prompted to specify an answer file name and title. Type **Test Answer File** in the **File name** box and accept the suggested **Title**. From this point on, this answer file will be used each time you test a variable or test assemble the template, and you will no longer be prompted to save your answers after each test. (You can, of course, choose a different answer file when you are at the assembly window.)

 For more information on using test answer files, see the HotDocs Help.

- If you click **Don't Save**, each time you test a variable or test assemble the template, you will use an empty, untitled answer file. You will also be prompted to save your answers each time you finish a test.

6 Click **OK** at the **Computation Editor**.

7 Click **Replace Once** at the **Variable Field** dialog box. The variable is inserted into the template.

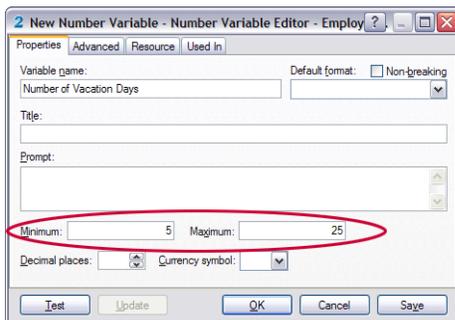
Set Minimum and Maximum Limits

The third number to be replaced, *TEN*, is in the third paragraph. All Hobble Creek Publishing employees receive at least five paid vacation days, but they can have no more than 25. You can use the **Minimum** and **Maximum** options to prevent users from entering a number smaller than five or larger than 25.

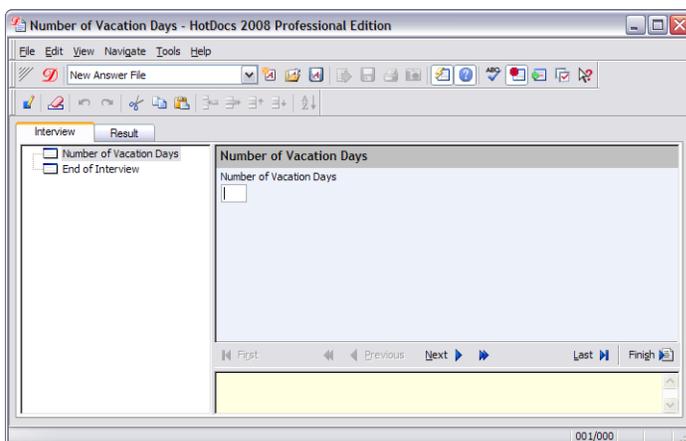
To set limits for a **Number variable**

- 1 Select **TEN** and click the **<<> Variable Field** button. The **Variable Field** dialog box appears.
- 2 Select **Number** and type **Number of Vacation Days** in the **Variable** box.
- 3 Click the **Edit Component** button. The **Number Variable Editor** appears.
- 4 Type **5** in the **Minimum** box and **25** in the **Maximum** box.

✔ Notice that, because you selected the template text *TEN*, HotDocs suggests *NINE* as the field-specific example format.



5 Click **Test**. A test assembly window appears.



6 Type **4** in the answer field and click the **Result** tab.

 You can use the  **Clear Answer** button in the test assembly window toolbar to clear answers.

The warning, *Please enter a number between 5 and 25*, appears.

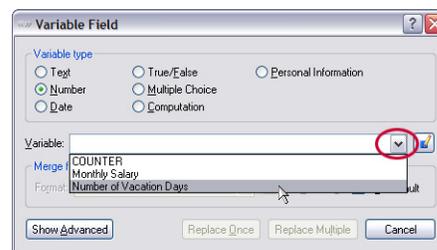
- 7 Click **OK** to return to the test assembly window.
- 8 Clear the answer and close the test assembly window.
- 9 Click **OK** at the **Number Variable Editor** and click **Replace Once** at the **Variable Field** dialog box. The variable is inserted into the template.

Insert Existing Variables

The next number, *10*, merges the number of vacation days in a different format. Instead of creating a new variable, you can use the variable you just created. Remember, the same variable can be merged several different places in a template, and the variable's format can be different for each field.

To replace the number with the variable you just created

- 1 Select **10** and click the **<> Variable Field** button.
- 2 Select **Number** and click the **Variable** drop-down button to see the list of variables.



- 3 Select **Number of Vacation Days** from the list.

The variable *Number of Vacation Days* appears in the **Variable** box. (Since *10* is the format HotDocs assumes by default, nothing appears in the **Format** box.)

- 4 Click **Replace Once** at the **Variable Field** dialog box.

On Your Own

The fourth paragraph also contains a number, *three*. Replace *three* with a Number variable named *Number of Seminar Days*.

You are now finished with this lesson. If you do not want to go on to Lesson 5 at this time, click the  **Save and Close** button to close the template.

If you are continuing on, click the  **Save** button to save your work.

Lesson 5: Create Conditional Text and Replace Multiple-Choice Text

Overview

This lesson teaches you how to make sections of text conditional, meaning the text will be included in the assembled document only if certain conditions exist. It also teaches you how to use Multiple Choice variables to give users several possible answers from which they can choose. And finally, this lesson teaches you how to include information for a variable that can help the user better understand what type of answer to enter.

If you are continuing immediately from Lesson 4, skip the instructions for opening the template and proceed to “Create Conditional Text.”

If you closed the template at the end of Lesson 4, complete the following steps.

To open the Employment Agreement template you used in Lesson 4

- 1 Open your word processor and click the  HotDocs 2008 button. The HotDocs library appears.
- 2 Select **Employment Agreement** and click  **Edit**. The template appears, ready for you to edit.

Create Conditional Text

In this template, the fourth paragraph contains information about the new employee’s privilege of attending training seminars throughout the year. The fifth paragraph contains information about the employee completing a trial period before he or she is officially hired by the company. However, not all Hobble Creek Publishing employees receive paid seminar days, nor are all new employees required to complete a trial period. In this template, you can make these paragraphs conditional, meaning they should be included in the finished document only when these conditions apply to the new employee.

To make text conditional, you use an IF instruction, which works by testing to see if one or more conditions are true. If so, the IF instruction merges the conditional text into the document.

The simplest way to create a condition is with a True/False variable, which asks a yes/no question. In this case, if the answer to the question is *yes*, the text is merged into the document.

To create an IF instruction

- 1 Select the fourth paragraph, including the blank line after it.

 When your paragraphs are separated by one or more blank lines (as they are in this template), you must consistently select either the blank lines above or the blank lines below each paragraph you make conditional. This will make the spacing in your documents uniform.

If you are using Word, the selected text should look like this:

In addition, «Employee Name» shall be allowed «Number of Seminar Days,nine» days each year to attend professional meetings or seminars, provided that he plans attendance at such meetings or seminars for minimum interference with the business of Hobble Creek Publishing.

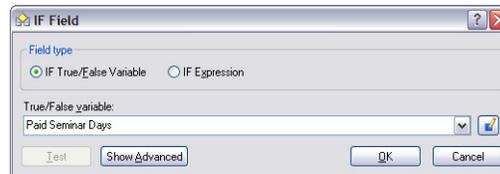
If you are using WordPerfect, the selected text should look like this:

In addition, «Employee Name» shall be allowed «Number of Seminar Days,nine» days each year to attend professional meetings or seminars, provided that he plans attendance at such meetings or seminars for minimum interference with the business of Hobble Creek Publishing.

- 2 Click the  IF Field button. The IF Field dialog box appears.



- 3 Make sure IF True/False Variable is selected and type Paid Seminar Days in the True/False variable box.



- 4 Click OK. The paragraph is now surrounded by an IF instruction.

«IF Paid Seminar Days»
 In addition, «Employee Name» shall be allowed «Number of Seminar Days,nine» days each year to attend professional meetings or seminars, provided that he plans attendance at such meetings or seminars for minimum interference with the business of Hobble Creek Publishing.
 «END IF»

On Your Own

Make the fifth paragraph conditional. Name the True/False variable you use for the condition *Trial Period* and remember to use the correct spacing between paragraphs. (Follow the instructions given for the *Seminar Days* paragraph.)

When you are finished, click the  Save button to save your work.

Replace Multiple-Choice Text

The fourth paragraph contains a pronoun (“provided that *he* plans attendance”). Pronouns are usually replaced with a Multiple Choice variable, which lets the user choose an answer from a list of pre-defined options.

To create a Multiple Choice variable that provides a pronoun for the employee

- 1 In the fourth paragraph, select **he** and click the **«» Variable Field** button. The **Variable Field** dialog box appears.
- 2 Select **Multiple Choice** and type **Employee Gender** in the **Variable** box.
- 3 Click the **Edit Component** button. The **Multiple Choice Variable Editor** appears.

The screenshot shows the 'New Multiple Choice Variable' dialog box. The 'Variable name' field contains 'Employee Gender'. The 'Default format' is set to 'Non-breaking'. The 'Option' column is empty. The 'Select' dropdown is set to 'One Only' and the 'Style' dropdown is set to 'Buttons'. The 'Test', 'Update', 'OK', 'Cancel', and 'Save' buttons are visible at the bottom.

- 4 Type **Male** in the first row of the **Option** column and **Female** in the second row.

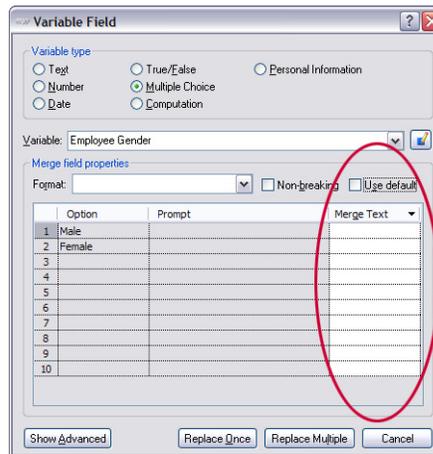
The screenshot shows the 'New Multiple Choice Variable' dialog box with the 'Option' column populated. The first row contains 'Male' and the second row contains 'Female'. The 'Option' column header is circled in red. The 'Select' dropdown is set to 'One Only' and the 'Style' dropdown is set to 'Buttons'. The 'Test', 'Update', 'OK', 'Cancel', and 'Save' buttons are visible at the bottom.

These are the options that will be presented to the user in the interview, but specific pronouns should be merged into the assembled document. To do this, you must enter merge text.

When you assign merge text to a Multiple Choice variable, you can assign it in one of two places: at the **Multiple Choice Variable Editor** (the dialog box you are now viewing), or at the **Variable Field** dialog box. When assigned at the **Multiple Choice Variable Editor**, the merge text is used every place the variable is used in the document. When assigned at the **Variable Field** dialog box, the merge text is used for that specific instance of the variable only. Since *Gender* must merge a different pronoun later in the template, you'll assign field-specific merge text.

- 5 Click **OK** to return to the **Variable Field** dialog box.

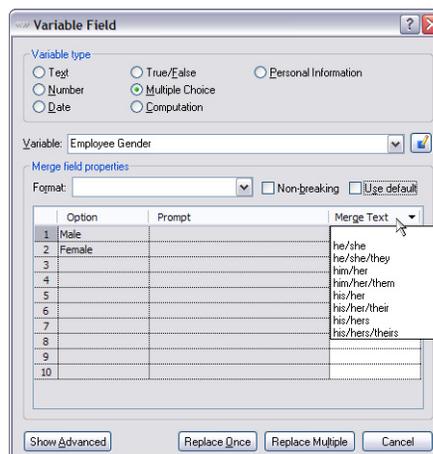
- 6 Clear Use default. The Merge Text column becomes active, allowing you to assign specific merge text for the field.



✓ HotDocs provides these lists of pronoun sets by default. They are available for every Multiple Choice variable you create. You can also create merge text of your own by typing the text in the **Merge Text** column.

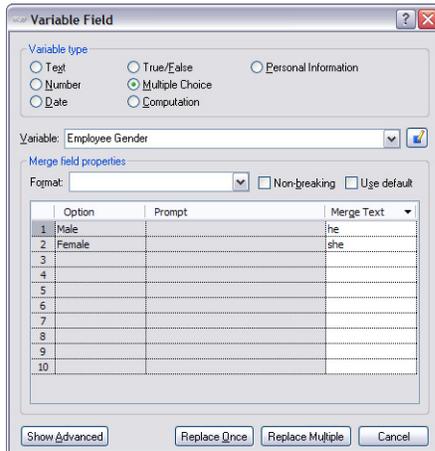
Normally when entering merge text, you would type each merge text option in the appropriate rows of the **Merge Text** column. However, HotDocs includes a list of common pronoun sets from which you can choose predefined options. When selected, these options are automatically merged in the column.

- 7 Click the Merge Text column heading. A list of pronoun sets appears.

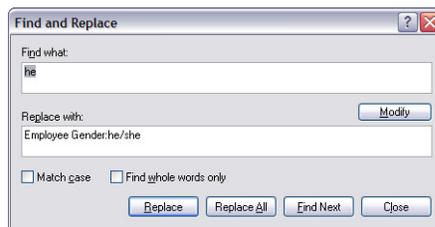


- 8 Select he/she.

The pronoun *he* appears in the first row of the **Merge Text** column and *she* appears in the second row.



9 Click **Replace Multiple**. The **Find and Replace** dialog box appears.



Even though there are no other instances of *he* in the template, choosing **Replace Multiple** allows you to see how you could search the entire template for instances and replace those instances.

10 Select **Find whole words only** and click **Replace All**. The pronoun is replaced with the variable. (Selecting **Find whole words only** prevents instances of *he* that are part of other words (such as *the*) from being replaced with the variable.)

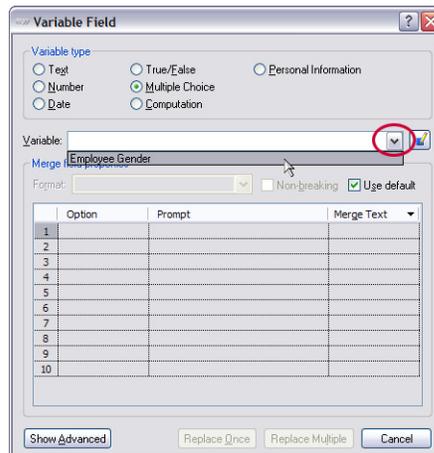
Choose Different Multiple Choice Merge Text

The fifth paragraph contains a different pronoun (“and upon *his* performance”) for the same person (the employee). Instead of creating a new variable to replace this pronoun, you can use the Multiple Choice variable you just created, but assign different merge text.

To replace another pronoun and change the merge text

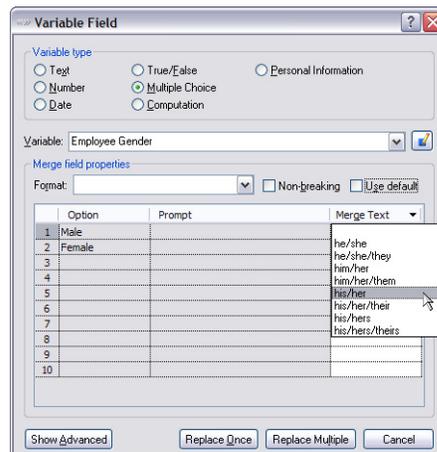
- 1 In the fifth paragraph, select *his* and click the **Variable Field** button. The **Variable Field** dialog box appears.
- 2 Select **Multiple Choice**.

- 3 Click the **Variable** drop-down button and select **Employee Gender** from the list.



- 4 Clear **Use default**.

- 5 Click the **Merge Text** column heading and select **his/her** from the list.



- 6 Click **Replace Once**. The variable is inserted into the template.

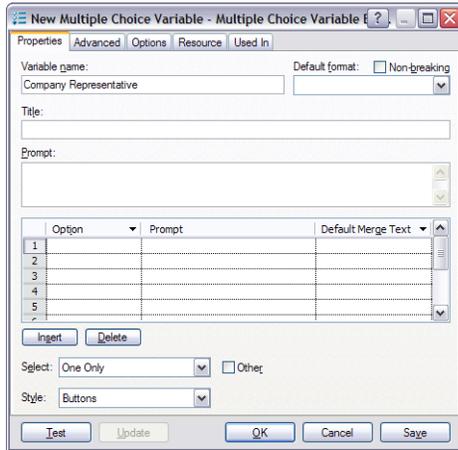
Add Resource Information

The final item of text that needs to be replaced in the template is the name of the company representative who signs the agreement. There are only three Hobbles Creek Publishing employees who sign Employment Agreements: Stephanie Walker, Ed Hall, and Kathryn Lee. You can present these options using a Multiple Choice variable.

To create a Multiple Choice variable and assign it a resource

- 1 Select **Stephanie Walker** (which is located near the end of the template) and click the **Variable Field** button. The Variable Field dialog box appears.
- 2 Select **Multiple Choice** and type **Company Representative** in the Variable box.

- 3 Click the  **Edit Component** button. The **Multiple Choice Variable Editor** appears.

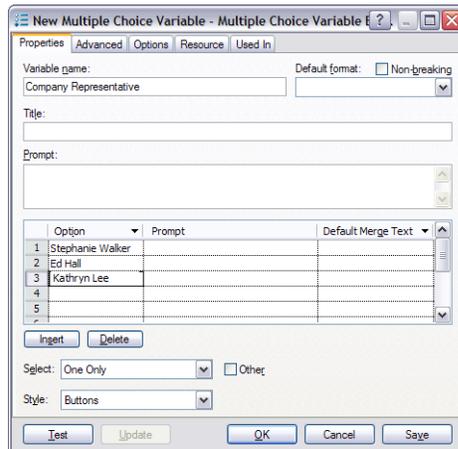


The screenshot shows the 'New Multiple Choice Variable' dialog box. The 'Variable name' is 'Company Representative'. The 'Option' column is empty. The 'Select' dropdown is set to 'One Only' and 'Style' is 'Buttons'.

Option	Prompt	Default Merge Text
1		
2		
3		
4		
5		

- 4 On separate rows in the **Option** column, type **Stephanie Walker**, **Ed Hall**, and **Kathryn Lee**.

If the cells in the **Default Merge Text** column are empty, the text from the **Option** column will be merged into the assembled document. Since you want the name you select to be merged into the document, do not enter any **Default Merge Text** for the options.

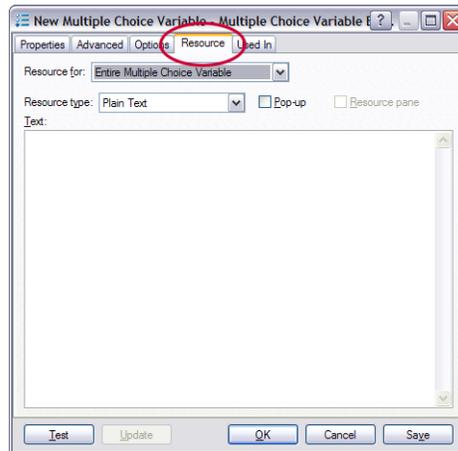


The screenshot shows the 'New Multiple Choice Variable' dialog box with the 'Option' column populated. The 'Default Merge Text' column is empty.

Option	Prompt	Default Merge Text
1	Stephanie Walker	
2	Ed Hall	
3	Kathryn Lee	
4		
5		

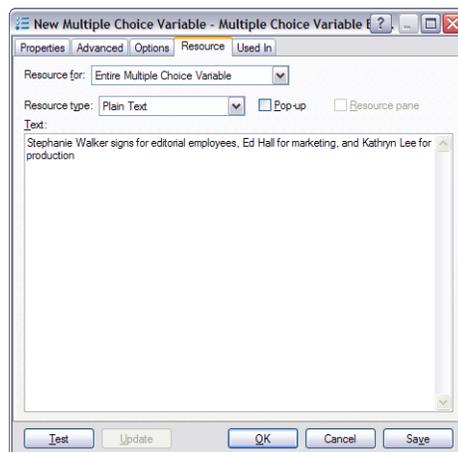
Users might not know which name to select for each agreement. You can add text to help them select the correct name.

- 5 Click the **Resource** tab. The window changes to show resource options.

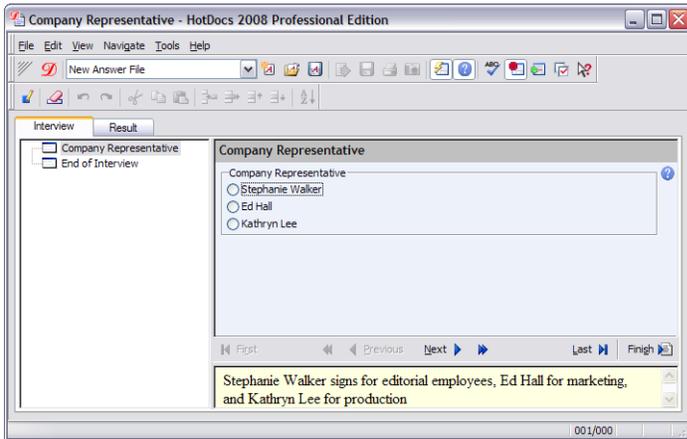


✓ To assign resources to each individual option (or company representative), select each option at the **Resource for** drop-down list and then enter the text.

- 6 Make sure **Entire Multiple Choice Variable** is selected at the **Resource for** drop-down list.
- 7 Make sure **Plain Text** is selected at the **Resource type** drop-down list.
- 8 In the **Text** box, type **Stephanie Walker signs for editorial employees, Ed Hall for marketing, and Kathryn Lee for production** in the **Text** box.



- 9 Click **Test**. A test assembly window appears and the information you just typed appears in the resource pane.



 If the resource pane isn't showing, select **Dialog Resource Pane** at the **View** menu.

- 10 Close the test assembly window and click **OK** at the **Multiple Choice Variable Editor**.

- 11 Click **Replace Once** at the **Variable Field** dialog box. The variable is inserted into the template.

You are now finished with this lesson. If you do not want to go on to Lesson 6 at this time, click the  **Save and Close** button to close the template.

If you are continuing on to the next lesson, click the  **Save** button to save your work.

Lesson 6: Test Assemble a Template

Overview

This lesson teaches you how to test assemble a document while you are editing a template. This allows you to make sure the template is working the way you want.

When you test assemble a template, HotDocs displays an interview for the template. This interview represents the questions users must answer in order to assemble a custom document.

If you are continuing immediately from Lesson 5, skip the instructions for opening the template and proceed to “Test Assemble the Document.”

If you closed the template at the end of Lesson 5, complete the following steps.

To open the Employment Agreement template you used in Lesson 5

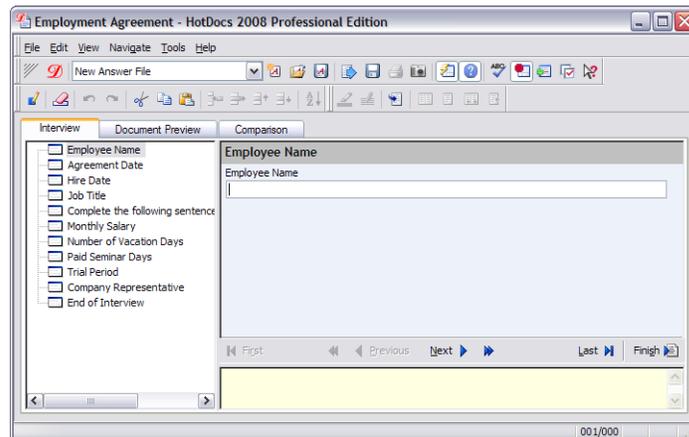
- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs library appears.
- 2 Select **Employment Agreement** and click  **Edit**. The template appears, ready for you to edit.

Test Assemble the Document

You will now test assemble the template.

To test the Employment Agreement template

- 1 At the template window, click the  **Assemble** button in the HotDocs toolbar. A test assembly window appears, showing a list of variables in the left pane. These variables appear in what are called *default* dialogs. Each question appears by itself in the right pane.

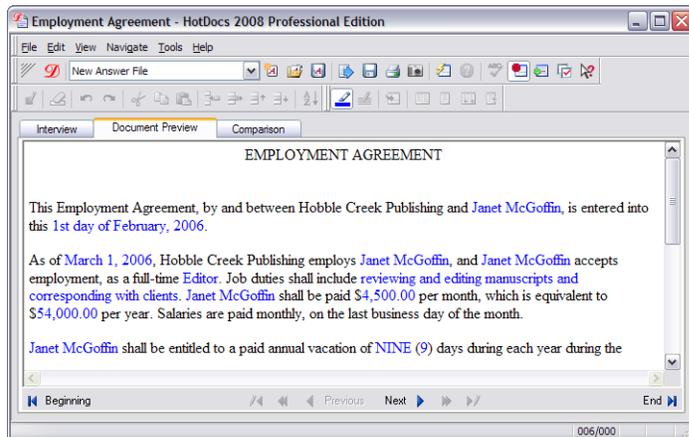


-  If you saved your answers in a test answer file in Lesson 4, you will notice that answers you have entered are now merged into the answer fields. If you want to use a new answer file for this portion of the tutorial, click the  **New Answers** button.

- At each dialog, enter an answer and click the **Next** button to go to the next dialog.

Notice that each default dialog contains a single variable, or question. In the next lesson, you will group multiple variables together in a custom dialog. Presenting groups of questions makes it easier for users to complete the interview.

- After you have provided all of the answers, click the **Document Preview** tab. A preview of the assembled document appears, allowing you to make sure answers were merged correctly.



- Choose **Close** (File menu) to close the test assembly window. You are returned to the template.

You are now finished with this lesson. If you do not want to go on to Lesson 7 at this time, click the **Save and Close** button to close the template.

Lesson 7: Create Custom Dialogs

Overview

This lesson teaches you how to group several related variables together in dialogs.

As you learned in the last lesson when you test assembled your template, HotDocs automatically asks all of the required questions in the template—with each question in its own default dialog. If this method of gathering information seems adequate, you do not need to create custom dialogs for the template. However, custom dialogs allow you to group related questions together and control the order in which questions are asked.

If you are continuing immediately from Lesson 6, skip the instructions for opening the template and proceed to “Create a Custom Dialog.”

If you closed the template at the end of Lesson 6, complete the following steps.

To open the Employment Agreement template you used in Lesson 6

- 1 Open your word processor and click the  HotDocs 2008 button. The HotDocs library appears.
- 2 Select **Employment Agreement** and click  **Edit**. The template appears, ready for you to edit.

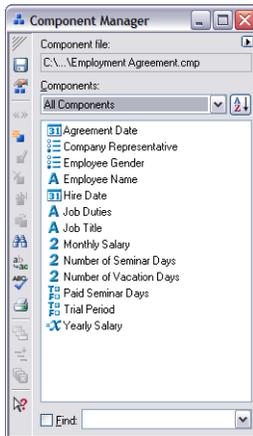
Create a Custom Dialog

How you group variables into dialogs depends on the template and your own preferences. Generally, you begin with the most basic information. In the *Employment Agreement*, the most basic information pertains to the employee.

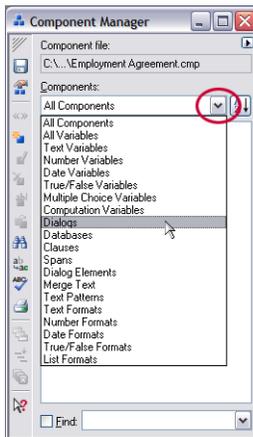
To create a custom dialog, you use Component Manager. Component Manager is a tool that lets you work with the different components and properties you use in your template.

To create a custom dialog containing employee information

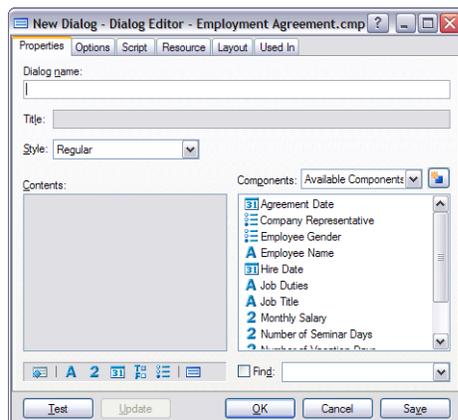
- 1 Click the  **Component Manager** button in the HotDocs toolbar. The **Component Manager** window appears.



- 2 Click the **Components** drop-down button and select **Dialogs** from the list.



- 3 Click the  **New Component** button. The **Dialog Editor** appears.

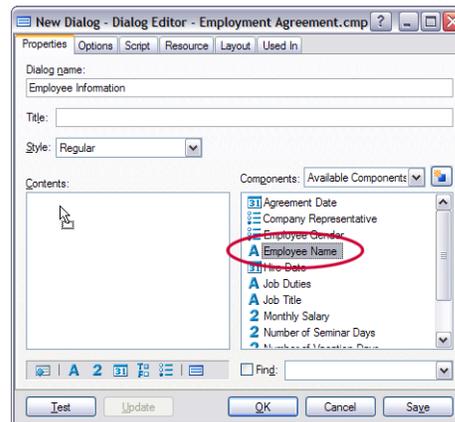


- 4 Type **Employee Information** in the **Dialog name** box.

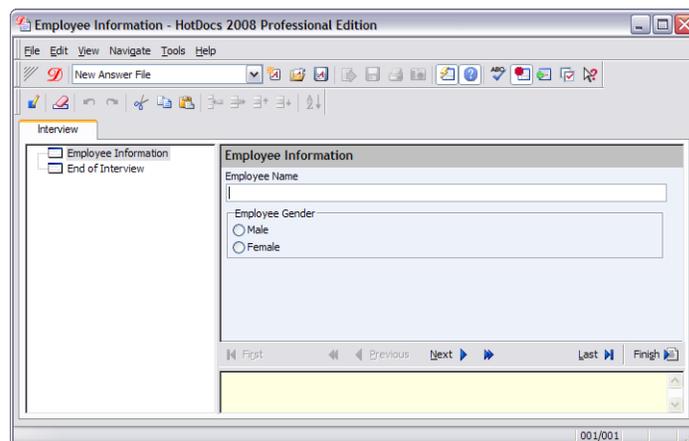
 To display **Component Manager** next to the word processor window, adjust the width of **Component Manager** to the size you want and click the  **Arrange** button in the Component Manager toolbar. HotDocs arranges the windows so you can work in both simultaneously.

✓ When you add a variable to a dialog, the variable becomes linked to the dialog. This means it cannot be used in any other dialogs unless you clear the **Link variables to this dialog** option at the **Options** tab. (See the HotDocs Help for more information about sharing variables between dialogs.) Linked variables appear grayed at the bottom of the **Components** list.

- 5 Drag **Employee Name** from the **Components** list into the **Contents** box.



- 6 Drag **Employee Gender** into the **Contents** box below **Employee Name**.
- 7 Click **Test**. A test assembly window appears.



The dialog looks the way it will when it is displayed during the interview.

- 8 Close the test assembly window, and then click **OK** at the **Dialog Editor**. The dialog *Employee Information* is now complete.

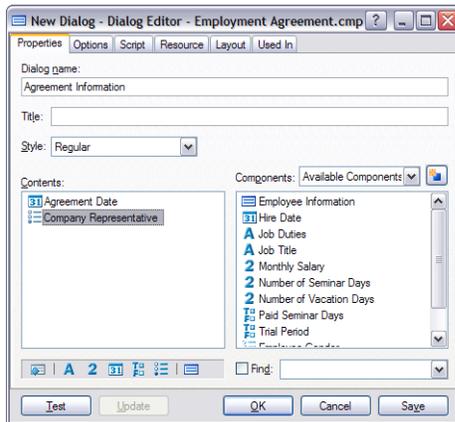
Create the Second Dialog

Another group of variables relates to information about the agreement.

To create a dialog containing agreement information

- 1 At Component Manager, click the **New Component** button. The **Dialog Editor** appears again.
- 2 Type **Agreement Information** in the **Dialog** name box.

3 Drag **Agreement Date** and **Company Representative** into the **Contents** box.



4 Click **OK**.

On Your Own

Create a dialog named *Job Information* that contains the variables *Hire Date*, *Job Title*, and *Job Duties*. (Follow the instructions given for the *Agreement Information* dialog, earlier.) Click **OK** at the **Dialog Editor** when you are finished.

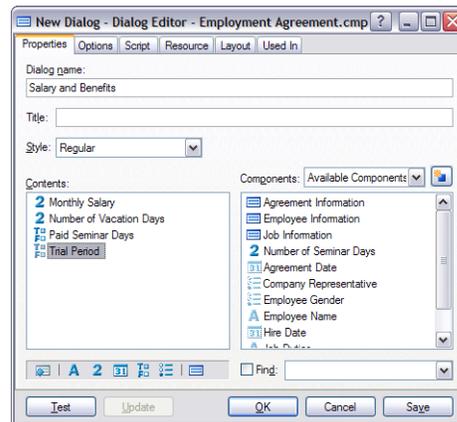
Group True/False Variables in a Dialog

Two of the remaining variables are True/False variables (*Paid Seminar Days* and *Trial Period*). By default, True/False variables appear in a dialog as yes/no questions. However, you can group True/False variables into single-selection or multiple-selection lists in a dialog. Single-selection lists allow users to choose one option, whereas multiple-selection lists allow users to choose multiple options.

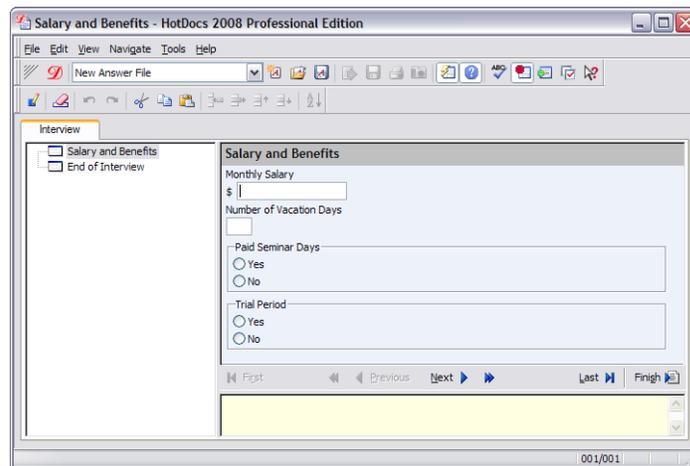
To create a dialog and group its True/False variables

- 1 Create a new dialog named **Salary and Benefits**.
- 2 Drag the following variables into the **Contents** box:
 - **Monthly Salary**
 - **Number of Vacation Days**
 - **Paid Seminar Days**

- Trial Period



3 Click Test. A test assembly window appears.

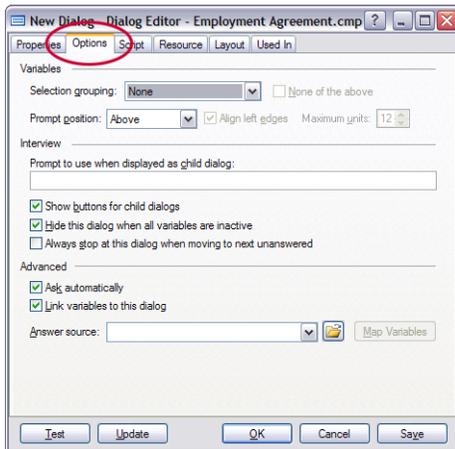


The two True/False variables have created two *yes/no* questions. If you select *Yes* for one of these questions, the corresponding conditional paragraph will be included in the assembled document.

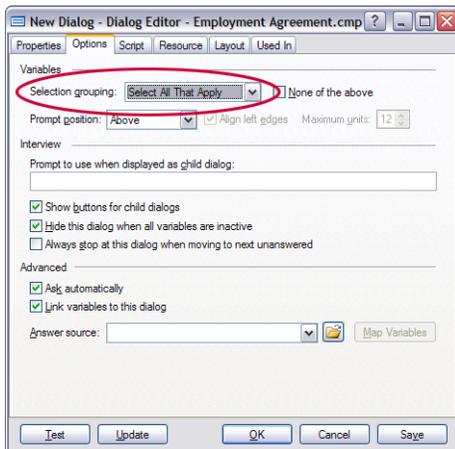
This dialog works as it is; however, you can make it more efficient by displaying the two variables in a multiple-selection list.

4 In the interview outline, click the **Salary and Benefits** dialog icon, and then click the  **Edit Component** button. The **Dialog Editor** comes to the front.

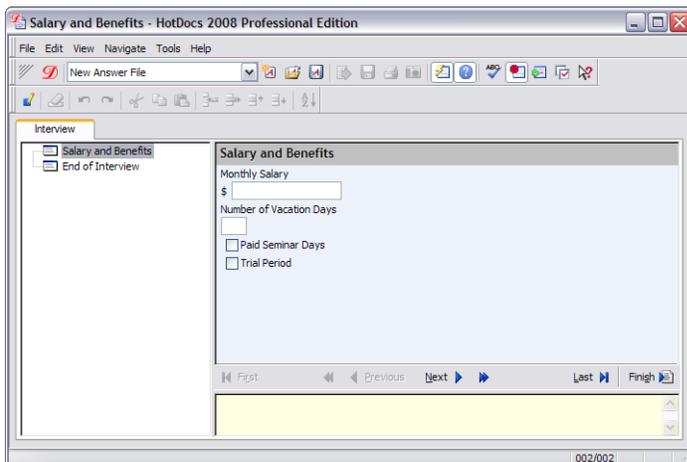
- 5 Click the **Options** tab. The window changes to show several custom options.



- 6 Click the **Selection grouping** drop-down button and select **Select All That Apply**.



- 7 Click **Update** to display the test assembly window again. The appearance of the dialog has changed—both True/False variables are represented by check boxes.



- 8 Close the test assembly window and return to the **Dialog Editor**.

✓ Another **Selection grouping** option, **Select One**, presents the True/False variables in a list where only one option can be chosen.

9 Click OK to close the **Dialog Editor**.

There is one remaining variable in the template, *Number of Seminar Days*. It should be asked only if *Paid Seminar Days* is true, so it can appear in its own dialog. Since HotDocs automatically asks variables not assigned to custom dialogs, there is no need to create a custom dialog for it.

You have finished automating the *Employment Agreement* template. In the next part of the lesson, you will test your automated template.

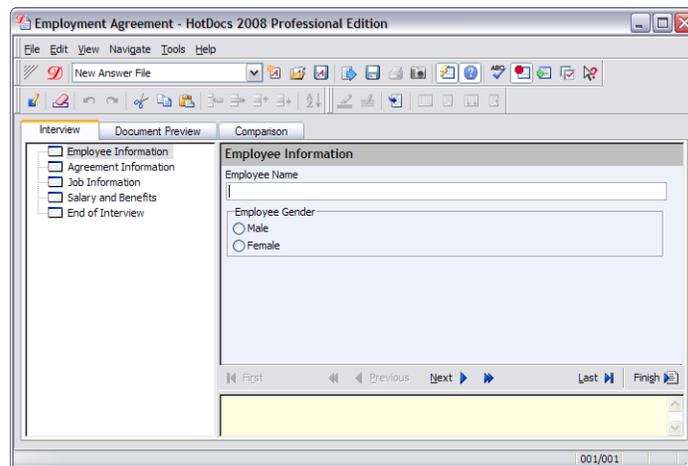
10 Click the X in the upper-right corner of **Component Manager** to close it.

11 Click the  **Save** button to save the changes you have made to the template and component file.

Test the Finished Template

To test the finished template and see the dialogs you created

1 Click the  **Assemble** button in the HotDocs toolbar. The test assembly window appears, showing the first dialog, *Employee Information*.

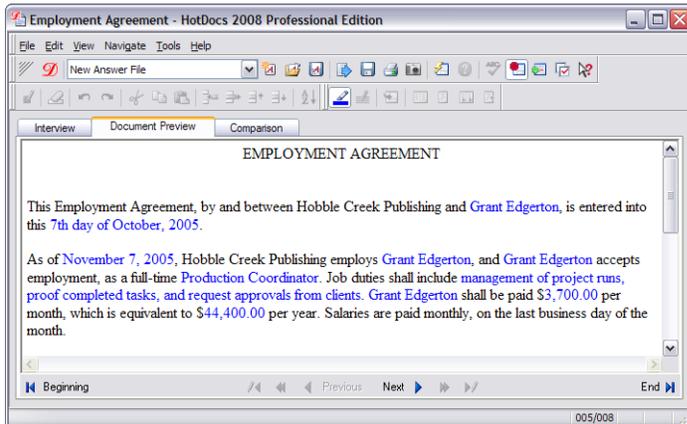


2 Answer the questions and click  **Next**. The *Agreement Information* dialog appears.

3 Continue answering questions as dialogs appear.

4 After you reach the *End of Interview* dialog, click the **Document Preview** tab to preview the assembled document. If you need to make changes to any answers, you can either change your answer right in the document (Word users), or you

can click the **Interview** tab again and make the change there (WordPerfect users).



- 5 Close the test assembly window.
- 6 At the template, click the  **Save and Close** button. The template is saved and closed.

You are now finished with this lesson.

In the next lesson, Lesson 8, you will create a clause library.

Lesson 8: Add a Clause Library to a Template Library

Overview

This lesson shows you how to create the *Agreement Clauses* clause library you used in Lesson 1.

A clause library is a collection of clauses, or blocks of text. Grouping clauses in a library allows users to select, organize, and insert any number of clauses into the document. HotDocs then proceeds to prompt users for any answers to variables contained in the clauses.

You can create a clause library either at a HotDocs library or directly in a template. For this lesson, you will create a clause library using text in a template.

When you create a clause library from template text, the first step is to put the text of all of the clauses you want in the library into a single HotDocs template. The next step is to replace any variable text with HotDocs variables. For this lesson, these two steps have already been completed.

The remaining steps (which you *will* complete in this lesson) include creating the clause components, adding them to the clause library, and then adding the clause library to the tutorial library. (For more information on creating template files and inserting variables, see Lessons 2 through 7.)

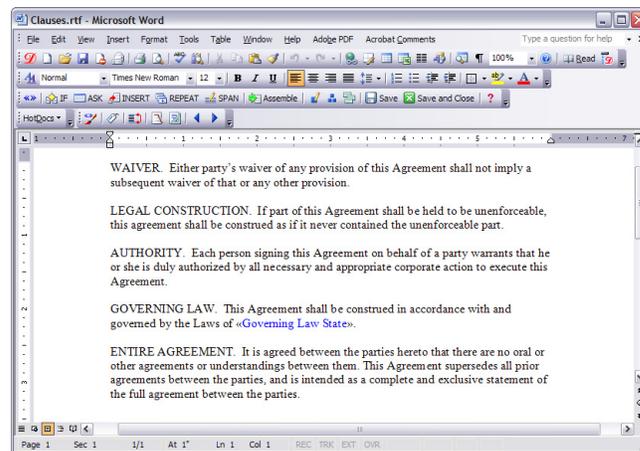
To open the Agreement Clauses template for editing

1 Open your word processor and click the  HotDocs 2008 button. The HotDocs library appears.

2 From the **HotDocs Tutorial Files** folder, select **Agreement Clauses**.

 If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library** (**File** menu), select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete step 2.

3 Click  **Edit**. The template appears, ready for you to edit.



The next step is to divide the text into HotDocs clauses. (You will notice that the *Governing Law* clause contains one variable, *Governing Law State*.)

Divide the Text into HotDocs Clauses

A HotDocs clause is a component that stores text.

To create a clause for the first paragraph and add it to the clause library

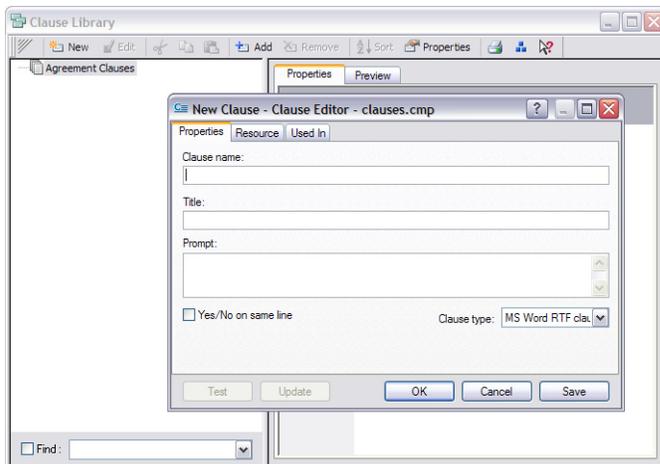
- 1 Select the first paragraph. If you are automating a Word template, select just the paragraph and *not* the blank line below it.

WAIVER. Either party's waiver of any provision of this Agreement shall not imply a subsequent waiver of that or any other provision.

However, if you're automating a WordPerfect template, select the entire blank line below the paragraph.

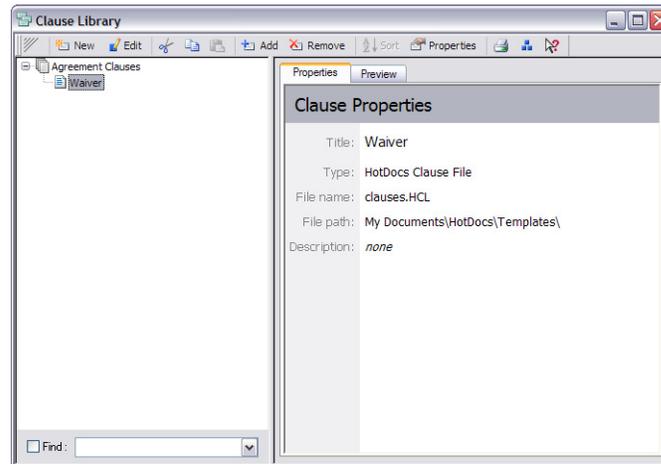
WAIVER. Either party's waiver of any provision of this Agreement shall not imply a subsequent waiver of that or any other provision.

- 2 Click the  **Clause Library** button in the HotDocs toolbar. The **Clause Library** window as well as the **Clause Editor** appear.



✓ When creating clauses, HotDocs automatically creates a clause library that has the same file name as the template in which you are creating clauses. This file has an .HDL file name extension, just like a template library.

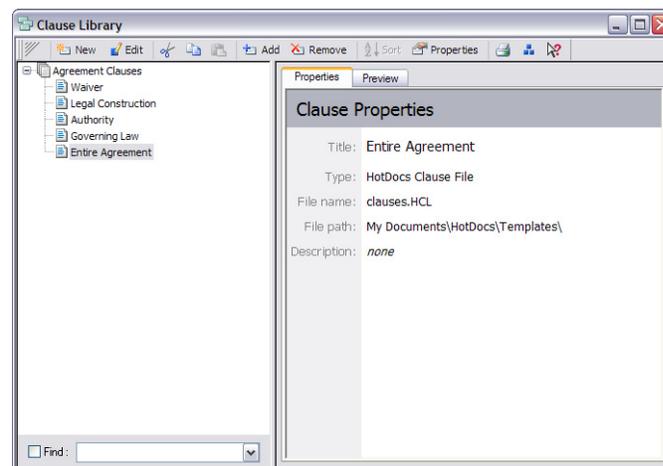
- 3 Type **Waiver** in the **Clause name** box and click **OK**. The clause is added to the clause library.



On Your Own

Create clauses and add them to the clause library for each of the remaining paragraphs. Name the clauses *Legal Construction*, *Authority*, *Governing Law*, and *Entire Agreement*, respectively. Even though the clause library is still displayed, you must click the  **Clause Library** button in the HotDocs Edit toolbar each time you want to add a clause to it. (Follow the instructions given for the *Waiver* clause, earlier.)

When you are finished, the clause library should look like this:



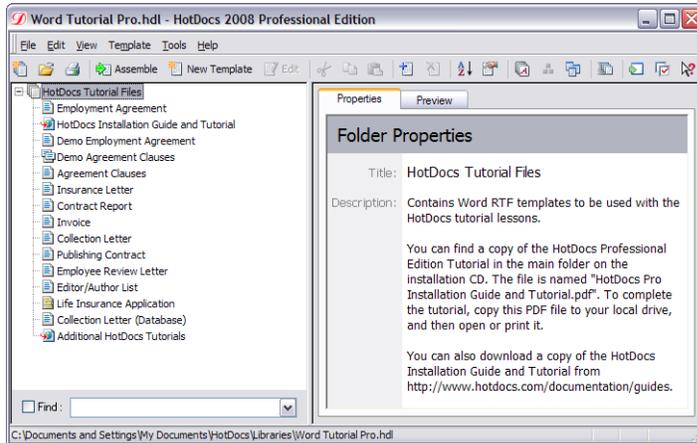
Close the clause library (click the **X** in the upper-right corner) and then click the  **Save and Close** button to close the template.

Add a Clause Library to a HotDocs Library

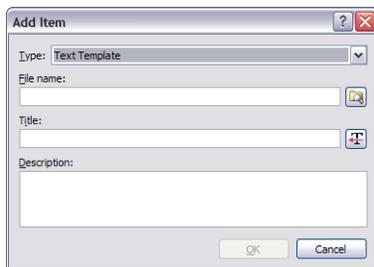
The next step is to add the clause library to the HotDocs library so you can insert the clauses into any word processor document.

To add the clause library you just created to the HotDocs library

- 1 Click the  **HotDocs 2008** button in your word processor toolbar. The HotDocs library comes to the front.
- 2 At the HotDocs template library, click the **HotDocs Tutorial Files** folder.



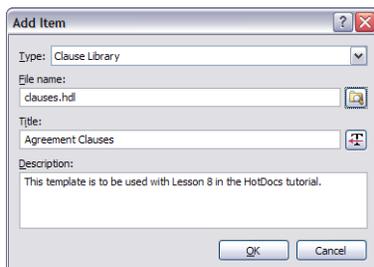
- 3 Click the  **Add Item** button in the template library toolbar. The **Add Item** dialog box appears.



- 4 Click the **Type** drop-down button and select **Clause Library**.
- 5 At the **File name** box, click the  **Browse** button to see a list of clause library files in the default *Templates* folder.

The clause library uses the same file name as the template from which it was created.

- 6 Select **clauses.hdl** and click **OK**. The **Add Item** dialog box appears again, showing the name of the clause library in the **File name** box and **Agreement Clauses** in the **Title** box.



- 7 Click **OK**. The clause library, *Agreement Clauses*, is added to the *HotDocs Tutorial Files* folder.

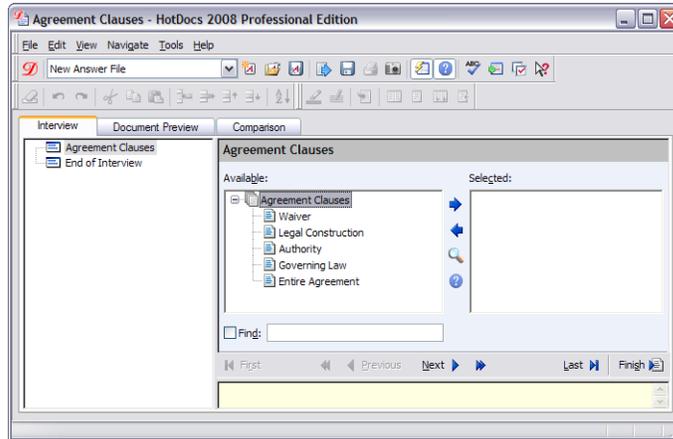
 You can also add a clause library to a template using the  **INSERT Field** button. Then, when the user assembles a document from the template, at the point where the clause library is inserted in the template, HotDocs displays the clause library and the user can choose which clauses to insert. (You will use this kind of clause library in “Lesson 13: Incorporate Clause Selection in a Template” on page 91.)

Test the Clause Library

You can test the clause library by using it to insert clauses.

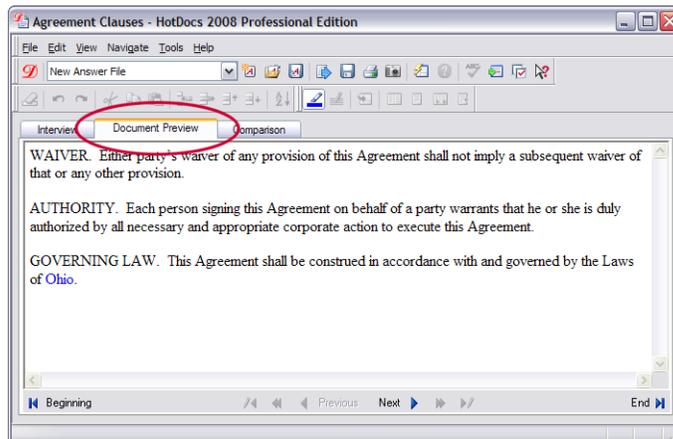
To test the finished clause library

- 1 At the template library, select the **Agreement Clauses** clause library.
- 2 Click  **Assemble**. The **Answer File** dialog box appears.
- 3 Accept the new answer file and click **OK**. The assembly window appears with the clause library showing in the dialog pane.



 Order the clauses by selecting a clause and dragging it up or down in the **Selected** list.

- 4 Select a clause from the **Available** list and click the  **Select** button to place it in the **Selected** list.
- 5 Add as many clauses as you want. When you are finished adding clauses, click  **Next**.
- 6 If you selected *Governing Law*, enter a state name at that dialog and click  **Next** to advance to the *End of Interview* dialog.
- 7 Click the **Document Preview** tab. HotDocs shows you a preview of the assembled document.



- 8 Close the assembly window. If prompted to save your answers and a copy of the assembled document, click **Don't Save**.

You are now finished with this lesson.

Lesson 9: Use Instruction and Expression Models

Overview

All new Hobbles Creek Publishing employees receive health insurance. *When* their insurance becomes effective, however, depends on which day of the month they are hired—employees who are hired on the first day of the month receive immediate coverage, while those who are hired *after* the first of the month must wait until the first day of the next month for coverage.

In this lesson, you will use the HotDocs scripting language to calculate the date when a new employee qualifies for insurance benefits.

When using the HotDocs scripting language, you must use certain instruction and expression keywords that HotDocs can recognize. *Instruction* keywords tell HotDocs to perform some sort of action, while *expression* keywords are predefined formulas you use to retrieve special values.

To help you use the correct keywords, you can use instruction and expression models. Many of these models include *placeholders*, which represent values that must be replaced for the script to work correctly. These values include literal values (or values you provide when writing the script), variables (which let the user provide the required value), and other models (which let you fine-tune your script).

There are two methods for writing a computation script:

- Using the instruction, expression, and component lists at the bottom of the **Computation Editor**, you can drag models and components from these lists and drop them into the script. This is useful if you are learning the scripting language, since it provides accuracy and helps you learn the syntax for using instructions and expressions. (You will use this method in this lesson.)
- You can type the script directly in the **Script** box. While doing this, you can access auto-complete lists that let you type a partial keyword or component name, then press a combination of keys to have HotDocs display a list of keywords or components it thinks you're trying to enter. You can then select the keyword you want and HotDocs will merge it into the script. This method is useful for those more familiar with the scripting language. (You will use this method in later lessons.)

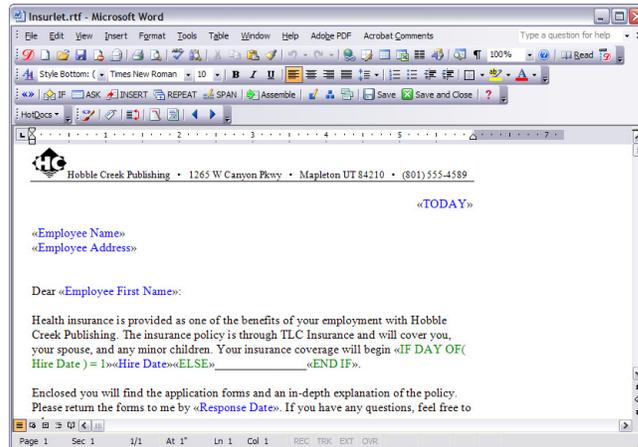
In this lesson, you will work with the *Insurance Letter* template.

To open the Insurance Letter template for editing

- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs library appears.
- 2 From the **HotDocs Tutorial Files** folder, select **Insurance Letter**.

 If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library** (**File** menu), select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete step 2.

- 3 Click  **Edit**. The template appears, ready for you to edit.



Most of the automation is complete, but the date the insurance becomes effective must be calculated.

Insert an Expression Model in a Computation Script

At the end of the first paragraph, an IF expression is used to merge the date an employee's insurance coverage becomes effective. If the employee was hired on the first day of the month, the coverage begins on the hire date. Otherwise, the coverage begins on the first day of the month following the hire date.

This second date is currently represented by a blank line. You must replace this line with a Computation variable that returns this date. You will use several expression models to create the variable.

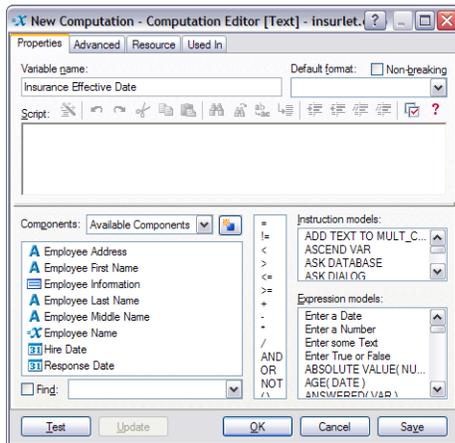
To insert a Computation variable that produces a date

- 1 Select the line between «ELSE» and «END IF».

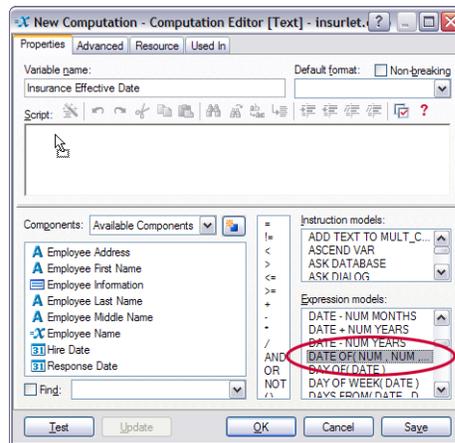
Health insurance is provided as one of the benefits of your employment with Hobbie Creek Publishing. The insurance policy is through TLC Insurance and will cover you, your spouse, and any minor children. Your insurance coverage will begin «IF DAY OF(Hire Date) = 1»«Hire Date»«ELSE» «END IF».

- 2 Click the  **Variable Field** button. The Variable Field dialog box appears.
- 3 Select **Computation** and type **Insurance Effective Date** in the Variable box.

- 4 Click the  **Edit Component** button. The **Computation Editor** appears.



- 5 Scroll through the **Expression models** list until you see **DATE OF(NUM, NUM, NUM)**. (Expression models are arranged alphabetically.)
- 6 Click the model, drag it into the **Script** box, and release the mouse button. The expression is inserted in the **Script** box.



 To make your work in this part of the tutorial easier, you can click and drag the lower-right corner of the **Computation Editor** to make it larger.

 The **DATE OF (NUM, NUM, NUM)** model lets you examine the different portions of a date (*day, month, and year*) and assign specific values to each portion. The resulting value can be merged into the assembled document.

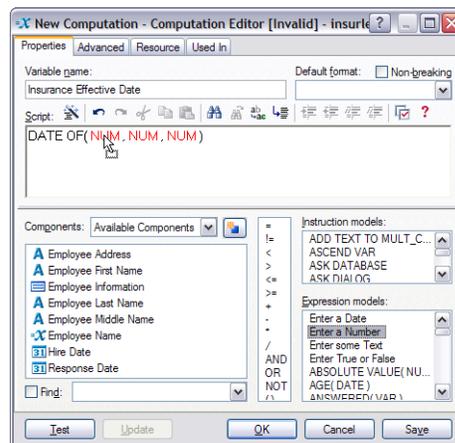
Replace a Script Placeholder with a Value

To help you identify which portion of the script you must replace with values, HotDocs assigns a different color to the placeholders. As you see, the **DATE OF** expression has three placeholders for required number values, which represent the day, month, and year, respectively.

You know the insurance needs to always start on the *first* day of the month, so you want the first **NUM** placeholder to be replaced with *1*. One way to enter a number in a computation is to use the **Enter a Number** expression model.

To replace the DATE OF expression with a Number

- 1 From the list of **Expression models**, drag **Enter a Number** (located near the top of the list) and drop it onto the first NUM placeholder.



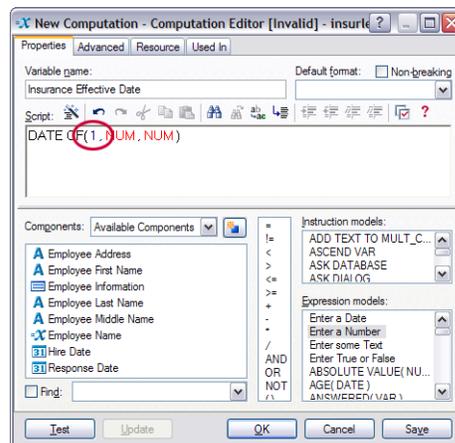
✓ When you use numbers in scripts, they must be formatted a certain way. Using the **Enter a Number** model lets you type a number and have HotDocs format it for you. This reduces the chance of error when HotDocs tries to process your script.

The **Enter a Number** dialog box appears.



- 2 Type **1** and click **OK**.

The number **1** replaces the first NUM placeholder.



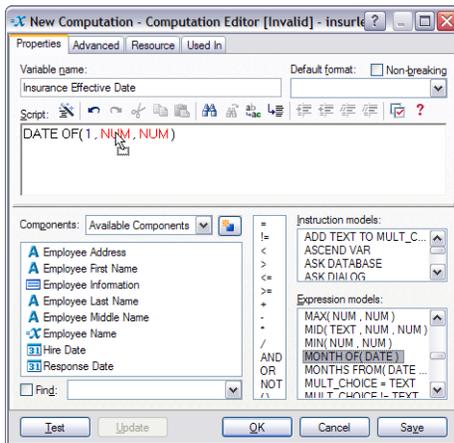
Replace Placeholders with Other Expressions

You can also replace placeholders with other expressions.

The second NUM placeholder represents the numeric equivalent of a month. You cannot simply enter a number like you did for the day, because the month you want depends on the hire date. Instead, you should use the **MONTH OF(DATE)** expression, which returns the numeric equivalent of a month.

To replace the placeholders in a computation with other expression models

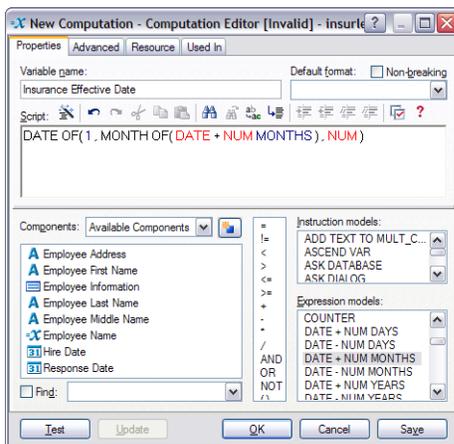
- 1 Drag **MONTH OF (DATE)** from the **Expression models** list onto the second **NUM** placeholder.



You want the **MONTH OF** expression to return the number for the month following the month of hire. In other words, you want to add one month to the hire date. The **DATE + NUM MONTHS** expression helps you do this.

- 2 Drag **DATE + NUM MONTHS** onto the **DATE** placeholder that is in the parentheses following **MONTH OF**.

Your script should look like this:



Replace a Placeholder with a Variable

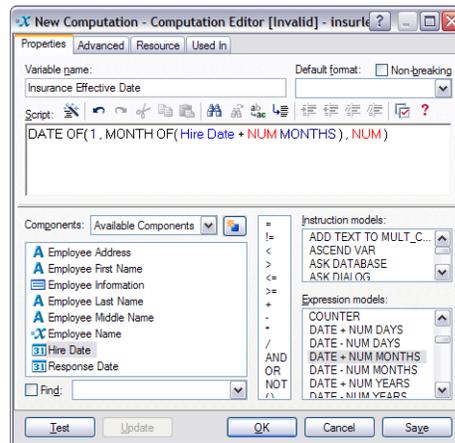
Placeholders are often replaced with HotDocs variables. The **DATE + NUM MONTHS** expression has two placeholders—**DATE** and **NUM**. (*MONTHS* is a keyword that remains in the finished script.) The **DATE** placeholder needs to be replaced with the date the employee was hired.

✓ The **MONTH OF (DATE)** model returns a number value that corresponds with the month portion of the date the user enters. For example, if a user types *July 1, 2007*, this expression returns a value of 7, since July is the 7th month of the year.

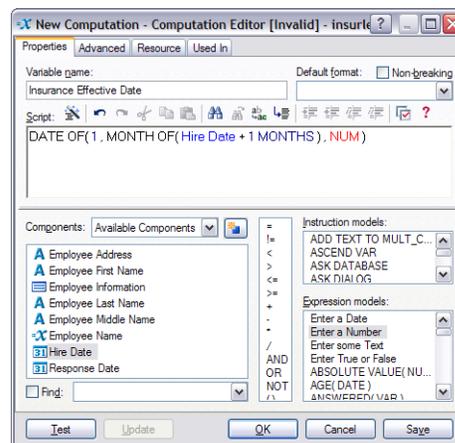
✓ The **DATE + NUM MONTHS** model takes a date and adds a specified number of months to it. For example, if a user's answer to a date question is *March 17, 2007* and one month needs to be added to it, then this expression returns the result of *April 17, 2007*.

To replace a placeholder in a computation with a variable

- 1 Drag **Hire Date** from the **Components** list onto the **DATE** placeholder.



- 2 Using the **Enter a Number** expression model, replace the **NUM** placeholder in the **DATE + NUM MONTHS** expression with the number **1**. HotDocs will add one month to the *Hire Date*.

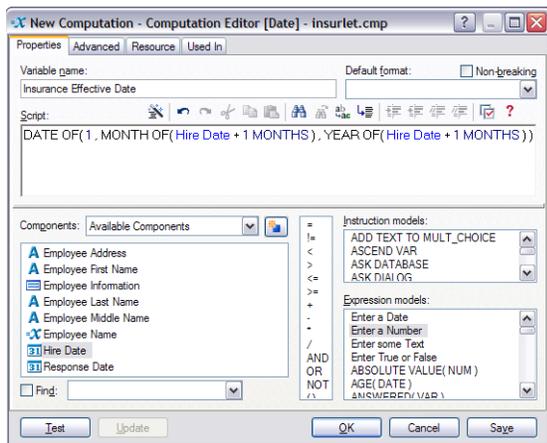


On Your Own

✓ The **YEAR OF (DATE)** model operates similar to **MONTH OF (DATE)** model. Here it is used to make sure the correct year is returned for the new date.

Replace the remaining **NUM** placeholder in the **DATE OF (NUM, NUM, NUM)** expression with **YEAR OF(Hire Date + 1 MONTHS)**. (Follow all of the instructions for replacing the second **NUM** placeholder in the expression given earlier.)

When you are finished, the script should look exactly like this:



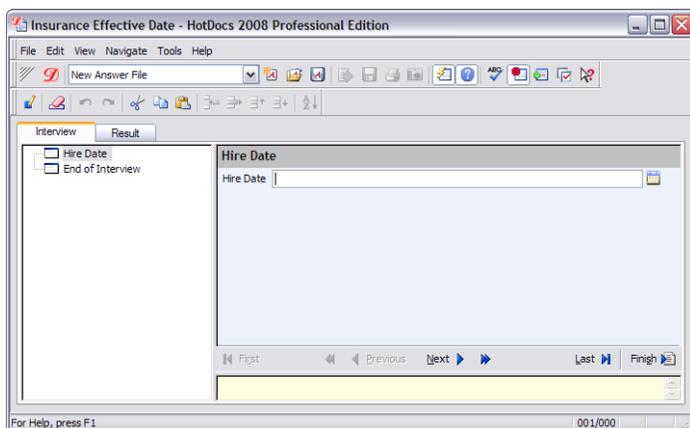
To summarize, this script produces a date that falls on the first day of the month immediately following the month the employee was hired. The first value in this script, *1*, specifies the first *day* of the month. The second value, *MONTH OF(Hire Date + 1 MONTHS)*, determines the *month* of the hire date and adds one month to it. The third value, *YEAR OF(Hire Date + 1 MONTHS)*, determines the *year* of the hire date (plus one month).

Test the Computation

Because some computation scripts can be quite complex, it is often helpful to test a computation script to make sure it calculates the correct result.

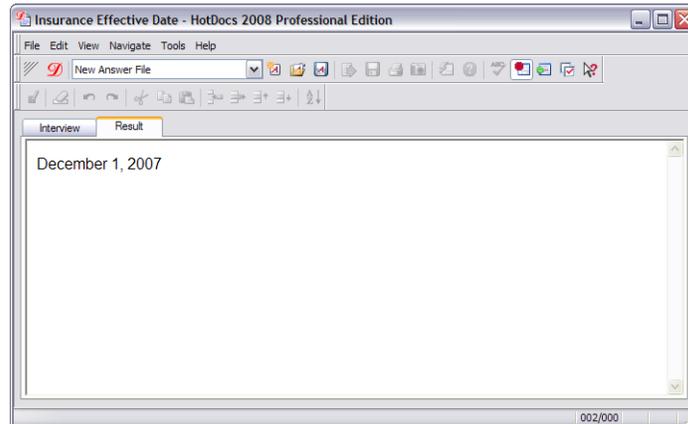
To test the Computation variable

- 1 At the **Computation Editor**, click **Test**. HotDocs opens a test assembly window showing the *Hire Date* variable.



- 2 Type *11/2/2007* in the **Hire Date** answer field and click the **Result** tab. HotDocs keeps you in the dialog to show you it has reformatted the answer you entered.

- 3 Click the **Result** tab again. HotDocs displays the answer *December 1, 2007*. This is the date the new employee's insurance coverage starts. This is correct.



- 4 Click the **Interview** tab and change the date to 12/2/2007.
- 5 Click the **Result** tab twice. HotDocs displays the answer *January 1, 2008*. HotDocs correctly calculates the date here as well, because your script uses the *Hire Date + 1 MONTHS* expression for both the month and the year.
- 6 Close the test assembly window to return to the **Computation Editor**, and then click **OK**.
- 7 At the **Variable Field** dialog box, click **Replace Once**. The variable is inserted in the template.

Now when users assemble this document, if the hire date falls after the first of the month, the new insurance effective date will be merged. Otherwise, the effective date will be the same as the hire date.

- 8 Click the  **Save and Close** button to close the template.

You are now finished with this lesson.

Lesson 10: Create Lists of Answers

Overview

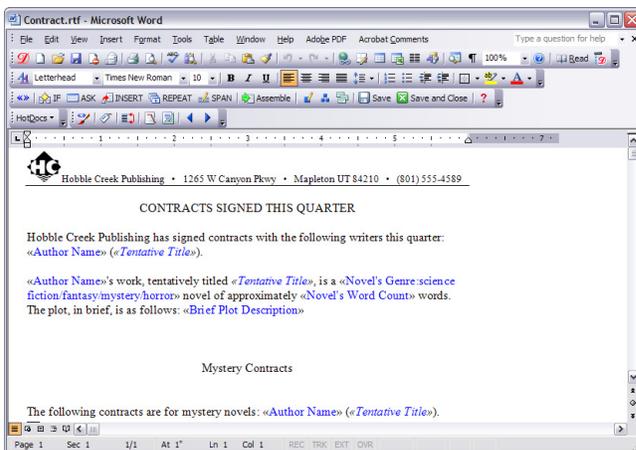
Sometimes when completing an interview, you need to enter two or more answers for a given question. These answers will then be used to generate a list within the document. For example, perhaps you need to include the names of several customers, rather than just a single customer. To accomplish this, you can repeat the *Customer Name* variable. This allows the user to enter as many answers as is necessary and each name will be merged.

In this lesson, you will generate a list of recently negotiated publishing contracts. Each contract will list the author, the tentative title for the book, and other information about the contract. To create these lists, you will surround the repeating text with a REPEAT instruction.

The first list of authors and tentative titles will be merged as a sentence. Individual paragraphs about each author will also be repeated. For this lesson, you will work with the *Contract Report* template.

To open the *Contract Report* template for editing

- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs library appears.
- 2 From the **HotDocs Tutorial Files** folder, select **Contract Report**.
 -  If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library (File menu)**, select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete step 2.
- 3 Click  **Edit**. The template appears, ready for you to edit.



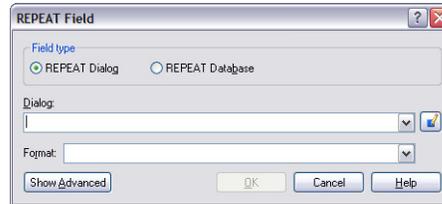
Most of the automation is complete, but the lists need to be created and the paragraph repeated.

Create a Variable-Length List Using Sentence Format

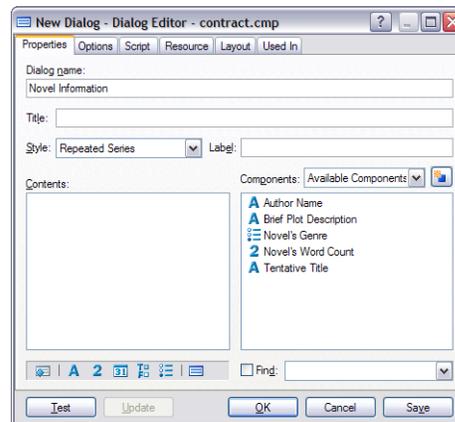
In the first paragraph, you want to merge the author's name and the book's tentative title for each contract that was signed.

To create the list of authors and titles

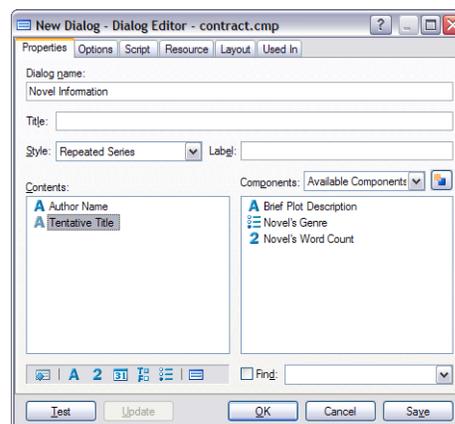
- 1 Select the text «Author Name» («*Tentative Title*») in the first paragraph. Do not select the period.
- 2 Click the  REPEAT Field button. The REPEAT Field dialog box appears.



- 3 Type **Novel Information** in the **Dialog** box and click the  **Edit Component** button. The **Dialog Editor** appears.



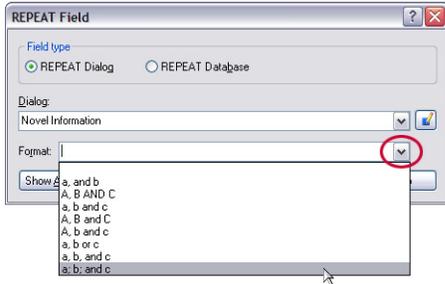
- 4 Drag **Author Name** and **Tentative Title** from the **Components** list to the **Contents** box.



- 5 Click **OK**. The **REPEAT Field** dialog box appears again.

You want the list to be merged into the sentence, with a semicolon after each entry and with the word *and* before the last entry. You can specify this format by selecting the corresponding example format.

- Click the **Format** drop-down button and select **a; b; and c** from the list. (This format is located at the bottom of the list.)



- Click **OK**. The REPEAT instruction is inserted into the template around «Author» («*Tentative Title*»).

Hobble Creek Publishing has signed contracts with the following writers this quarter:

«REPEAT Novel Information;a; b; and c»
«Author Name» («*Tentative Title*»)«END REPEAT»

✓ You can specify an option that keeps HotDocs from inserting returns after REPEAT and IF instructions if the text you are working with is within a paragraph. (Any returns HotDocs inserts when creating instructions, HotDocs will remove.) See the HotDocs Help for more information.

Test a Portion of the Template

Instead of testing the entire template, you can test just the portion of the template you repeated to make sure it works correctly.

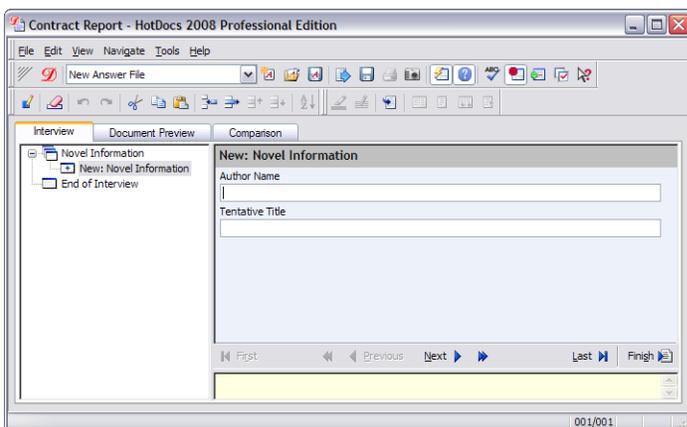
To test the list of authors and titles

- Select the first paragraph. Be sure to include the final period.

Hobble Creek Publishing has signed contracts with the following writers this quarter:

«REPEAT Novel Information;a; b; and c»
«Author Name» («*Tentative Title*»)«END REPEAT»

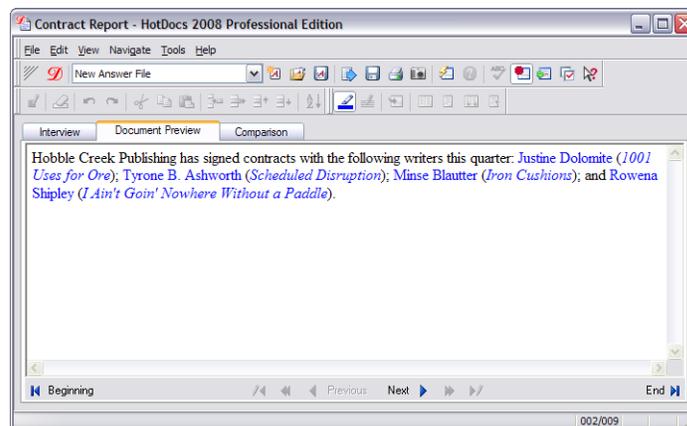
- Click the  **Assemble** button. HotDocs opens a test assembly window, which shows a new repetition of the dialog, *Novel Information*.



- 3 Enter answers for the first novel and click **Next**. The empty dialog, *Novel Information*, appears again.

Notice that as you provide answers at each dialog, HotDocs adds repeated dialog icons to the interview outline. If you later decide to add more information, you can click the last dialog, *New: Novel Information*, and enter your answers there.

- 4 Enter answers for as many novels as you want. When you are finished, click **Next** at an empty dialog.
- 5 Click the **Document Preview** tab to see the generated list. Notice how the list is punctuated.



- 6 Close the test assembly window. HotDocs returns you to the template.

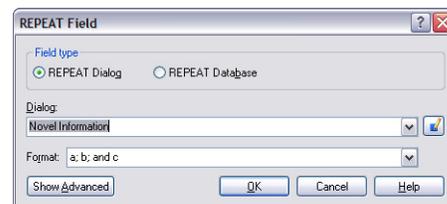
Sort a List

✓ Sorting allows you to alphabetize a list. You can either sort from *A to Z* or from *Z to A*. (Number values sort from *1 to 9* or *9 to 1*.)

If you want answers in the list to appear in a specific order, you can sort the list in alphanumeric order. In this case, you will sort the list alphabetically by the book's tentative title.

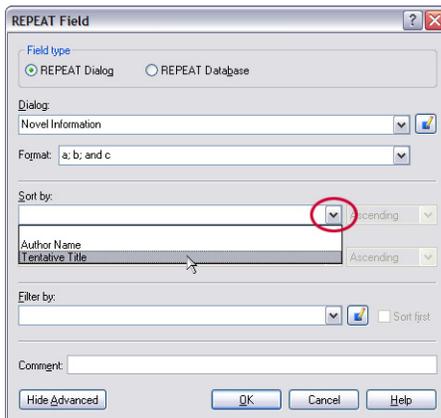
To sort the list alphabetically by the author's last name

- 1 Position the cursor in the REPEAT instruction and click the **REPEAT Field** button. The **REPEAT Field** dialog box for the instruction appears.



- 2 Click **Show Advanced**. The dialog box expands to show sorting options.

- Click the **Sort by** drop-down button to display the list of variables in the repeated dialog and select **Tentative Title**.



- Make sure **Ascending** is selected and then click **OK**.

You can test the list again to see the information appear in alphabetical order, based on the title. If you do, once you are finished, close the test assembly window and return to the template.

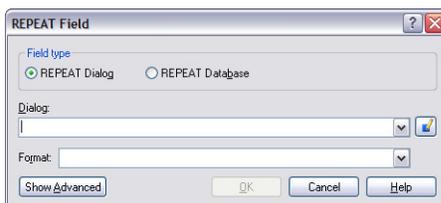
Repeat a Paragraph

You can use the same technique you used for the list in the first paragraph to repeat a larger section of a template, such as a paragraph. The paragraph that follows the list you just created needs to be merged into the document once for each novel.

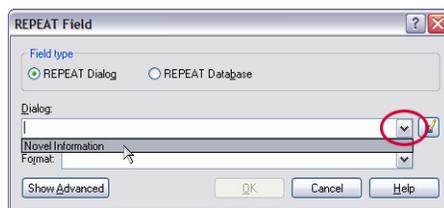
The paragraph includes the same variables you repeated for the previous list, so you can use the dialog you created for that list with the paragraph. However, because the paragraph includes more information than the list, you must edit the dialog and add those variables to it.

To repeat a paragraph

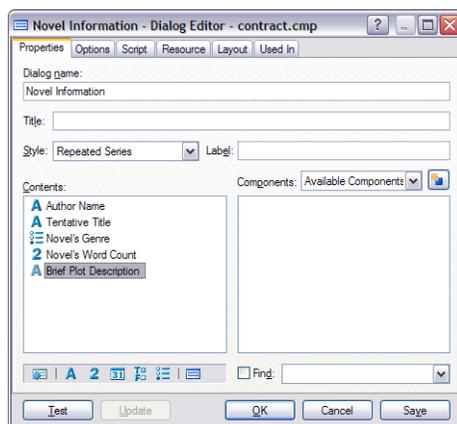
- Select the second paragraph. (If you are automating a Word template, also select the blank line immediately after the paragraph. If you're automating a WordPerfect template, select the entire line of text below the paragraph.) This will separate each repeated paragraph with a blank line.
- Click the  **REPEAT Field** button. The **REPEAT Field** dialog box appears.



- 3 Click the **Dialog** drop-down button and select **Novel Information**.



- 4 Click the  **Edit Component** button to edit the dialog. The **Dialog Editor** appears with the dialog *Novel Information* displayed.
- 5 Drag the following variables from the **Components** list to the bottom of the **Contents** box: **Novel's Genre**, **Novel's Word Count**, and **Brief Plot Description**.



- 6 Click **OK**. You are returned to the **REPEAT Field** dialog box.
- 7 Click **OK**. The **REPEAT** instruction is merged into the template around the paragraph.

«REPEAT Novel Information»
 «Author Name»'s work, tentatively titled «Tentative Title», is a «Novel's Genre: science fiction/fantasy/mystery/horror» novel of approximately «Novel's Word Count» words.
 The plot, in brief, is as follows: «Brief Plot Description»

«END REPEAT»

If you would like, you can test assemble this portion of the template. If you do, you will see that each author's information appears in its own paragraph.

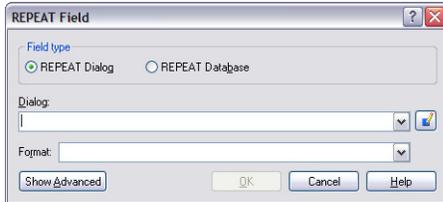
Create a Filtered List

In the last section of the template, you must create a list of only those contracts for mystery novels. This requires you to write a computation script that performs an action based on whether a condition is *true* or *false*. In this case, if an author has written a mystery novel, HotDocs will add the author's information to a list of just mystery novels. If the novel isn't a mystery novel, it will not be included in the list.

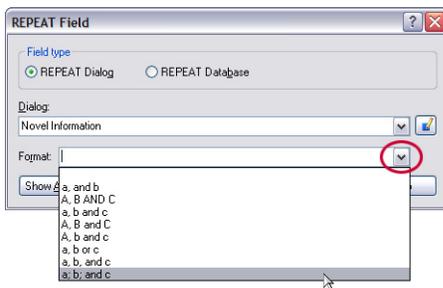
First, however, you must insert the **REPEAT** instruction.

To create the REPEAT instruction

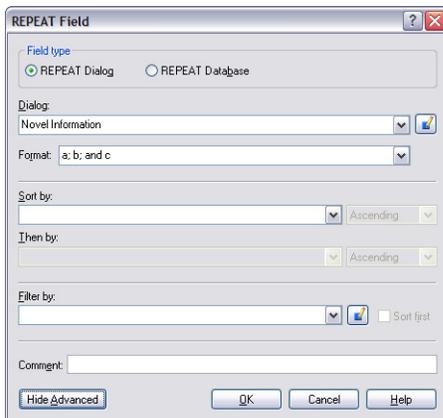
- 1 Select the text «Author» («*Tentative Title*») in the *last* paragraph. Do not select the period.
- 2 Click the  REPEAT Field button. The REPEAT Field dialog box appears.



- 3 Select **Novel Information** from the **Dialog** drop-down list and select **a; b; and c** from the **Format** drop-down list.



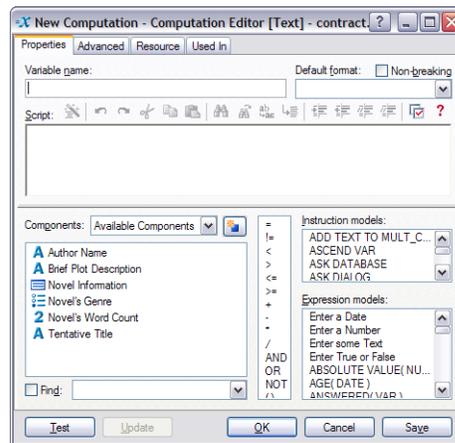
- 4 Click **Show Advanced**. The dialog box changes to show filtering options.



Now that you have created the REPEAT instruction, you can create the filter.

To create a filter

- 1 Click the  **Edit Component** button next to the **Filter** by drop-down list. The **Computation Editor** appears.

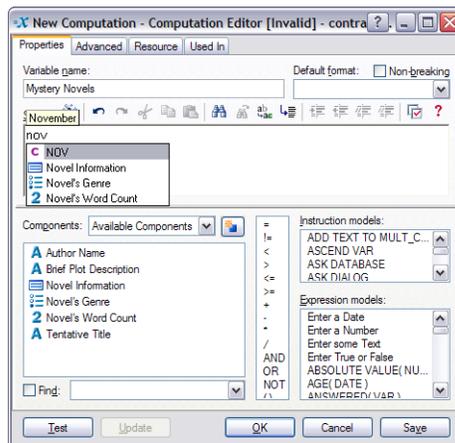


✓ A filter must result in a *true/false* (or *yes/no*) value. In this instance, it will filter all mystery novels and insert them into the template.

- 2 Type **Mystery Novels** in the **Variable name** box.

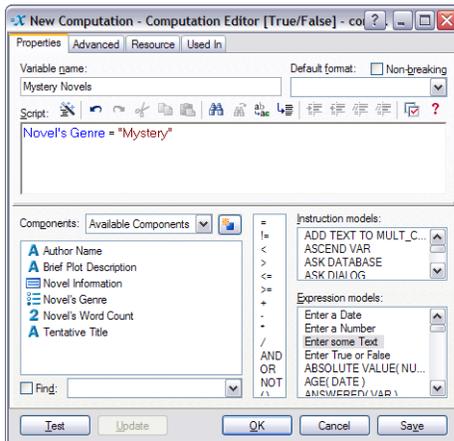
In Lesson 9, you used instruction and expression *models* to write your computation script. To write this script, however, you will use HotDocs' auto-complete feature to help you find and enter the correct instructions, expressions, and keywords.

- 3 Place your cursor in the **Script** box and press and hold the **Ctrl** key as you press the **Spacebar**. HotDocs displays a list of all components and keywords.
- 4 Type the letters **N, O,** and **V.** HotDocs filters the list to show only those items in the auto-complete list that contain that specific combination of letters.



- 5 Press the **DOWN ARROW** key to select the Multiple Choice variable, **Novel's Genre**.
- 6 Once it's selected (highlighted), press **Enter**. The keyword is merged into the script.
- 7 Enter a space character and then type an *equals* operator (=).
- 8 Enter another space character and press **Ctrl+Spacebar**. A list of options appears.

- 9 Press the DOWN ARROW key to select **Mystery**. The option is merged in the script, between quotation marks. Your script should look like this:



- 10 Click **OK** at the **Computation Editor** and click **OK** at the **REPEAT Field** dialog box. The REPEAT instruction is merged into the template.

Mystery Contracts

The following contracts are for mystery novels: «REPEAT Novel Information;a; b; and c::Mystery Novels»
«Author Name» («Tentative Title»)«END REPEAT»

Now, during the interview, whenever a user enters information about the signed contracts, HotDocs will merge only those contracts for mystery novels in this paragraph. (If you'd like, you can test assemble the template to see how this works. Once you are finished, close the test assembly window.)

Click the  **Save and Close** button to close the template.

You are now finished with this lesson.

Lesson 11: Use a Table to Display a List of Answers

Overview

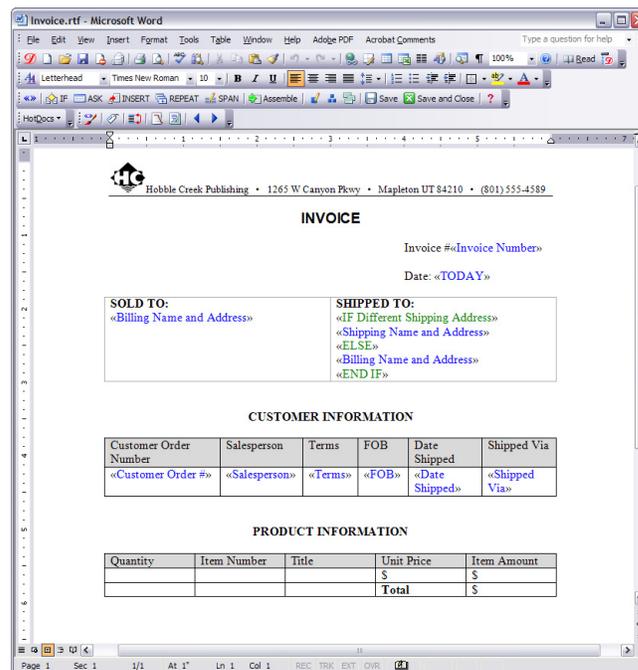
Sometimes you must insert a list of information using a word processor table, with each row representing an entry in the list. You can insert variables in a table row and then repeat that row using a REPEAT instruction. Just like variables in REPEAT instructions in a regular section of text, variables in a repeated table row can merge different values each time they are repeated.

For information on repeating sections of text, see “Lesson 10: Create Lists of Answers” on page 69. For additional information on REPEAT instructions, see “Overview: Include Lists in Your Documents” in the *HotDocs Help*.

You will work with the *Invoice* template for this part of the tutorial.

To open the Invoice template

- 1 Open your word processor and click the  HotDocs 2008 button. The HotDocs library appears.
- 2 From the **HotDocs Tutorial Files** folder, select **Invoice**.
 -  If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library** (**File** menu), select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete step 2.
- 3 Click  **Edit**. The template appears, ready for you to edit.



Most of the automation is complete, but you still must insert the variables for the *Product Information* table, which will be repeated.

Insert Variables in a Table

You can place HotDocs variables (which have already been created) into the *Product Information* table cells.

To insert variables in word processor table cells

- 1 Position your cursor in the first empty row of the **Quantity** column and click the **«» Variable Field** button. The **Variable Field** dialog box appears.
- 2 Select **Number** and select **Quantity** from the **Variable** drop-down list.



- 3 Click **OK**. The variable is inserted in the table cell.

PRODUCT INFORMATION

Quantity	Item Number	Title	Unit Price	Item Amount
«Quantity»			\$	\$
			Total	\$

- 4 Repeat this process for the following variables:

Item Number (Text)

Title (Text)

Unit Price (Number)

Item Amount (Computation)

When you are finished, your table should look like this:

PRODUCT INFORMATION

Quantity	Item Number	Title	Unit Price	Item Amount
«Quantity»	«Item Number»	«Title»	\$«Unit Price»	\$«Item Amount»
			Total	\$

Repeat a Table Row

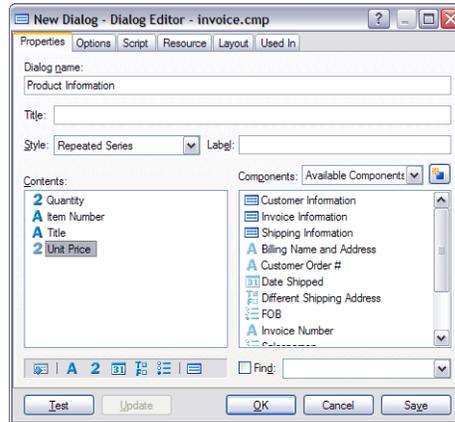
Once you have inserted your variables into the table, you need to tell HotDocs to repeat the table row containing the variables. REPEAT instructions are always linked to a HotDocs dialog that contains the repeated variables, or, in the case of a repeated Computation variable, a dialog that contains all of the variables used in the computation.

To repeat the second table row

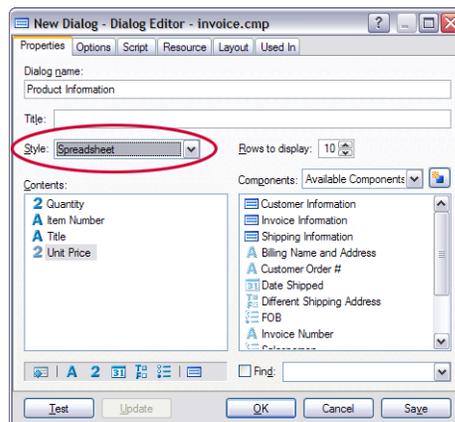
- 1 Position the cursor before the variable **«Quantity»**, in the second row of the table.
- 2 Click the **REPEAT Field** button. The **REPEAT Field** dialog box appears.

✓ The final repeated variable, *Item Amount*, is not included in the dialog because it is a Computation variable and cannot be asked. Instead, the values used to calculate the total (*Quantity* and *Unit Price*) appear in the dialog.

- 3 Type **Product Information** in the **Dialog** box and click the **Edit Component** button. The **Dialog Editor** appears.
- 4 Drag the following variables from the **Components** list into the **Contents** box: **Quantity**, **Item Number**, **Title**, and **Unit Price**.



- 5 Click the **Style** drop-down button and select **Spreadsheet**.



- 6 Click **OK** at the **Dialog Editor**, and click **OK** again at the **REPEAT** Field dialog box. The **REPEAT** instruction is inserted in the table.

PRODUCT INFORMATION

Quantity	Item Number	Title	Unit Price	Item Amount
«REPEAT Product Information»«Quantity»	«Item Number»	«Title»	\$«Unit Price»	\$«Item Amount»
			Total	\$

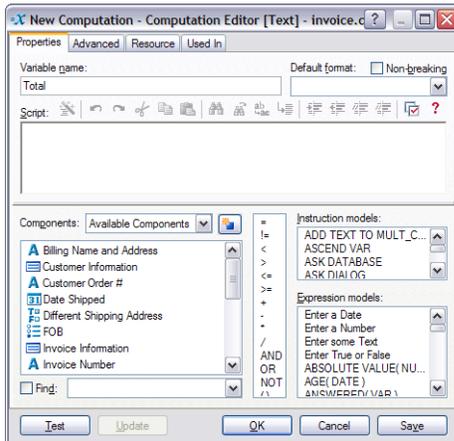
⚠ When inserting a **REPEAT** instruction in a table, no **END REPEAT** instruction will be merged at the end of the row. This is because HotDocs sees the end of the row as the end of the **REPEAT**. This means you cannot repeat text on more than one row in the table.

Find the Sum of a Table Column

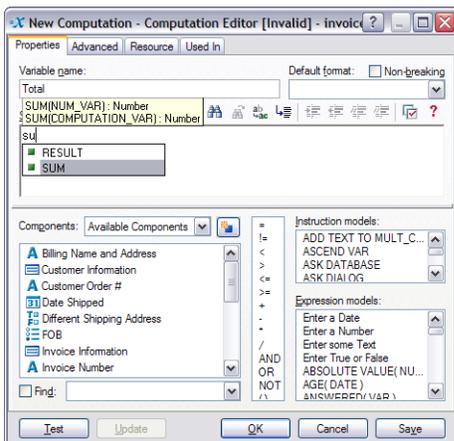
You can find the sum total of a column that contains either a Number variable or a Computation variable. For the invoice, you will add the different amounts for the *Item Amount* column.

To find the total sum of Item Amount

- 1 Position your cursor in the last cell of the last row. (This is the field after *Total*.)
- 2 Click the **«» Variable Field** button. The **Variable Field** dialog box appears.
- 3 Select **Computation** and type **Total** in the **Variable** box.
- 4 Click the **Edit Component** button. The **Computation Editor** appears.



- 5 Place your cursor in the **Script** box and press **Ctrl+Spacebar**. HotDocs displays the auto-complete list of instruction and expression keywords, as well as component names.
- 6 Type the letters **S** and **U**. HotDocs filters the list to show only those models or components that contain that specific combination of letters. It highlights the expression keyword, **SUM**.



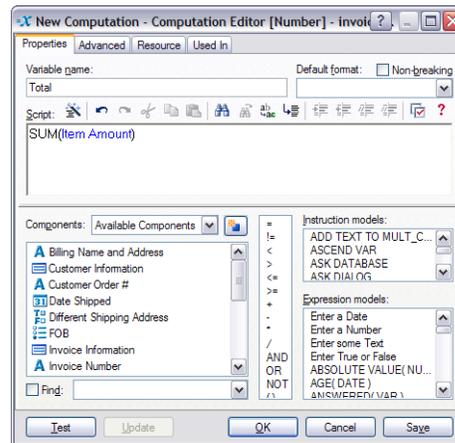
- 7 Press **Enter**. HotDocs merges the keyword **SUM** into the script.
- 8 Type an opening parenthesis. HotDocs adds the parenthesis and displays a syntax hint.

Syntax hints show you the instruction or expression in its entire syntactical format, including any placeholders you must replace with values. Because there are two **SUM** expressions (one for simple numbers and one for computed), HotDocs displays two hints.

- 9 Press the **F5** key. This displays a list of components only.

10 Start typing the name of the Computation variable, **Item Amount**.

11 When **Item Amount** is highlighted in the list, press **Enter** to add it to the script and then type the closing parenthesis.



12 Click the **Default format** drop-down button and select **9,999.00**. This ensures the answer is formatted properly when it's merged into the assembled document.

13 Click **OK** at both the **Computation Editor** and the **Variable Field** dialog box.

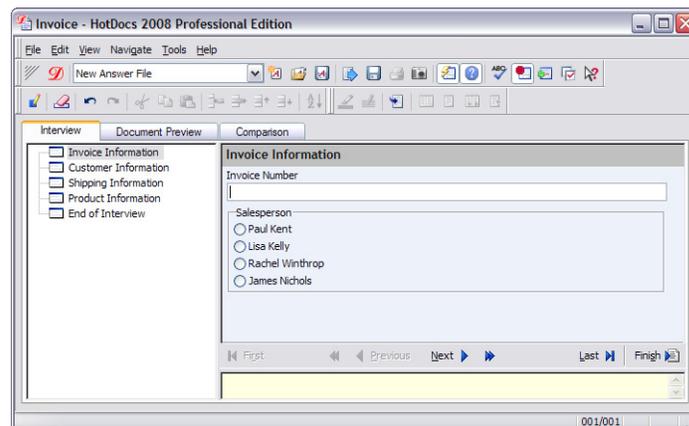
14 Click the **Save** button to save your work.

Test the Table

You can test the table and subsequent Computation variable by test assembling the document.

To test your work with the table and the computation

1 At the template, click the **Assemble** button. A test assembly window appears, which displays the dialog, **Invoice Information**.



2 Enter information and click **Next** to advance. The dialog **Customer Information** appears.

3 Continue answering questions until the dialog **Product Information** appears.

4 In separate rows of the spreadsheet, enter answers for as many items as you want. When you are finished, click  **Next**.

5 Once you reach the **End of Interview** dialog, click the  **Send the assembled document to the word processor** button. HotDocs opens your word processor and displays the assembled document in it.

The table row appears once for each item entered. The final table row contains the sum of the rightmost column.

6 Close the assembled document as well as the test assembly window. (You don't need to save the document.) Then click the  **Save and Close** button to close the template.

You are now finished with this lesson.

Lesson 12: Use Complex IF Instructions

Overview

In Lesson 5, you used IF instructions to merge single conditional paragraphs into an assembled document. This method worked well since your paragraphs were unrelated.

Sometimes, however, you have a group of related paragraphs, only one of which should be merged into a final document. You can use a series of IF expressions to control the entire group of paragraphs.

In this lesson, you will work with a *Collection Letter* template. When the collection letter is assembled, the text of the document changes depending on how late the customer's payment is.

For additional information on IF instructions, see "Overview: Make Parts of Templates Conditional" in the HotDocs Help.

To open the *Collection Letter* template for editing

- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs library appears.
- 2 From the **HotDocs Tutorial Files** folder, select **Collection Letter**.

 If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library** (**File** menu), select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete step 2.

3 Click  **Edit**. The template appears, ready for you to edit.



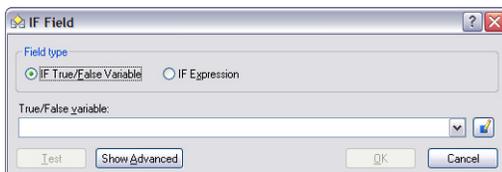
In this template, there are four different versions of the second paragraph. You will create a series of expressions to merge the correct paragraph, based on how late the customer's payment is.

Begin the IF Expression

In this template, you want the first of these alternate paragraphs merged into the document only when the invoice is no more than 30 days old. This is the paragraph beginning *Please review your records*.

To make the paragraph conditional

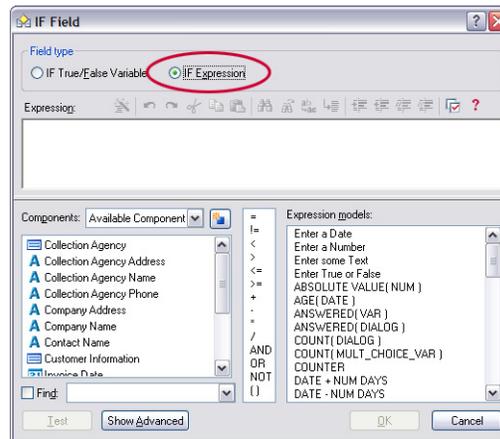
- 1 Select the second paragraph. (If you are automating a Word template, also select the blank line immediately after the paragraph. If you're automating a WordPerfect template, select the entire line of text below the paragraph.)
- 2 Click the  **IF Field** button. The **IF Field** dialog box appears.



You do not want to create a True/False variable because that would require the user to answer an additional question. Instead, you want to write an expression that uses information you already know to test how old the invoice is.

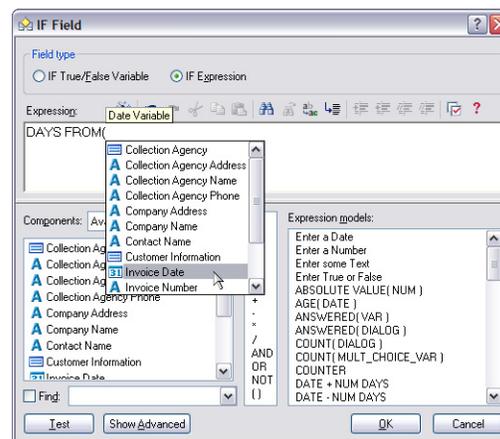
✓ An IF expression must result in a *true/false* value. In this expression, you will determine whether the number of days between the payment date and the current date is less than or equal to 30 days. If true, HotDocs will insert the paragraph requesting immediate payment.

- 3 Select **IF Expression**. The dialog box changes to show a scripting box for the expression.



You want to test whether the date of the invoice is no more than 30 days from today.

- 4 Place your cursor in the **Expression** box and press **Ctrl+Spacebar**. HotDocs displays the auto-complete list.
- 5 Select (highlight) **DAYS FROM** and press **Enter**. HotDocs merges the expression keywords into the **Expression** box.
- 6 Type an opening parenthesis. The syntax hint appears, reminding you that this expression needs two date values.
- 7 Press the **F5** key and choose **Invoice Date** from the component list.



✓ The **TODAY** keyword inserts the current date, based on your computer's system clock.

- 8 Type a comma, a space, and then type the keyword **TODAY**.
- 9 Type a closing parenthesis, followed by a space and the *less than or equal to* operator (**<=**). (Make sure there is a space character on either side of the operator.)

10 Type 30 after the operator and click OK. The IF expression is merged into the template around the paragraph.

```
«IF DAYS FROM(Invoice Date, TODAY) <= 30»
Please review your records and send your payment for $«Invoice Total:9,999.99» in
today. If you have any questions, please call me.

«END IF»
```

Add an ELSE IF to the IF Instruction

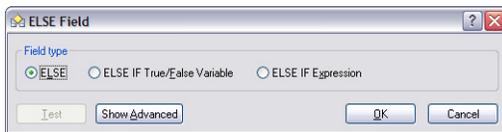
The next paragraph needs to be merged into the document when the invoice is more than 30 days old but less than or equal to 60 days old. When you create this part of the instruction, you can use information you've already determined by using the first IF instruction.

To add an ELSE IF to the IF instruction

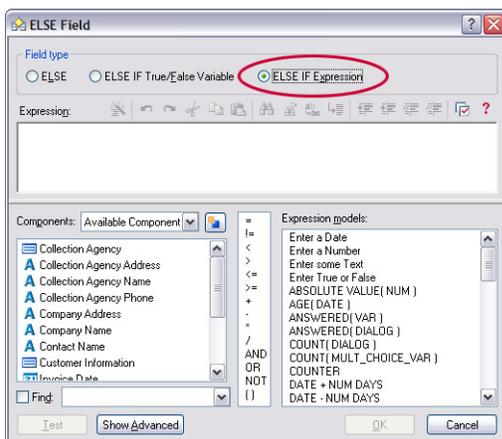
1 Highlight the «END IF» instruction and the next paragraph (including the blank line below it).

```
«END IF»
A late fee of $«Late Fee:9,999.99» has been added to the invoice total of $«Invoice
Total:9,999.99», making a new total of $«New Total:9,999.99». In order to preserve your
good credit, please send full payment today or contact me to make other suitable
arrangements.
```

2 Click the  IF Field button. The ELSE Field dialog box appears.

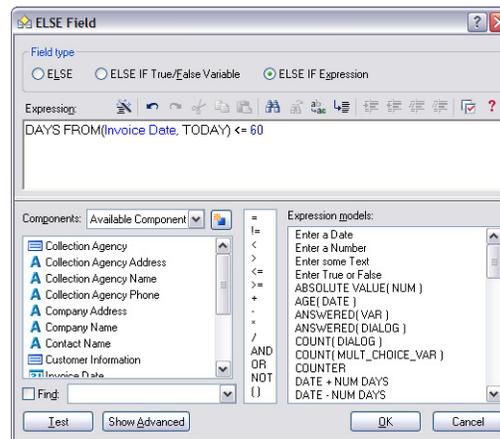


3 Select ELSE IF Expression. The dialog box changes to show a scripting box for the expression.



You've already tested whether the invoice is more than 30 days old. Now you must test whether it is no more than 60 days old. If it is, this next paragraph will be merged using an ELSE IF instruction.

- 4 Using the auto-complete functionality, enter the script `DAYS FROM(Invoice Date, TODAY) <= 60` in the Expression box.



- 5 Click OK. The second part of the IF instruction is inserted around the paragraph.

On Your Own

Following the instructions given earlier for adding an ELSE IF to the IF instruction, make the next paragraph conditional on the invoice date being less than or equal to 90 days from today, using the following expression:

```
DAYS FROM( Invoice Date, TODAY ) <= 90
```

Add the Final Paragraph

You could write an expression to control the final paragraph as well, but that is unnecessary in this case because you want this paragraph merged automatically whenever all of the other conditions are *false* (or if the invoice is more than 90 days old).

To make the final paragraph conditional

- 1 Select «END IF» and the final paragraph, including the *Collection Agency* variables and the blank line following them.
- 2 Click the  IF Field button. The ELSE Field dialog box appears.



- 3 Leave **ELSE** selected and click **OK**. The **ELSE** instruction is inserted around the final paragraph.

«ELSE»

Your account has been referred to a collection agency. Please direct all further correspondence to:

«Collection Agency Name»
 «Collection Agency Address»
 «Collection Agency Phone»

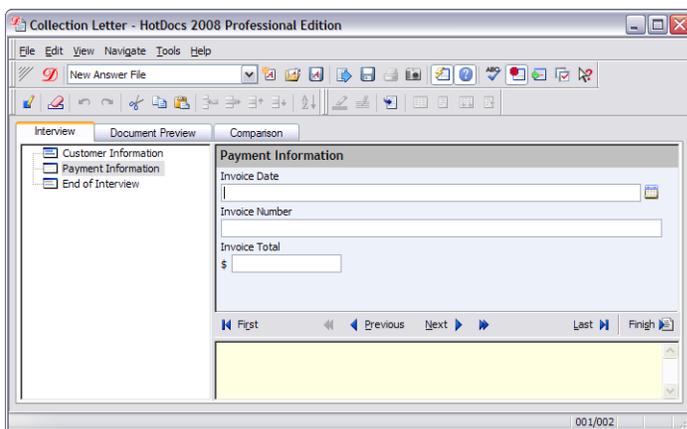
«END IF»

Test the Expressions

Now that you have conditioned each paragraph, you will test the template to make sure the correct paragraph will be inserted, based on the invoice date.

To assemble the template

- 1 At the template, click the  **Assemble** button. A test assembly window appears, showing the first dialog, **Customer Information**.
- 2 Answer the questions in the dialog and click **Next**. The **Payment Information** dialog appears.

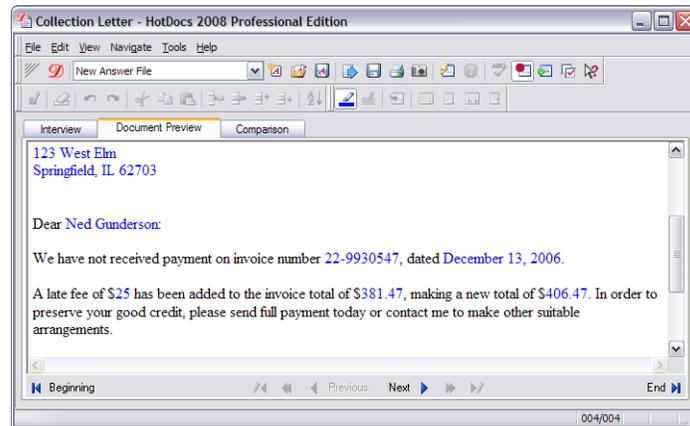


- 3 Click the  **Calendar** button next to the **Invoice Date** answer field. The **Calendar** appears.

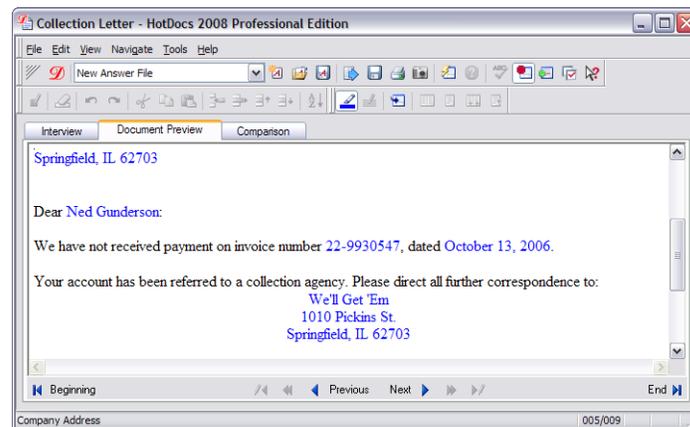


- 4 Click the **Left Arrow** next to the **Month** field to navigate back one month and click **OK**.
- 5 Answer the remaining questions in the dialog.

- 6 Click **Next** and continue answering questions in the interview. The **End of Interview** dialog appears.
- 7 Click the **Document Preview** tab. Note that the paragraph about late fees has been merged.



- 8 Click the **Interview** tab again and navigate to **Payment Information**.
- 9 Click the **Invoice Date** calendar icon again and click the **Left Arrow** for the month two more times.
- 10 Answer the remaining questions in the interview and click the **Document Preview** tab. Note how the paragraph has changed and now includes information about the collection agency.



- 11 Close the test assembly window
 - 12 At the template, click the  **Save and Close** button.
- You are now finished with this lesson.

Lesson 13: Incorporate Clause Selection in a Template

Overview

In Lesson 8, you created a clause library and used it to assemble clauses that could be inserted into any document, whether the document was assembled using HotDocs or not. In this lesson, you will create a clause library containing clauses that are intended for insertion at a specific point in a document.

During the interview, HotDocs will present the clause library to the user so the user can select the clauses to insert and reorder them, if desired. HotDocs will then insert the clauses into the document at the specified place.

The process of preparing a clause library that is to be used for clause selection is much the same as the process you used in Lesson 8 to prepare a clause library for insertion into a template library.

The first step is to place the text for all of the clauses you might want in the finished document into one template. This means if there are two versions of a paragraph that could go in, you would put both versions in the template. You also must insert all of the variables you need. In this lesson, this has already been done for you.

The second step is to divide the template text into clauses and add the clauses to the clause library. Finally, you must insert the clause library file into the template text using an INSERT instruction. You will complete these two steps in this lesson.

In this lesson, you will work with the *Publishing Contract* template.

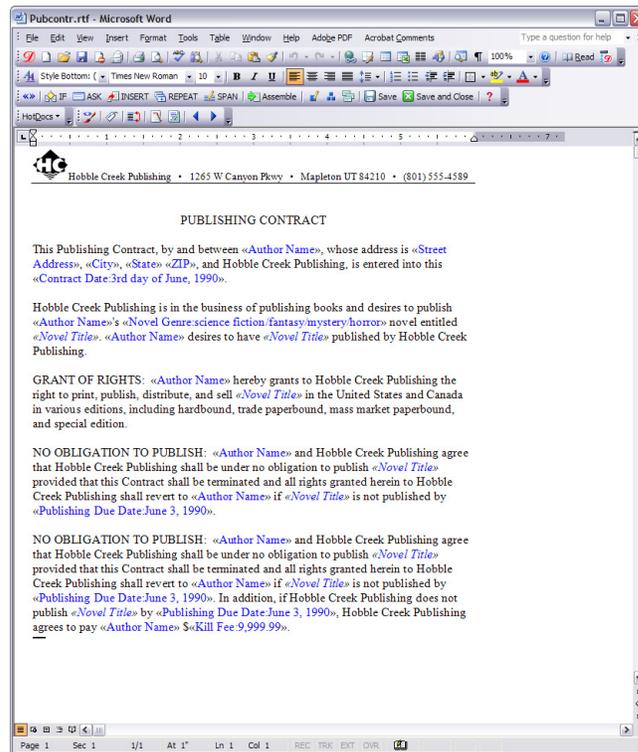
To open the Publishing Contract template for editing

- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs library appears.
- 2 From the **HotDocs Tutorial Files** folder, select **Publishing Contract**.

 If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library (File menu)**, select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete step 2.

 Do not group variable questions together in custom dialogs unless you know that all of the questions in the dialog will always be in the same clause.

- 3 Click the  Edit button. The template is opened for editing.



Most of the automation is complete, but the text needs to be divided into clauses and added to the clause library.

Divide the Text into Clauses and Add Them to the Clause Library

You must divide the text into clauses and then add those clauses to the clause library.

To create a clause for the third paragraph

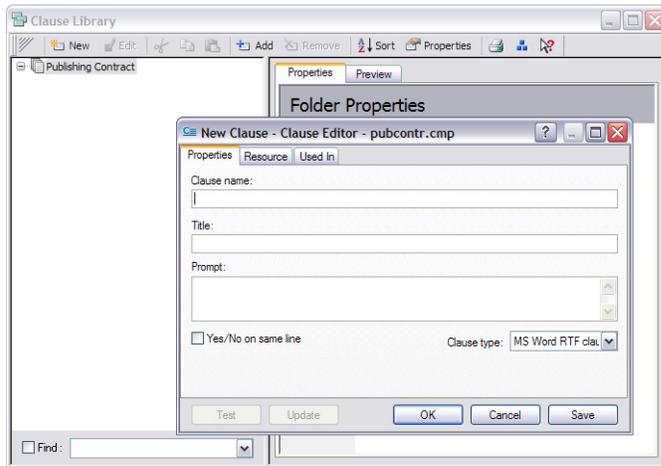
- 1 Select the third paragraph. If you are automating a Word template, select just the paragraph and *not* the blank line below it, like this:

GRANT OF RIGHTS: «Author Name» hereby grants to Hobble Creek Publishing the right to print, publish, distribute, and sell «Novel Title» in the United States and Canada in various editions, including hardbound, trade paperbound, mass market paperbound, and special edition.

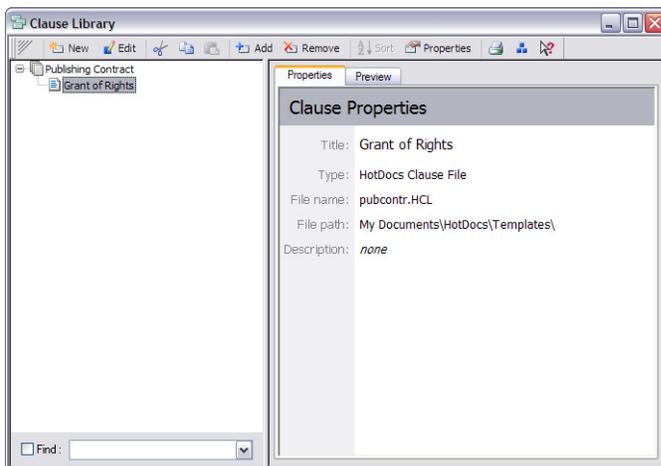
If you are automating a WordPerfect template, select the entire blank line below the paragraph, like this:

GRANT OF RIGHTS: «Author Name» hereby grants to Hobble Creek Publishing the right to print, publish, distribute, and sell «Novel Title» in the United States and Canada in various editions, including hardbound, trade paperbound, mass market paperbound, and special edition.

- 2 Click the  **Clause Library** button. The **Clause Library** and the **Clause Editor** appear.



- 3 Type **Grant of Rights** in the **Clause name** box.
4 Click **OK**. The clause is added to the clause library.



You will see that HotDocs has removed the *Grant of Rights* paragraph from the template.

On Your Own

Create clauses for the remaining two paragraphs and add them to the clause library. Name the clauses *No Obligation*, and *No Obligation with Kill Fee*, respectively.

When you are finished, close the clause library (but not the template) by clicking the X in the upper-right corner of the clause library window. (If the clause library is not visible, click its icon in the Windows taskbar.)

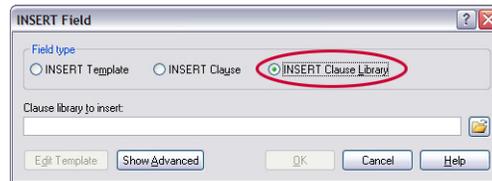
 Adjust the height of the **Clause Library** window and click the **Arrange** button. This arranges the library with the template, allowing you to work in both windows simultaneously.

Insert the Clause Library into the Template File

The final step is to insert the clause library file into the template file. You should insert the clause library at the place in the template where you want the selected clauses to be inserted.

To insert the clause library in the Publishing Contract

- 1 Place your cursor two lines below the second paragraph.
- 2 Click the  INSERT Field button. The INSERT Field dialog box appears.
- 3 Select INSERT Clause Library.



- 4 Click the  Open button next to the Clause library to insert box.
- 5 Select **pubcontr.hdl** and click OK. HotDocs displays the INSERT Field dialog box again, showing the file you just selected in the Clause library to insert box.



- 6 Click OK at the INSERT Field dialog box.

An instruction to insert the clause library file is added to the template.



Hobble Creek Publishing • 1265 W Canyon Pkwy • Mapleton UT 84210 • (801) 555-4589

PUBLISHING CONTRACT

This Publishing Contract, by and between «Author Name», whose address is «Street Address», «City», «State» «ZIP», and Hobble Creek Publishing, is entered into this «Contract Date:3rd day of June, 1990».

Hobble Creek Publishing is in the business of publishing books and desires to publish «Author Name»'s «Novel Genre:science fiction/fantasy/mystery/horror» novel entitled «Novel Title». «Author Name» desires to have «Novel Title» published by Hobble Creek Publishing.

«INSERT "pubcontr.hdl"»

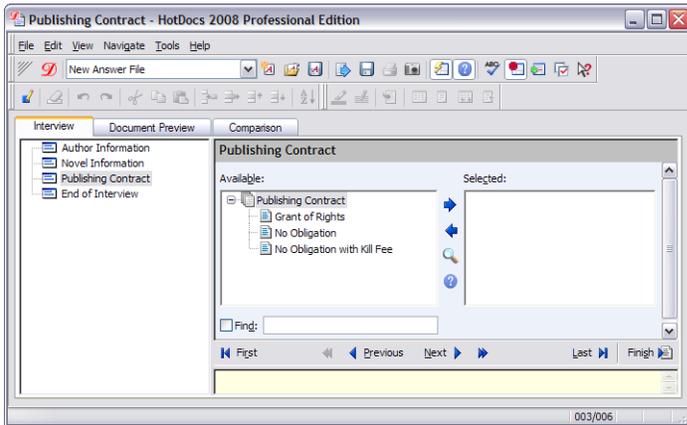
Test the Clause Library

You can test clause insertion by testing the template that contains the clause library.

To test the finished template

- 1 At the template, click the  Assemble button. HotDocs displays the first dialog in the interview, **Author Information**.

- 2 Answer questions and advance through the dialogs until you come to the **Publishing Contract** clause library.



- 3 Select the clauses you want included in the final document and add them to the **Selected** box. If you want the clauses in a different order, drag and drop them within this box.
- 4 When you have selected and ordered the clauses you want, click **Next**. HotDocs displays dialogs to gather the variable information for the clauses.
- 5 Enter the information and click **Next** to advance to each new dialog. Keep answering questions as dialogs appear.
- 6 When you finish answering questions, click the **Document Preview** tab. All of the clauses you selected are included in the document in the order they were listed in the **Selected** box.
- 7 Close the test assembly window and return to the template. Then click the **Save and Close** button to close the template.

You are now finished with this lesson.

✓ In addition to using the **Select** button to add clauses to the **Selected** list, you can select a clause and press **Enter** or **Insert**.

Lesson 14: Change the Layout of Dialogs

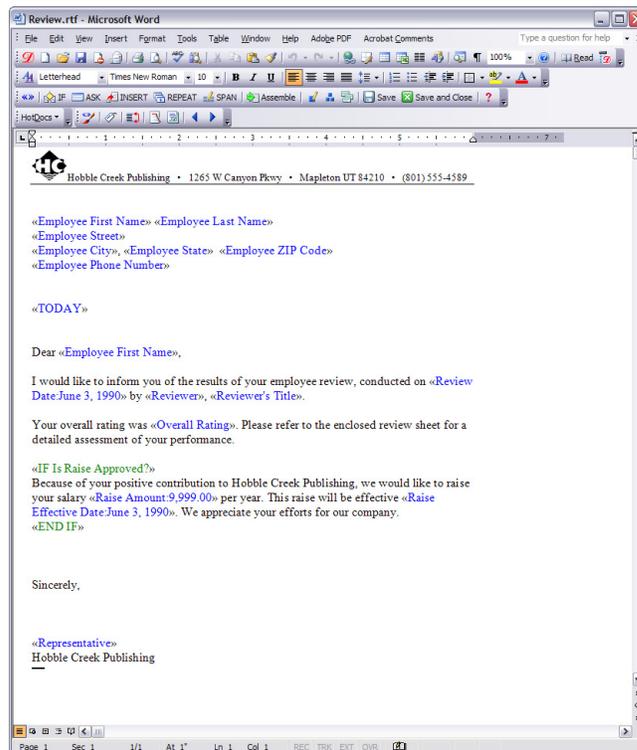
Overview

When you create a custom dialog, each variable you add to the dialog is placed on its own line in the dialog. Sometimes, however, it makes more sense to place two or three variables, side by side, on the same line. To do this, you can change the position of the variable's answer fields as you are editing the dialog.

In this lesson, you will visually arrange variables in a dialog that is used in the *Employee Review Letter* template.

To open the Employee Review Letter template for editing

- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs library appears.
- 2 From the **HotDocs Tutorial Files** folder, select **Employee Review Letter**.
 -  If the *HotDocs Tutorial Files* folder does not appear, choose **Open Library** (**File** menu), select the HotDocs library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete Step 2.
- 3 Click  **Edit**. The template appears, ready for you to edit.

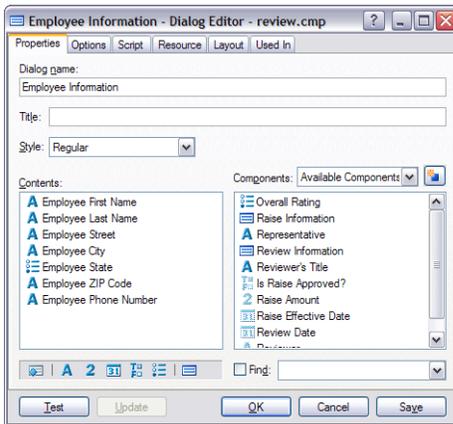


Test the Dialog to See How Variables Are Arranged

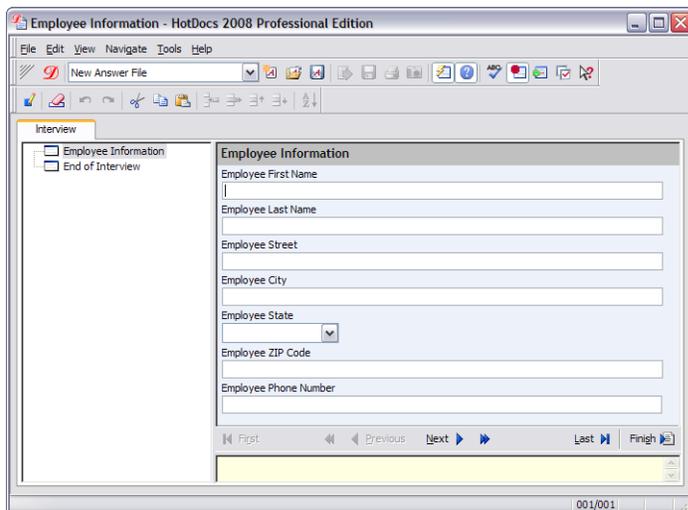
By default, HotDocs places each variable in a dialog on its own line. However, you can place two or three variables side by side in a dialog.

To open the dialog and test it

- 1 At the template, click the  **Component Manager** button. The **Component Manager** window appears.
- 2 Select the dialog **Employee Information** from the component list and click the  **Edit Component** button. The **Dialog Editor** appears.



- 3 Click **Test**. A test assembly window appears.



Each answer field appears on its own line. However, to better organize the variables, you want *Employee City*, *Employee State*, and *Employee ZIP Code* to be side by side on the same line.

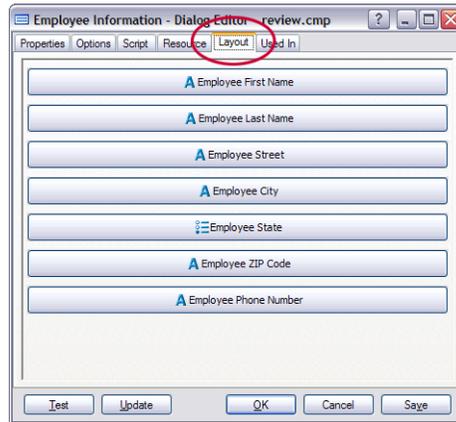
Place Answer Fields on the Same Line in a Dialog

You can use the **Dialog Editor** to change the layout of a dialog.

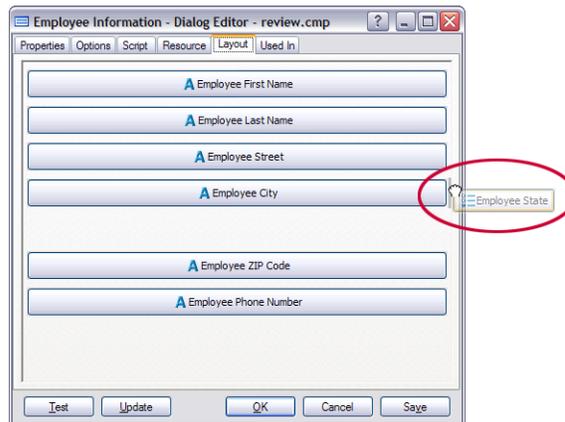
 Adjust the width of **Component Manager** and click the  **Arrange** button. HotDocs will arrange **Component Manager** and the template window so you can view both windows simultaneously.

To change the layout of a dialog

- 1 In the interview outline of the test assembly window, click the **Employee Information** dialog icon and click the  **Edit Component** button. The **Dialog Editor** comes to the front.
- 2 Click the **Layout** tab. The window changes to show icons for each variable in the dialog.

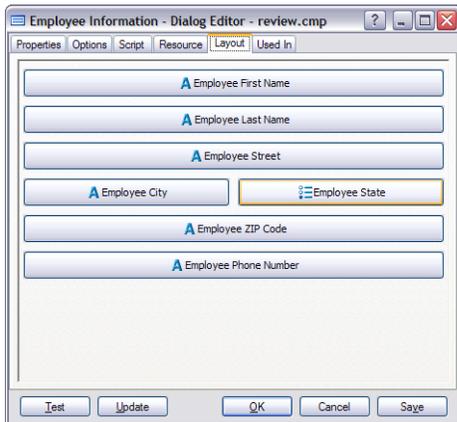


- 3 Click on the **Employee State** icon and drag it to the right of the **Employee City** icon.



As you are dragging the variable to its new location, you see horizontal and vertical lines, indicating where the variable will be placed when you release the mouse button.

- 4 When you see a vertical line to the right of the **Employee City** icon, release the mouse button. HotDocs places the two variable icons on the same line.



When you do this, you will notice that the test assembly window automatically updates to show the new placement of the variable.

On Your Own

At the **Dialog Editor**, drag **Employee ZIP Code** to the same line as **Employee City** and **Employee State**. (It should appear next to **Employee State**.) Follow the instructions for moving **Employee State**.

When you are finished, click **Update** to view the test assembly window again. The three variables appear on the same line.

Close the test assembly window and click **OK** at the **Dialog Editor**.

If you are not continuing on to Lesson 15, close **Component Manager** and then click the  **Save and Close** button to close the template.

If you are continuing on to Lesson 15, click the  **Save** button to save your work. (Leave **Component Manager** open.)

 If you're placing only two variables on a single line, you can make one of the variables wider than the other. To do this, right-click the variable you want to be wider and select **Wide** from the shortcut menu.

Lesson 15: Create Scripts for Your Dialogs

Overview

You can add scripts to dialogs that allow you to hide or gray variables, require that certain variables be answered before the user can proceed, set a variable to a value that cannot be changed by the user, and set a limit to the number of answers a user can enter for a REPEAT instruction. You can also make instructions in a script conditional using IF instructions or expressions.

In this lesson, you will create a script for the dialog *Raise Information*. This dialog is in the *Employee Review Letter* template you used in Lesson 14.

If you are continuing immediately from Lesson 14, skip the instructions for opening the template and proceed directly to “Test the Raise Information Dialog to See How it Appears.”

To open the Employee Review Letter template

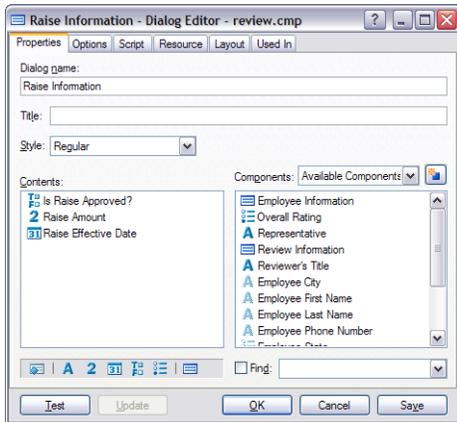
- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs library appears.
- 2 From the **HotDocs Tutorial Files** folder, select **Employee Review Letter**.
 If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library** (**File** menu), select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then complete Step 2
- 3 Click  **Edit**. The template appears, ready for you to edit.
- 4 At the template, click the  **Component Manager** button to open Component Manager.

Test the Raise Information Dialog to See How it Appears

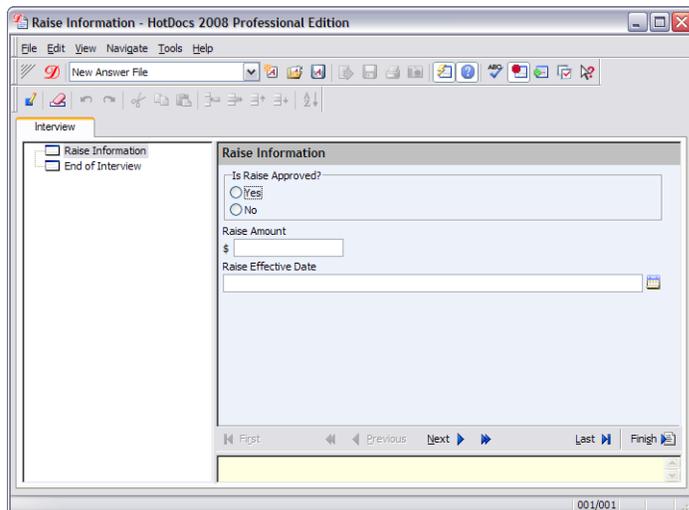
In this section, you will test the *Raise Information* dialog to see how the variables in the dialog appear. Often, you can get ideas about ways to customize a dialog after you've seen how it looks.

To test the Raise Information dialog

- 1 At the **Component Manager** window, select the dialog **Raise Information** from the component list and click the  **Edit Component** button. The **Dialog Editor** appears.



- 2 Click **Test**. A test assembly window appears.



This dialog first asks if the employee was approved for a raise. It then asks the raise amount and the date the raise will go into effect.

You will create a script so that if the raise was approved, HotDocs requires the user to answer two questions about the raise before he or she can proceed with the interview. Otherwise, if the raise is *not* approved, the questions are grayed so answers can't be entered.

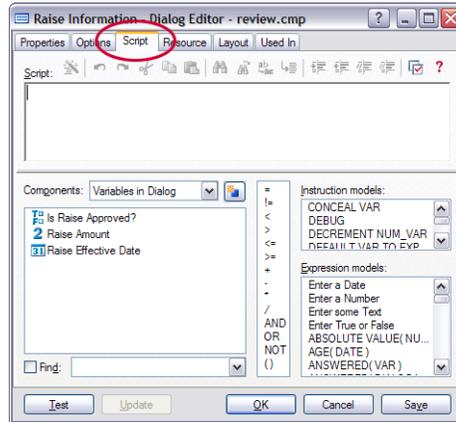
- 3 Close the test assembly window and return to the **Dialog Editor**.

Create a Script

Because you need certain variables to be available only if certain conditions are met, you must create a script for the dialog.

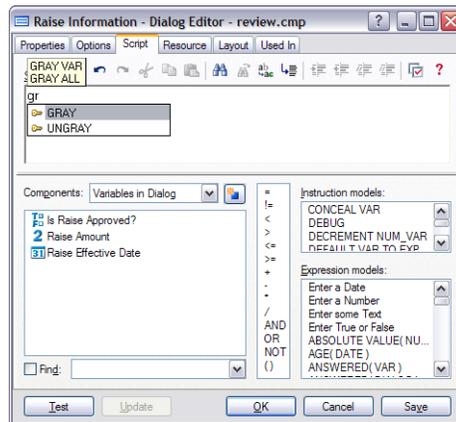
To create a script to control what variables are asked

- 1 At the **Dialog Editor**, click the **Script** tab.

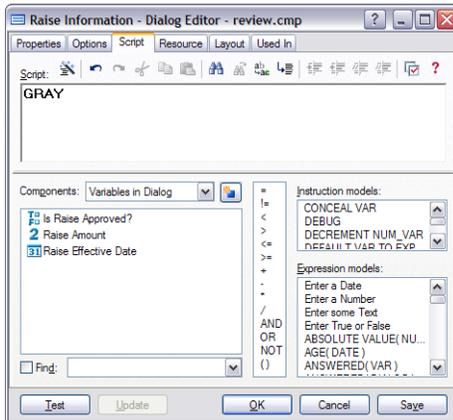


First, you will insert an instruction that will gray the second and third variable questions. Then, you will create a condition so that if the employee is approved for a raise, the questions will be ungrayed and the user will be required to answer them. This way, the user can only enter the raise amount and effective date if the raise was approved.

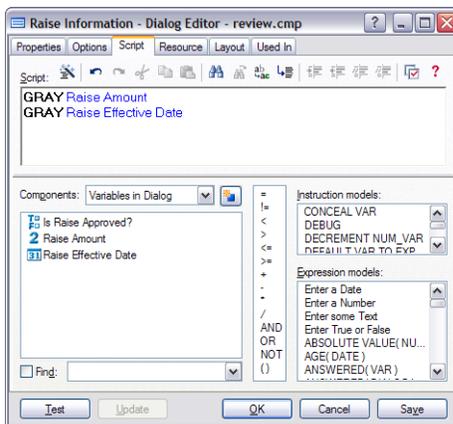
- 2 With your cursor in the **Script** box, press **Ctrl+Spacebar** and then type the letters **GR**. HotDocs highlights the **GRAY** instruction in the list.



- 3 Press **Enter**. The instruction is merged into the script.



- 4 Enter a space and then press the **F5** key. HotDocs displays a list of components.
- 5 Select **Raise Amount** and press **Enter**. HotDocs merges the Number variable **Raise Amount** into the script.
- 6 On the next line, enter the instruction **GRAY Raise Effective Date**. (Either use the auto-complete feature, or type the script directly in the **Script** box.)

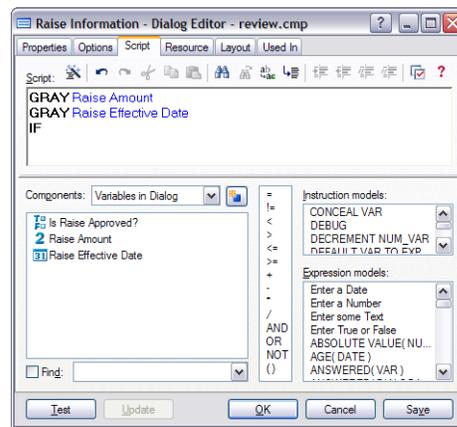


Insert a Condition

Now you must use an **IF** instruction to test whether the raise is approved for the employee. If so, the related variables will be enabled (ungrayed) and required.

To add a conditional instruction to the script

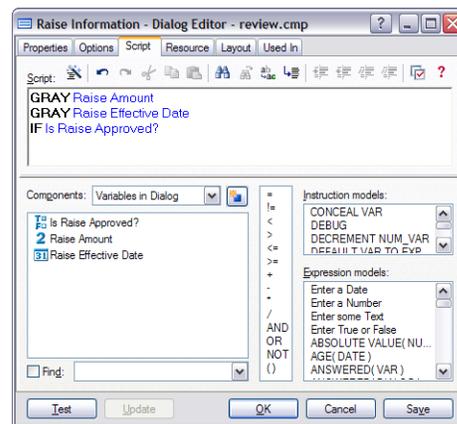
- 1 On the next line in the script, type the instruction keyword IF and press the spacebar.



✓ Using an IF instruction in a script creates the same kind of conditional test you used when you inserted conditional paragraphs in lessons 5 and 12. In this case, if a user answers *yes* to a variable, he or she can answer other questions in the dialog.

- 2 Press the F5 key to view the list of components and select **Is Raise Approved?** from the list.

You have now created the first part of the condition. The Script box should look like this:



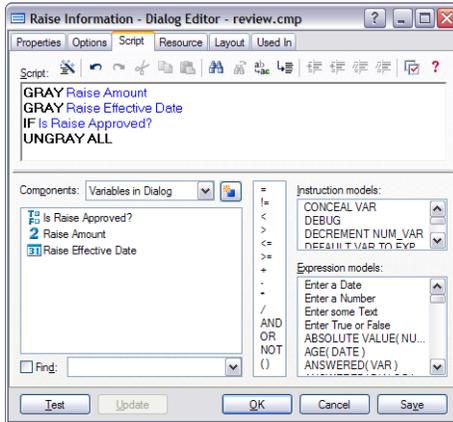
Control the Appearance of Variables in the Script

Next, you will use the UNGRAY ALL and REQUIRE VAR instructions to ungray the two variables and require that they be answered if the condition is true.

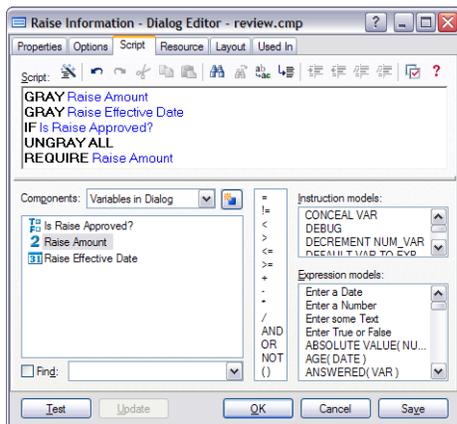
To use instructions to control the functionality of variables

- 1 After the instruction, **IF Is Raise Approved?**, press the **Enter** key.

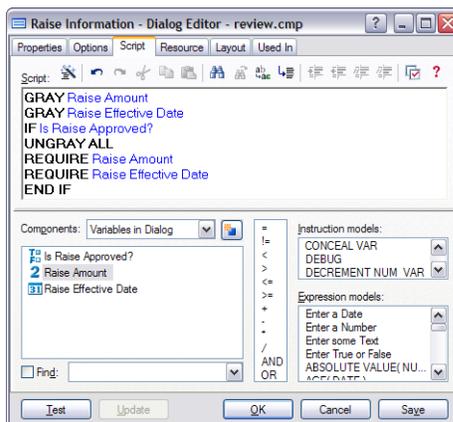
- 2 Using the auto-complete feature (**Ctrl+Spacebar**), enter the **UNGRAY** instruction, followed by the keyword **ALL**. This will ungray all of the variables that have been grayed.



- 3 On the next line of the script, enter the script, **REQUIRE Raise Amount**.



- 4 On the next line, enter the instruction, **REQUIRE Raise Effective Date**.
5 Enter the **END IF** instruction on the next line. Your script should look like this:



✓ You could use the **UNGRAY VAR** instruction to ungray each variable individually. However, **UNGRAY ALL** is more efficient.

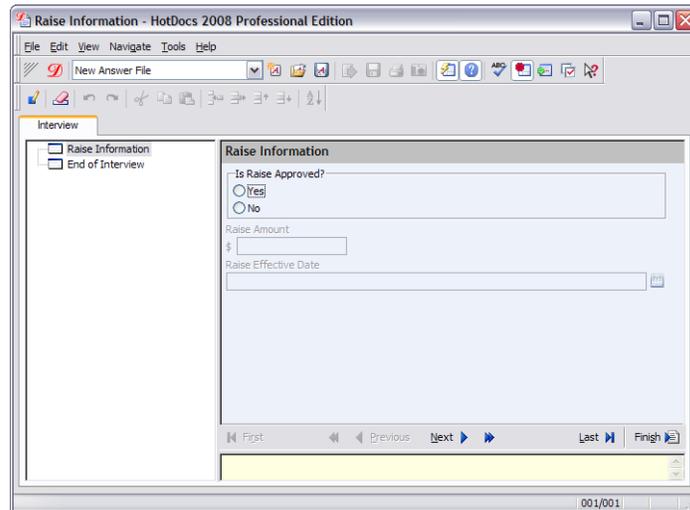
✓ To format the script so that any instructions or variables within an IF instruction are indented, click the **Auto Format** button.

Test the Script

Once you have created a script, it is a good idea to test your work. You do this by clicking **Test** at the **Dialog Editor**.

To test your dialog script

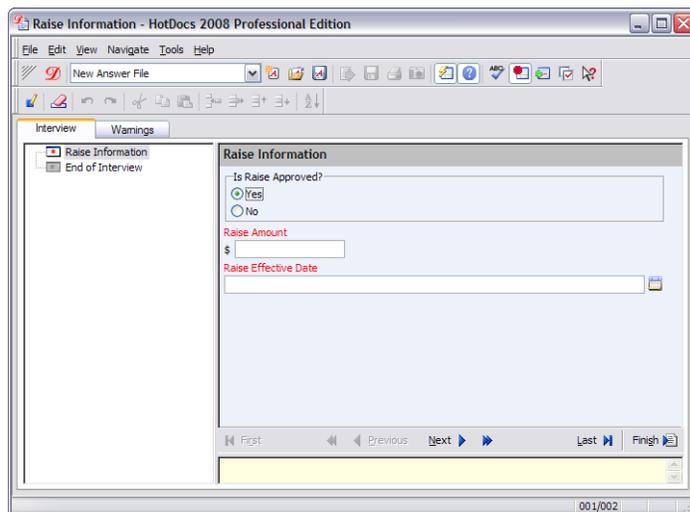
- 1 Click **Test**. A test assembly window appears.



The second and third variable questions should be grayed. To see if the conditional part of the script works, you must meet the condition and then try to exit the dialog without answering the required questions.

- 2 Select **Yes** to answer the variable question *Is Raise Approved?* and to meet the condition specified by the IF instruction.

The second and third questions are ungrayed and HotDocs marks the required questions in both the interview outline and in the dialog. This means that questions in this dialog must be answered before you can move to the next dialog.



- 3 Do not answer any questions and click **Next**.

HotDocs displays a message saying you must answer the questions before proceeding to the next dialog in the interview.

- 4 Click **OK** and enter the required answers. Notice that you are able to advance to the next dialog, and the text changes to black, indicating that answers have been entered for them.
- 5 Close the test assembly window to return to the **Dialog Editor**.
- 6 Click **OK** to close the **Dialog Editor** and then close **Component Manager**.
- 7 Click the  **Save and Close** button to close the template.

You are now finished with this lesson.

Lesson 16: Create Lists Within a List

Overview

In this lesson, you will make it possible for users to enter sublists within a list by placing REPEAT instructions inside each other. You will use a template called *Editor/Author List*, which keeps a record of the authors each editor has worked with at Hobble Creek Publishing. The list will also include a list of each author's books that each editor has edited.

You will create this three-level list by nesting REPEAT instructions inside each other. There are two parts to nesting REPEAT instructions:

- Insert REPEAT instructions in the template.
- Insert repeated dialogs into each other.

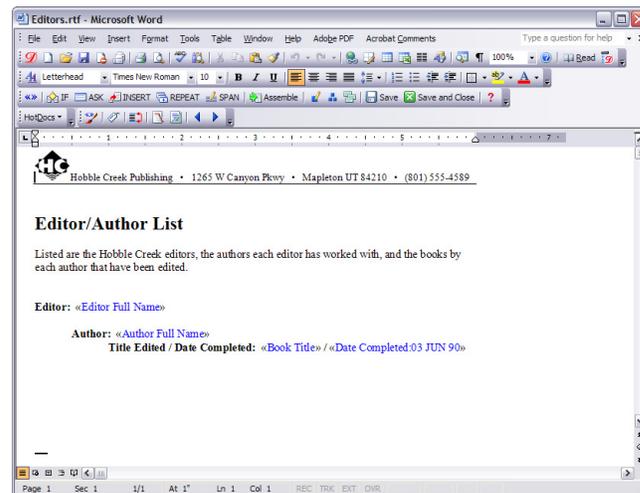
You will work with the *Editor/Author List* template for this part of the tutorial.

To open the Editor/Author List template

- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs library appears.
- 2 From the **HotDocs Tutorial Files** folder, select **Editor/Author List**.

 If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library** (**File** menu), select the HotDocs library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click Open. Then complete Step 2.

- 3 Click  **Edit**. The template appears, ready for you to edit.



Insert REPEAT Instructions

As the template is now, you can enter only one answer for each variable. However, you may want to enter multiple answers for each variable. For example, you may want to include a list of editors and under each editor, list several authors who have

worked with each editor. Likewise, each author may have written more than one book. To create such a list, you must first insert the necessary REPEAT instructions into the template.

To insert the REPEAT instructions into the template

- 1 Select the three lines of text that begin with **Editor**, **Author**, and **Title Edited/ Date Completed**, followed by the blank line below the last line of text. (Selecting the blank line will separate editors' information with an additional line.)

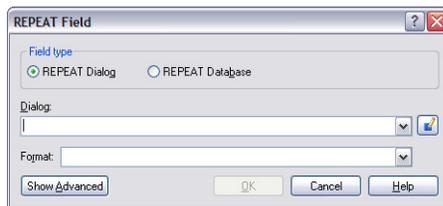
If you are using Word, the selected text should look like this:

```
Editor: «Editor Full Name»
Author: «Author Full Name»
Title Edited / Date Completed: «Book Title» / «Date Completed:03 JUN 90»
```

If you're using WordPerfect, the selected text should look like this:

```
Editor: <Editor First Name> <Editor Last Name>
Author: <Author First Name> <Author Last Name>
Title Edited / Date Completed: <Book Title> / <Date Completed:03 JUN 90>
```

- 2 Click the  REPEAT Field button. The REPEAT Field dialog box appears.



- 3 Click the **Dialog** drop-down button, select **Editor Information**, and then click **OK**. The *Editor Information* REPEAT instruction appears around all three levels of the list.

```
«REPEAT Editor Information»
Editor: «Editor Full Name»
Author: «Author Full Name»
Title Edited / Date Completed: «Book Title» / «Date Completed:03 JUN 90»
«END REPEAT»
```

- 4 Starting with the line containing the second level of the list, **Author**, select the text and variables up to the *«END REPEAT»*.

If you're using Word, the selected text should look like this:

```
«REPEAT Editor Information»
Editor: «Editor Full Name»
Author: «Author Full Name»
Title Edited / Date Completed: «Book Title» / «Date Completed:03 JUN 90»
«END REPEAT»
```

✓ The *Editor Information* dialog, which has already been created, contains the variable for the first level of the list, *Editor Full Name*.

If you're using WordPerfect, select the entire first line, starting at the left margin. The selected text should look like this:

```
<REPEAT Editor Information>
Editor: <Editor First Name> <Editor Last Name>
Author: <Author First Name> <Author Last Name>
Title Edited / Date Completed: <Book Title> / <Date Completed:03 JUN 90>
<END REPEAT>
```

✓ The *Author Information* dialog contains the variable for the second level of the list, *Author Full Name*.

- 5 Click the  REPEAT Field button. The REPEAT Field dialog box appears.
- 6 Select **Author Information** from the Dialog drop-down list and click OK.



The REPEAT *Author Information* instruction appears around the text and variables starting with the sublist *Author*.

- 7 If you are using Word, position your cursor before the «REPEAT Author Information» instruction and press the **Backspace** key so the REPEAT instruction is aligned with the left margin. (Doing this ensures the sublists are properly tabbed in the assembled document.) If you are using WordPerfect, do nothing.

On Your Own

✓ The *Book Information* dialog contains the variables for the third level of the list, *Book Title* and *Date Completed*.

Insert a REPEAT instruction for the sublist *Title Edited / Date Completed*. Use *Book Information* as the repeated dialog, and, if you're using Word, make sure the REPEAT instruction is also aligned with the left margin—just like *Author Information*.

This is how the template should look after you've inserted all three REPEAT instructions. Note how the REPEAT instructions are nested three levels deep.

```
<<REPEAT Editor Information>>
Editor: <<Editor Full Name>>

<<REPEAT Author Information>>
  Author: <<Author Full Name>>

<<REPEAT Book Information>>
  Title Edited / Date Completed: <<Book Title>> / <<Date Completed:03 JUN 90>>

<<END REPEAT>>
<<END REPEAT>>
<<END REPEAT>>
```

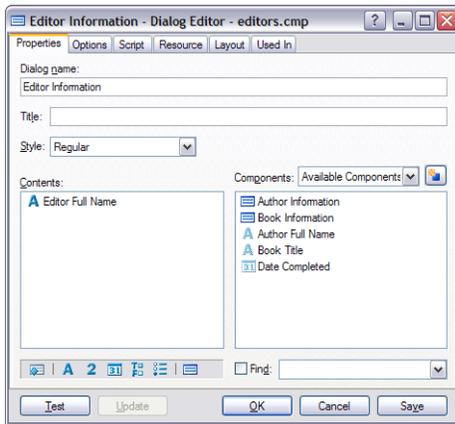
Insert the Author Information Dialog into the Editor Information Dialog

Now that you have created nested REPEAT instructions in the template, you must nest the actual dialog components so they represent the nesting shown in the template.

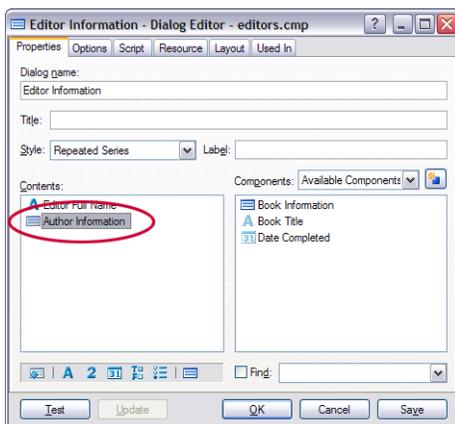
You can do this by inserting the second and third-level dialog components into the dialogs that precede them. In other words, you must insert the *Book Information* dialog into the *Author Information* dialog, and the *Author Information* dialog into the *Editor Information* dialog.

To insert the *Author Information* dialog into the *Editor Information* dialog

- 1 Click the  **Component Manager** button. The **Component Manager** window appears.
- 2 Select the dialog, **Editor Information**, from the list and click the  **Edit Component** button. The **Dialog Editor** appears.



- 3 Click the **Style** drop-down button and select **Repeated Series**. This will cause the dialog to repeat as a series of questions.
- 4 Drag **Author Information** from the **Components** list to the bottom of the **Contents** box.



Insert the Book Information Dialog into the Author Information Dialog

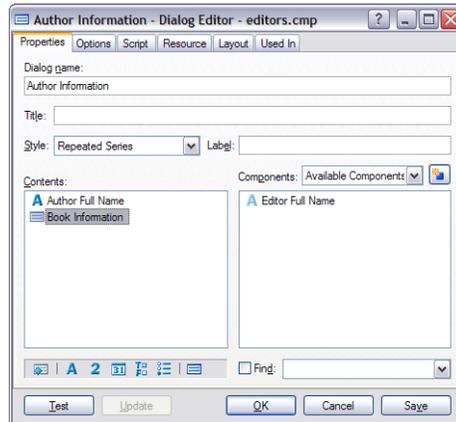
Finally, you must insert the last dialog, *Book Information*, into the *Author Information* dialog.

 During an interview, an inserted dialog appears on its parent dialog as an icon. Users click the icon and the inserted dialog appears. When they finish answering questions in the inserted dialog, they are returned to the parent dialog.

 **Repeated Series** and **Spreadsheet** are two style options for repeated dialogs. One of these two options must be selected in order for repeated dialogs to work.

To add Book Information dialog to Author Information

- 1 At the **Dialog Editor for Editor Information**, double-click the **Author Information** icon in the **Contents** list. The **Dialog Editor** appears, showing the **Author Information** dialog.
- 2 Click the **Style** drop-down button and select **Repeated Series**.
- 3 Drag **Book Information** from the **Components** list into the **Contents** box.



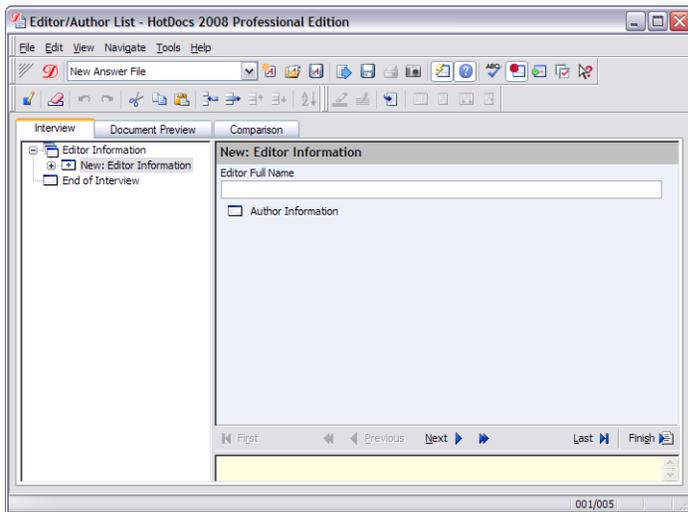
- 4 Double-click **Book Information**. The **Dialog Editor for Book Information** appears.
- 5 Click the **Style** drop-down button and select **Spreadsheet on Parent**. (Selecting **Spreadsheet on Parent** displays the dialog—in spreadsheet format—directly on the *Author Information* dialog, thus eliminating an extra mouse click for the user.)
- 6 Click **OK**.
- 7 At both the **Author Information** and the **Editor Information** dialog boxes, click **OK**. Then close Component Manager.

Test the Nested Repeat Instructions

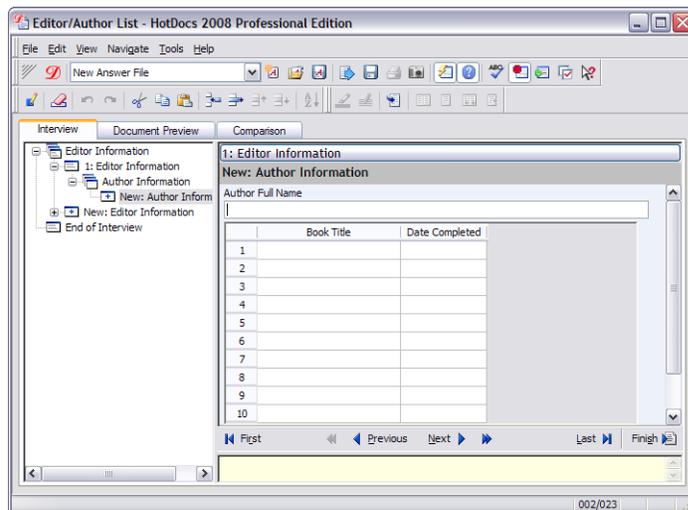
As you already know, it is always useful to test any complex work you do when automating a template to make sure your template works correctly.

To test a nested repeat instruction

- 1 At the template, click the  **Assemble** button. A test assembly window appears.



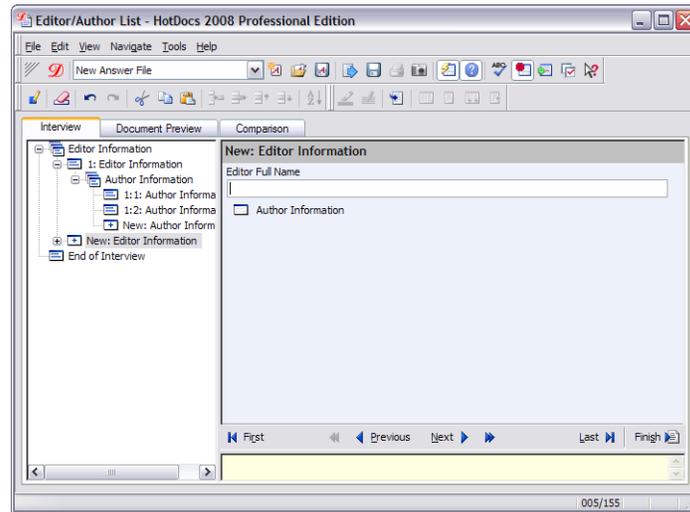
- 2 Enter an editor at the **Editor Information** dialog, and click the **Author Information** child dialog icon to enter the author information. The **Author Information** dialog appears.



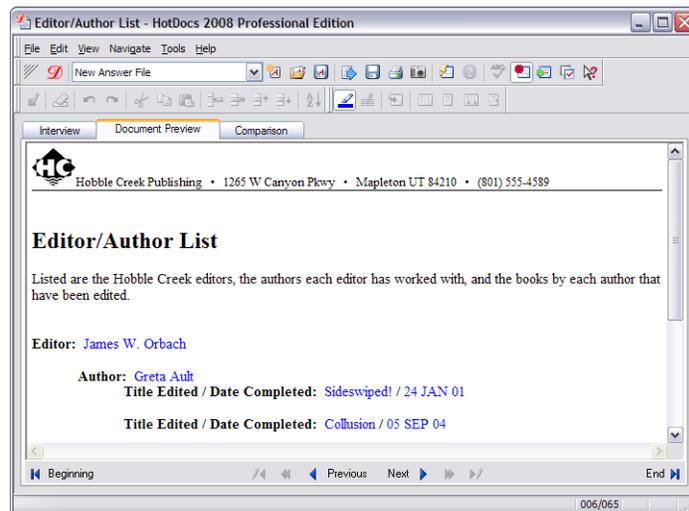
Notice that instead of a child dialog icon appearing for the *Book Information* dialog, a spreadsheet appears instead. This is because you selected **Spreadsheet on Parent** at the **Style** drop-down list.

- 3 Enter the author information, along with a list of book titles and their completion dates.
- 4 Click  **Next** to enter information for a second author along with that author's book information.

- 5 After you've entered the information for the second author's books, click **Next** at an empty dialog. You are taken to the next repetition of the **Editor Information** dialog.



- 6 Enter information for a second editor, along with that editor's author and book information.
- 7 Continue this process for as many editors (and corresponding authors and books) as you want. Eventually, you will see several levels of answered dialogs in the interview outline.
- 8 Click the **Document Preview** tab when you're finished entering author and book information for the last editor in your list. A preview of the assembled document appears, with a three-level list.



- 9 Choose **Close** (File menu) to close the test assembly window. You don't need to save your answers.
- 10 Click the **Save and Close** button to close the template.

You are finished with the HotDocs tutorial. You can now complete the HotDocs Automator tutorial, where you will learn how to automate PDF templates.

For additional information on using HotDocs, please refer to the HotDocs Help.

Chapter 3: HotDocs Automator Tutorial

Overview

This tutorial introduces you to important concepts and features available in HotDocs Automator. Before completing this tutorial, you should complete the lessons in Chapter 2, since many of the ideas and concepts you see in the following lessons build upon the things you learned in Chapter 2.

These lessons are short and are structured so you can complete each lesson in separate sessions. However, you must complete the lessons in the order they are presented.

 The document used in this tutorial was drafted to suit the purposes of the tutorial. It is not provided as a valid legal document.

Lesson 1: Create a Form Template File

Overview

Using HotDocs® Automator, you can create templates based on electronic forms. (Forms are documents that contain underlying text and graphics that cannot be changed. Examples include loan applications, tax forms, and pre-printed court forms.)

An electronic form can be created in almost any Windows application from which you can print, such as a word processor or a document design program. However, to automate one of these electronic documents, the document must first be converted to PDF (using the HotDocs PDF printer driver) and then saved as a form template.

 Using Windows Explorer, you can also change the file name extension of the PDF file to HPT. Then, when you open the file, it is already in a format you can automate.

In this first lesson, you will open a PDF file using HotDocs Automator and then save it as a form template file. You will use this form template throughout the rest of this tutorial.

 To complete Lesson 1 and create a form template file, you must use HotDocs PDF Advantage, Professional Edition. If you do not have HotDocs PDF Advantage but you would still like to learn to automate a form template, please skip to Lesson 1a, where you will prepare an already created form template for automation. Then, please contact your HotDocs sales representative for more information on obtaining a license for PDF Advantage.

Save a PDF Document as a PDF Form Template File

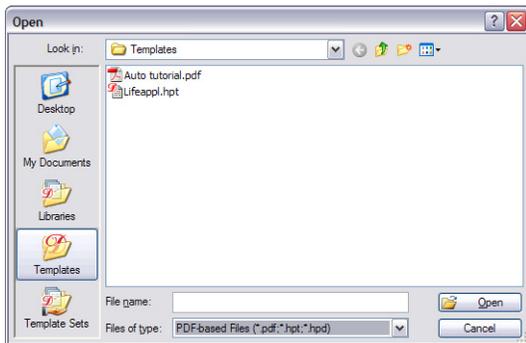
If you are using HotDocs Professional with HotDocs PDF Advantage, you can open an existing PDF document in HotDocs Automator and save it as a PDF template.

 Complete instructions for generating PDF documents from forms you design yourself are included in the HotDocs Help.

To open a PDF document and save it as a template file

- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.

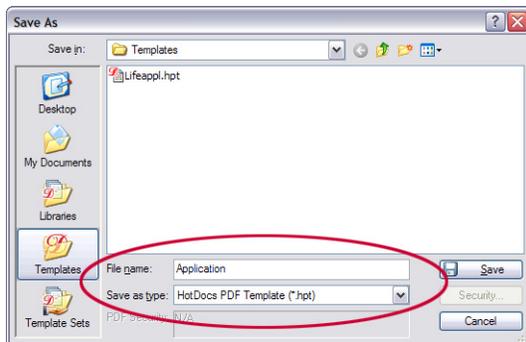
- 3 Click the **Files of type** drop-down button and select **PDF Files**. HotDocs displays a list of PDF files saved to the default *Templates* folder.



- 4 Select **auto tutorial.pdf** and click **Open**. HotDocs opens the *Life Insurance Application* in HotDocs Automator.

The file is still in PDF format, so you must save it as a form template file.

- 5 Choose **Save As** (File menu). The **Save As** dialog box appears.
- 6 In the **File name** box, type **Application**.



- 7 Click **Save**. HotDocs saves the file as a PDF form template. It is now ready for you to automate.

If you do not want to go on to Lesson 2 at this time, click **Exit** (File menu) to close the template. Otherwise, proceed to Lesson 2.

 You can use the Windows **Find** command (**Start > Search > For Files or Folder**) to find the file if it is not in this location.

Lesson 1a: Use an Already-Created Form Template



Complete Lesson 1a only if you do not have a license for HotDocs PDF Advantage but you still want to learn how to automate a form template.

Overview

In earlier versions of HotDocs, users had the option of creating *new* form templates using the HotDocs HFT driver. However, use of this driver was only supported on Windows 98 and Windows Me operating systems. Starting with the release of HotDocs 2006, HotDocs is no longer supported for use on Windows 98/Me, which means support for the HFT driver was also eliminated. However, users can still automate *existing* form template (.HFT) files.

For the purposes of this tutorial, we have included an HFT file for you to automate. You must rename this template before you can proceed to Lesson 2.

To rename the Auto Tutorial.hft template

- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. **HotDocs Automator** opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Auto tutorial.hft** from the default *Templates* folder and click **Open**. The **Life Insurance Application** form opens.
- 4 Click **Save As** (File menu). The **Save As** dialog box appears.
- 5 Type **Application.hft** in the **File name** box and click **Save**. The file is saved.

If you do not want to go on to Lesson 2 at this time, click **Exit** (File menu) to close the template. Otherwise, proceed to Lesson 2.

Lesson 2: Create Fields and Attach Text Variables

Overview

The underlying text of a form template is static, meaning the text and graphics cannot be changed, and nothing can be entered or changed on the form. In order to enter information on the form, you must create fields that overlay the template. These fields are dynamic and information (answers) entered in them can be modified as needed.

Once you create fields, you can attach variables to each field. A variable determines the kind of answer that can be merged into a field. There are seven types of variables: Text, Number, Date, True/False, Multiple Choice, Computation, and Personal Information.

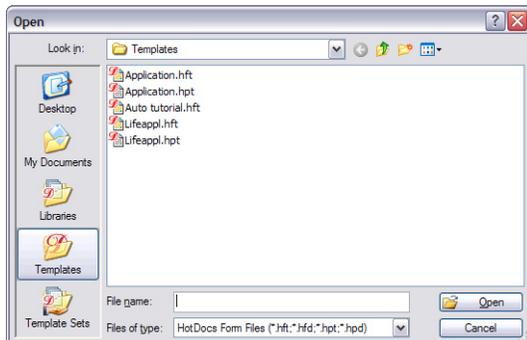
In this lesson, you will learn how to create variable fields. You will also learn how to create Text variables and attach them to fields. (Other lessons will teach you how to create the other types of variable fields.)

If you are continuing immediately from Lesson 1, skip the instructions for opening the template and proceed to “Create Simple Text Fields.”

If you closed the template at the end of Lesson 1, complete the following steps.

To open the Life Insurance Application template you created in Lesson 1

- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.



- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears, ready for you to automate.

Create Simple Text Fields

The first place where information needs to be added to the form is the *Name* blank.

 If HotDocs is already running, choose **HotDocs Automator** from the library window **Tools** menu.

 You can also open a form template by pressing **Ctrl+O** at the Automator window.

To create a HotDocs field in the *Name* blank

- 1 Make sure the  **Select Tool** button is selected in the HotDocs Automator toolbar and place the cursor directly below the word *Name*.

LIFE INSURANCE APPLICATION

Name	SSN	Application Date
Address	Phone No.	Birth Date
	Age	Gender

- 2 Press and hold down the left mouse button and move the cursor to the lower-right corner of the blank. Field borders will appear as you drag the cursor.

LIFE INSURANCE APPLICATION

Name	SSN	Application Date
Address	Phone No.	Birth Date
	Age	Gender

- 3 Release the mouse button. The new field is created and the field is automatically selected, as indicated by the handles on the field borders.

LIFE INSURANCE APPLICATION

Name	SSN	Application Date
Address	Phone No.	Birth Date
	Age	Gender

If you want to adjust the size of the field, click one of the handles and drag.

On Your Own

 You can use the **Page Up/Page Down** keys or the **Home/End** keys to adjust the height and width of a field, respectively. To move a field up, down, left, or right, use the arrow keys on your keyboard.

The second blank that needs a field is the *Address* blank. Create a field in the *Address* blank. (Follow the instructions for creating the field for the *Name* blank, earlier.)

Make sure the field height is large enough to contain a three-line answer. Remember, you can resize the field by clicking one of the handles and dragging.

Auto-Detect Fields

You can create a field by manually drawing the field with your mouse cursor (like you did for the *Name* and *Address* blanks), or you can have HotDocs create a field by automatically detecting the borders surrounding the blank. You will use this method to place a field in the *SSN* blank.

To auto-detect a field for the *SSN* blank

- Click in the *SSN* blank and click the  **Detect Field** button. (Or, double-click in the *SSN* blank.) HotDocs creates a field in the blank.

On Your Own

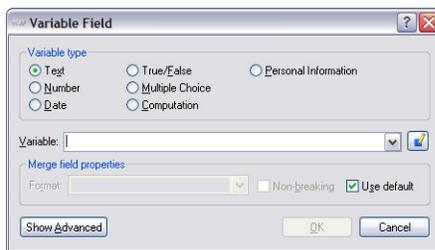
The next blank that needs a field is the *Phone No.* blank. Create a field in the *Phone No.* blank using the auto-detect feature. (Follow the instructions for creating the field for the *SSN* blank, earlier.)

Attach Text Variables to Fields

Once you have created a field, you can attach a variable to it. (Since fields don't have names, the tutorial refers to fields by the label of the corresponding blank on the form. For example, the field in the *Name* blank will be called the *Name* field.)

To attach a variable to the *Name* field

- 1 Click the *Name* field to select it and click the **Variable Field** button. (Or, double-click the field.) The **Variable Field** dialog box appears.



The dialog box lists several types of variables. The *Name* field should contain text, so you will attach a Text variable to it.

- 2 Make sure **Text** is selected from the **Variable type** group.
- 3 Type **Applicant Name** in the **Variable** box.



- 4 Click **OK**. The variable *Applicant Name* is created and attached to the *Name* field.

LIFE INSURANCE APPLICATION

Name	SSN	Application Date
Applicant Name	Phone No.	Birth Date
Address	Age	Gender

During assembly, the variable prompts the user for an answer, which is then merged into that field. Other fields that use the same variable will also use that answer.

✓ With existing fields, you can select a field and click **Detect Field** to have HotDocs resize it. This can make the field fit more tightly in its allotted space.

✓ If you have completed the tutorials for automating a text template, you will notice that variable and other component-editing dialog boxes are the same in both Automator and in the word processor.

✓ To attach a variable to a field, you can also select a field and press **Enter**.

✓ You can also right-click on any field and select **Edit Component** from the shortcut menu.

On Your Own

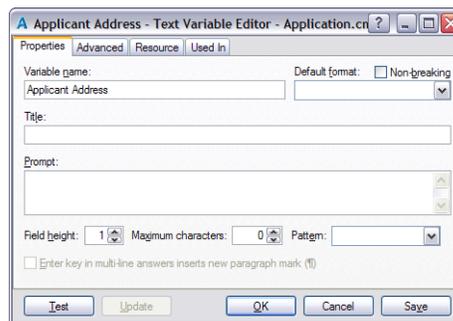
The *Address* field also needs a Text variable attached to it. Create the Text variable *Applicant Address*. (Follow the instructions for creating the variable *Applicant Name*, earlier.)

Allow Multi-Line Text Answers

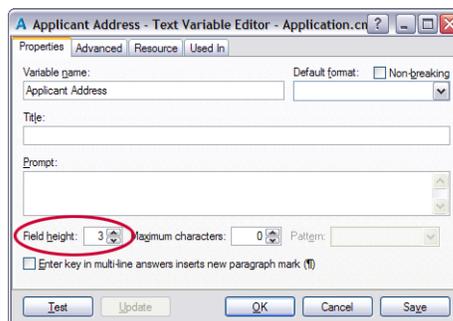
When you created the *Address* field, you made it tall enough to allow multiple lines of text. However, when you attach a Text variable to the field, the variable, by default, will only allow you to enter a single line of text. This means you can't enter the different address lines. You must change the default setting to allow multiple lines.

To allow hard returns in a text answer field

- 1 Select the *Address* field and click the  **Edit Component** button (found in the Automator toolbar). The **Text Variable Editor** appears.



- 2 At the **Field height** box, click the up arrow until the number **3** appears.



- 3 Click **OK** at the **Text Variable Editor**.

Now, users can enter hard returns to enter multiple lines of text in the answer.

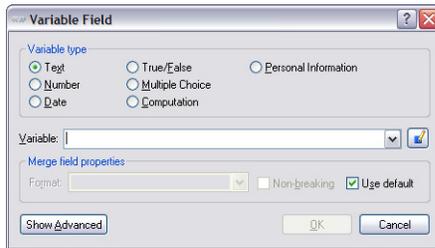
Format Answers

You might think that the *SSN* field should have a Number variable attached to it. However, Number variables are usually reserved for numbers that can be added, multiplied, divided, etc. In the case of Social Security numbers, telephone numbers, and other non-calculated numbers, you should use Text variables.

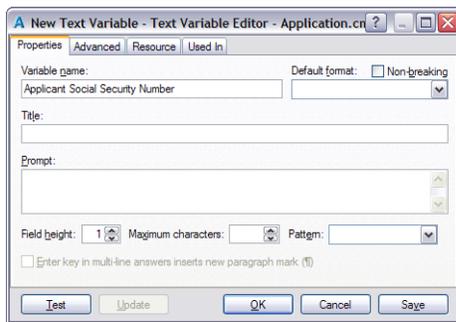
You could create a basic Text variable for the SSN field and rely on users to type the hyphens in the appropriate places. However, you can specify a pattern that controls what users can type, how the answer looks as they type it, and how the answer will be merged into the assembled document.

To create a Text variable with a pattern and attach it to the SSN field

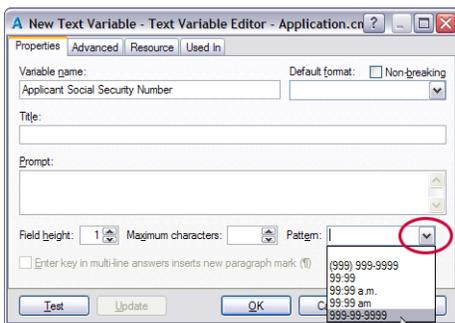
- 1 Select the SSN field and click the **Variable Field** button. (Or, double-click the field.) The **Variable Field** dialog box appears.



- 2 Make sure **Text** is selected, and type **Applicant Social Security Number** in the **Variable** box.
- 3 Click the **Edit Component** button, which is located to the right of the **Variable** box. The **Text Variable Editor** appears.



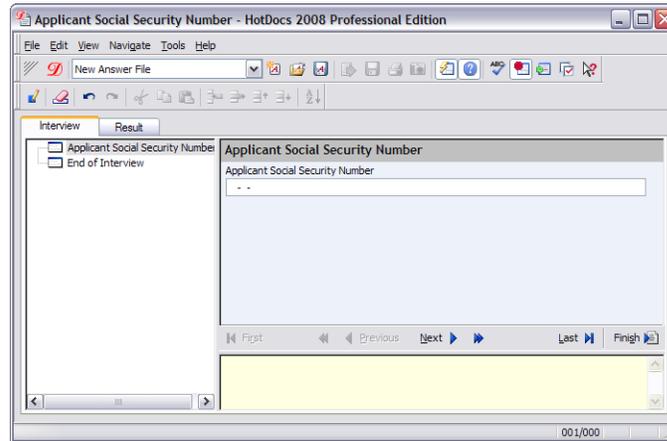
- 4 Click the **Pattern** drop-down button to display a list of patterns.



- 5 Select **999-99-9999**.

✓ You can create your own pattern by typing it in the **Pattern** box. (See the HotDocs Help for specific information on creating custom patterns.)

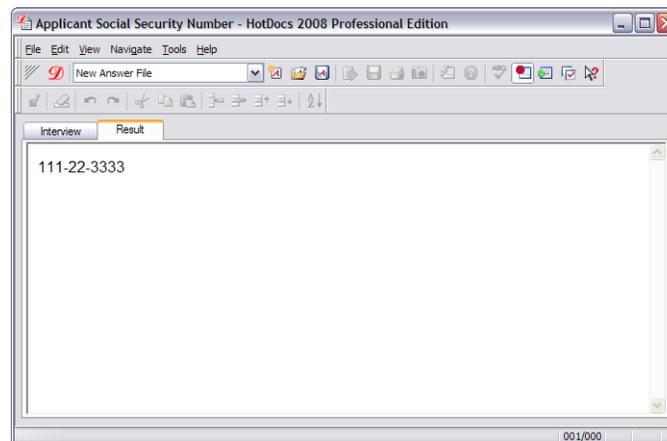
- 6 Click **Test**. A test assembly window appears, showing the Social Security number question. Two hyphens appear in the answer field, just as they will appear in the field on the form during direct-fill assembly.



- 7 Enter 111223333.

HotDocs automatically puts the first three characters before the first hyphen, the next two characters between the two hyphens, and the last four characters after the second hyphen.

- 8 Click the **Result** tab. HotDocs shows you how the answer will be merged into the assembled document.



- 9 Choose **Close** (File menu) to close the test assembly window. HotDocs asks if you want to save your answers.

- 10 Decide whether to save your answers, based on the following information:

- If you click **Save** or **Save As**, you will be prompted to specify an answer file name and title. Type **Test Answer File** in the **File name** box and accept the suggested **Title**. From this point on, this answer file will be used each time you test a variable or test assemble the template, and you will no longer be prompted to save your answers after each test. (You can, of course, choose a different answer file when you are at the test assembly window.)

✓ For more information on using test answer files, see the HotDocs Help.

- If you click **Don't Save**, each time you test a variable or test assemble the template, you will use an empty, untitled answer file. You will also be prompted to save your answers each time you finish a test.

11 Click **OK** at both the **Text Variable Editor** and the **Variable Field** dialog box. The variable is created and attached to the field.

On Your Own

The *Phone No.* field also needs a variable with a pattern. Create the Text variable *Applicant Telephone Number* with the pattern *(999) 999-9999*. (Follow the instructions for creating the variable *Applicant Social Security Number*, earlier.)

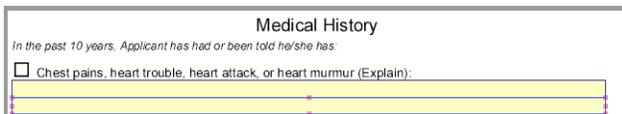
Group Two Text Fields as a Run-On

Sometimes an answer for a Text variable must span two fields, meaning an answer must wrap from one field to the next. To allow for this, you must group the separate fields as a run-on group. When two or more fields are grouped, HotDocs treats the grouping as if it were one field. Once the fields are grouped, you can attach a Text variable to it.

In the *Medical History* section of the template, the user must enter explanations for any existing health conditions. For each health condition, you will create two text fields, group them as a single text field, and then assign a Text variable to the group.

To group two text fields

- 1** On the line below *Chest pains, heart trouble, heart attack, or heart murmur*, create a field. (Remember, you can either double-click on the line, or you can click on the line and click the  **Detect Field** button.)
- 2** On the second line, create another field.



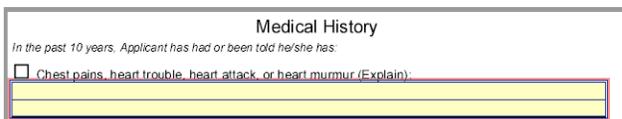
Medical History

In the past 10 years, Applicant has had or been told he/she has:

Chest pains, heart trouble, heart attack, or heart murmur (Explain):

Two yellow text fields are shown below the prompt, one on each line.

- 3** With the second field already selected, press the **Ctrl** key and click the first field. Both fields should now be selected.
- 4** Click the **Field** menu and select **Group**. A bounding box appears around both fields, indicating that the fields are now grouped.



Medical History

In the past 10 years, Applicant has had or been told he/she has:

Chest pains, heart trouble, heart attack, or heart murmur (Explain):

The two yellow text fields are now enclosed in a red rectangular bounding box, indicating they are grouped.

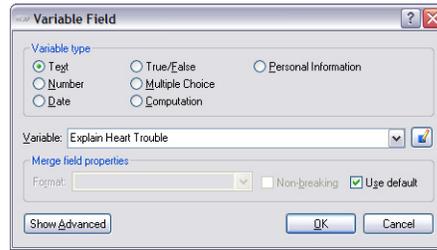
The next step is to attach a Text variable to the field.

To attach a Text variable to the run-on group

- 1** With the grouped fields selected, click the  **Variable Field** button. The **Variable Field** dialog box appears.

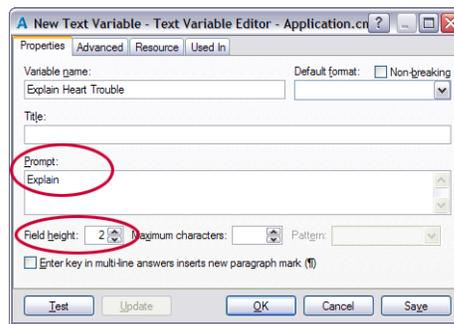
 You can also right-click on a field and choose **Group** from the shortcut menu.

- 2 Make sure **Text** is selected from the **Variable type** group and type **Explain Heart Trouble** in the **Variable** box.



- 3 Click the **Edit Component** button. The **Text Variable Editor** appears.

- 4 Type **Explain** in the **Prompt** box, and enter the number **2** in the **Field height** box.



- 5 Click **OK** at the **Text Variable Editor** as well as at the **Variable Field** dialog box.

On Your Own

Repeat this process for each remaining medical condition by creating and grouping text fields, and then attaching Text variables to the grouped fields. Use the variable names, prompts, and field heights given in the following table. (Follow the instructions in “Group Two Text Fields as a Run-On” on page 125.)

Variable Name	Prompt	Field Height
Explain High Blood Pressure	Explain	2
Explain Diabetes	Explain	2
Explain AIDS	Explain	2

When you are finished, the *Medical History* section of the template should look like this:

The screenshot shows a form titled "Medical History" with the instruction "In the past 10 years, Applicant has had or been told he/she has:". Below this are four checkboxes, each followed by a text field for explanation. The first checkbox is "Chest pains, heart trouble, heart attack, or heart murmur (Explain):" with a text field containing "Explain Heart Trouble". The second is "High blood pressure, cancer, or tumors (Explain):" with a text field containing "Explain High Blood Pressure". The third is "Diabetes, pneumonia, or disorder of the lymph system (Explain):" with a text field containing "Explain Diabetes". The fourth is "AIDS, AIDS-related complex, or immune system disorder (Explain):" with a text field containing "Explain AIDS".

Click the  **Save Form** button to save your work.

If you do not want to go on to Lesson 3 at this time, click **Exit** (**File** menu) to close the template.

Lesson 3: Attach Date and Computation Variables

Overview

This lesson teaches you how to attach Date and Computation variables to fields. Date variables store dates. Computation variables store formulas used to calculate values, such as a time period subtracted from a date, or two numbers multiplied together.

If you are continuing immediately from Lesson 2, skip the instructions for opening the template and proceed to “Attach Date Variables.”

If you closed the template at the end of Lesson 2, complete the following steps.

To open the Life Insurance Application template you used in Lesson 2

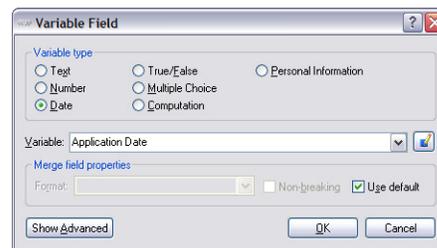
- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears, ready for you to automate.

Attach Date Variables

The first place that needs a Date variable attached is the *Application Date* blank.

To attach a Date variable to the *Date* blank

- 1 Create a field in the *Application Date* blank. (Review the first part of Lesson 2 if you don't remember how to do this.)
- 2 With the field selected, click the **<> Variable Field** button. (Or, double-click the field.) The **Variable Field** dialog box appears.
- 3 Select **Date** in the **Variable type** box, and type **Application Date** in the **Variable** box.



- 4 Click **OK** at the **Variable Field** dialog box. The variable is attached to the *Application Date* field.

On Your Own

Create a field for the *Birth Date* blank and then create and attach the Date variable *Applicant Birth Date*. (Follow the instructions for creating the Date variable *Application Date*, earlier.)

Compute Numbers

One of the most useful features of HotDocs is its ability to compute values. In this form, you must calculate the age of the applicant.

When writing computation scripts, HotDocs requires you to use a specific syntax, or language. To help you learn this syntax, HotDocs provides you with instruction and expression models. Instruction models tell HotDocs to perform various kinds of tasks, while expression models help HotDocs generate values or answers.

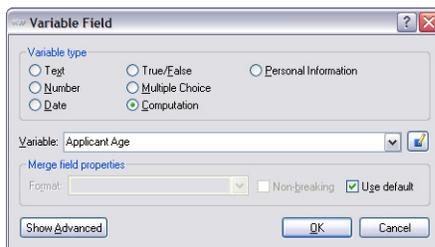
Models usually contain placeholders, which you must replace with a value. Often, these values are variables, but they can also be literal values you type, or other expressions or instructions. A model cannot work until all of its placeholders are replaced with values.

As you are writing a computation script, you can access lists of instruction and expression models. These lists appear at the bottom of the Computation Editor. To help you quickly type the script, you can also use an auto-complete list of instruction and expression keywords.

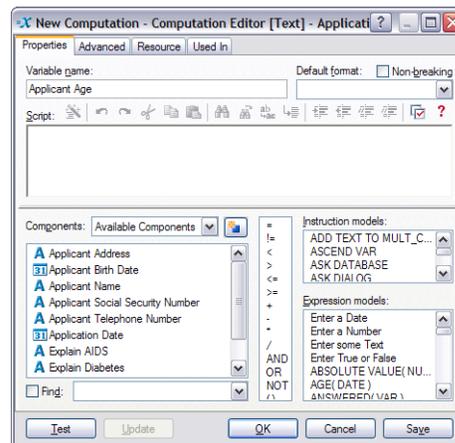
 For more information on creating computation scripts, see “Lesson 9: Use Instruction and Expression Models” on page 61.

To create the Computation variable

- 1 Create a field in the *Age* blank.
- 2 With the field selected, click the **Variable Field** button. (Or, double-click the field.) The **Variable Field** dialog box appears.
- 3 Select **Computation** and type **Applicant Age** in the **Variable** box.



- 4 Click the  **Edit Component** button. The **Computation Editor** appears.

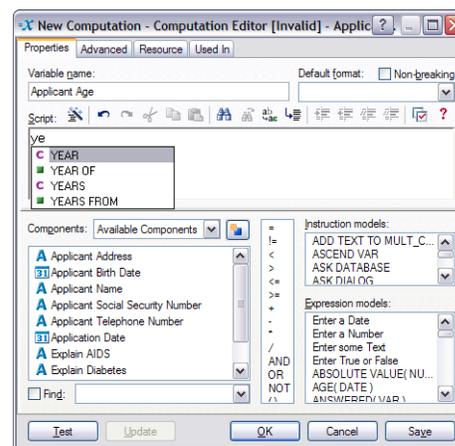


✓ Another expression that would calculate the same result is the `AGE(DATE)` expression.

To calculate the applicant's age, you will use the `YEARS FROM(DATE, DATE)` expression, which computes the number of years between two dates.

To compute the age of the applicant

- 1 In the **Script** box, press **Ctrl+Spacebar**, then type the letters **Y** and **E**. HotDocs displays a list of keywords and components containing this combination of letters.



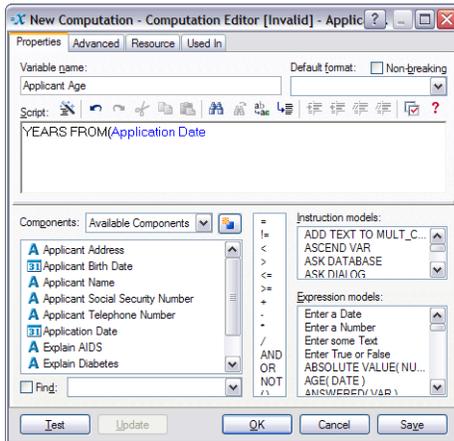
✓ You can also display syntax hints by pressing the **F7** key whenever the cursor is in the instruction or expression keyword.

- 2 Use the **DOWN ARROW** key to select (highlight) **YEARS FROM** and press **Enter**. HotDocs merges the expression into the **Script** box
- 3 Type an opening parenthesis. When you do this, HotDocs displays a syntax hint.

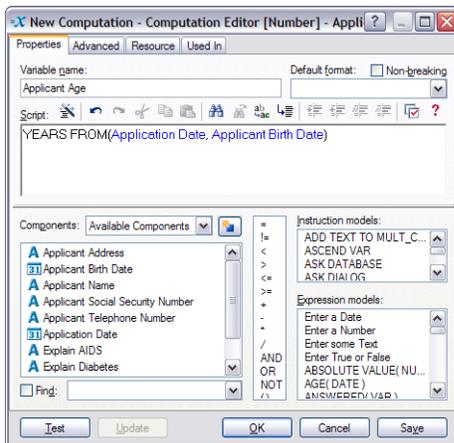
Syntax hints show you the instruction or expression in its entire syntactical format, including showing any placeholders you must replace with values. This expression requires two date values.

- 4 Press the **F5** key. HotDocs displays a list of all the components in the template.

- 5 Using the DOWN ARROW key, select **Application Date** and press **Enter**. HotDocs merges the variable into the script.



- 6 Type a comma, followed by a space, and then press **F5** again. HotDocs displays a list of all the components in the template.
- 7 Select **Applicant Birth Date** and press **Enter**. The variable is merged.
- 8 Type a closing parenthesis. The script should look like this:



Now, when a user provides a date for both the *Application Date* and the *Applicant Birth Date*, HotDocs will use this script to determine how many years there are between these two dates.

Test the Computation

You can test the computation to make sure it works correctly.

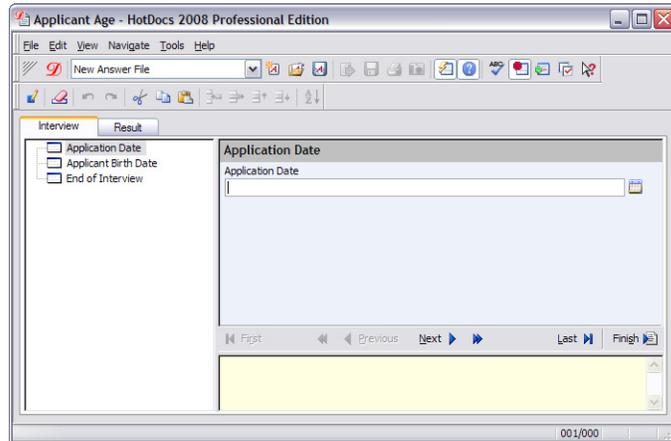
✓ This variable calculates the applicant's age by computing the number of years between the applicant's birth date and the application date.

✓ To quickly enter today's date, type the letter **t** in the answer field.

✓ To save your work, you can also press **Ctrl+S**.

To test the computation script

- 1 At the **Computation Editor**, click **Test**. A test assembly window appears, displaying the *Application Date* answer field.



- 2 Enter today's date and click **Next**. HotDocs reformats your answer and keeps you at the dialog so you can verify the change.
- 3 Click **Next** again. The *Applicant Birth Date* answer field appears.
- 4 Enter your own birth date and click the **Result** tab. HotDocs again reformats your answer.
- 5 Click the **Result** tab again. You should see your age displayed.
- 6 Choose **Close** (File menu).
- 7 Click **OK** at the **Computation Editor**.
- 8 Click **OK** at the **Variable Field** dialog box.
- 9 Click the **Save Form** button to save your work.

If you do not wish to go on to Lesson 4 at this time, click **Exit** (File menu).

Lesson 4: Attach Multiple Choice Variables

Overview

This lesson teaches you how to attach Multiple Choice variables to fields. A Multiple Choice variable lets the user choose an answer from a list of options. Specifically, in this lesson, you will create a Multiple Choice variable that merges a selected option into an answer field. You will also create a Multiple Choice variable that allows the user to select one of a group of check boxes.

If you are continuing immediately from Lesson 3, skip the instructions for opening the template and proceed to “Create a Multiple Choice Variable.”

If you closed the template at the end of Lesson 3, complete the following steps.

To open the Life Insurance Application template you used in Lesson 3

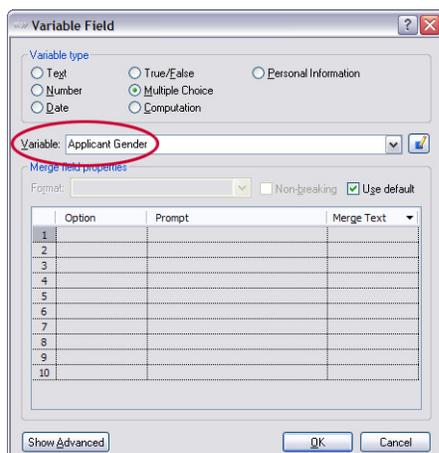
- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears, ready for you to automate.

Create a Multiple Choice Variable

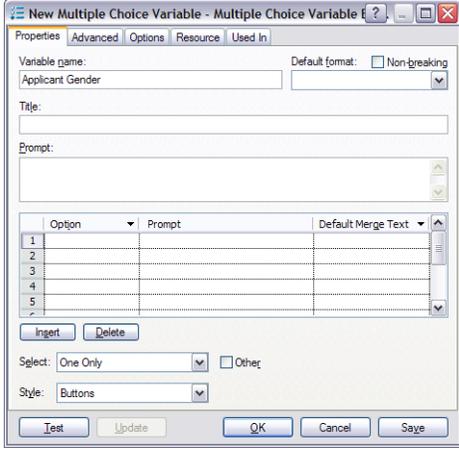
Multiple Choice variables are useful when the user’s answer has to be one of a limited number of answers. You will use a Multiple Choice variable to specify the gender of the applicant.

To create a Multiple Choice variable

- 1 Create a field in the *Gender* blank and open the **Variable Field** dialog box.
- 2 Select **Multiple Choice** and type **Applicant Gender** in the **Variable** box.



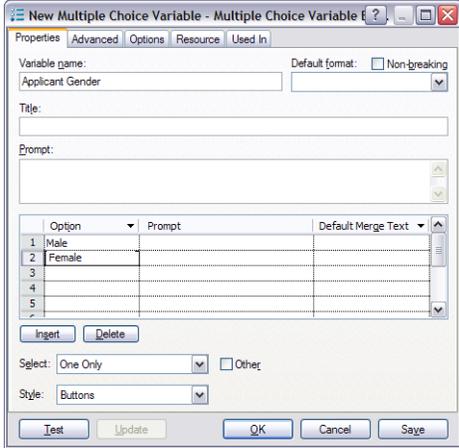
- 3 Click the  **Edit Component** button. The **Multiple Choice Variable Editor** appears.



The dialog box shows the following fields and options:

- Variable name: Applicant Gender
- Default format: Non-breaking
- Title: (empty)
- Prompt: (empty)
- Table with columns: Option, Prompt, Default Merge Text
- Buttons: Insert, Delete
- Select: One Only, Other
- Style: Buttons
- Buttons: Test, Update, OK, Cancel, Save

- 4 Type **Male** in the first row of the **Option** column and **Female** in the second row. These are the options from which the user will choose.



The dialog box shows the following fields and options:

- Variable name: Applicant Gender
- Default format: Non-breaking
- Title: (empty)
- Prompt: (empty)
- Table with columns: Option, Prompt, Default Merge Text
- Buttons: Insert, Delete
- Select: One Only, Other
- Style: Buttons
- Buttons: Test, Update, OK, Cancel, Save

- 5 Click **OK** at both the **Multiple Choice Variable Editor** and the **Variable Field** dialog box.

The variable is attached to the *Gender* field. Because you did not enter merge text (or text that will be inserted into the document if the user chooses that particular option), if the user chooses *Male*, the word *Male* will be merged into the field. If the user chooses *Female*, the word *Female* will be merged.

Attach Multiple Choice Variables to Check Boxes

In addition to merging text into an edit field, you can use a Multiple Choice variable to let the user select one of a group of check boxes. This form uses five check boxes to specify the applicant's marital status. Each check box's label should be specified as an option for the variable.

To group check box fields

- 1 Place the mouse pointer in the *Single* check box and double-click. HotDocs creates a check-box field.

Check-box fields are a different color than the other fields you have created. HotDocs determines whether a field should be a check box based on the size of the field.

- 2 Create the remaining *Marital Status* check boxes.
- 3 When all the check box fields have been created, select all of them by holding down the **Ctrl** key and clicking each one.

		Age Applicant	Gender Applicant Gender
Amount of Coverage \$		Monthly Contribution to Cash Account \$	
Marital Status		<input type="checkbox"/> Applicant is Smoker	
<input checked="" type="checkbox"/> Single	<input checked="" type="checkbox"/> Married	<input checked="" type="checkbox"/> Separated	<input checked="" type="checkbox"/> Divorced <input checked="" type="checkbox"/> Widowed
Medical History			

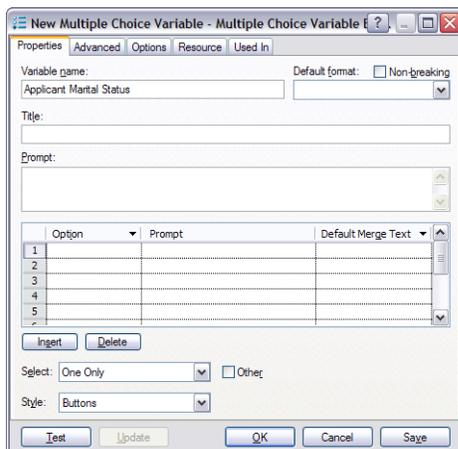
- 4 With the fields selected, click **Group** (Field menu). A bounding box appears around the five fields.

		Age Applicant	Gender Applicant Gender
Amount of Coverage \$		Monthly Contribution to Cash Account \$	
Marital Status		<input type="checkbox"/> Applicant is Smoker	
<input checked="" type="checkbox"/> Single	<input checked="" type="checkbox"/> Married	<input checked="" type="checkbox"/> Separated	<input checked="" type="checkbox"/> Divorced <input checked="" type="checkbox"/> Widowed
Medical History			

Now that you have grouped the check box fields, you can assign a Multiple Choice variable to the group. (Like grouped text fields, grouped check-box fields are treated as one field and can have only one variable attached to them.)

To attach a Multiple Choice variable to the *Marital Status* check boxes

- 1 Open the **Variable Field** dialog box.
- 2 Select **Multiple Choice** and type **Applicant Marital Status** in the **Variable** box.
- 3 Click the  **Edit Component** button. The **Multiple Choice Variable Editor** appears.

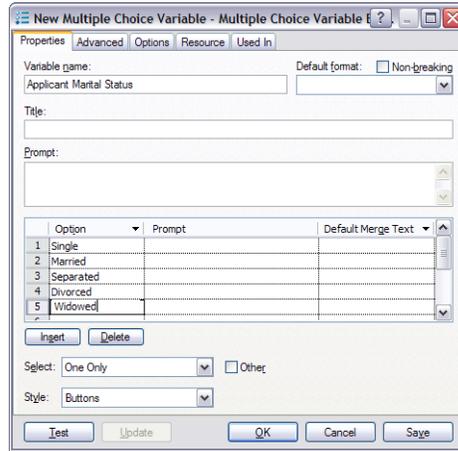


Option	Prompt	Default Merge Text
1		
2		
3		
4		
5		

- 4 Type **Single** in the first row of the **Option** column.

 Another way to select multiple fields is to press **Ctrl** as you use the mouse to draw a box around the fields.

- 5 Type **Married** in the second row of the **Option** column.
- 6 In rows three, four, and five, type the remaining options, **Separated**, **Divorced**, and **Widowed**, in this order.



- 7 Click **OK** at both the **Multiple Choice Variable Editor** and the **Variable Field** dialog box.

The Multiple Choice variable is now attached to the five *Marital Status* check boxes. When a user chooses one of the options, an *X* will be merged into the corresponding check-box field.

When you are finished, click the  **Save Form** button to save your work.

If you do not want to go on to Lesson 5 at this time, click **Exit** (File menu).

Lesson 5: Attach Number Variables

Overview

This lesson teaches you how to attach Number variables to fields.

If you are continuing immediately from Lesson 4, skip the instructions for opening the template and proceed to “Create Number Variables.”

If you closed the template at the end of Lesson 4, complete the following steps.

To open the Life Insurance Application template you used in Lesson 4

- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears, ready for you to automate.

Create Number Variables

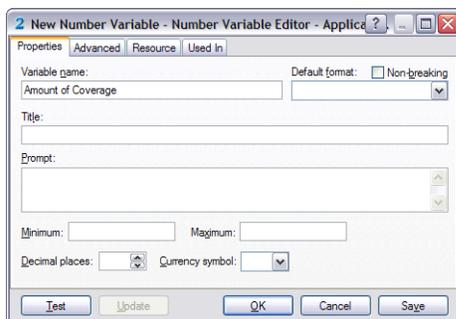
The *Amount of Coverage* blank needs a Number variable attached to it.

To create a Number variable for the *Amount of Coverage* blank

- 1 Manually create a field for the *Amount of Coverage* blank by using the  **Select Tool** button.

 Automatically detecting the field here will not work because the **\$** in the field keeps HotDocs from detecting the underlying field borders properly. This is why you must manually create the field.

- 2 Open the **Variable Field** dialog box and select **Number**.
- 3 Type **Amount of Coverage** in the **Variable** box and click the  **Edit Component** button. The **Number Variable Editor** appears.



By default, HotDocs only allows the user to enter whole numbers in the field, which, in this case, would be whole dollar amounts. To allow the user to enter cents, you can specify decimal places.

- 4 Enter 2 in the **Decimal places** box.

- 5 Select \$ from the **Currency symbol** drop-down list.

You want two decimal places to appear in the answer field even if the answer is a whole number. (By default, HotDocs merges only the decimal places if they have a value.) You can use a format to merge the entire dollar amount, even if the user doesn't enter any cents.

- 6 Click the **Default format** drop-down button and select 9,999.00.

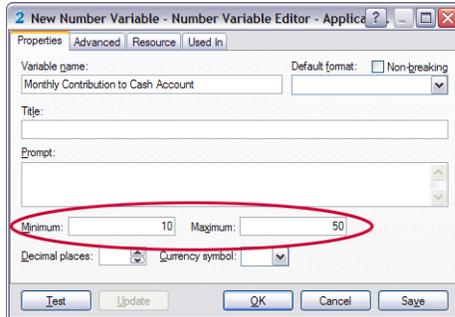
- 7 Click **OK** at both the **Number Variable Editor** and the **Variable Field** dialog box.

Set Minimum and Maximum Limits

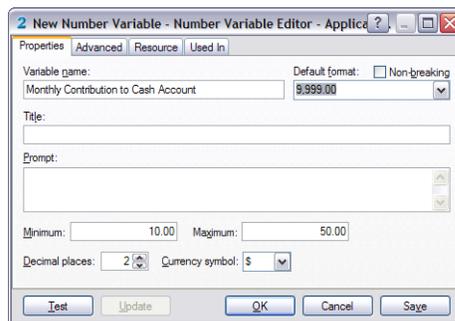
Each applicant must contribute no less than \$10 but no more than \$50 a month to a special cash account. You can help the user enter the correct number by setting minimum and maximum limits for a Number variable. When completing an interview, users cannot move to the next answer field or dialog if their answers are not within the limits.

To create a Number variable with limits for the *Monthly Contribution to Cash Account* blank

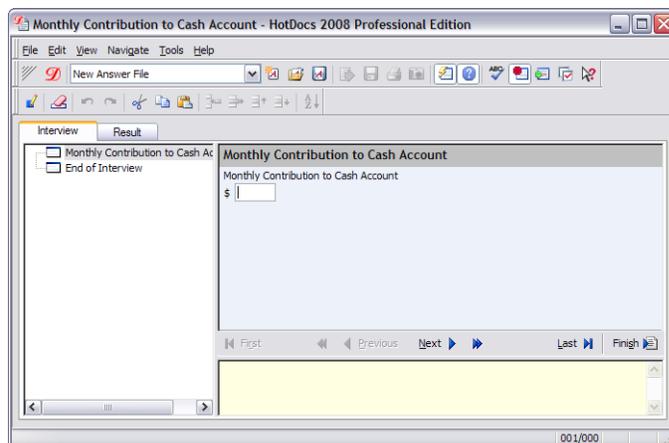
- 1 Manually create a field for the *Monthly Contribution* blank and open the Variable Field dialog box.
- 2 Select **Number** and type **Monthly Contribution to Cash Account** in the Variable box.
- 3 Click the  **Edit Component** button. The Number Variable Editor appears.
- 4 Type **10** in the **Minimum** box and **50** in the **Maximum** box.



- 5 Enter **2** in the **Decimal places** box.
- 6 Select **\$** from the **Currency symbol** drop-down list.
- 7 Select **9,999.00** from the **Default format** drop-down list.



- 8 Click **Test**. The test assembly window appears.



- 9 Type **60** and click the **Result** tab.

 Currency symbols are not merged into the assembled document. They are only used to prompt the user for the correct type of information.

 You can use the  **Clear Answer** button in the toolbar to clear the answer field.

The warning, *Please enter a number between 10 and 50*, appears.

10 Click **OK** to return to the test assembly window.

11 Clear your answer from the answer field and close the test assembly window.

12 Click **OK** at both the **Number Variable Editor** and the **Variable Field** dialog box.

13 Click the  **Save Form** button to save your work.

If you do not want to go on to Lesson 6 at this time, click **Exit** (**File** menu).

Lesson 6: Attach True/False Variables and Make Fields Conditional

Overview

This lesson teaches you how to attach True/False variables to fields. It also teaches you how to make fields conditional, which means a user can enter an answer in the field only if a certain condition is met.

If you are continuing immediately from Lesson 5, skip the instructions for opening the template and proceed to “Attach True/False Variables.”

If you closed the template at the end of Lesson 5, complete the following steps.

To open the Life Insurance Application template you used in Lesson 5

- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears, ready for you to automate.

Attach True/False Variables

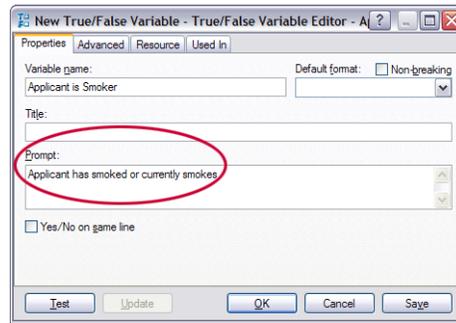
In this part of the lesson, you will use True/False variables to merge check marks into single check boxes, based on how users answer Yes/No questions. If the answer is *yes*, an X is merged in the box. If the answer is *no*, nothing is merged in.

To create a True/False variable for the Applicant is Smoker check box

- 1 Attach a field to the *Applicant is Smoker* check box and open the **Variable Field** dialog box.
- 2 Select **True/False** and type **Applicant is Smoker** in the **Variable** box.
- 3 Click the  **Edit Component** button. The **True/False Variable Editor** appears.



- 4 Type **Applicant has smoked or currently smokes** in the Prompt box.

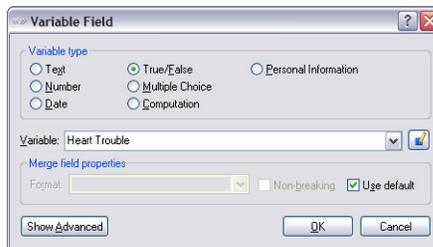


- 5 Click **OK** at both the **True/False Variable Editor** and the **Variable Field** dialog box. The variable is attached to the check box.

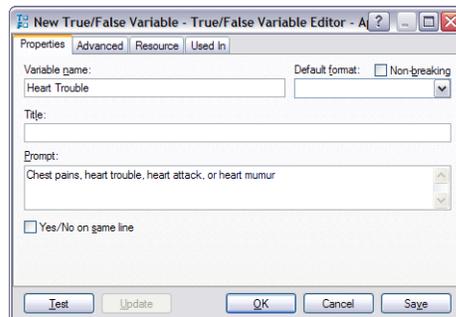
In the *Medical History* section of the application, there are four more check box fields that must be created. You will attach True/False variables to these fields. Additionally, you will specify prompts for these variables.

To create True/False variables for the health conditions check boxes

- 1 Create a field for the *Chest pains, heart trouble, heart attack, or heart murmur* check box.
- 2 Open the **Variable Field** dialog box and select **True/False**.
- 3 Type **Heart Trouble** in the **Variable** box.



- 4 Click the **Edit Component** button. The **True/False Variable Editor** appears.
- 5 Type the following in the **Prompt** box: **Chest pains, heart trouble, heart attack, or heart murmur**.



- 6 Click **OK** at both the **True/False Variable Editor** and at the **Variable Field** dialog box.

On Your Own

Using the rest of the check boxes, create fields and then attach True/False variables. Assign the corresponding prompt text to each variable:

True/False variable	Prompt text
High Blood Pressure	High blood pressure, cancer, or tumors
Diabetes	Diabetes, pneumonia, or disorder of the lymph system
AIDS	AIDS, AIDS-related complex, or immune system disorder

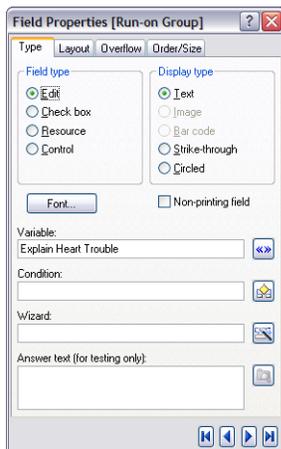
Follow the instructions for creating the True/False variable *Heart Trouble*, earlier.

Make Fields Conditional

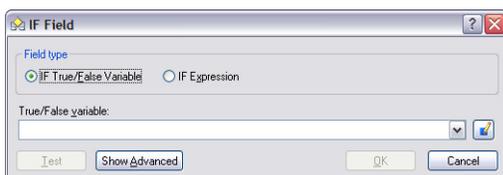
In the *Medical History* section of the template, users should only enter an explanation of a health problem if the corresponding health problem's check box is selected. To specify this requirement, you will condition the text fields on the corresponding True/False variables being true.

To make the first health problem explanation field conditional

- 1 Select the grouped field, **Explain Heart Trouble**.
- 2 Click the  **Field Properties** button. The **Field Properties** dialog box appears.



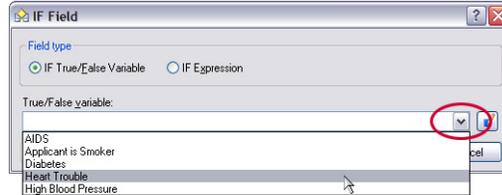
- 3 Click the  **IF Field** button next to the **Condition** box. The **IF Field** dialog box appears.



 For the most part, the properties you specify at the **Properties** dialog box control how answers entered during assembly are processed and displayed in the assembled form document.

✓ The **IF True/False Variable** option is used when the conditions are a simple *either/or* choice. The **IF Expression** option is used when the conditions require more complicated logic.

- You want the user to enter an explanation of heart trouble only if the applicant has had chest pains, heart trouble, a heart attack, or heart murmur—in other words, only if the True/False variable *Heart Trouble* is true.
- 4 With **IF True/False Variable** selected from the **Field type** group, select **Heart Trouble** from the True/False variable drop-down list.



- 5 Click **OK** at the **IF Field** dialog box.

On Your Own

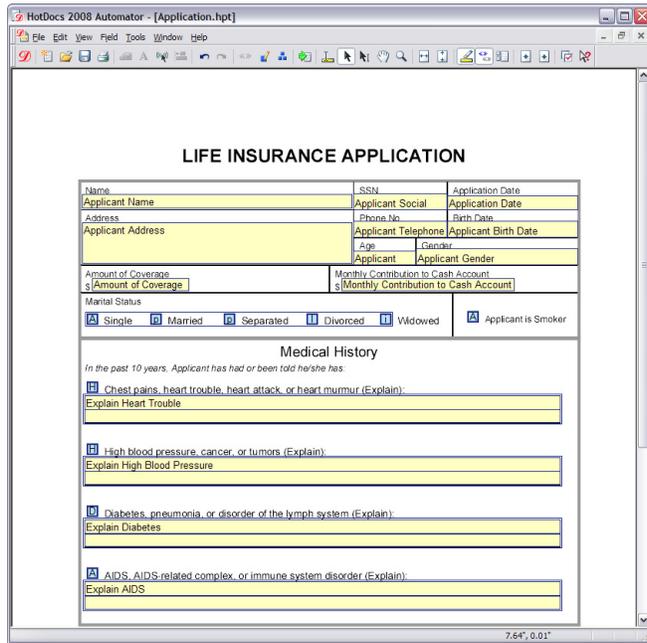
You can leave the **Field Properties** dialog box open as you modify the field properties for the remaining fields.

Follow the instructions in “Make Fields Conditional” and condition each remaining text field upon its preceding check-box field. Specifically:

- The *Explain High Blood Pressure* field should be conditioned upon the *High Blood Pressure* True/False variable.
- The *Explain Diabetes* field should be conditioned upon the *Diabetes* True/False variable.
- The *Explain AIDS* field should be conditioned upon the *AIDS* True/False variable.

When you are finished, close the **Field Properties** dialog box by clicking the **X** in the upper-right corner of the box.

Click the  Save Form button to save your work. The form template should now look like this:



LIFE INSURANCE APPLICATION

Name	SSN	Application Date
Applicant Name	Applicant Social	Application Date
Address	Phone No.	Birth Date
Applicant Address	Applicant Telephone	Applicant Birth Date
	Age	Gender
Amount of Coverage	Applicant	Applicant Gender
Amount of Coverage	Monthly Contribution to Cash Account	Monthly Contribution to Cash Account

Marital Status
 Single Married Separated Divorced Widowed Applicant is Smoker

Medical History
 In the past 10 years, Applicant has had or been told he/she has:

Chest pains, heart trouble, heart attack, or heart murmur (Explain):
 Explain Heart Trouble

High blood pressure, cancer, or tumors (Explain):
 Explain High Blood Pressure

Diabetes, pneumonia, or disorder of the lymph system (Explain):
 Explain Diabetes

AIDS, AIDS-related complex, or immune system disorder (Explain):
 Explain AIDS

If you do not want to go on to Lesson 7 at this time, click **Exit** (File menu).

Lesson 7: Create a List Using a Table

Overview

This lesson shows you how to automate tables in forms so they can gather and display lists of answers. To do this, you must first add the repeated variables to a repeated dialog. You then must attach the repeated dialog to the fields in the table.

If you are continuing immediately from Lesson 6, skip the instructions for opening the template and proceed to “Create Fields in the Table Cells.”

If you closed the template at the end of Lesson 6, complete the following steps.

To open the Life Insurance Application template you used in Lesson 6

- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears, ready for you to automate.

Create Fields in the Table Cells

The first step in automating a table is to create fields on all of the lines in the table.

To create fields in the Beneficiaries table’s cells

- One at a time, click on each line in the table and then click the  **Detect Field** button. (Or, one at a time, double-click on each line in the table.) When you are finished, the table should look like this:

Beneficiaries		
<i>List your beneficiaries in the order you want them to benefit. The second beneficiary will only benefit if the first cannot. Likewise, the third beneficiary will only benefit if the first and second cannot.</i>		
Name	Phone Number	Relationship to Applicant
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>

 While variables are only assigned to the first row of the table, answers gathered with a REPEAT instruction are merged into as many rows of the table as needed.

Attach Variables to Tables

The next step in automating a table is to attach variables to the first row of fields in the table.

To attach variables to the Beneficiaries table

- 1 Select the first field in the *Name* column.
- 2 Click the  **Variable Field** button. The **Variable Field** dialog box appears.
- 3 Make sure **Text** is selected and type **Beneficiary Name** in the **Variable** box.

- 4 Click OK. The variable is attached to the field.

Beneficiaries		
<small>List your beneficiaries in the order you want them to benefit. The second beneficiary will only benefit if the first cannot. Likewise, the third beneficiary will only benefit if the first and second cannot.</small>		
Name	Phone Number	Relationship to Applicant
Beneficiary Name		

On Your Own

Create two Text variables, *Beneficiary Telephone Number* and *Relationship to Applicant*, and attach them to the top fields in the other two columns. When creating the *Beneficiary Telephone Number* variable, specify the telephone number pattern. (Click the  **Edit Component** button to edit the component properties.)

When you are finished, the table should look like this:

Beneficiaries		
<small>List your beneficiaries in the order you want them to benefit. The second beneficiary will only benefit if the first cannot. Likewise, the third beneficiary will only benefit if the first and second cannot.</small>		
Name	Phone Number	Relationship to Applicant
Beneficiary Name	Beneficiary Telephone	Relationship to Applicant

Group the Fields

The third step in automating a table is to group the fields as a table. Before you can group the fields, however, you must first select every field in the table.

To select and then group the table fields

- 1 Press and hold the **Shift** key. Notice that the mouse cursor changes to show a pointer with a bounding box.
- 2 While still pressing **Shift**, click and hold the mouse button, dragging it from the top-left corner of the table to the bottom-right corner.

As you do this, HotDocs displays a bounding box around the group of fields.

- 3 Release the mouse button first, and then release the **Shift** key. Every field in the table is selected.
- 4 With the fields selected, choose **Group as Table** (Field menu). HotDocs places a group box around the fields.

Beneficiaries		
<small>List your beneficiaries in the order you want them to benefit. The second beneficiary will only benefit if the first cannot. Likewise, the third beneficiary will only benefit if the first and second cannot.</small>		
Name	Phone Number	Relationship to Applicant
Beneficiary Name	Beneficiary Telephone	Relationship to Applicant

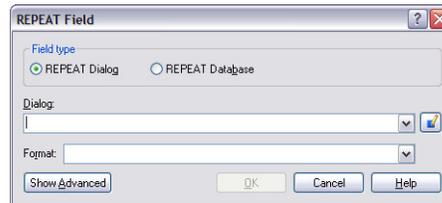
 When grouping table fields, you can also right-click and select **Group as Table** from the shortcut menu.

Create the Dialog

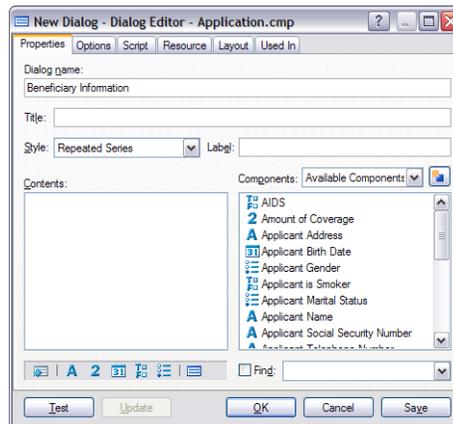
The final step is to create a repeated dialog that contains the variables used in the table. Creating the dialog identifies the variables in the table as repeating variables, or variables that store lists of answers.

To create the dialog

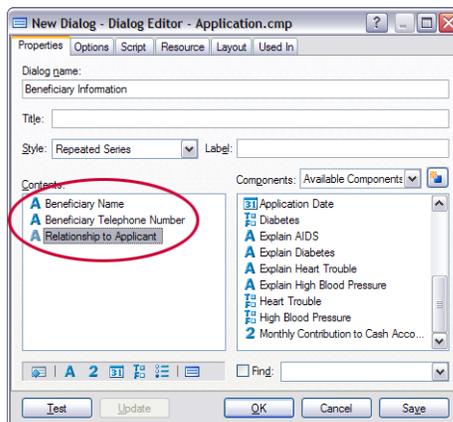
- 1 Double-click the table. The REPEAT Field dialog box appears.



- 2 Type **Beneficiary Information** in the Dialog box and click the  **Edit Component** button. The Dialog Editor appears.



- 3 From the **Components** list, drag **Beneficiary Name** into the **Contents** box. (You may have to scroll through the list to see the variable.)
- 4 Drag **Beneficiary Telephone Number** and **Relationship to Applicant** into the **Contents** box.



- 5 Make sure **Repeated Series** is selected at the **Style** drop-down list.
- 6 Click **OK** at both the **Dialog Editor** and the **REPEAT Field** dialog box.
- 7 Click the  **Save Form** button to save the template.

The form template is now completely automated for direct-fill assembly. In the next lesson, you will test this assembly method.

If you do not want to go on to Lesson 8 at this time, click **Exit** (**File** menu) to close the template.

 The **Repeated Series** style causes HotDocs to display the same information-gathering dialog over and over until all answers are entered. Another style, **Spreadsheet**, displays all of the variables in a spreadsheet so the user can enter answers in one dialog without clicking **Next** to get to the next repetition.

Lesson 8: Type Your Answers Directly on the Form

Overview

In this lesson, you will test assemble the form to see if the template is working the way you want. When you do this, you will enter your answers directly on the form fields.

If you are continuing immediately from Lesson 7, skip the instructions for opening the template and proceed to “Fill the Form Directly.”

If you closed the template at the end of Lesson 7, complete the following steps.

To open the Life Insurance Application template you used in Lesson 7

- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the HotDocs template folder and click **Open**. The template appears.

Fill the Form Directly

To fill the Life Insurance Application form

- 1 Click the  **Test Assemble** button. A test assembly window appears, displaying the document. The first field in the document, the *Name* field, should be selected so that you can enter an answer.

 If you saved your answers in a test answer file in Lesson 2, you will notice that answers you have entered are now merged into the form's fields. If you want to use a new answer file for this portion of the tutorial, click the  **New Answers** button.

- 2 Type an answer in the *Name* field and press **Tab** to move to the next field.
- 3 Continue entering answers in fields. Some fields may require a different kind of answer. In the following fields, you'll need to do something besides enter a standard text, date, or number answer:

Field	Do This
Age	Answer the two date questions and then tab to this field. HotDocs automatically calculates the answer and merges it into the field.
Gender	Click the drop-down button and select an answer.
Marital Status and Smoker check boxes	Click the check box, or press the space bar to select the check box.
Health question check boxes	Click the check box for each true statement, or press the space bar to select it.
Health explanations	Click in the corresponding check box to activate the field before you type.

- 4 When you are through directly filling the form, choose **Close** from the **File** menu of the assembly window. You are returned to the template.

If you do not want to go on to Lesson 9 at this time, click **Exit** (**File** menu).

Lesson 9: Create an Answer Wizard

Overview

This lesson teaches you how to create an answer wizard—a series of one or more dialogs that helps users provide an answer for a particular field or group of fields on the form. When a field has an answer wizard attached, a wizard button appears next to the field when the user tabs to it during direct-fill assembly. Users can click the button to see the dialog or series of dialogs so they can enter answers all at once.

You can create an answer wizard for any field. Some types of fields that commonly have wizards are fields grouped as a table, fields that appear as a computed value, and fields that have a condition attached. In this lesson, you will create an answer wizard for the *Beneficiaries* table.

If you are continuing immediately from Lesson 8, skip the instructions for opening the template and proceed to “Create an Answer Wizard for a Table.”

If you closed the template at the end of Lesson 8, complete the following steps.

To open the Life Insurance Application template you used in Lesson 8

- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears, ready for you to automate.

Create an Answer Wizard for a Table

When you test assembled the form document by entering answers directly in the fields, you could enter only three beneficiaries—one for each row of the table. If you create an answer wizard for the table, you can enter as many beneficiaries as you need. If you enter more answers than can fit in the table, you can send the answers to an addendum.

Creating an answer wizard for a table is easy because you have already created the dialog that contains all the repeated variables. All you need to do is designate that dialog as the wizard.

To create an answer wizard

- 1 Click the table to select the group of fields.

- 2 Click the  **Field Properties** button. The **Field Properties** dialog box appears.

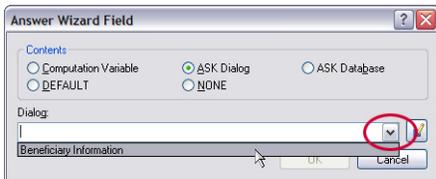


- 3 Click the  **Wizard** button. The **Answer Wizard Field** dialog box appears.



- 4 Select **ASK Dialog**.

- 5 Click the **Dialog** drop-down button and select **Beneficiary Information**.



- 6 Click **OK** at the **Answer Wizard Field** dialog box.

- 7 Close the **Field Properties** dialog box by clicking the X in the upper-right corner of the dialog box.

Test the Answer Wizard

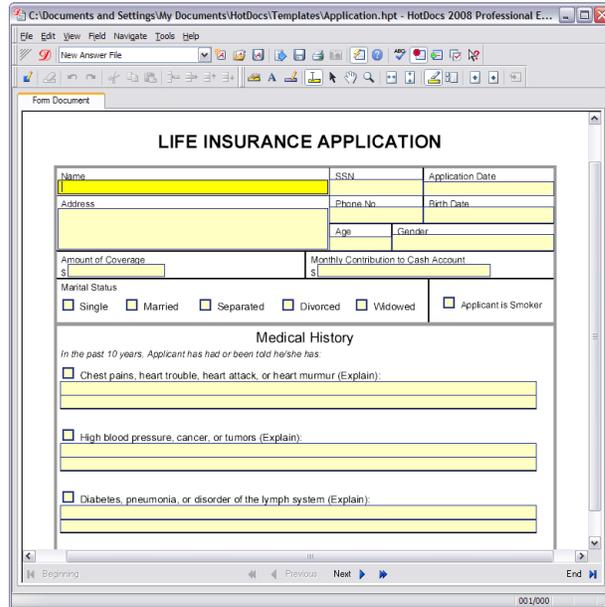
You can test the answer wizard you just created.

 To access the **Field Properties** dialog box, you can also select a field and press **Alt+Enter**.

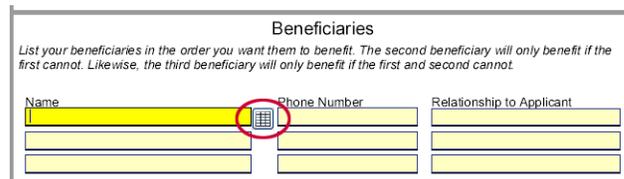
 Wizards can also be used to calculate answers. For example, when you attach a Computation variable to a field, users cannot type directly in the field during assembly. In such a situation, you could attach an answer wizard to the field that asks the variables required to perform the calculation.

To test the answer wizard

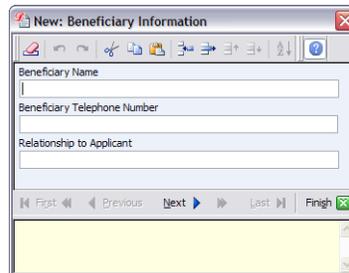
- 1 Click the  Test Assemble button. A test assembly window appears.



- 2 Click the first field in the table. A  table wizard button appears to the right of the field.



- 3 Click the **wizard** button. The **Beneficiary Information** dialog box appears.



- 4 Enter information for the first beneficiary and click Next.
- 5 Repeat this process for three more beneficiaries (for a total of four).
- 6 After entering the fourth answer, click Finish. HotDocs merges the first three sets of answers into the table.

- 7 Click anywhere outside the table. The **Overflow Status** dialog box appears.



This dialog box warns you that you have entered more answers than can fit in the table and gives you options for working with the overflow.

- 8 Select **Send the contents of the table to the addendum** and click **Close**. HotDocs sends the beneficiary information to the addendum and inserts the reference *See 1 in Addendum* in the first table field.
- 9 Click the scroll bar on the right side of the assembly window and scroll down to the addendum page. The answers from the table are listed.
- 10 Close the test assembly window to return to the template.
- 11 Click the  **Save Form** button to save your work.

If you do not want to go on to Lesson 10 at this time, click **Exit** (File menu).

Lesson 10: Create Answer-Gathering Dialogs

Overview

When you test assembled the form in the preceding lessons, you entered your answers directly in the fields on the form. In addition to this method, you can provide users with an alternate way to assemble a form document by presenting questions in an interview. This includes displaying groups of questions in dialogs. (This is how users answer questions when assembling text documents.)

When HotDocs creates a default interview, it does so by asking variables in the order it reads them in the template, which is from left to right, and top to bottom. If the variable is linked to a dialog, HotDocs displays the dialog instead.

In this lesson, you will create the dialogs for the *Life Insurance Application*. The next lesson, *Lesson 11*, shows you how to have HotDocs generate a default interview.

If you are continuing immediately from Lesson 9, skip the instructions for opening the template and proceed to “Gather Variables into Dialogs.”

If you closed the template at the end of Lesson 9, complete the following steps.

To open the *Life Insurance Application* template you used in Lesson 9

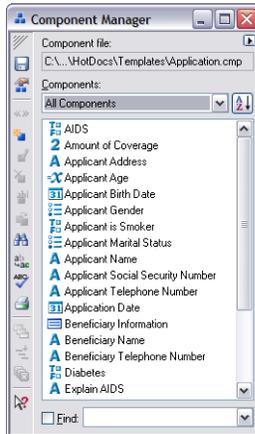
- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears, ready for you to automate.

Gather Variables into Dialogs

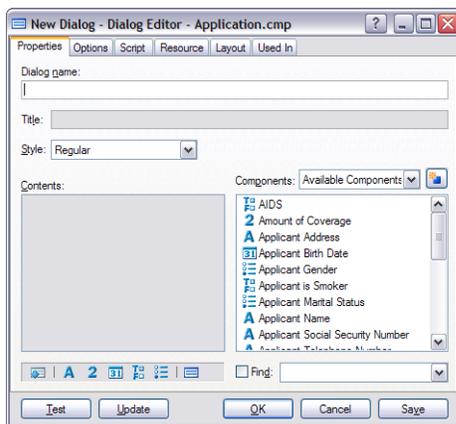
How you divide variables into dialogs depends on the template and your own preferences. Generally, you begin with the most basic information, which, in this template, is the applicant’s personal information.

To create a custom dialog containing the applicant's basic information

- 1 At the template, click the  **Component Manager** button. The **Component Manager** window appears.



- 2 Click the **Components** drop-down button and select **Dialogs**. The list shows the dialog that has already been created.
- 3 Click the  **New Component** button. The **Dialog Editor** appears.



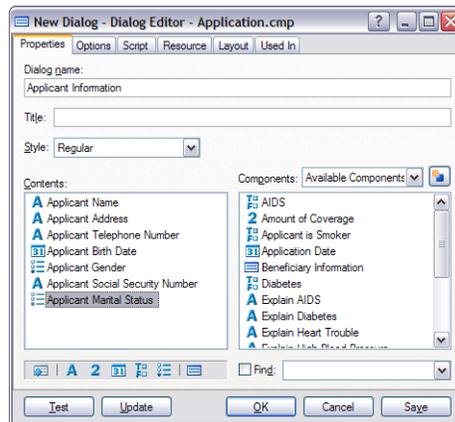
- 4 Type **Applicant Information** in the **Dialog name** box.
- 5 Drag **Applicant Name** from the **Components** list into the **Contents** box.
- 6 Drag the following variables into the **Contents** box, in this order:

Applicant Address
Applicant Telephone Number
Applicant Birth Date
Applicant Gender
Applicant Social Security Number

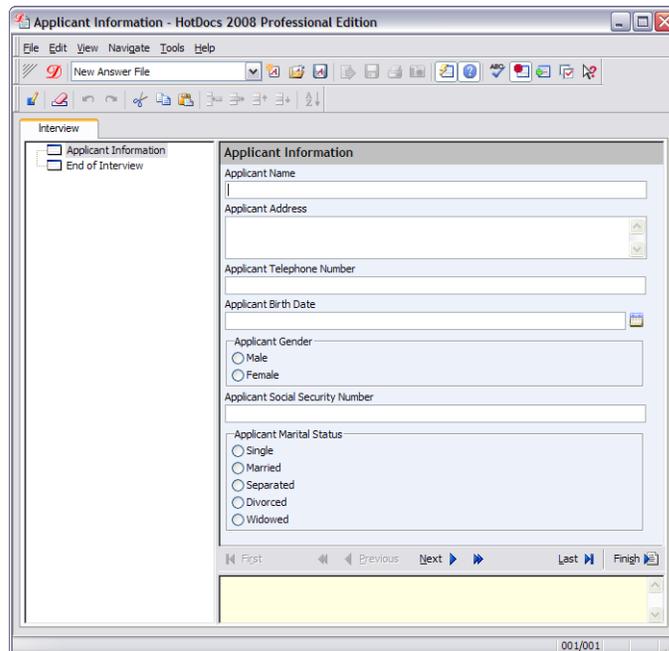
 To display **Component Manager** next to the Automator window, adjust the width of Component Manager and click the  **Arrange Windows** button. HotDocs arranges the windows so you can work in both simultaneously.

 You can assign a dialog title to a dialog by typing it in the **Title** box. The title replaces the dialog name when the dialog is presented during the interview.

Applicant Marital Status



- 7 Click **Test**. A test assembly window appears.



The dialog looks exactly the way it will during the interview.

- 8 Close the test assembly window and click **OK** at the **Dialog Editor**. (Do not close Component Manager.)

The *Applicant Information* dialog is now complete.

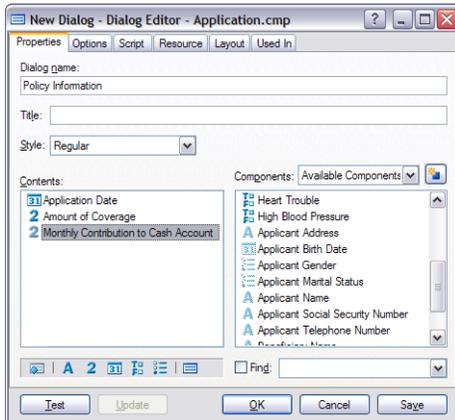
Create the Second Dialog

The next group of variables provides information about the life insurance policy.

To create a dialog containing policy information

- 1 Create a new dialog and type **Policy Information** in the **Dialog name** box. (Follow the instructions for creating the *Applicant Information* dialog.)

- 2 Drag **Application Date**, **Amount of Coverage**, and **Monthly Contribution to Cash Account** from the **Components** list into the **Contents** box.



- 3 Click **OK** when you are finished.

On Your Own

Create a dialog named **Medical History** that contains the following variables, in this order:

- Applicant is Smoker
- Heart Trouble
- Explain Heart Trouble
- High Blood Pressure
- Explain High Blood Pressure
- Diabetes
- Explain Diabetes
- AIDS
- Explain AIDS

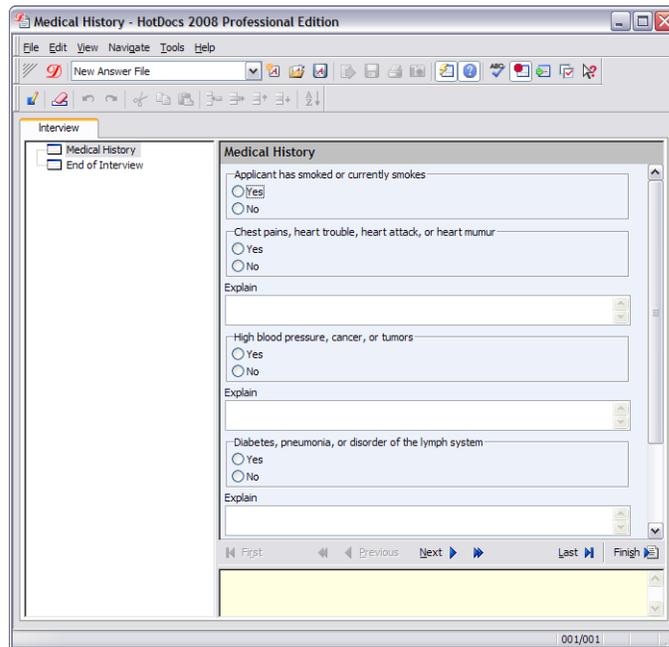
When you are finished, do *not* close the **Dialog Editor**.

Test the dialog

As you have learned, it is useful to test your work. In this part of the lesson, you will test the dialog you have just created to see how it will appear to the user.

To test the Medical History dialog

- 1 At the **Dialog Editor**, click **Test**. A test assembly window appears.



The five True/False variables create five yes/no questions. If you select *Yes* for one of these questions, an *X* will appear in the corresponding check box on the document.

This dialog will work as it is; however, you can make it more efficient by grouping the True/False variables into one multiple-selection list.

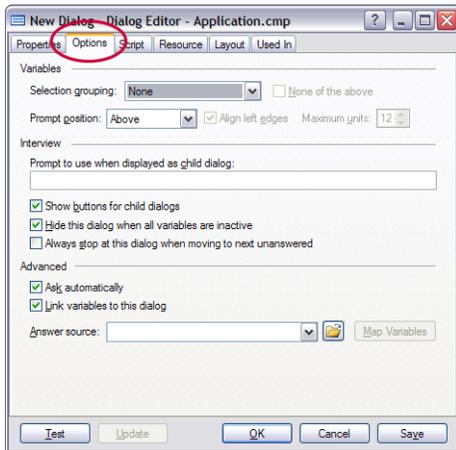
Group True/False Variables in a Dialog

You can put True/False variables into single-selection or multiple-selection groups, depending on what makes the most sense for the form.

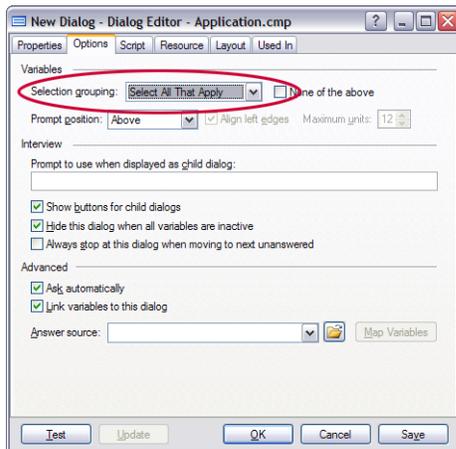
To group True/False variables

- 1 Right-click on the **Medical History** icon in the interview outline and choose **Edit Component** from the shortcut menu. The **Dialog Editor** comes to the front.

- 2 Click the **Options** tab of the **Dialog Editor**. The window changes to show several custom options.

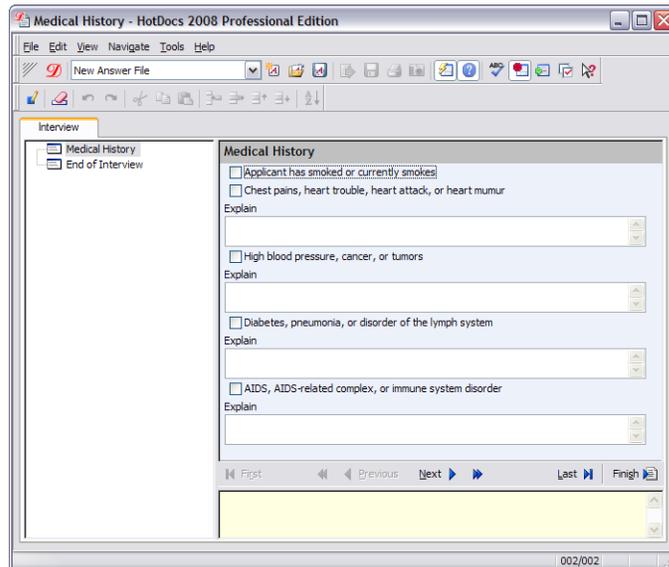


- 3 Click the **Selection grouping** drop-down button and choose **Select All That Apply**.



✓ The other **Selection grouping** option, **Select One**, groups questions so users can select only one option.

- 4 Click **Update**. The test assembly window appears again, and the appearance of the dialog has changed.



Instead of yes/no questions, each health condition is preceded by a single check box.

- 5 Close the test assembly window, and return to the **Dialog Editor**. Do *not* close the **Dialog Editor**.

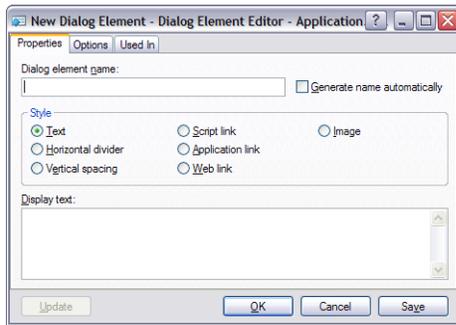
Add Dialog Elements to the Dialog

The *Medical History* dialog could be confusing to users who are new to the life insurance application. Just as you can add prompts to variables, you can add text to the dialog to guide users in answering the questions. You can also add elements such as white space to a dialog to visually arrange questions in the dialog.

To add white space to a dialog

- 1 At the **Dialog Editor**, click the **Properties** tab.
- 2 Click the  **Create New Dialog Element** button and drag it above the **Heart Trouble** variable in the **Contents** box. You will see a horizontal line between **Applicant is Smoker** and **Heart Trouble**. This indicates where the dialog element will be inserted.

- Release the mouse. The **Dialog Element Editor** appears.

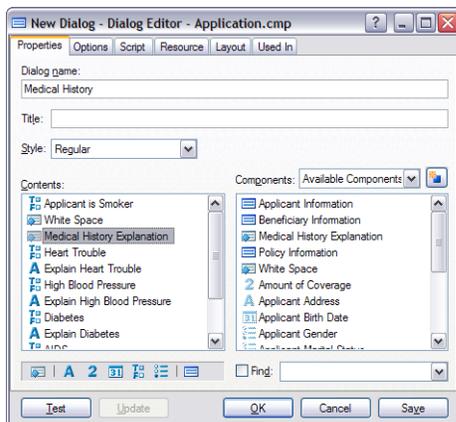


- Enter **White Space** in the **Dialog element name** box.
- In the **Style** group, select **Vertical spacing**.
- Leave the **Spacing** percentage at **100%** and click **OK** at the **Dialog Element Editor**. The dialog element is added to the dialog.

The next step is to add explanatory text to the dialog. You will do this by creating another dialog element.

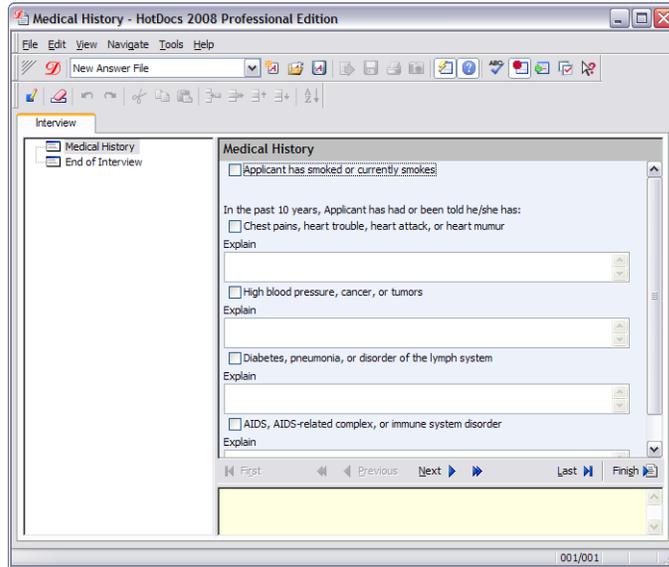
To add explanatory text to the dialog

- Click the **Create New Dialog Element** button again and drag it above the **Heart Trouble** variable but just below the **White Space** dialog element in the **Contents** box.
- Release the mouse button. The **Dialog Element Editor** appears.
- Enter **Medical History Explanation** in the **Dialog element name** box.
- In the **Display text** box, type **In the past 10 years, Applicant has had or been told he/she has:**.
- Click **OK**. The dialog element is created and added to the dialog.



- Click **Test**.

The two dialog elements appear before the list of True/False variables in the dialog. The white space separates the question about smoking from the questions about existing medical conditions.



7 Close the test assembly window.

Because you have made a lot of changes to the dialog, it is a good idea to save your work.

8 Click **Save** at the **Dialog Editor**.

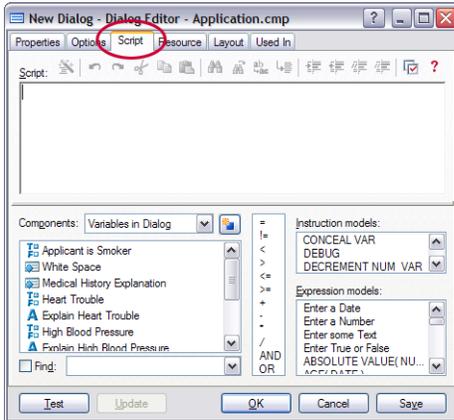
Make Variables in a Dialog Conditional

In Lesson 6, you made the text fields in the *Medical History* section of the template conditional upon their corresponding check boxes—if a user selects a check box, he or she can tab to the text field and enter an explanation for the health condition. You can also use conditional logic to make answer fields in a dialog conditional upon other questions being answered.

In this lesson, you will create a dialog script that grays the answer fields for the health explanations if the user doesn't select the preceding options.

To create a dialog script

- 1 At the **Dialog Editor**, click the **Script** tab. The window changes to show a script-writing box.



Notice that the **Components** list contains a list of all of the components used in this particular dialog. They are listed in the same order as they were added to the dialog.

- 2 Click in the **Script** box, and enter the following script:

```
GRAY Explain Heart Trouble
GRAY Explain High Blood Pressure
GRAY Explain Diabetes
GRAY Explain AIDS
```

Remember, you can quickly access the list of components by pressing either the **F5** key or **Ctrl+Spacebar**.

These instructions gray the explanation fields so the user can't enter any answers.

- 3 After the last GRAY instruction, press **Enter** and enter the following script:

```
IF Heart Trouble
UNGRAY Explain Heart Trouble
END IF
```

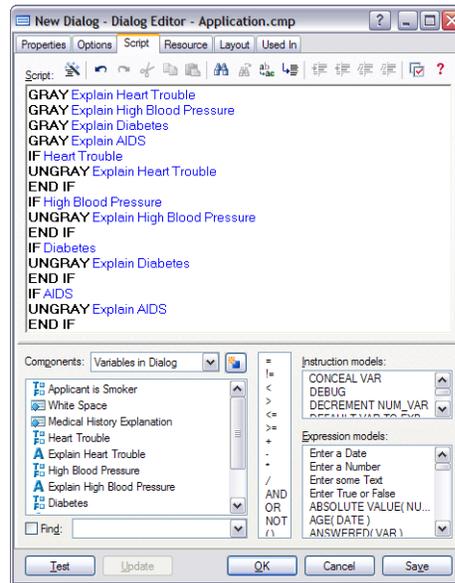
This tells HotDocs that if the user selects the **Heart Trouble** True/False variable, it should ungray the Text variable so the user can answer it.

On Your Own

Using the condition for *Heart Trouble* as an example, ungray the three remaining Text variables (*Explain High Blood Pressure*, *Explain Diabetes*, and *Explain AIDS*) based on their corresponding True/False variables being *true*.

 Drag the bottom border of the **Script** box to make the script-writing area bigger. This allows you to see more of your script.

When you are finished, the dialog script should look like this:



 To format the script so that any instructions or variables within an IF instruction are indented, click the **Auto Format** button.

If you would like, you can test the dialog to see how the script works. When you are finished testing, close the test assembly window and then click **OK** at the **Dialog Editor**.

Click the  **Save Form** button to save your work.

If you do not want to go on to Lesson 11 at this time, close the **Component Manager** (by clicking the X in the upper-right corner) and then choose **Exit** (**File** menu).

Lesson 11: Instruct HotDocs to Create a Default Interview

Overview

In this lesson, you will select a component file property that will instruct HotDocs to create a default interview for the *Life Insurance Application*. This interview will be generated using the dialogs you created in Lesson 10.

When creating a default interview, HotDocs presents to the user variables and dialogs in the order it reads them in the template, which is from left to right, and from top to bottom. In doing this, it also generates an interview outline in the left pane of the assembly window. This outline lists the names of the dialogs and gives visual information about the answered status of the variables in the dialog. You can click an item in the outline and see its corresponding answer field or fields in the dialog pane (or the right-hand pane of the window).

If you are continuing immediately from Lesson 10, skip the instructions for opening the template and proceed to “Have HotDocs Generate a Default Interview.”

If you closed the template at the end of Lesson 10, complete the following steps.

To open the *Life Insurance Application* template you used in Lesson 10

- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears, ready for you to automate.
- 4 Click the  **Component Manager** button. The **Component Manager** window opens.

Have HotDocs Generate a Default Interview

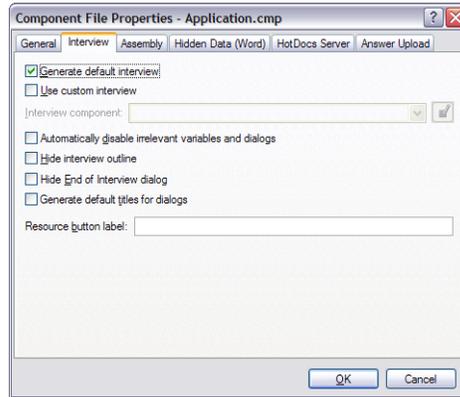
When you instruct HotDocs to generate a default interview, it does so by reading through the text of the template and asking variables in the order they appear. If the variables are linked to a dialog, it asks the dialog instead. You specify this property at the component file.

To select the property that specifies a default interview

- 1 At **Component Manager**, click the  **Component File Properties** button. The **Component File Properties** dialog box appears.
- 2 Click the **Interview** tab. The view changes to show interview options.

 You can create a custom interview for a template. Doing this would allow you to control the order dialogs are asked in the interview. For information on creating a custom interview, please see the HotDocs Help.

3 Select **Generate default interview**.



4 Click **OK**.

5 Close **Component Manager** by clicking the **X** in the upper-right corner.

Now when users assemble the document, they will, by default, enter answers using an interview. You will test this assembly method in the next lesson. (During the interview, users can still click the **Form Document** tab and enter their answers directly on the form.)

Click the  **Save Form** button to save the template, which is now completely automated.

If you do not want to go on to Lesson 12 at this time, click **Exit** (**File** menu).

Lesson 12: Assemble a Document Using an Interview

Overview

Now you can test assemble the form document using the default interview.

If you are continuing immediately from Lesson 11, skip the instructions for opening the template and proceed to “Assemble a Form Document Using an Interview.”

If you closed the template at the end of Lesson 11, complete the following steps.

To open the Life Insurance Application template you used in Lesson 11

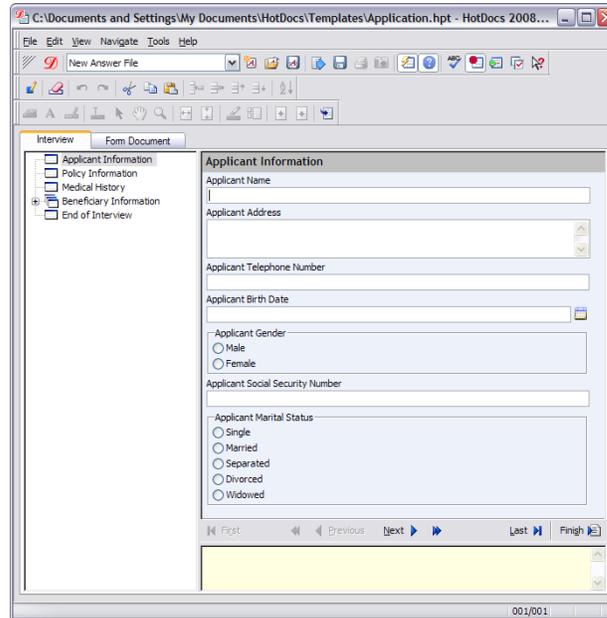
- 1 Choose **HotDocs 2008 > HotDocs Automator** from **Programs** on the **Start** menu. HotDocs Automator opens.
- 2 Click the  **Open Form** button. The **Open** dialog box appears.
- 3 Select **Application.hpt** (or **Application.hft**) in the default *Templates* folder and click **Open**. The template appears.

Assemble a Form Document Using an Interview

Once you have specified a default interview, HotDocs adds an **Interview** tab to the assembly window. (Direct-fill assembly shows only the **Form Document** tab.) The **Interview** tab shows an interview outline and the dialog pane. The interview outline displays icons for the dialogs you have created. You can click any dialog icon and view the corresponding variable or variables in the dialog pane. You enter answers in the dialog pane.

To test assemble a document using a default interview

- 1 Click the  **Test Assemble** button. A test assembly window appears, displaying the *Applicant Information* dialog in the right pane of the window. The left pane shows the interview outline.

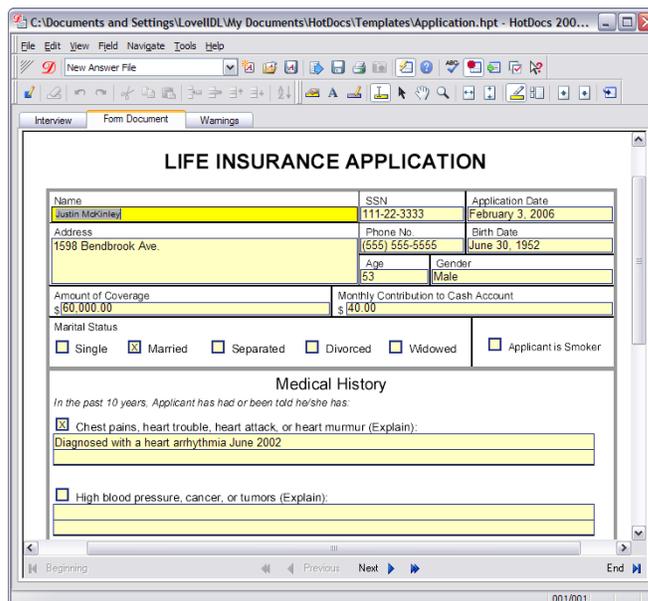


The screenshot shows the HotDocs Test Assemble window. The left pane displays the interview outline with the following items: Applicant Information, Policy Information, Medical History, Beneficiary Information, and End of Interview. The right pane shows the Applicant Information dialog with the following fields and options:

- Applicant Name: [Text Field]
- Applicant Address: [Text Field]
- Applicant Telephone Number: [Text Field]
- Applicant Birth Date: [Date Picker]
- Applicant Gender:
 - Male
 - Female
- Applicant Social Security Number: [Text Field]
- Applicant Marital Status:
 - Single
 - Married
 - Separated
 - Divorced
 - Widowed

Navigation buttons at the bottom include First, Previous, Next, Last, and Finish.

- 2 Enter the information in the answer fields and click **Next** to see the next dialog.
- 3 When you are finished, click the **Form Document** tab. The assembled form document appears, showing the answers you entered.



The screenshot shows the HotDocs Test Assemble window with the **Form Document** tab selected. The assembled form document is titled "LIFE INSURANCE APPLICATION" and contains the following information:

Name	SSN	Application Date
Justin McKinley	111-22-3333	February 3, 2006
Address	Phone No.	Birth Date
1598 Bendbrook Ave.	(555) 555-5555	June 30, 1952
	Age	Gender
	53	Male
Amount of Coverage	Monthly Contribution to Cash Account	
\$60,000.00	\$40.00	
Marital Status		
<input type="checkbox"/> Single <input checked="" type="checkbox"/> Married <input type="checkbox"/> Separated <input type="checkbox"/> Divorced <input type="checkbox"/> Widowed <input type="checkbox"/> Applicant is Smoker		
Medical History		
In the past 10 years, Applicant has had or been told he/she has:		
<input checked="" type="checkbox"/> Chest pains, heart trouble, heart attack, or heart murmur (Explain):		
Diagnosed with a heart arrhythmia June 2002		
<input type="checkbox"/> High blood pressure, cancer, or tumors (Explain):		

Navigation buttons at the bottom include Beginning, Previous, Next, and End.

- 4 Close the test assembly window to return to the template, and then click **Exit** (**File** menu) to close the template.

Now that you have automated the template and test assembled it to make sure it creates a correctly assembled document, you can add the template to a HotDocs template library. You will do this in the next lesson.

Lesson 13: Add Templates to a Library

Overview

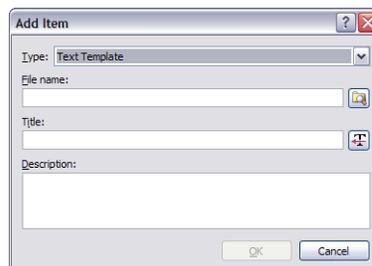
The final step in creating a form template is to add the template to a HotDocs library. A library allows you to group related templates and provide descriptions and longer titles for the templates to make them easier to find and edit. Also, if you are distributing the template to other users, placing it in a library will make it easier for them to access.

You can easily create a new library for the template, but for this lesson you will add the template to the existing library.

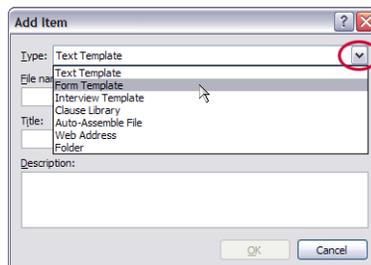
To add the template to the existing library

- 1 Choose **HotDocs 2008 > HotDocs Professional** from **Programs** on the **Start** menu. (Or, if the HotDocs icon appears in the Windows taskbar, click it.) The HotDocs library opens.
- 2 At the library, click on the **HotDocs Tutorial Files** folder.

 If the **HotDocs Tutorial Files** folder does not appear, click the  **Open Library** button, select the tutorial library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**. Then select the **HotDocs Tutorial Files** folder.
- 3 Click the  **Add Item** button in the template library toolbar. The **Add Item** dialog box appears.



- 4 Click the **Type** drop-down button and select **Form Template**.

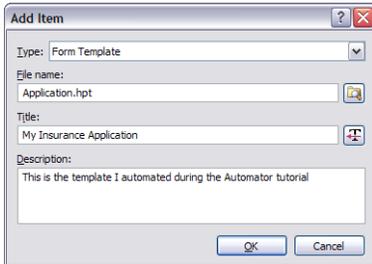


- 5 Click the  **Browse** button next to the **File name** box. The **Add Item File Name** dialog box appears.
- 6 Select **Application.hpt** (or **Application.hft**) and click **OK**.
- 7 Click in the **Title** box.

 HotDocs libraries contain references (or shortcuts) to the actual files on the local disk or network.

When you click in the **Title** box, HotDocs automatically suggests *Application* as the template title. However, you may want something more descriptive.

- 8** Delete *Application* from the **Title** box and type **My Life Insurance Application**.
- 9** In the **Description** box, type **This is the template I automated during the Automator tutorial**.



- 10** Click **OK**. The template is added to the library.

Now that the template is in the library, you can select it and click  **Assemble** to assemble a form document, or click  **Edit** to open the template for editing.

You have completed the HotDocs Automator Tutorial. For additional information on using HotDocs Automator, please see the HotDocs Help.

Chapter 4: HotDocs Database Connectivity Tutorial

Before You Start

You should be familiar with HotDocs before completing this tutorial. If you are not, you should complete the HotDocs tutorial earlier in this manual before continuing.

This tutorial introduces you to the basic features of connecting your template to a database. Specifically, you will prepare a Collection Letter (Database) that will retrieve data from two database tables using an ADO connection string. You will then link variables in the template to fields in each database table. Finally, you will set up filtering options that will limit the number of records with which you are working.

The tutorial is divided into six lessons, which you must complete in order. However, you can complete each lesson in separate sessions.

 The template and database files used in the tutorial were drafted for instructional purposes only.

Overview: Understand Database Connectivity

As you know, each time you use a text or form template to assemble a document, HotDocs prompts you for the information the template requires. Once you have answered the questions, HotDocs merges your answers and carries out your instructions to produce the finished, assembled document.

There are three ways to provide HotDocs with the information it needs during the interview. You can use any (or a combination) of these methods:

- Enter the information manually.
- Retrieve the information from an answer file.
- Retrieve the information from a database using a database component.

The last option allows you to retrieve answers during an interview from a database during document assembly. That way, information from a company data store or other database-driven application, such as a case manager, can also be used to assemble a HotDocs document.

For a more technical explanation of database connection, please see the topic, “Understand Database Connectivity” in the HotDocs Helps.

Lesson 1: Create an ADO Connection String

Overview

The *Collection Letter (Database)* template you are using in this tutorial has already been automated. It uses variables that ask information about the client and any invoices the client may have. You could assemble the document as it is now and provide this information by manually entering it. However, using a database component, you can retrieve the answers you already know from a database.

Before you can do this, you must define the association between your template and the database by first creating a database component and then by linking the database component to the database file. You will do this using an ADO connection string.

In this lesson, you will link a Microsoft Access database (*Tutorial.mdb*) to the *Collection Letter (Database)* template. This database file contains two tables, *CLIENT* and *INVOICE*. You will retrieve data from both of these tables in your template.

 If you don't have Microsoft Access installed, HotDocs can still retrieve data from the database file using the Microsoft Jet 4.0 OLE DB Provider, which was automatically installed with HotDocs.

To open the tutorial library

- Choose **HotDocs 2008 > HotDocs Professional** from **Programs** on the **Start** menu. The HotDocs tutorial library appears.

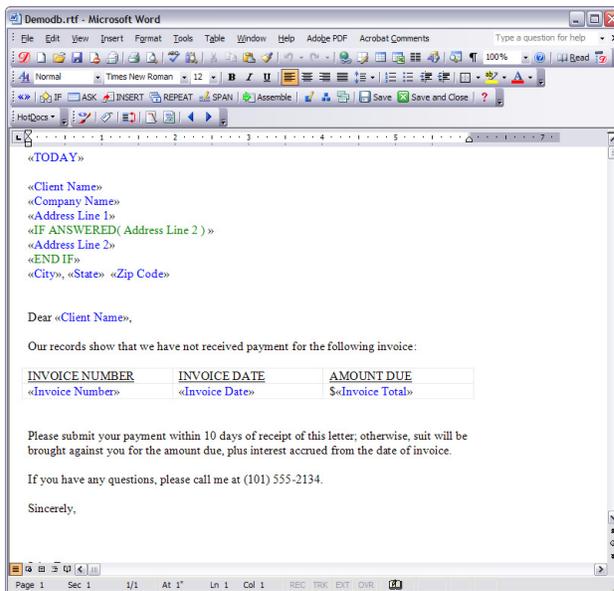
 If the **HotDocs Tutorial Files** folder does not appear, choose **Open Library (File menu)**, select the library for your word processor in the default **Libraries** folder (for example, Word Tutorial Pro.hdl), and click **Open**.

Create a Database Component

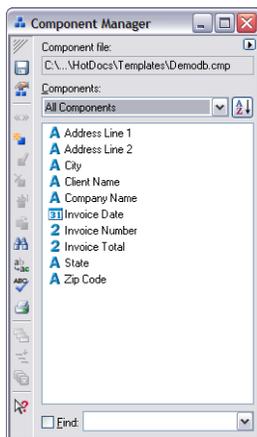
In order to create the connection, you must first create the database component that will store the connection information.

To create a database component

- 1 At the template library, select **Collection Letter (Database)** and click  **Edit**. The template appears, ready for you to edit.



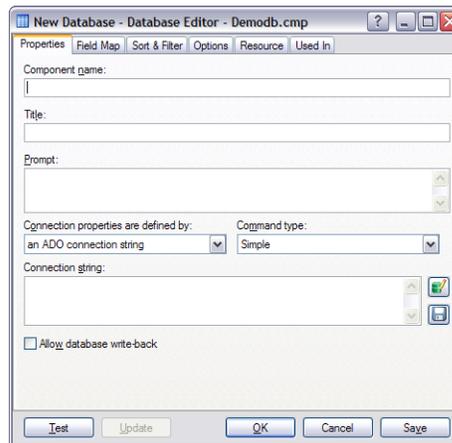
- 2 Click the  **Component Manager** button in the HotDocs toolbar of the word processor. The **Component Manager** window appears.



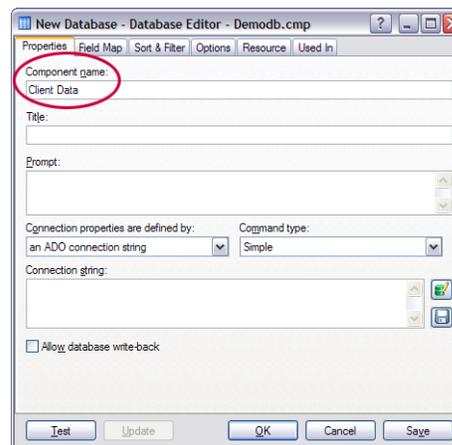
- 3 Click the **Components** drop-down button and select **Databases** from the list. The list changes to show only database components. (Because you have not yet created any database components, this list is empty.)

 You can have HotDocs arrange the template and Component Manager windows so you can view both simultaneously. To do this, adjust Component Manager to the width you want and click the  **Arrange Windows** button in the Component Manager toolbar.

- 4 Click the  **New Component** button. The **Database Editor** appears.



- 5 Type **Client Data** in the **Component name** box.



Now that you have created the database component and assigned a name to it, you must define a connection string, which contains the information HotDocs needs to connect to the database.

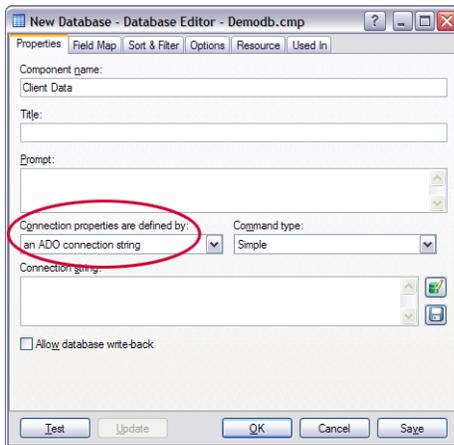
Define the ADO Connection String

An ADO connection string defines the relationship HotDocs has to the database. It contains information about the OLE DB provider, the location of the actual database file, the name of the table from which you will be retrieving data, and any other information about the connection HotDocs needs to know to maintain its relationship with the database.

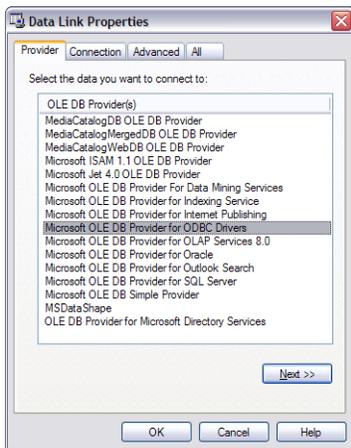
You can use a simple wizard to help you define the connection string.

To build the ADO connection string

- 1 With the Database Editor displayed, make sure an ADO connection string is selected at the Connection properties are defined by drop-down list.



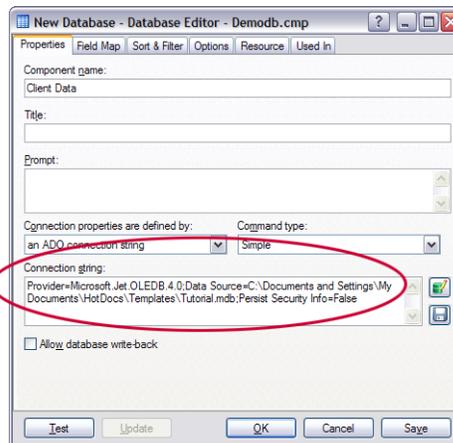
- 2 Click the  Edit button next to the Connection string box. The Data Link Properties dialog box appears.



- 3 Select **Microsoft Jet 4.0 OLE DB Provider** from the list of providers and click **Next**. HotDocs displays the **Connection** tab of the **Data Link Properties** dialog box.

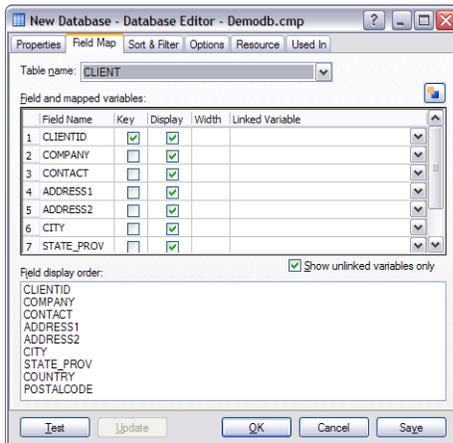


- 4 Click the ... **Browse** button next to the **Database name** box and locate the database file, **Tutorial.mdb**. (This file should be located in the default *Templates* folder, for example, *C:\Documents and Settings\UserName\My Documents\HotDocs\Templates*.)
- 5 Click **Open**. The **Data Link Properties** dialog box appears again.
- 6 Click **OK**. The **Database Editor** appears again, showing the ADO connection string that will link your template to the database.



- 7 Click the **Field Map** tab.

- 8 Click the **Table name** drop-down button and select **CLIENT**. The fields from the *CLIENT* table appear in the **Field Name** column.



- 9 Click **OK**.

HotDocs creates an association between the template file and the database, particularly the *CLIENT* database table. In Lesson 2 you will link fields in this table to variables in the template.

On Your Own

Create a second database component using an ADO connection string. Follow the instructions in both “Create a Database Component” on page 176 and “Define the ADO Connection String” on page 178. Name this component **Invoice Data**, and use the same database file, **Tutorial.mdb**. When selecting the database table at the **Field Map** tab, choose **INVOICE**.

Once you have created the second database component and defined its connection string, you are now ready to link variables in your template to fields in these two database tables. You will do this in Lesson 2.

If you do not want to go on to Lesson 2 at this time, close **Component Manager** (by clicking the X in the upper-right corner of the Component Manager window), and then click the  **Save and Close** button to close the template.

If you are continuing on to Lesson 2, click the  **Save** button to save your work.

Lesson 2: Link Variables to Fields in a Database Table

Overview

As you already know, to create an interview, HotDocs reads through the template and displays the variables (or dialogs if the variables are linked to the dialogs) in the order it reads them. When variables are linked to a database table (rather than a dialog), HotDocs displays the database table in the interview so you can choose a record. The selected data is then retrieved from the database file and merged into the assembled document.

In this lesson, you will link variables in the template to fields in both the *CLIENT* and *INVOICE* tables. You will also designate *key fields* in each table. Key fields help HotDocs remember which records have been selected during an interview.

If you are continuing immediately from Lesson 1, skip the instructions for opening the template and proceed to “Link Variables to Fields.”

If you closed the template at the end of Lesson 1, complete the following steps.

To open the Collection Letter (Database) template

- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs template library appears.
- 2 Select **Collection Letter (Database)** and click  **Edit**. The template appears, ready for you to edit.
- 3 Click the  **Component Manager** button. The **Component Manager** window appears.

Link Variables to Fields

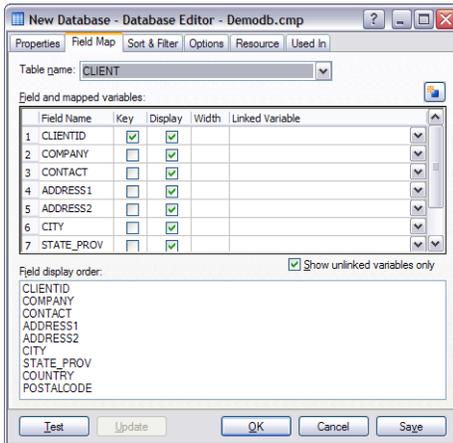
Once the connection to your database has been defined, you can link fields in the database table to variables in your template. These links allow HotDocs to retrieve information from the database table each time the template is used to assemble a document.

To link variables in the template to fields in the database

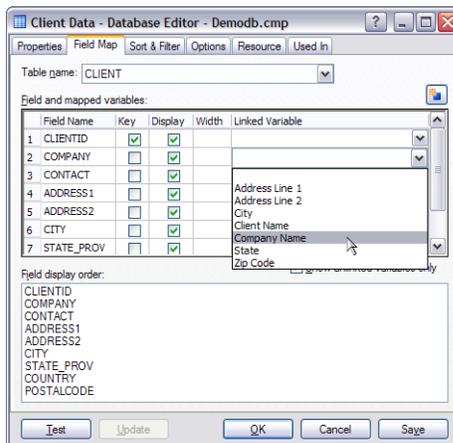
- 1 With **Component Manager** displayed, select **Client Data** from the list of components and click the  **Edit Components** button. The **Database Editor** appears.

 When linking variables to fields, data types must match. For example, if the field type is text, you must link to either a Text variable or a Multiple Choice variable (since Multiple Choice variables are text values).

- 2 Click the **Field Map** tab. The **CLIENT** table appears, with a list of all the fields in the table.



- 3 In the **Linked Variable** column, click the drop-down button that corresponds to the **COMPANY** field and select **Company Name**.



The HotDocs variable *Company Name* is now linked to the *COMPANY* field in the *CLIENT* database table.

- 4 Repeat step 3 to link the following variables to fields in the database table:

Field	Variable
CONTACT	Client Name
ADDRESS1	Address Line 1
ADDRESS2	Address Line 2
CITY	City
STATE_PROV	State
POSTALCODE	Zip Code

The fields in the database table are now linked to variables in the template. Now that you have done this, you must make sure one of the fields is designated as the key field.

✓ By default, once you link a variable to a field, HotDocs removes the variable from the list. You can view the entire list of variables—linked and unlinked—by clearing **Show unlinked variables only**. When you do this, variables that are already linked are surrounded by brackets and are sorted to the bottom of the variable list.

Designate a Specific Field as the Key Field

During the interview, when HotDocs connects to a database, it displays a table of records. When you select a record and move to another dialog, HotDocs can remember from where in the database table the record was selected so that if you later review the table, your record will still appear selected.

For HotDocs to operate this way, you must make one of the fields in the database component the *key field*. A key field contains information that is unique to a record—for example, an invoice number or a customer identification number. Without a key field, HotDocs has no way to remember from where in the table the record was retrieved and therefore, cannot retain this information once you navigate to a new dialog. This may be confusing to some users since even though answers may appear in the assembled document, the interview shows that no record has been selected.

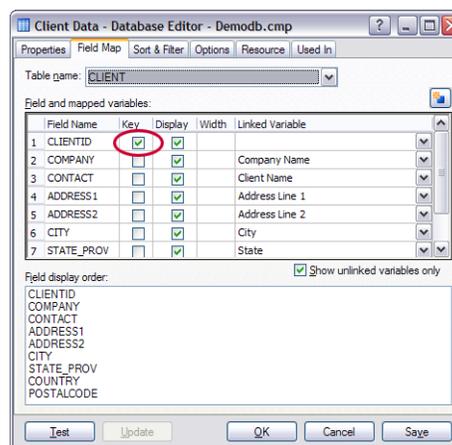
Selecting a key field also makes it easier to reuse an answer file that contains selected database records, for the reasons explained above.

If you are connecting to a database table using a native OLE DB provider, HotDocs recognizes the key field that was designated when the database table was first created and automatically assigns it as the key field for the database component. Even so, it is always a good idea to make sure the correct key field is specified. Because a client's identification number will always be unique, you must make sure the CLIENTID field is designated as the key field.

✓ HotDocs does not recognize existing key fields when you connect to the database using the OLE DB provider for ODBC, or when your database component is connected to a database view. You must manually identify the key using the **Database Editor**.

To ensure CLIENTID is designated as the key field

- In the **Key** column, make sure the box next to the **CLIENTID** field is selected.



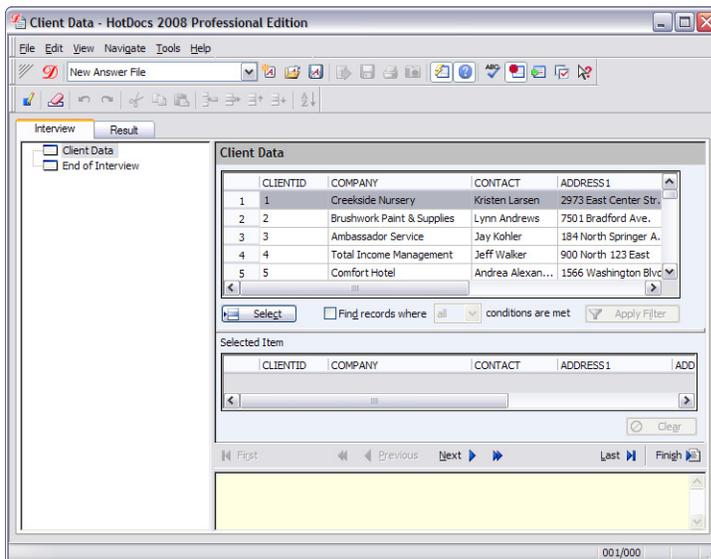
Now that you have linked fields in the CLIENT table to corresponding variables in the template, you can test how the database component will appear during assembly.

Test the Database

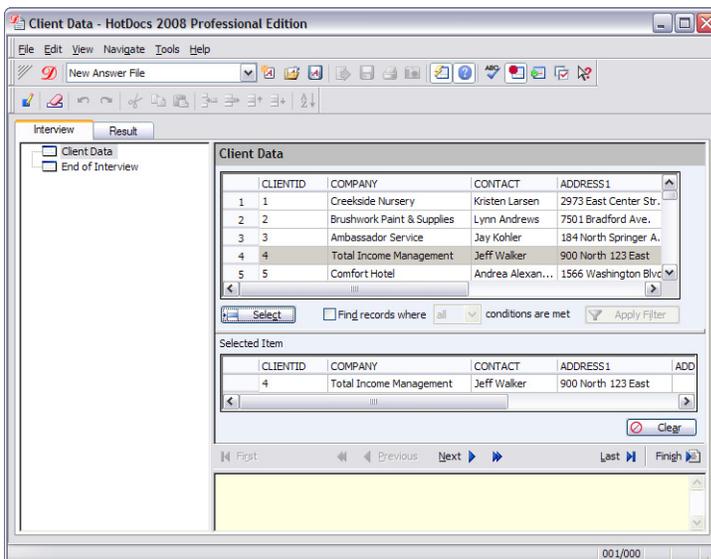
Just as you can test variables and dialogs as you create them, you can test database components.

To test the database component

- 1 At the Database Editor, click **Test**. A test assembly window appears, displaying the **Client Data** table.

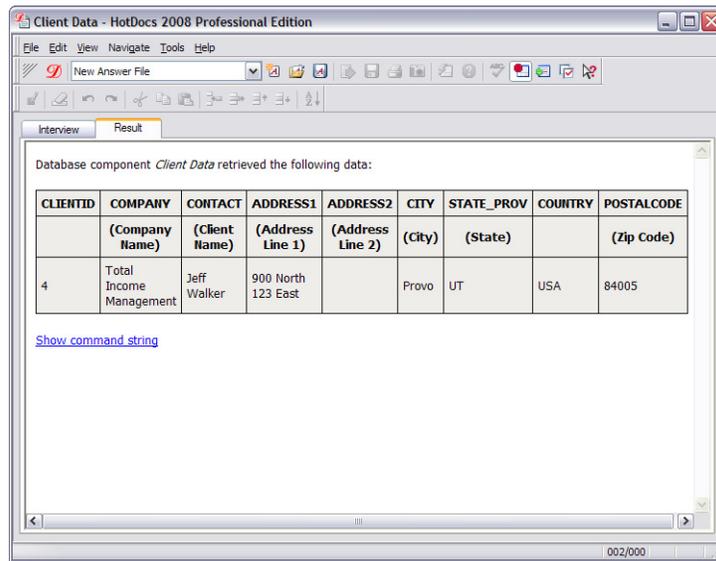


- 2 Click a record in the table and click **Select**, located below the database spreadsheet. HotDocs adds the record to the **Selected Item** list, at the bottom of the dialog pane.



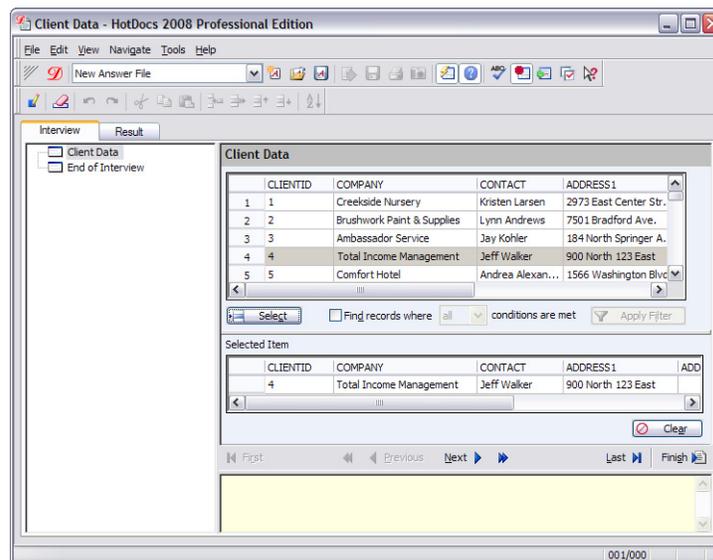
✓ If you click the **Show command string** link, HotDocs displays the database command string it used to retrieve this data.

- Click the **Result** tab. The window changes to show the information the database component retrieved.



The first row in the table shows the names of the fields in the actual database table, while the second row shows the names of the HotDocs variables to which these fields are linked. The third row shows the particular answers that were retrieved from the database table.

- Click the **Interview** tab again. Notice that the record is still listed in the **Selected Items** list. This is because the database component has a key field. If no key field had been assigned, no records would appear selected.



- Close the test assembly window by choosing **Close** from the **File** menu. HotDocs asks you to save your answers.

✓ For more information on using test answer files, see the HotDocs Help.

- Decide whether to save your answers, based on the following information:
 - If you click **Save** or **Save As**, you will be prompted to specify an answer file name and title. Type **Test Answer File** in the **File name** box and accept the

suggested **Title**. From this point on, this answer file will be used each time you test a component or test assemble the template, and you will no longer be prompted to save your answers after each test. (You can, of course, choose a different answer file when you are at the assembly window.)

- If you click **Don't Save**, each time you test a component or test assemble the template, you will use an empty, untitled answer file. You will also be prompted to save your answers each time you finish a test.

7 Click **OK** at the **Database Editor**.

On Your Own

Open the *Invoice Data* database component, click the **Field Map** tab, and link the following variables to their corresponding fields in the *INVOICE* table. Follow the instructions given in “Link Variables to Fields” on page 182.

Field	Table Field
INV_NUMBER	Invoice Number
INV_DATE	Invoice Date
INV_TOTAL	Invoice Total

After linking the variables, make sure *INV_NUMBER* is designated as the key field. (Follow the instructions given in “Designate a Specific Field as the Key Field” on page 184.) Then, when you are finished, click **OK** to close the **Database Editor**.

If you do not want to go on to Lesson 3 at this time, close **Component Manager** (by clicking the **X** in the upper-right corner of the Component Manager window), and then click the  **Save and Close** button to close the template.

If you are continuing on to Lesson 3, click the  **Save** button to save your work.

Lesson 3: Select Fields and Records for Display

Overview

Sometimes a database table contains fields that have no corresponding variables in the template. Even if the fields contain information, you may not want them to appear during assembly—especially if there are already a large number of fields in a record. Also, perhaps a user assembling a document needs to see only the information in one or two fields, such as the company name and company contact, in order to select a record. In such situations, it makes sense to display only those fields, rather than all the fields.

In this lesson, you will hide fields that have little or no meaning to the user. You do this at the Database Editor.

If you are continuing immediately from Lesson 2, skip the instructions for opening the template and proceed to “Choose Which Fields Will be Presented During Assembly.”

If you closed the template at the end of Lesson 2, complete the following steps.

To open the Collection Letter (Database) template

- 1 Open your word processor and click the  HotDocs 2008 button. The HotDocs template library appears.
- 2 Select **Collection Letter (Database)** and click  **Edit**. The template appears, ready for you to edit.
- 3 Click the  **Component Manager** button. The **Component Manager** window appears.

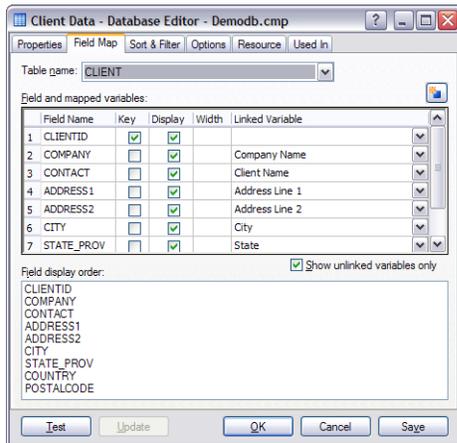
Choose Which Fields Will be Presented During Assembly

You can customize the database component to show only the required fields during the interview.

To select which fields will be shown in the table during assembly

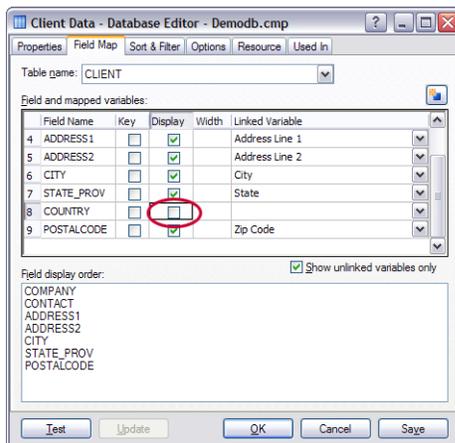
- 1 With **Component Manager** displayed, select **Client Data** from the list of components and click the  **Edit Component** button. The **Database Editor** appears.

- Click the **Field Map** tab. The window changes to show all the fields in the *CLIENT* table.



Notice that in the **Display** column, each field has a check mark. This means that HotDocs will display each of these fields during the interview. However, some fields are unnecessary for the user to see.

- In the **Display** column, clear the check marks for the following database fields: **CLIENTID** and **COUNTRY**



- Click **Test**. HotDocs displays the test assembly window and shows the seven remaining fields in the table.
- Close the test assembly window by choosing **Close** from the **File** menu. HotDocs displays the **Database Editor** again.
- Click **OK** to close the **Database Editor**.

On Your Own

Edit the *Invoice Data* database component so that the following fields are *not* displayed during assembly:

CLIENTID
 SHIP_DATE
 DATE_PAID

✓ Notice that as you “remove” fields, the field name is also removed from the **Field display order** box. The order of fields in this box indicates the order of columns in the spreadsheet when it’s displayed during the interview.

Follow the instructions given in “Choose Which Fields Will be Presented During Assembly” on page 188. When you are finished, click **OK** to close the **Database Editor**.

If you do not want to go on to Lesson 4 at this time, close **Component Manager** and then click the  **Save and Close** button to close the template.

If you are continuing on to Lesson 4, click the  **Save** button to save your work.

Lesson 4: Filter the List of Records

Overview

By default, when HotDocs presents a table of records during the interview, every record in the table is displayed. Depending on the size of the table, this may mean the table with which you are working could make retrieving data from it sluggish.

You can limit the number of records retrieved from a database by creating a filter. A filter compares the value in a database field with another value or with a variable in the template. Only the records that meet the comparison are displayed.

In this template, the *INVOICE* table contains many records, but you want to generate letters for only those invoices that have not been paid. You can create a filter that displays only the records whose *DATE_PAID* field is empty.

If you are continuing immediately from Lesson 3, skip the instructions for opening the template and proceed to “Create the Database Filter.”

If you closed the template at the end of Lesson 3, complete the following steps.

To open the Collection Letter (Database) template

- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs template library appears.
- 2 Select **Collection Letter (Database)** and click  **Edit**. The template appears, ready for you to edit.
- 3 Click the  **Component Manager** button. The **Component Manager** window appears.

Create the Database Filter

You must create a filter that displays only the records whose *DATE_PAID* field is empty. Then, during the interview, users can choose from only unpaid accounts.

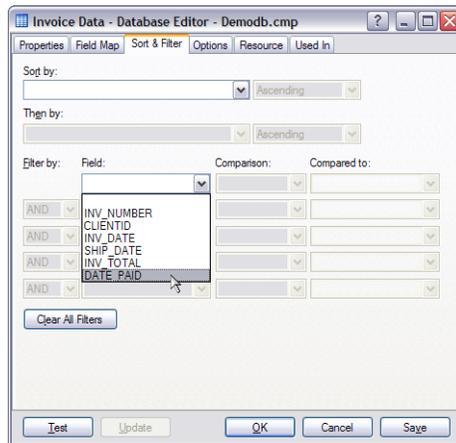
To filter the records

- 1 With **Component Manager** displayed, edit the **Invoice Data** component. The **Database Editor** appears.

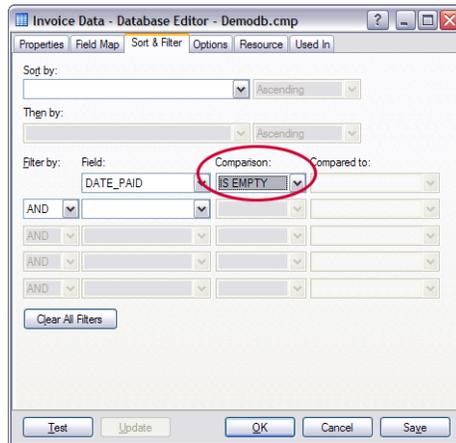
- 2 Click the **Sort & Filter** tab. The window changes to show sorting and filtering options.



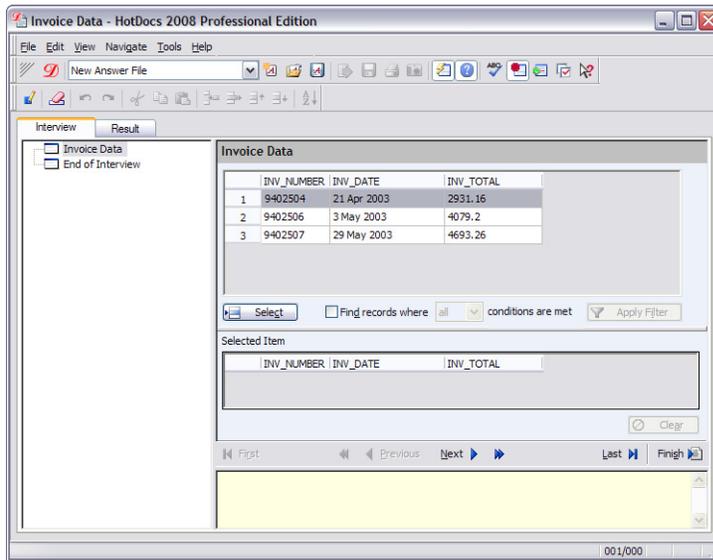
- 3 Click the **Filter by Field** drop-down button and select **DATE_PAID**. The **Comparison** field becomes active.



- 4 Click the **Comparison** drop-down button and select **IS EMPTY**.



5 Click **Test**. HotDocs displays the test assembly window.



Notice that only three records are listed. This is because the *DATE_PAID* field for these three invoices is empty, meaning these accounts are delinquent on their payments.

6 Close the test assembly window and click **OK** at the **Database Editor**.

If you do not want to go on to Lesson 5 at this time, close **Component Manager**, and then click the  **Save and Close** button to close the template.

If you are continuing on to Lesson 5, click the  **Save** button to save your work.

Lesson 5: Select Related Database Records

Overview

Sometimes your template may link to more than one database table. If so, and if both tables have at least one field in common, you can create a filter based on this association so that, during assembly, you can select a record from one table and have the corresponding records from the other table automatically selected.

For example, both the *INVOICE* and *CLIENT* tables used in the *Collection Letter (Database)* template have a field named *CLIENTID*. Using this commonality, you can set up a filter so that when the user selects an unpaid invoice from the *INVOICE* table, HotDocs automatically retrieves, from the *CLIENT* table, the client whose name is on the unpaid account.

If you are continuing immediately from Lesson 4, skip the instructions for opening the template and proceed to “Link the Common Field to a Variable.”

If you closed the template at the end of Lesson 4, complete the following steps.

To open the Collection Letter (Database) template

- 1 Open your word processor and click the  HotDocs 2008 button. The HotDocs template library appears.
- 2 Select **Collection Letter (Database)** and click  **Edit**. The template appears, ready for you to edit.
- 3 Click the  **Component Manager** button. The **Component Manager** window appears.

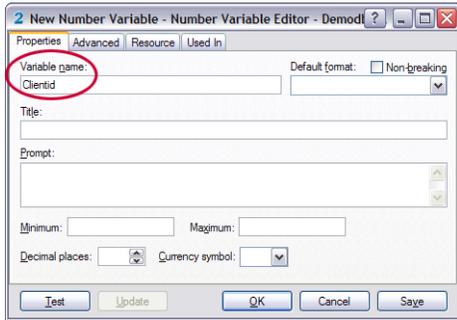
Link the Common Field to a Variable

In order to associate the *CLIENT* and *INVOICE* tables, you must first link the common field, *CLIENTID*, to a variable in the component file. However, since this variable does not yet exist, you must create it. Once created, you can filter records based on the answer to this variable.

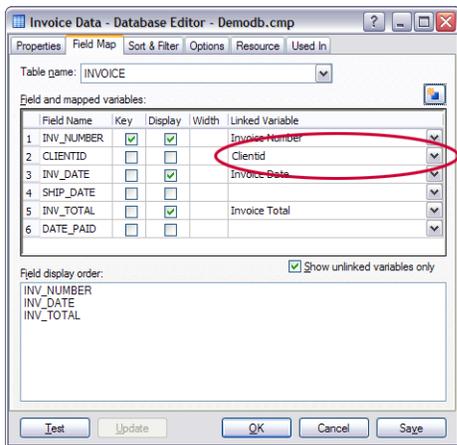
To create a variable and link your common field to it

- 1 From Component Manager, edit the **Invoice Data** database component. The **Database Editor** appears.
- 2 Click the **Field Map** tab. The window changes to show which variables in the template are linked to which fields in the table.
- 3 In the **Linked Variable** column, click the empty cell next to the **CLIENTID** field and click the  **New Component** button. The **Number Variable Editor** appears.

Because *CLIENTID* is a number field, HotDocs automatically creates a Number variable and assigns a name to it, based on the name of the table field.



- 4 Click OK. HotDocs creates the variable (*Clientid*) and links it to the field in the table.



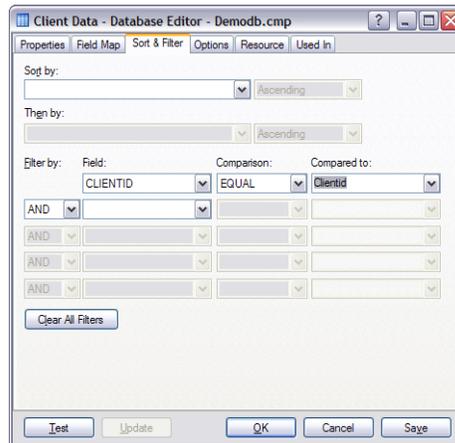
- 5 Click OK to close the Database Editor.

Use the Linked Variable as a Filter

The next step in this process is to use the HotDocs variable you just linked to the *INVOICE* table to filter the same company record from the *CLIENT* table. The filter uses the answer for *CLIENTID* (which is answered when the user selects an invoice) and returns the record from the *CLIENT* table that matches that value.

- 1 Using Component Manager, edit the Client Data database component. The Database Editor appears.
- 2 Click the Sort & Filter tab. HotDocs displays options for sorting and filtering the records in the database.
- 3 Click the Filter by Field drop-down button and select CLIENTID.
- 4 Click the Comparison drop-down button and select EQUAL.

5 Click the **Compared to** drop-down button and select **Clientid**.



6 Click **OK** to save your work and close the **Database Editor**.

7 Close **Component Manager**.

In the next lesson, you will test the template and see how, when a record is selected from the *INVOICE* table during assembly, HotDocs automatically selects the record in the *CLIENT* table that corresponds with the chosen invoice.

If you do not want to go on to Lesson 6 at this time, click the  **Save and Close** button to close the template.

If you are continuing on to Lesson 6, click the  **Save** button to save your work.

Lesson 6: Test the Template

Overview

As with any template you are automating, it is always a good idea to test assemble it to make sure it assembles the way you expect. In this lesson, you will test assemble the *Collection Letter (Database)* template you have automated.

If you are continuing immediately from Lesson 5, skip the instructions for opening the template and proceed to “Test the Template.”

If you closed the template at the end of Lesson 5, complete the following steps.

To open the Collection Letter (Database) template

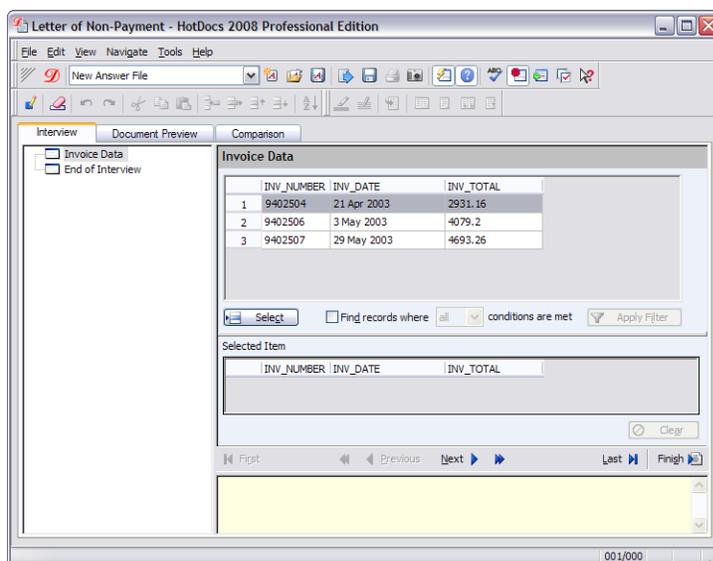
- 1 Open your word processor and click the  **HotDocs 2008** button. The HotDocs template library appears.
- 2 Select **Collection Letter (Database)** and click  **Edit**. The template appears, ready for you to edit.

Test the Template

Based on the automation you have done in the template, when you test the *Collection Letter (Database)* template, HotDocs should present a list of unpaid invoices. When you select a specific invoice, it should automatically select and merge the information about the client whose name is on the invoice.

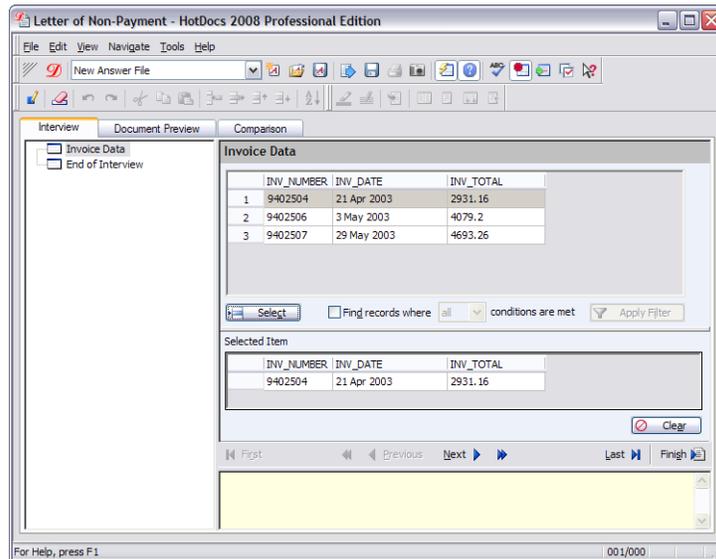
To test the template

- 1 At the template, click the  **Test Assemble** button. A test assembly window appears, showing the list of unpaid invoices.

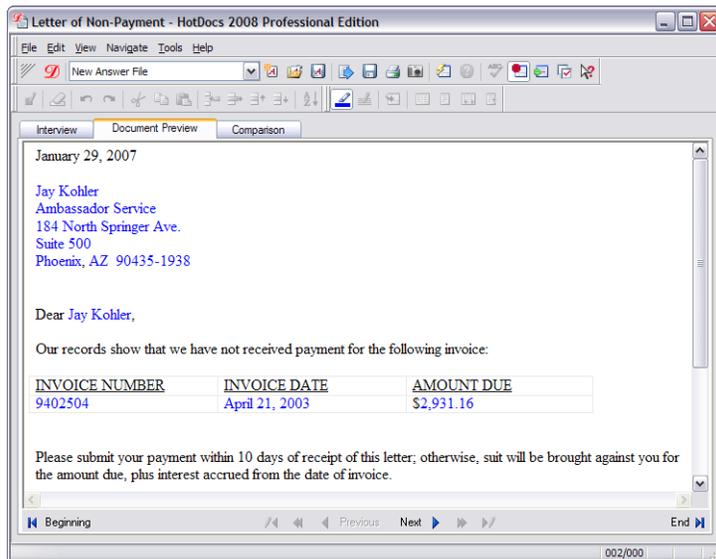


✓ You can also double-click a record and HotDocs will add it to the **Selected Item** list. To clear a selection, click **Clear**.

- 2 Click the first record and then click  **Select**. HotDocs adds the record to the **Selected Item** list.



- 3 Click the **Document Preview** tab. HotDocs displays a copy of the assembled document with all the correct information merged into the document, including the client information which was automatically retrieved from the database.



- 4 Click the **Interview** tab again to return to the database selection dialog.
- 5 Select either the second or third record (by choosing the record and clicking the  **Select** button) and click the **Document Preview** tab. The information in the assembled document changes, based on the new selection you have made.
- 6 Close the test assembly window, and then click the  **Save and Close** button to close the template.

This completes the Database Connection tutorial. Please refer to the HotDocs Help for additional information about using the Database Connection.