Up and Running with Microsoft® Visual InterDev™

Table of Contents

Companies, names and/or data used in screens and sample output are fictitious, unless otherwise noted.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Lesson 1 - Getting Started

Welcome to *Up and Running with Microsoft® Visual InterDev™*, a step-by-step tutorial that shows you how to use the main features of Visual InterDev. As you work through the tutorial, you will construct a simple intranet application for Friendship Insurance, a fictitious insurance company.

The Friendship Insurance intranet application includes the following:

- A dynamically generated HTML report with a listing of current Friendship customer records (stored in a Microsoft SQL Server ™ or Microsoft Access customer database).
- An HTML data form for insurance agents to directly edit customer records in the database.
- A schedule tracking page that displays the appointment schedules for insurance agents on a given day.

Before running the tutorial, you should install Visual InterDev and set up a Web server for the Friendship Insurance project.

Required Hardware and Software

Microsoft Visual InterDev is based on a distributed development environment. Visual InterDev projects are located on Web servers, while the development takes place on developers' workstations. You can complete the tutorial on a standalone computer running the Personal Web Server for Microsoft Windows 95™ or the Peer Web Services on Microsoft Windows NT™ Workstation 4.0. As an alternative, you can set up your Web server on a separate dedicated computer running Windows NT Server 4.0 and the Microsoft Internet Information Server (IIS).

Web Server Configuration

If you choose to develop against a dedicated Internet Information Server, the following server configuration is recommended:

- Intel® Pentium™-class server with at least 32MB RAM
- Microsoft Windows NT Server 4.0
- Microsoft Internet Information Server 2.0
- Microsoft Visual InterDev server components (see below for installation instructions)
- Microsoft SQL Server 6.5 with Service Pack 1 for SQL Server 6.5 installed (optional)

Note If you complete the tutorial using Microsoft SQL Server, you can also run the SQL Server database on a dedicated database server computer separate from the Web server.

Development Workstation Configuration

The following development workstation configuration is recommended:

- Intel Pentium-class workstation with at least 32MB RAM (24MB for Win95)
- Microsoft Windows NT Workstation 4.0 or Microsoft Windows 95
- Microsoft Visual InterDev client components (see below for installation instructions)

Installing the Visual InterDev Server and Client

To install the Visual InterDev server

1 Run the Visual InterDev master setup program, **Setup.exe**.

2 Open the installation instructions for the server components from the Master Setup.

3 If you want to develop against the **Personal Web Server for Windows 95**, install this optional component (Windows 95 only). You can also develop against the Peer Web Services in Windows NT Workstation 4.0.

4 Install **Active Server Pages**.

5 Install the **Microsoft FrontPage™ 97 Server Extensions**.

6 Install the **Visual InterDev server**.

To install the Visual InterDev client

7 Run the Visual InterDev master setup program, **Setup.exe.**

8 Open the installation instructions for the client components from the Master Setup.

9 Install the Visual InterDev client.

1**Note** You have the option to install Microsoft® Image Composer, Media Manager, and Music Producer.

10 Choose **Exit** to close the **Master Setup** screen.

- **Note** If not already installed, the Microsoft Internet Explorer 3.01 Web browser will be installed as part of Setup. While you can view pages developed in Visual InterDev using any standard Web browser, Internet Explorer 3.01 is used as the basis for InfoViewer, the HTML help system and browser integrated into Visual Studio 97 itself. Therefore, you will be viewing help topics in InfoViewer (on the **InfoView** tab in the Project Workspace window).

Registering the ActiveX Calendar Control

Before running the tutorial, you need to register a developer license for an ActiveX™ calendar control you'll be using in Lesson 7, “Working with Client-Side Components.”

To install the Calendar control

- Run C:\GuidedTour\Setup.bat.

Installing the Friendship Insurance Database

You can install either a SQL Server or Microsoft Access version of the Friendship Insurance database to use with the tutorial. However, you must use a Microsoft SQL Server database to complete Lesson 9, “Working with the Database Designer.”

Installing the SQL Server Friendship Insurance Database

You can install SQL Server on any server machine on your network, including the same machine running your Internet Information Server (IIS) Web server.

Note If you install on the same machine as your Web server, install SQL Server *before* the Visual InterDev server components. This ensures that the proper version of the ODBC 3.0 drivers are installed on your server.

To install the SQL Server Friendship Customers database

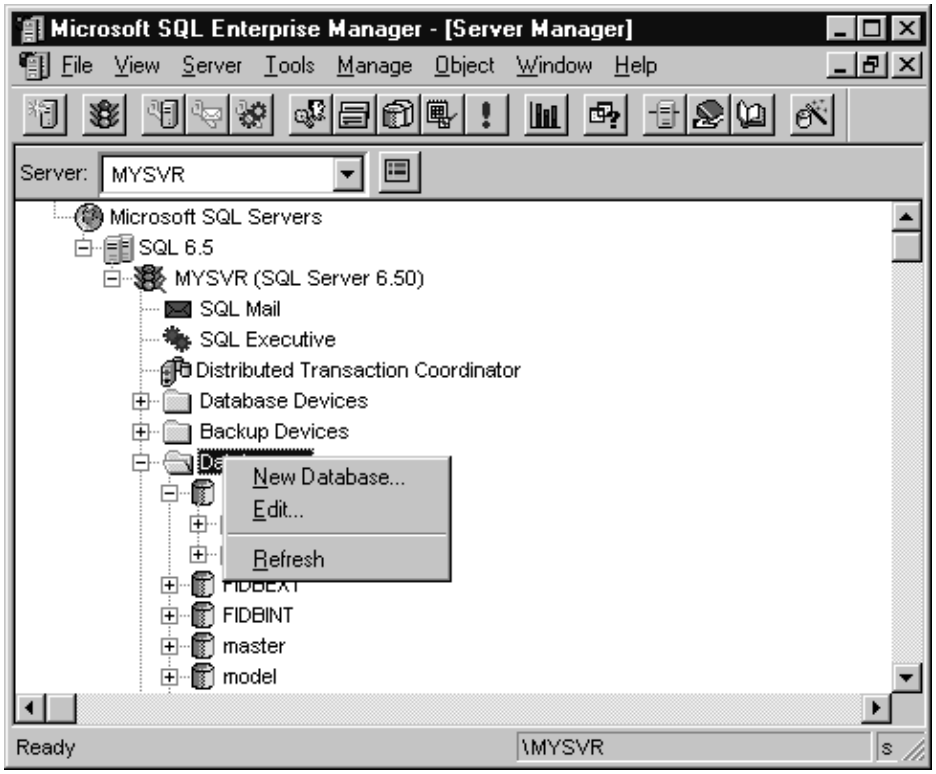
11 On your SQL Server 6.5 database server, click the **Start** menu, and then click **SQL Enterprise Manager** on the **Microsoft SQL Server** programs menu.

12 In the SQL Enterprise Manager, click the plus (+) sign adjacent to the server name.

2**Note** You may need to choose your server first from the server list box. If your server is not started, you can right-click the traffic signal icon, and then choose **Start**.

13 Right-click the **Databases** folder, and then click **New Database**.

1



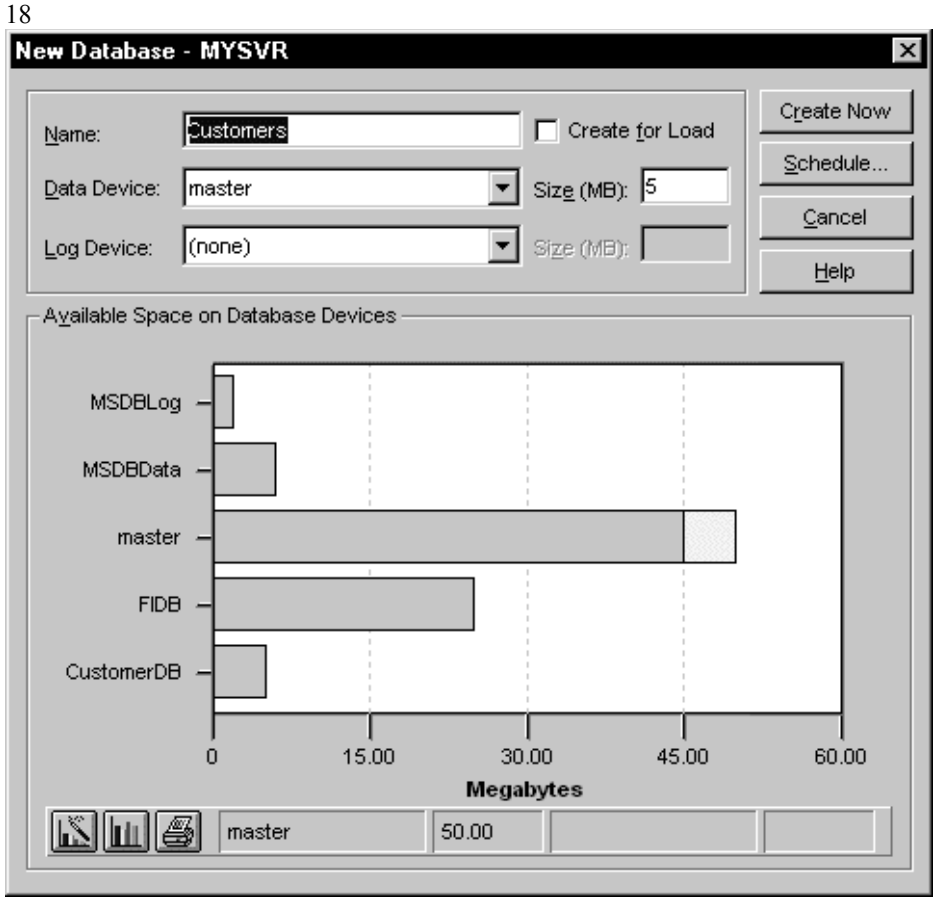
14 In the **Name** box, enter “Customers.”

15 In the **Data Device** box, select the **master** device.

16 In the **Size** box, enter “5.”

17 Click **Create Now**.

18

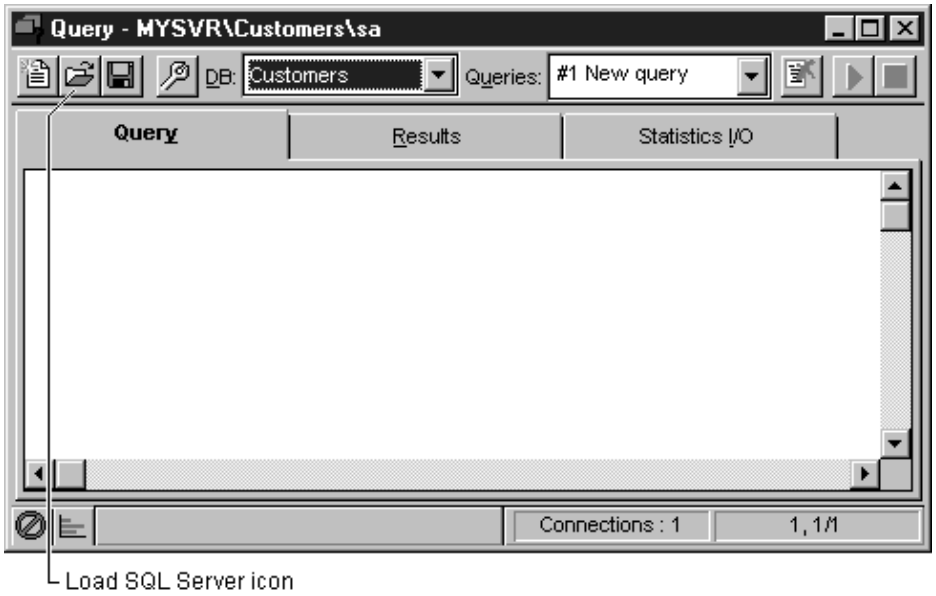


19 Open the **Databases** folder, and select the **Customers** database.

20 Use the Windows NT® Explorer to move or copy C:\GuidedTour\Database\customer.sql to the C:\mssql\binn folder.

21 From the **Tools** menu in the SQL Enterprise Manager, choose **SQL Query Tool**. In the **DB** field, and make sure "Customers" is selected.

2



22 Click the **Load SQL Script** icon on the toolbar. Select **Customer.SQL** from the C:\mssql\binn directory.

23 Click the **Execute Query** icon on the toolbar.

1The script is complete when the color of the **Execute Query** icon returns to green.

24 From the **File** menu, choose **Exit** to close the SQL Enterprise Manager.

Installing the Microsoft Access Friendship Insurance Database

You can also install the Microsoft Access version of the database. Note that if you use the Microsoft Access database for the tutorial, you will not be able to complete Lesson 9, "Working with the Database Designer."

To install the Microsoft Access Friendship Customers database

25 Create a directory on your Web server called C:\Database.

26 Right-click the C:\Database folder in Windows Explorer, and then click **Sharing**. Name the share **Database** so that it is available to Visual InterDev development workstations as *Web server name*\Database.

27 Copy C:\GuidedTour\Database\customers.mdb from your development workstation to the C:\Database folder on your Web server.

Lesson 2 - Creating Your First Web Application

A Visual InterDev Web application consists of a virtual root on a Web server, and all of its folders and files. The simplest way to create a new Web application is to use the Web Project Wizard. The Web Project Wizard communicates with the specified Web server and automatically creates a new Visual InterDev Web application.

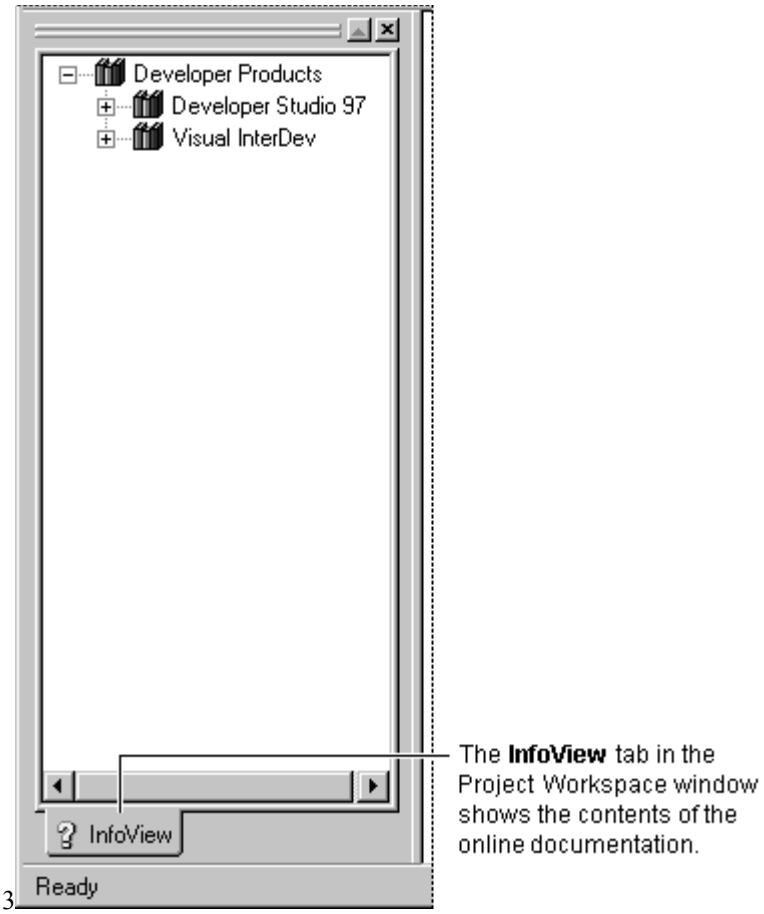
In this lesson, you'll create the Web application for Friendship Insurance.

Using the Web Project Wizard

To create a new Web application with Visual InterDev

28 On your development workstation, start Visual InterDev.

2By default, the **InfoView** tab is displayed in the Project Workspace window.



3

2

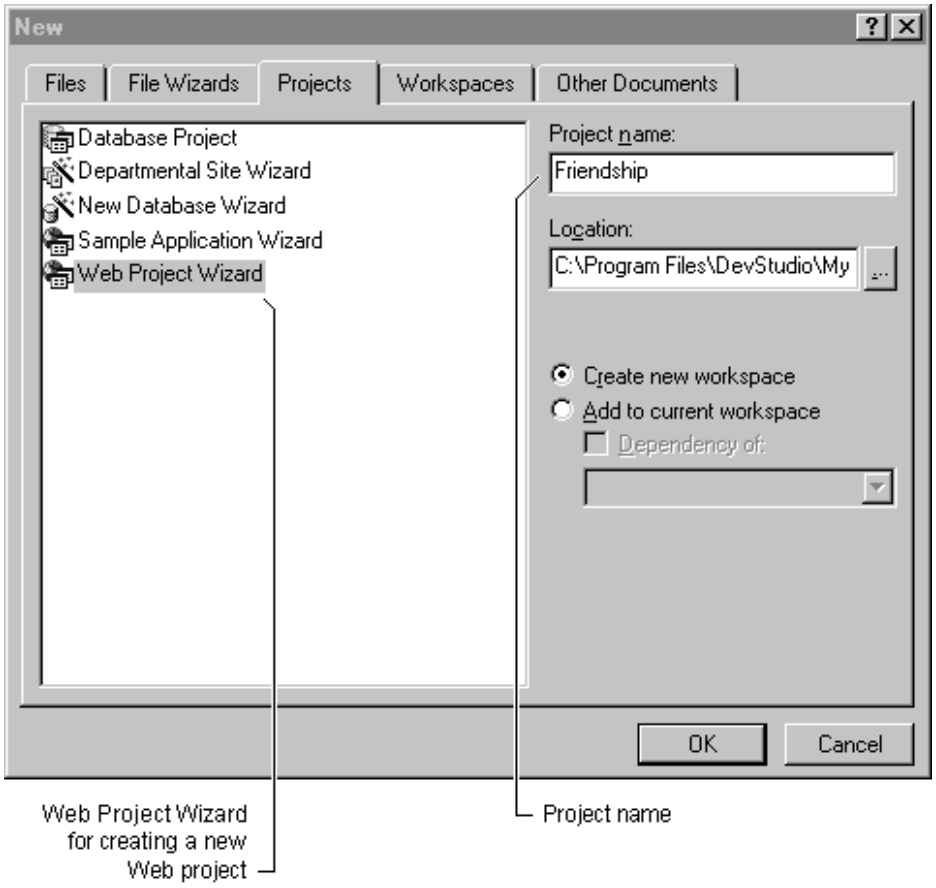
29 On the **File** menu, click **New**.

30 On the **Projects** tab, select **Web Project Wizard**.

31 In the **Project Name** box, type “Friendship,” and then click **OK**.

3

4



4

3This displays Step 1 of the Web Project Wizard.

32 In the **Server Name** box, type the name of your IIS Web server.

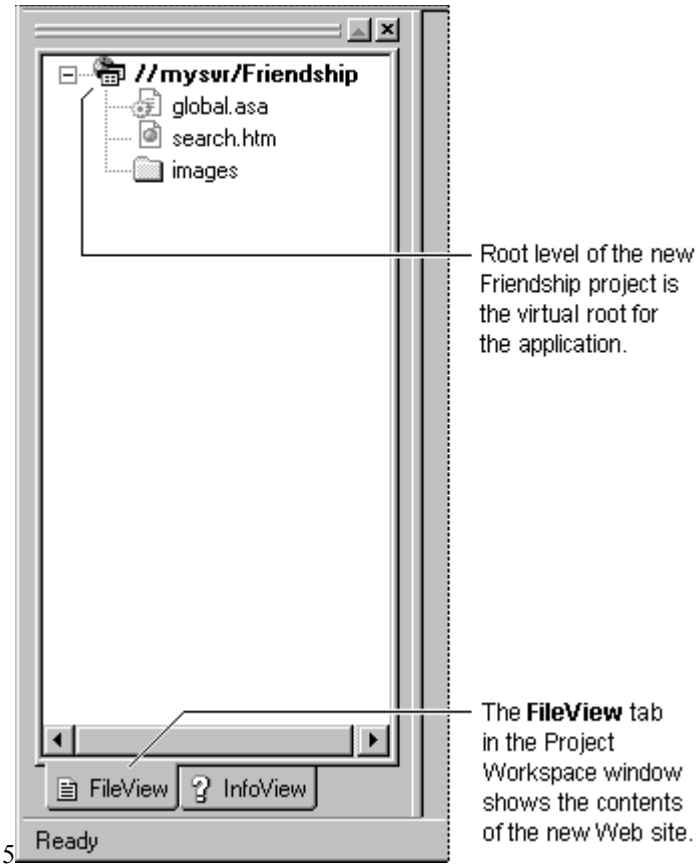
33 Verify that the **Connect using SSL** box is clear, and then click **Next**.

4This displays Step 2 of the Web Project Wizard.

34 By default, the **Create a New Web** option is selected, so just click **Finish**. (You can specify that Visual InterDev automatically index your site for full-text searches.)

5After the Web Project Wizard finishes, you'll have successfully set up your first Visual InterDev Web application at http://[*your server name*]/Friendship.

5



5

6

Tip If the Web Project Wizard can't connect to the specified Web server, that server's FrontPage™ Server extensions may not be set up correctly. Visit the Visual InterDev Web site at http://www.microsoft.com/vinterdev and refer to the Technical FAQ to help you troubleshoot common setup issues.

1

Creating a Simple Active Server Page

The basic element for any Active Server application is the *Active Server Page*, or .asp file. An Active Server Page (ASP) is an HTML document that contains embedded server-side scripting. Microsoft Internet Information Server 3.0 executes the server-side scripting and strips it out of the ASP before sending the page to a browser. From the client-side perspective, the ASP is a standard HTML document, viewable on any platform using any Web browser.

Visual InterDev's controls and wizards automate the process of creating Active Server Pages, but for purposes of this tutorial, let's construct your first simple Active Server Page by hand.

To create an Active Server Page

35 On the **File** menu, click **New**.

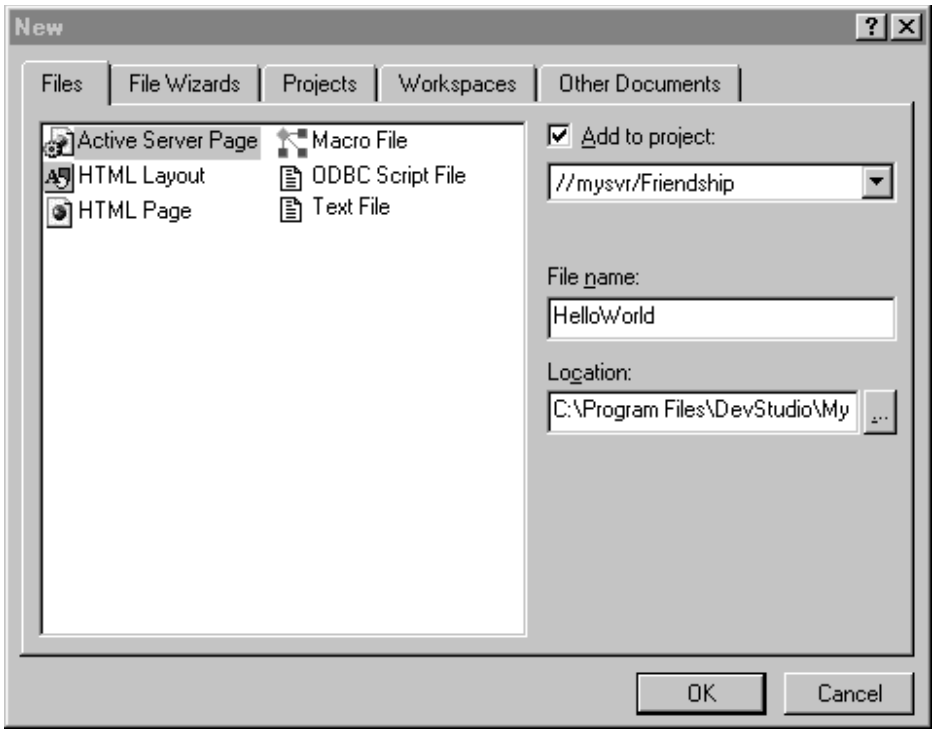
36 On the **Files** tab, select **Active Server Page**.

37 Verify that the **Add to Project** check box is selected.

38 In the **File Name** box, type "HelloWorld," and then click **OK**.

7

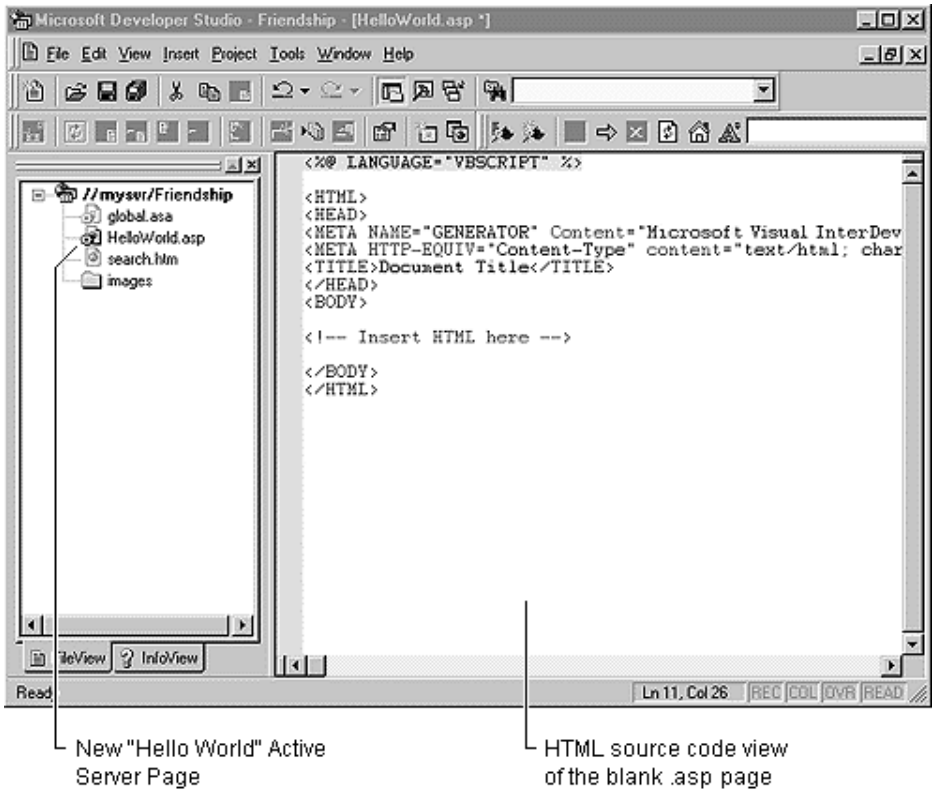
6



8

Visual InterDev creates the new ASP, adds it to your project, and then opens the new ASP in the HTML Source Editor.

9



Note By default, Visual InterDev generates client- and server-side script in Visual Basic® Scripting Edition (VBScript), but you can easily change the script language to Microsoft's JavaScript-compatible scripting language (JScript™) by choosing **Options** on the **Tools** menu.

Working in the Source Editor

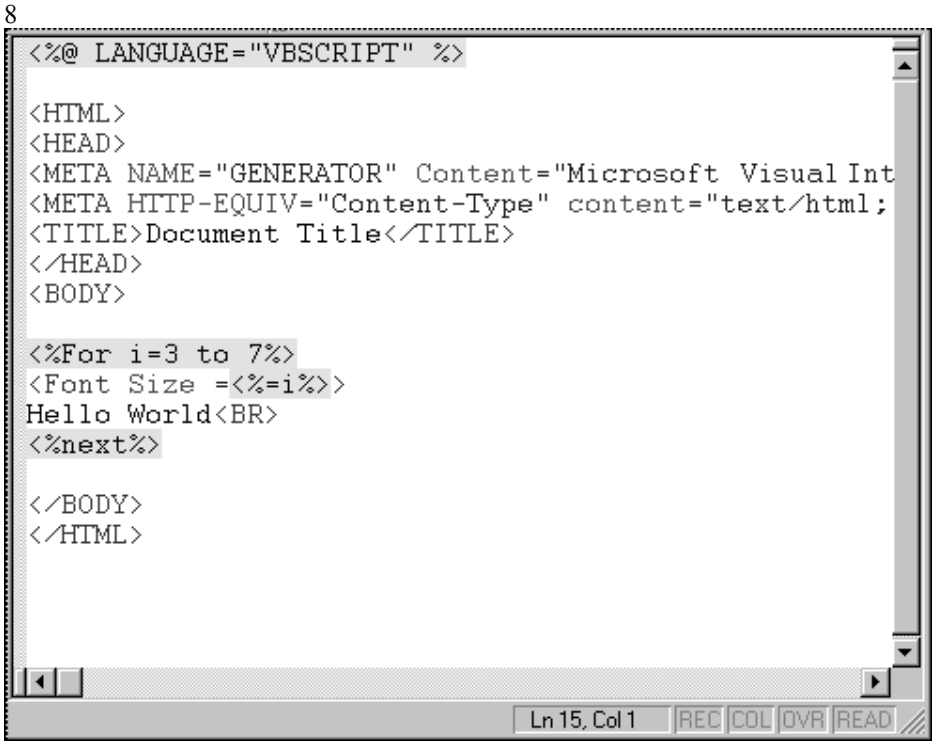
To add HTML code in the Source Editor

39 In the HTML <TITLE> tag, select **Document Title** and replace it by typing "HelloWorld."

40 Position the pointer after the first HTML <BODY> tag, then type the following HTML and server-side script:

```
<% FOR i = 3 TO 7 %>
<FONT SIZE=<%= i %>>
Hello World!<BR>
<% Next %>
```

8



As you type, the HTML Source Editor highlights everything within HTML <%></%> script tags in yellow. This indicates that the script will execute on the Web server, but won't be sent to the browser.

In the example cited above, the script in the ASP will perform a loop on the server, dynamically generating the text, "Hello World," in an increasing font size in the HTML document sent to the browser.

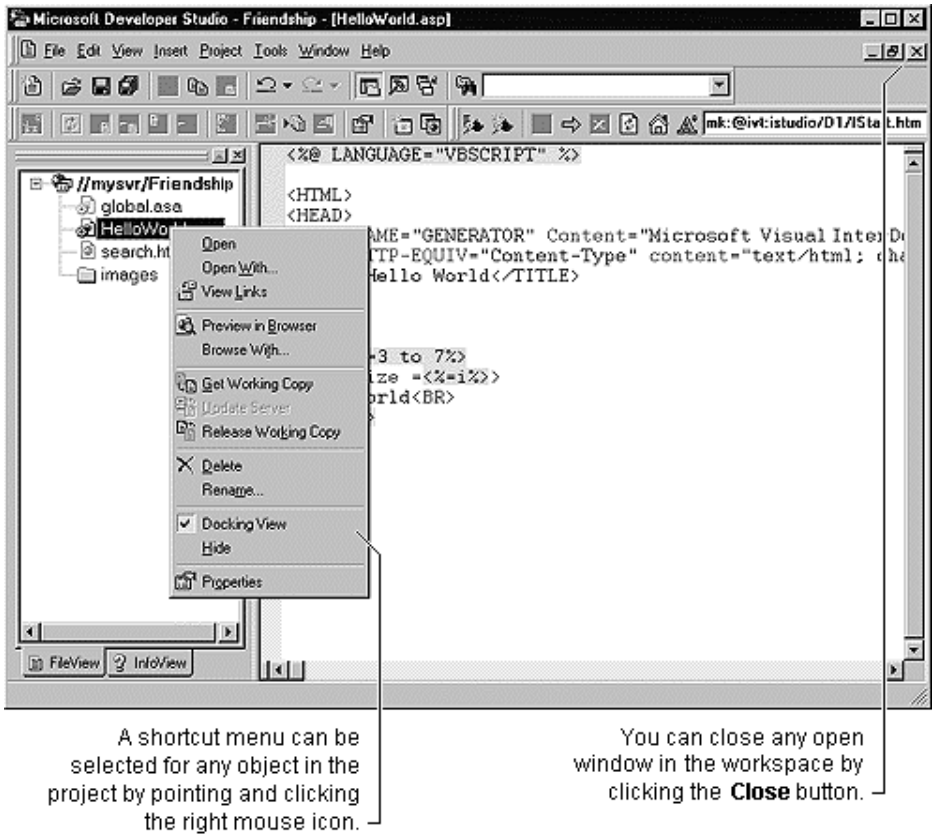
Saving and Previewing the Page

To save and preview the Active Server Page

41 On the **File** menu, click **Save**.

42 In **FileView**, right-click **HelloWorld.asp** to display the shortcut menu, and then click **Preview in Browser**.

9



There are three ways you can browse a file under development:

- By using InfoViewer to browse without leaving the Visual InterDev IDE (**Browse With, InfoViewer**).
- By using the default browser installed on your computer (**Preview in Browser**).
- **Note** Both the **Browse With** and **Preveiw in Browser** commands are on the project and file shortcut menus. In addition, the **Preview in Browser** command is located on the File menu.
- By choosing from a list of all the browsers installed on your computer (**Browse With**). You can select multiple browsers from the list.

13

Note Both Microsoft Internet Explorer and InfoViewer implement content caching by default. When you preview a page that you've previewed previously, be sure to click **Refresh** on the toolbar to ensure that you're looking at the latest version of your page.

4

10

14

If you view the HTML source for that page in your Web browser, you'll see that all of the server-side script has been stripped out of the document by the server. Only HTML is sent to the browser, so that you can view ASP files using any standard Web browser running on any platform.

Releasing the Working Copy

In **FileView**, you'll notice that the icon for HelloWorld.asp is now in color, while the icons for the other elements in the Web project are gray. A color icon means that you have a local working copy of the file on your hard drive. A gray icon means that the file exists only on the Web server, and you don't have a local working copy.

When you're done editing a file, you can easily release the working copy back to the server.

To release the working copy back to the server

- In **FileView**, right-click **HelloWorld.asp** to display the shortcut menu, and then click **Release Working Copy**.

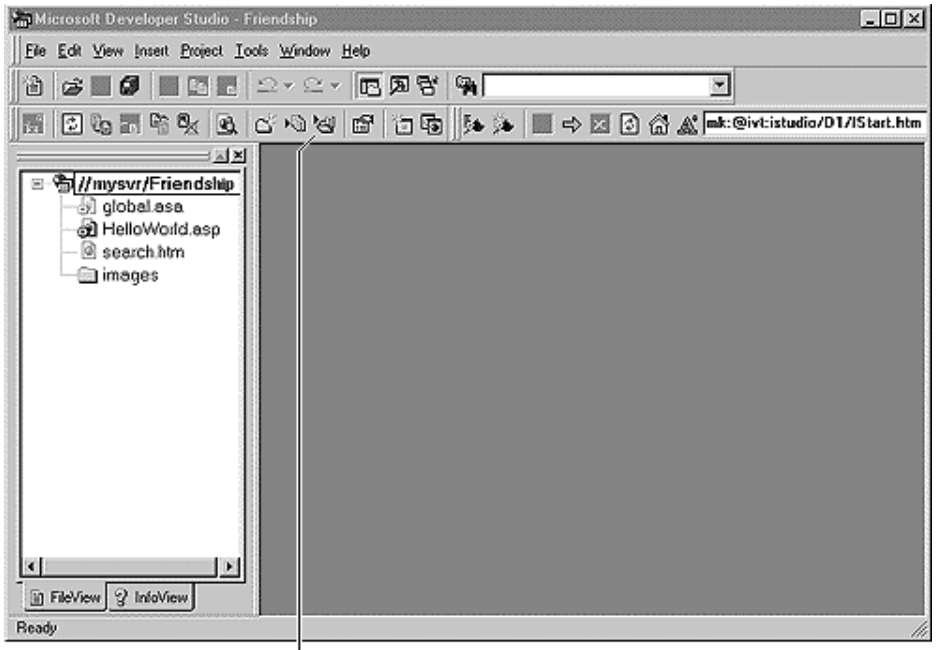
  6Visual InterDev releases the local working copy of the file back to the server, and then shades the icon gray.

You can retrieve a working copy of any file and open it in the default editor for that file type by double-clicking the file in **FileView**.

Tip The Visual InterDev toolbar also provides easy access to most of the Web project commands. Try releasing the working copy by clicking the corresponding toolbar button. You can select which toolbars you want displayed using the **Customize** command on the **Tools** menu.

15

11



Adding External Content Files to a Web Project

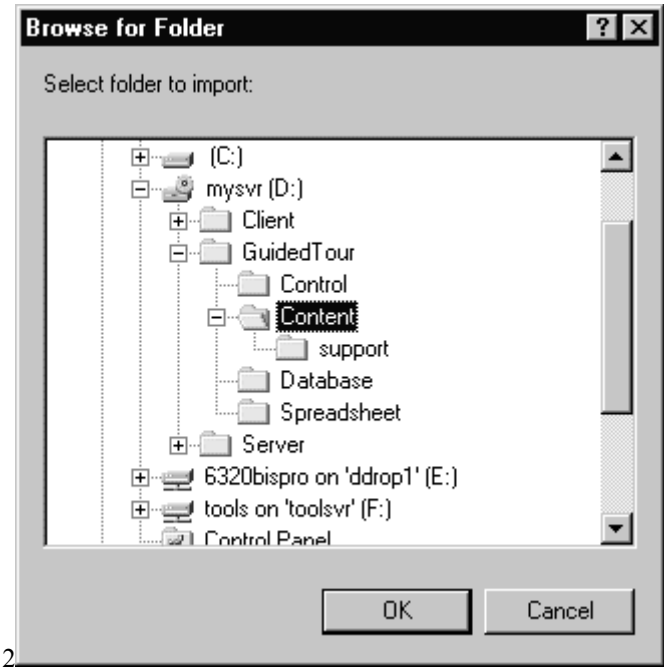
Most developers have existing content that they'll want to add to their Web projects. Visual InterDev makes this task easy.

To add external files to a Web project

43 In **FileView**, right-click the root level of the project, and then click **Add Folder Contents**.

44 In the **Browse for Folder** dialog box, locate the folder \\GuidedTour\Content, and then select it. (Make sure you select just the **Content** subfolder, and not the entire GuidedTour folder.)

16

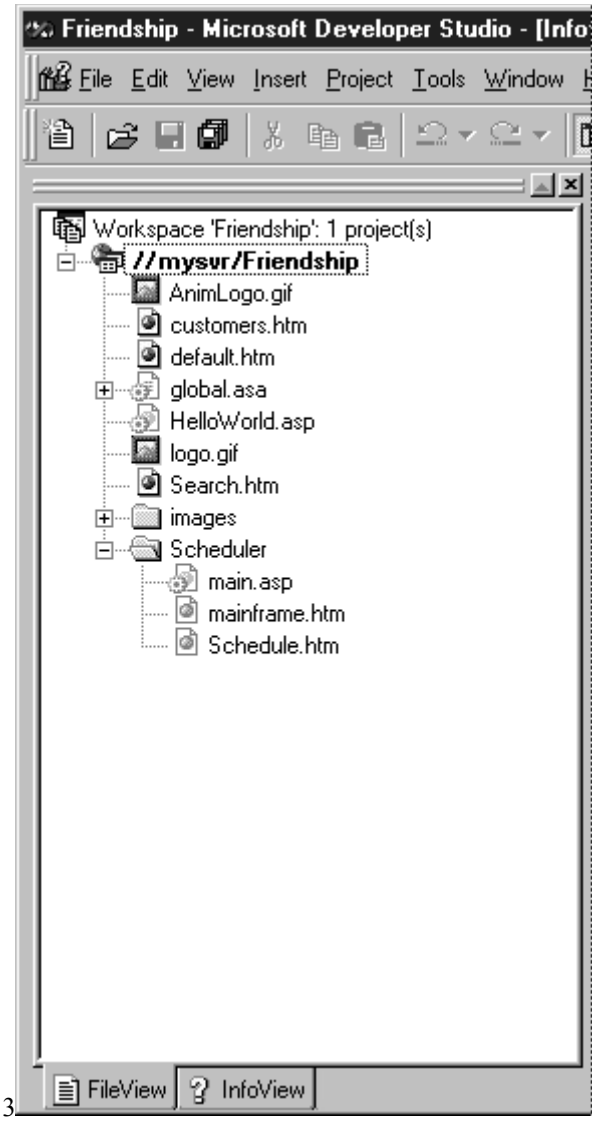


12

17

45 Click **OK** to add the contents of this folder to your Web project. Visual InterDev copies the contents automatically, via HTTP, to your Web server.

18



13

19

Tip You can also drag files and folders from Windows Explorer® directly into a Web project, or into a folder within a project. This will publish the files, over HTTP, to the appropriate location on the Web server.

Lesson 3 - Using the FrontPage Editor for Visual InterDev

The FrontPage Editor for Visual InterDev lets developers edit HTML pages in WYSIWYG mode, without having to manually type HTML tags. In this lesson, you'll use the FrontPage Editor to create a display template for an Active Server Page. In a subsequent lesson, you'll use the ASP to display records from the Friendship Insurance customer database.

Opening and Editing the Customer Listing Page

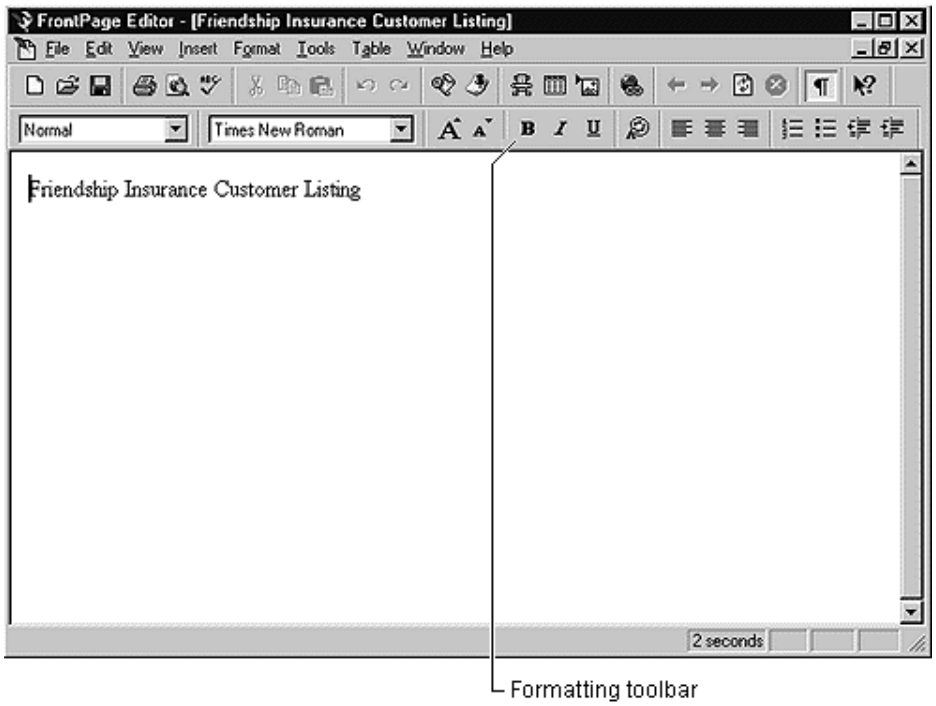
46 In **FileView**, right-click **Customers.htm** to display the shortcut menu, and then click **Open With**.

7This displays the **Open With** dialog box, which lets you choose the editor you want to use to open the specified file.

47 In the list of editors, click **Microsoft FrontPage Editor (Visual InterDev Edition)**, and then click **Open**.

20

14



21

8The FrontPage Editor opens, showing the partially completed Web page in WYSIWYG editing mode.

48 Select the heading **Friendship Insurance Customer Listing**.

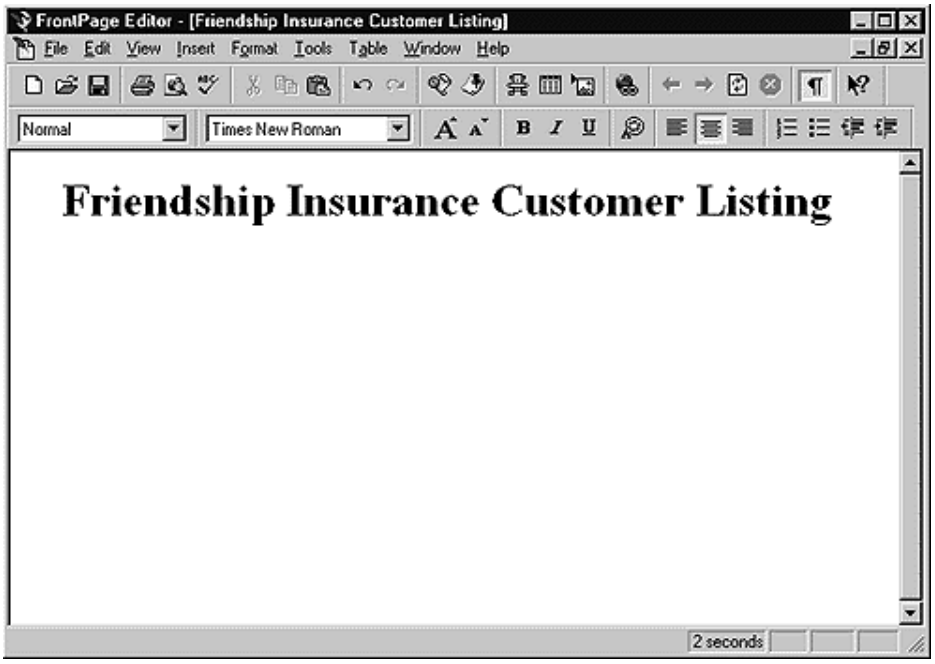
49 On the **Formatting** toolbar, click **Bold** to make the heading bold.

50 Click the **Increase Font Size** button three times to increase the size of the heading font.

51 Click the **Center Text** button to center the heading. The page should now resemble the following figure.

22

15

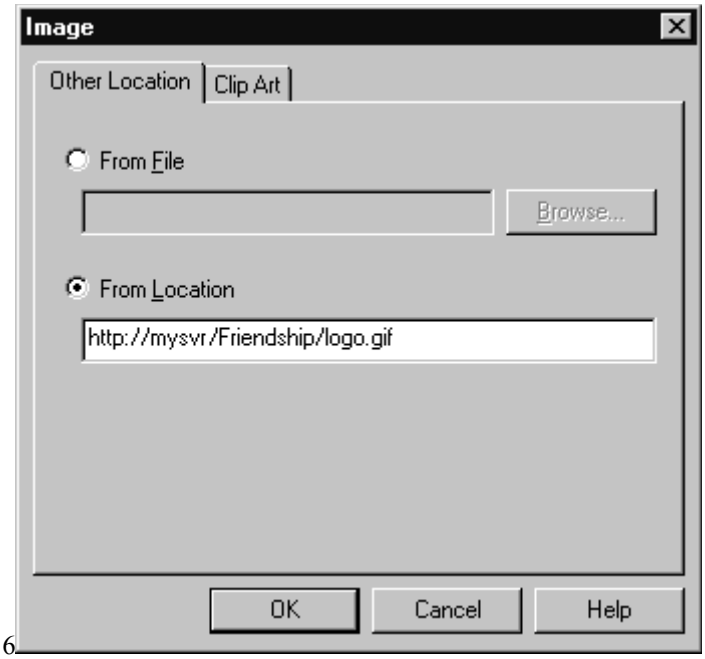


23

52 Position the cursor at the end of the heading line, and then press **Enter** to start a new line after the heading.

53 On the **Insert** menu, click **Image**.

24



16

25

54 Select **From Location**, and then after the **http://** prefix, type **[*your server name*]/friendship/logo.gif** and click **OK**. The FrontPage Editor will insert the Friendship Insurance logo into the page based on the URL to your Web application.

Tip You can also enter a relative URL in the **From Location** box. For example, you could enter logo.gif in the **From File** box. If you have a working copy of logo.gif on your machine, the image will appear on the page as you edit it. If you don't have a working copy of logo.gif, you can still view the image when you preview the page in the browser.

26

Inserting and Editing a Table

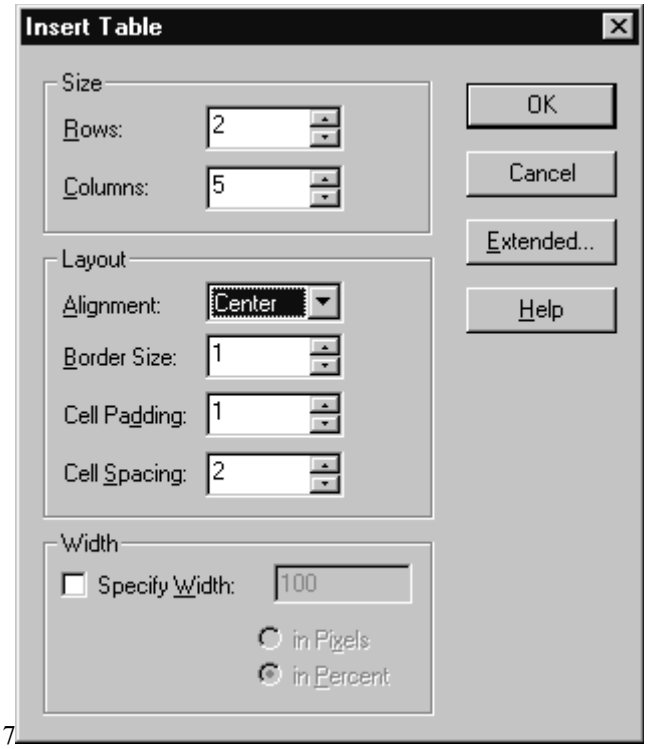
55 In the FrontPage Editor window for **Customers.htm**, press **Enter** to start a new line after the image you just inserted.

56 On the **Table** menu, click **Insert Table**.

9This displays the **Insert Table** dialog box.

57 Specify the initial table values as follows: in the **Columns** box, type "5"; in the **Alignment** box, choose **Center**; and then in the **Border Size** box, type "1."

27



17

28

58 Click **OK** to insert the table using the specified values.

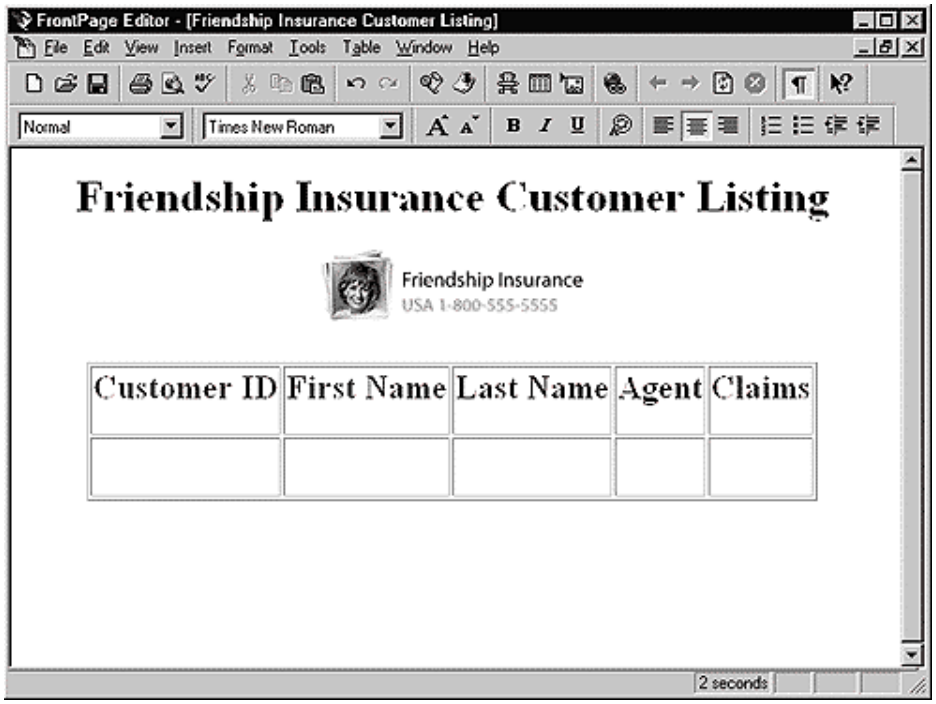
59 In the FrontPage Editor window, select the top left cell of the table you just inserted, and then type the column heading “Customer ID.”

60 In the remainding four cells of the first row of the table, type the following column headings: “First Name,” “Last Name,” “Agent,” and “Claims.”

61 Select the entire heading row, and then format it as follows: On the **Formatting** toolbar, click the **Bold** button, click the **Increase Font Size** button twice, and then click the **Center Text** button.

29

18



30

62 Select the second row of the table (the empty cells), and then click the **Center Text** button. This will center the customer data when it appears in the table.

63 On the **File** menu, click **Save**, and then click **Exit** to close the FrontPage Editor.

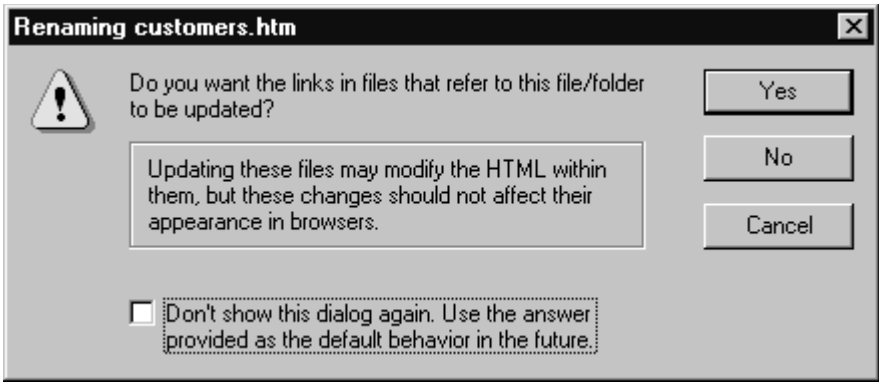
31

64 In **FileView**, right-click **Customers.htm** to display the shortcut menu, and then click **Release Working Copy**.

Renaming a File

Now that you've saved **Customers.htm**, you'll change it to an Active Server Page by renaming it with the .asp extension. To do so, in the **FileView** tab, right-click **Customers.htm** to display the shortcut menu, and then click **Rename**. In the **To** box, change the name to **Customers.asp**, and then click **OK**. If you have automatic link repair turned on, you'll see the following dialog box.

32



33

As you rename or move elements of your Web site, Visual InterDev can repair hyperlinks automatically. In the example above, Visual InterDev will automatically repair any links within your Web site to the newly renamed Customer.asp after you click **Yes**. To view or change Visual InterDev's default settings for repairing links, on the **Tools** menu, click **Options**, and then click the **Web Projects** tab.

Note You can turn on automatic link repair for a site by right-clicking the project root and then choosing **Properties**. Select the **On** button under **Link Repair**.

34

Working with Third-Party Content Editors

In Visual InterDev, you can open a file using one of the default editors, such as the HTML Layout Editor, or you can use any other editor installed on your computer.

The **Open With** dialog box lets you open a file in a third-party editor, add a third-party editor to the list of default editors, or set a third-party editor as the default for the specified file type.

To open a file in a third-party editor

65 In **FileView**, right-click the file you want to edit, and then click **Open With** on the shortcut menu. (For example, to open the image file Logo.gif in Adobe® PhotoShop®, right-click the file in **FileView**, and then choose **Open With**.)

66 Click **Add** to browse for the PhotoShop program .exe and add it to the list of editors. You can also click **Set as Default** to specify PhotoShop as the default editor for .gif files.

You can use this same procedure to associate any file type with your favorite editor.

As you get and release working copies of the files, Visual InterDev ensures that the files are updated on the server, and also manages the links between files on the server. If you're using source control on the project for team-based development, Visual InterDev will also manage checking files in and out.

Lesson 4 - Connecting to a Database

Now that you have created a template for displaying customer information on the Customer Listing page, you can add script to display dynamic information from the Friendship SQL Server customer database. The first step is to establish a connection to the customer database within the project.

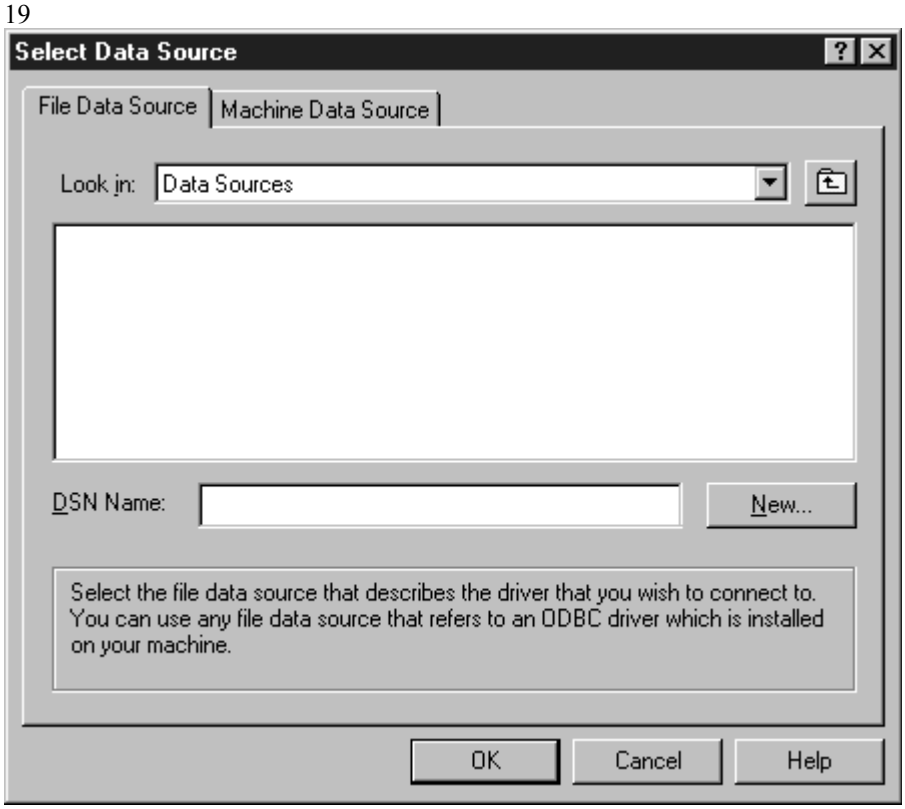
To add a data connection to a Web project

67 In **FileView**, right-click the root level of the project, and then choose **Add Data Connection**.

68 In the **Select Data Source** dialog box, click the **File Data Source** tab.

69 Click **New** to create a new file data source.

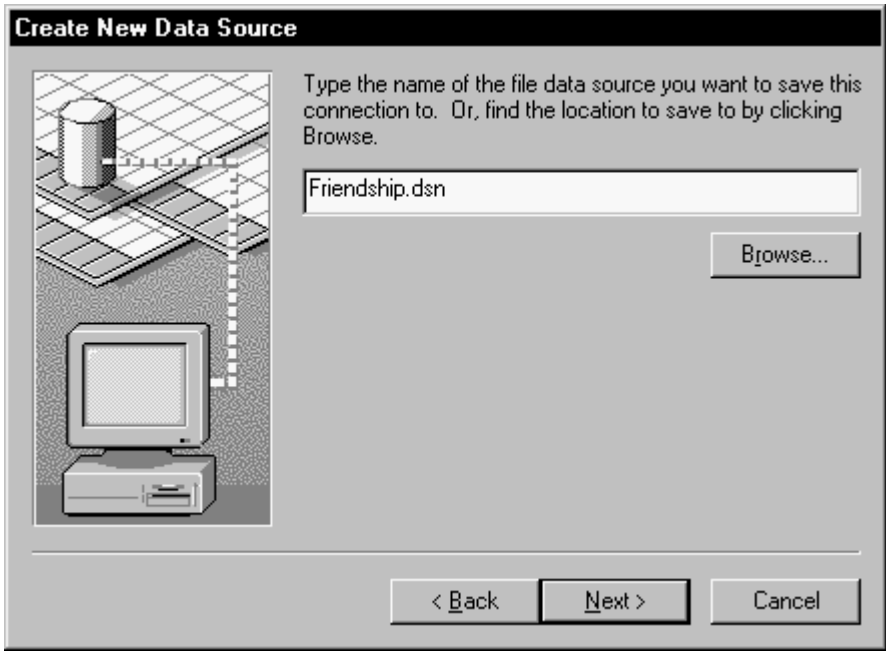
19



70 In the **Create New Data Source** dialog box, select the SQL Server driver, and then click **Next**.

71 Enter **Friendship.dsn** in the file data source box, and then click **Next**.

20



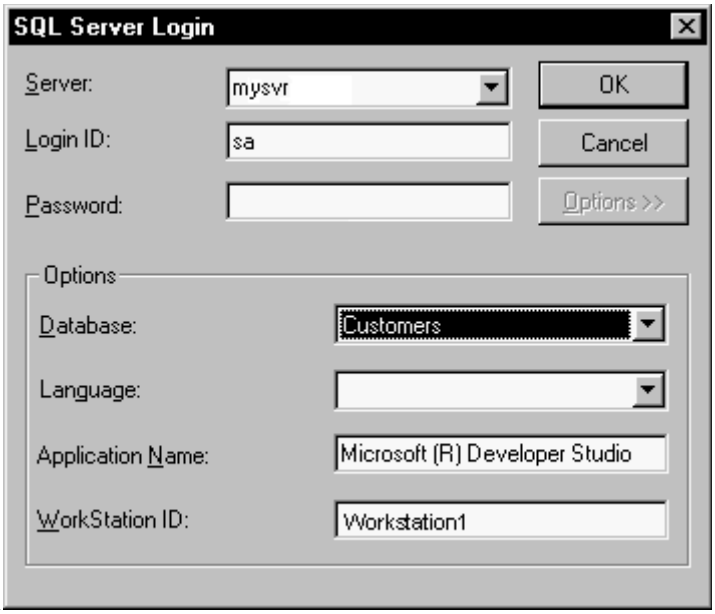
72 Click **Finish**.

10The SQL Server login dialog is displayed.

73 In the **Server** list box, select the database server.

74 In the **Login ID** and **Password** fields, enter your login ID and password, and then click the **Options** button.

75 In the **Database** field, select the **Customers** database.



21

76 Click **OK** to close the dialog box.

11 You are returned to the **Select Data Source** dialog box.

77 Select **Friendship.dsn** from the list of data sources, and then click **OK**.

78 In the SQL Server login dialog box, click **OK**.

12 The property page for the data connection is displayed.

79 In the **Name** field, enter **CustomerDB** for the database connection.

80 In the **Run Time** tab of the property page, enter the SQL Server user ID and password that you want to use as the default login ID when customers browse your Web site.

Note If you are using the Access version of the Friendship Insurance database, you don't need to enter a login ID or password.

5

You must enter a valid ID and password for your database if it is secured (for example, if you have a unique password for the SQL Server account and want users to log on under a different ID). The login ID and password information is hidden and inaccessible to users browsing your site. If you leave the **Run Time** tab blank, an attempt is made to establish a connection at run time using the design-time ID entered in step 4, but with no password.

Once you establish a connection to the database, the global.asa file is copied to your local drive. In **FileView**, a database connection icon (for example, CustomerDB) appears below the Global.asa file. This file is the "front door" to your Web application, and allows the Web server to initialize application variables such as

database connections whenever a user enters your Web site for the first time. The Global.asa file is executed once for each user no matter which page in your application is viewed first.

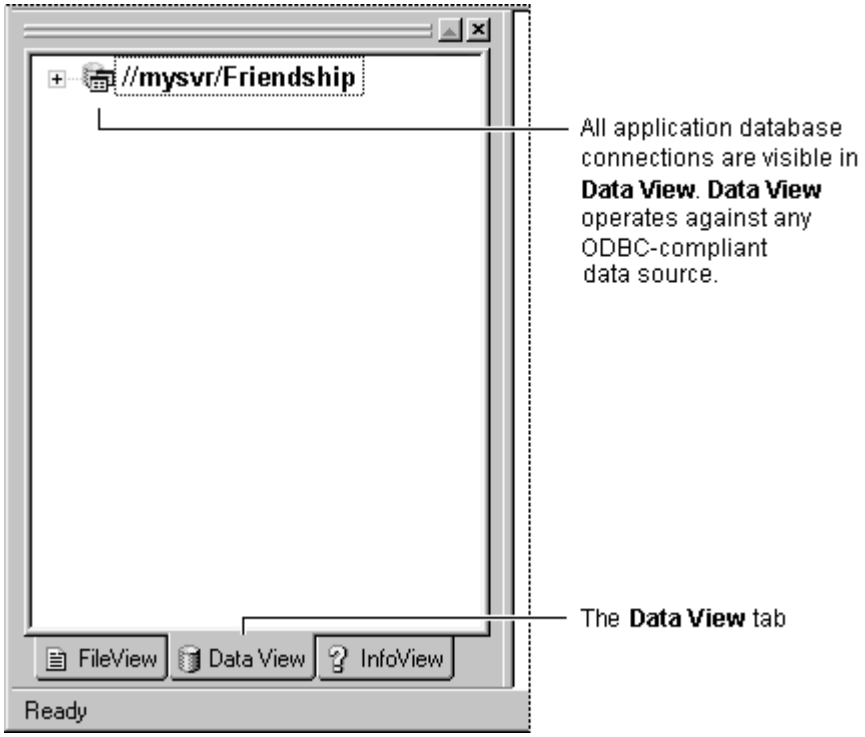
Working in Data View

Microsoft Visual InterDev provides easy access to a range of powerful database development tools, all integrated directly within the development environment. One of these database features is Data View, which allows developers to view detailed information on each database they are using in the Web application.

To view information on the Friendship Insurance Customers database

81 Click the **Data View** tab in the Project Workspace window.

22



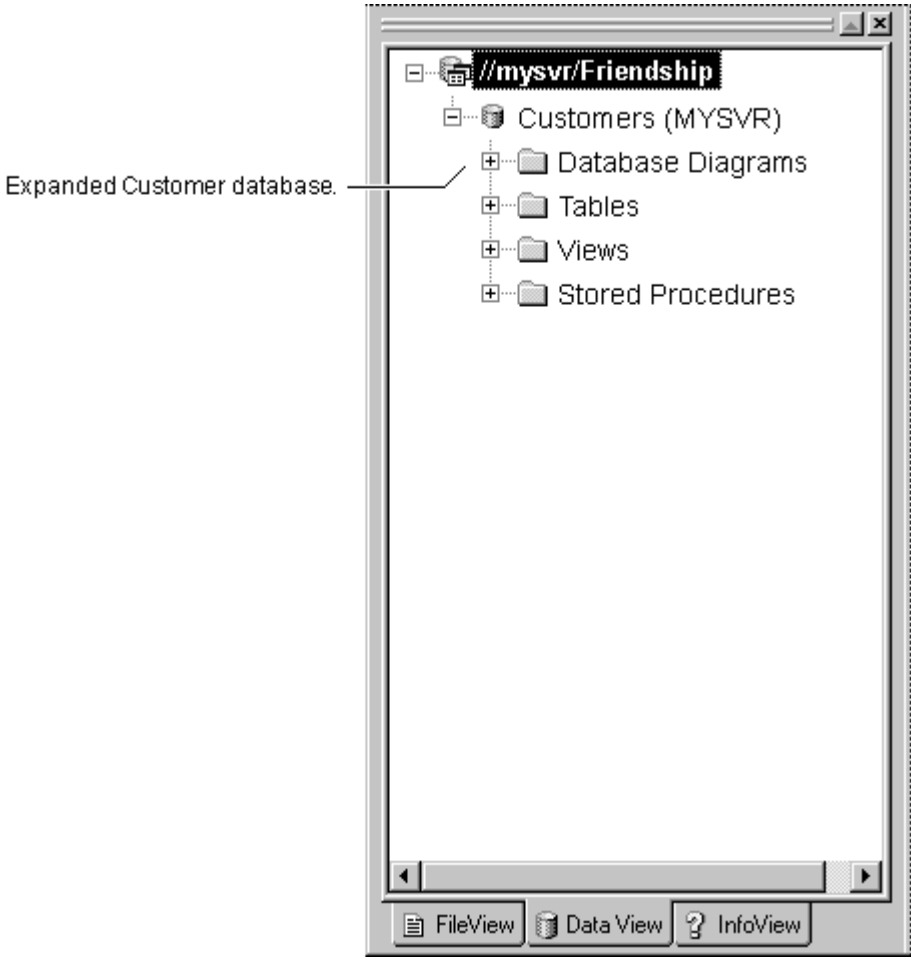
82 Click the plus (+) sign adjacent to the left of the **Friendship** project name.

13The Customers database name appears below the database connection.

83 Click the plus (+) sign adjacent to the **Customers** database name.

14Folders for the diagrams, tables, views, and stored procedures in the database are displayed.

23



84 Click the plus (+) sign adjacent to the **Tables** folder to view the tables in the Customers database.

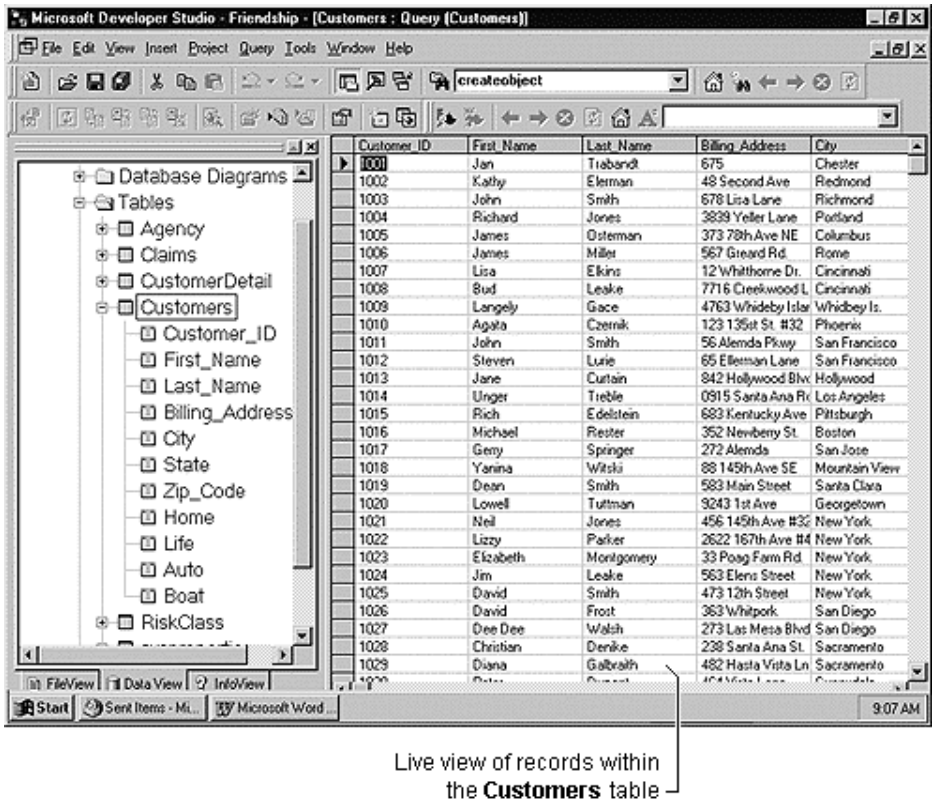
24 

85 Click the plus (+) sign adjacent to the **Customers** table to view the columns contained within that table.

86 Double-click the **Customers** table to see a live view of the current recordset for that table.

15If you have write privileges for the database, you can enter new records, modify existing records, or delete records from within the development environment.

25



87 Close the recordset by clicking the **X** (**Close** button) in the upper-right corner of the window.

Lesson 5 - Working with the Query Designer

Visual InterDev includes a Query Designer that allows developers to visually build and test SQL statements against any ODBC database. In this lesson, you'll use the Query Designer to build a query that returns records from the Customer database.

Using Design-Time ActiveX Controls

While the Query Designer can be used standalone to build and test SQL statements within the development environment, special data-access controls called *design-time ActiveX controls* can be used with the Query Designer to generate server-side script that executes a query on the Web server and returns the results on a dynamic Web page.

Design-time ActiveX controls generate standard HTML and script that can be viewed in any browser on any platform. While Microsoft Visual InterDev includes a number of design-time ActiveX controls, third parties can build their own controls to extend Visual InterDev with new functionality.

Adding a Data Range Header Control

An example of a design-time ActiveX Control is the **Data Range Header** control, which lets you build an SQL query in the Query Designer, and then automatically builds the server script to display the returned records in HTML. This allows users to page forward and backward through the returned records.

To add a Data Range Header control to the Customer Listing page

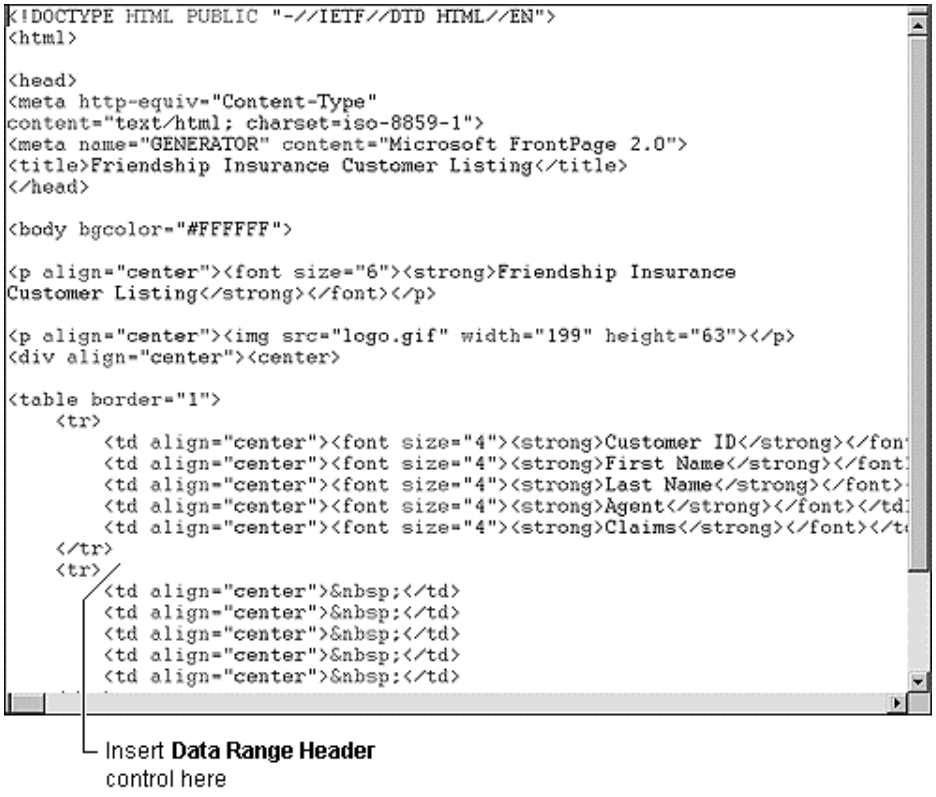
88 In **FileView**, double-click the Customers.asp page.

16 The page opens in the Text Editor.

89 Locate the end of the first row definition in the table (that is, the first closing </tr> tag).

90 Place the text insertion pointer directly after this </tr> tag, and click **Enter** to start a new line.

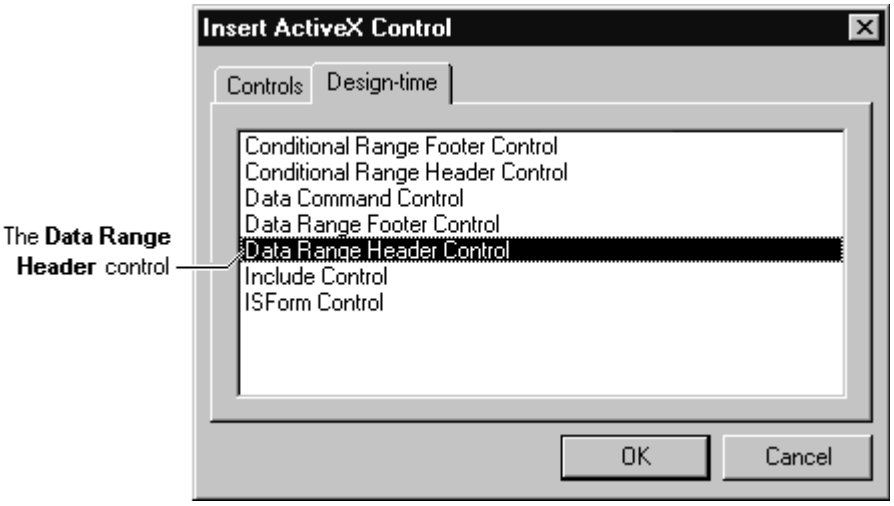
26



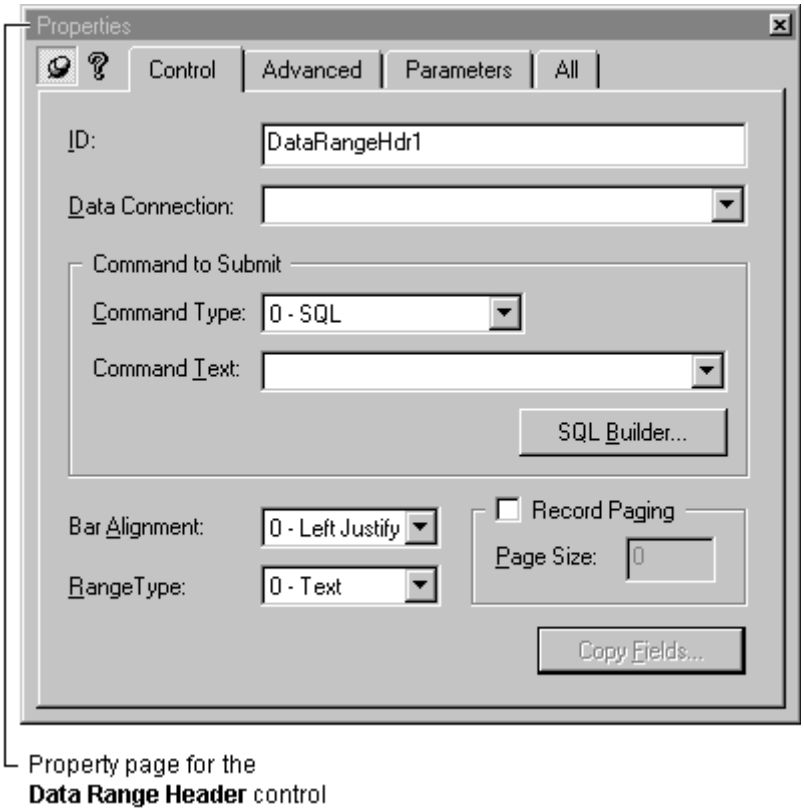
91 From the **Insert** menu, click **Into HTML**, and then click **ActiveX Control**.

92 Click the **Design-time** tab, and then choose **Data Range Header Control**.

27



17The Object Editor opens the **Data Range Header** control and displays its property page.



28

93 Click the **Advanced** tab on the property page. In the **Cursor Type** list box, choose **1-Keyset**.

94 Click the **Control** tab on the property page. In the **Data Connection** list box, choose the **CustomerDB** connection.

95 In the **Bar Alignment** list box, choose **Centered**.

18This setting centers the paging buttons (buttons used to page through database records) inserted at the bottom of the page.

96 Select the **Record Paging** check box and enter “10” in the **Page Size** box.

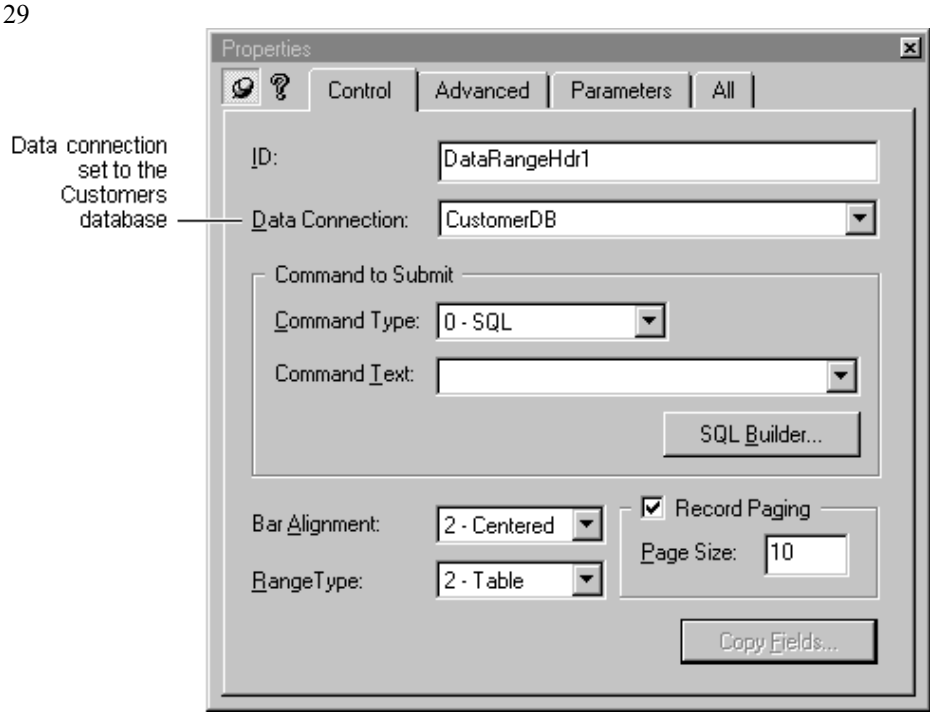
19These settings enable customers to page through records rather than display all of the records in a single scrolling table.

97 In the **Range Type** list box, choose **Table**.

20This setting indicates that results will be displayed in an HTML table.

21

29



Using the Query Designer

The Query Designer enables developers to build SQL statements against any ODBC database. Using a group of graphical design panes, developers can drag and drop tables to interactively build and test their queries.

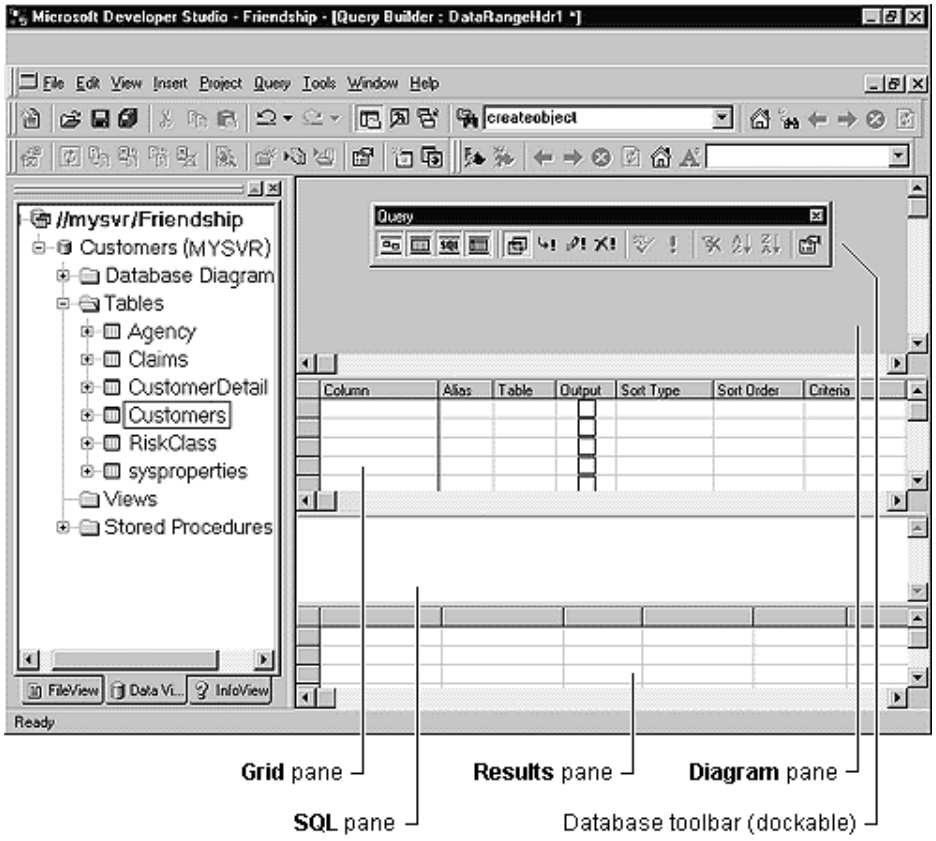
You can start the Query Designer from the property page for the **Data Range Header** control.

To build a query using the Query Designer

98 On the **Control** tab in the **Data Range Header** control's property page, click the **SQL Builder** button.

22The Query Designer opens in the right window.

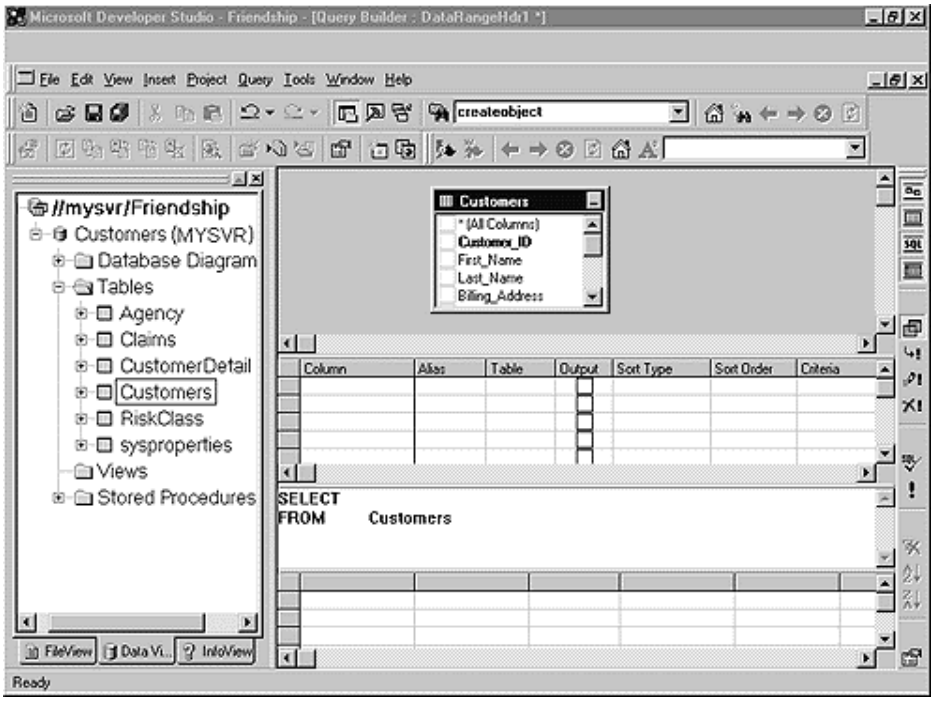
30



99 Drag the floating Query Designer toolbar to the right side of the Query Designer, if desired.

100In **Data View**, drag the **Customers** table into the **Diagram** pane of the Query Designer.

31



101 In the **Customer** table, select the boxes adjacent to the **Customer_ID**, **First_Name,** and **Last_Name** fields.

32

102 Test the query by clicking the **Run** icon on the **Database** toolbar.

33

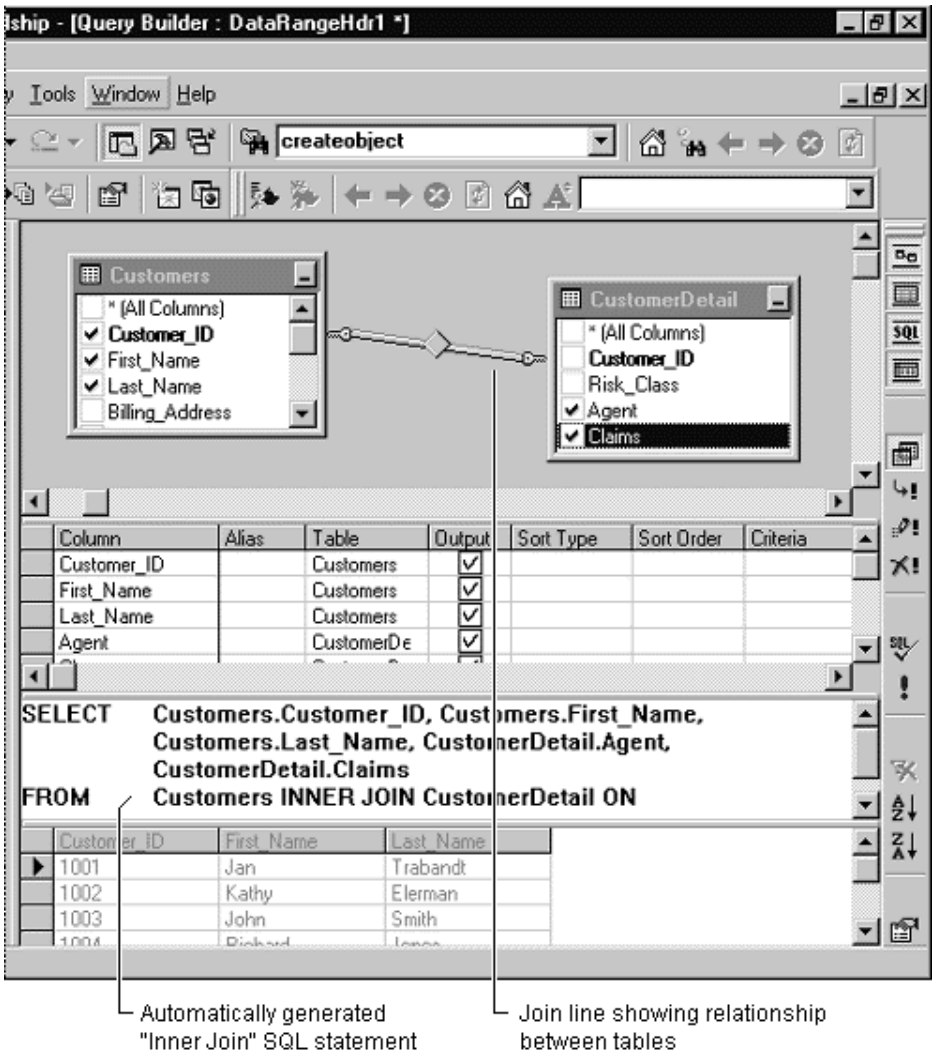
23 The results are displayed in the **Results** pane of the Query Designer.

103 Drag the **CustomerDetail** table into the **Diagram** pane.

24Notice how the Query Designer recognizes the relationship between the tables and joins them with a line. The SQL statement for the query appears in the **SQL** pane.

104Select the boxes adjacent to the **Agent** and **Claims** boxes in the **CustomerDetail** table.

34



105Right-click the **Last_Name** column in the **Customers** table, and choose **Sort Ascending** to sort customers by last name.

106Click the **Run** button on the **Database** toolbar to test the query.

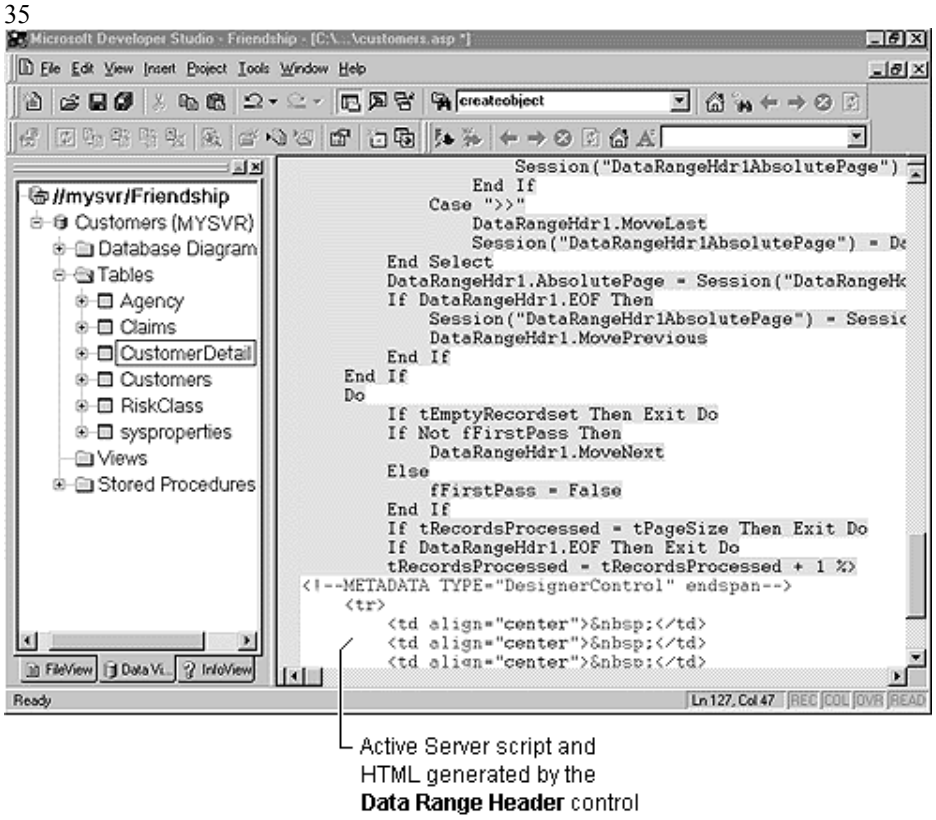
107Once you are satisfied with the query, close the Query Designer.

108Choose **Yes** when prompted to update the database connection.

109From the **File** menu, choose **Close** to close the **Data Range Header** control.

25The **Data Range Header** control automatically generates the HTML and server-side script to execute the SQL query, and inserts it into the Customer Listing page.

35



Note To edit an existing design-time ActiveX control, right-click in the generated script, and then click **Edit Design-time Control**.

Adding the Records to Display

The next step is to add the records you would like to display in the HTML table. The **Data Range Header** control automatically loops through the available records in the recordset, dynamically extending the table for each row returned.

To add records to display in the HTML table

110In customers.asp, find the table row that will contain the database fields returned.

1.

```
            Session("DataRangeHdr1AbsolutePage") = Sessi
            DataRangeHdr1.MovePrevious
        End If
    End If
    Do
        If tEmptyRecordset Then Exit Do
        If Not fFirstPass Then
            DataRangeHdr1.MoveNext
        Else
            fFirstPass = False
        End If
        If tRecordsProcessed = tPageSize Then Exit Do
        If DataRangeHdr1.EOF Then Exit Do
        tRecordsProcessed = tRecordsProcessed + 1 %>
<!--METADATA TYPE="DesignerControl" endspan-->
    <tr>
        <td align="center"> </td>
        <td align="center"> </td>
        <td align="center"> </td>
        <td align="center"> </td>
        <td align="center"> </td>
    </tr>
</table>
</center></div>
</body>
</html>
```

Row definition to be dynamically filled with database information

111Replace the existing table row definition with the following:

```
<tr>
    <td align="center"><%= DataRangeHdr1("Customer_ID") %></td>
    <td align="center"><%= DataRangeHdr1("First_Name") %></td>
    <td align="center"><%= DataRangeHdr1("Last_Name") %></td>
    <td align="center"><%= DataRangeHdr1("Agent") %></td>
    <td align="center"><%= DataRangeHdr1("Claims") %></td>
</tr>
```

Tip Instead of typing in the script to display field names in an HTML table as shown above, you can use the **Copy Fields** button in the **Data Range Header** control's property page to copy the correct script to the Clipboard. The script can then be pasted directly into the HTML code for the table.

Adding a Data Range Footer Control

The **Data Range Footer** control is used in conjunction with the **Data Range Header** control to write the server script that finishes the loop to move through all the records in the recordset. To complete the Customer Listing page, you need to insert a **Data Range Footer** control at the bottom of the table definition.

To insert a Data Range Footer control into the Customer Listing page

112 Locate the </table> tag that ends the table definition. Position the pointer just before this tag, but after the </tr> tag that immediately precedes it.

113 From the **Insert** menu, click **Into HTML** and then click **ActiveX Control**.

114 Click the **Design-time** tab, and then choose **Data Range Footer Control**.

115 Close the property page for the control.

116 Close the Object Editor.

26 The **Data Range Footer** control automatically generates the HTML and server-side script to end the data range loop and inserts it in the page.

117 Save the completed Customer Listing page.

118 Close the Customers.asp page.

119 In **FileView**, right-click Customers.asp and then click **Preview in Browser**.

The completed page shows live database records in a dynamically generated HTML page. The user can page through the records using the navigation bar at the bottom of the page. When a table contains hundreds or even thousands of records, paging through records is faster than having to wait for all the records to be displayed on a page.

2.


Friendship Insurance Customer Listing - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites Print Font Mail Links
Friendship Insurance Customer Listing
Friendship Insurance
USA 1-800-555-5555
Customer ID First Name Last Name Agent Claims
1001 Jan Trabandt Sean Chai 0
1002 Kathy Elerman Lisa Smith 1
1003 John Smith Jim Porter 2
1004 Richard Jones John Jones 0
1005 James Osterman Sean Chai 1
1006 James Miller Sean Chai 0
1007 Lisa Elkins Lisa Smith 1
1008 Bud Leake Jim Porter 0
1009 Langely Gace Lisa Smith 0
1010 Agata Czernik John Jones 1
<< < > >> Requery
Page: 1

Lesson 6 - Using the Data Form Wizard

In the previous lesson, you created a custom page that returns a listing of customer records from a database. Web developers might also want to build HTML forms that allow users to modify database information directly through their Web browser—for example, inserting, updating and deleting database records in a table. Normally, building HTML forms bound to data is a complex task, but Visual InterDev's Data Form Wizard simplifies this task.

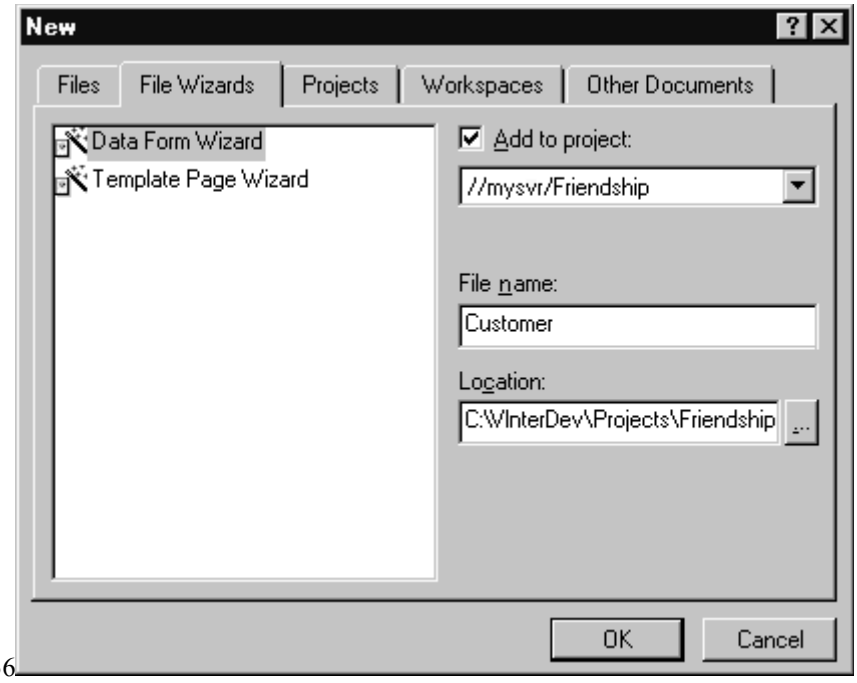
In this lesson, you'll use the Data Form Wizard to build a customized HTML form that allows Friendship insurance agents to modify customer records stored in a SQL Server database.

Note The Data Form Wizard works with any ODBC database, not just SQL Server.

To start the Data Form Wizard

120 From the **File** menu, click **New**.

121 Click the **File Wizards** tab.



36

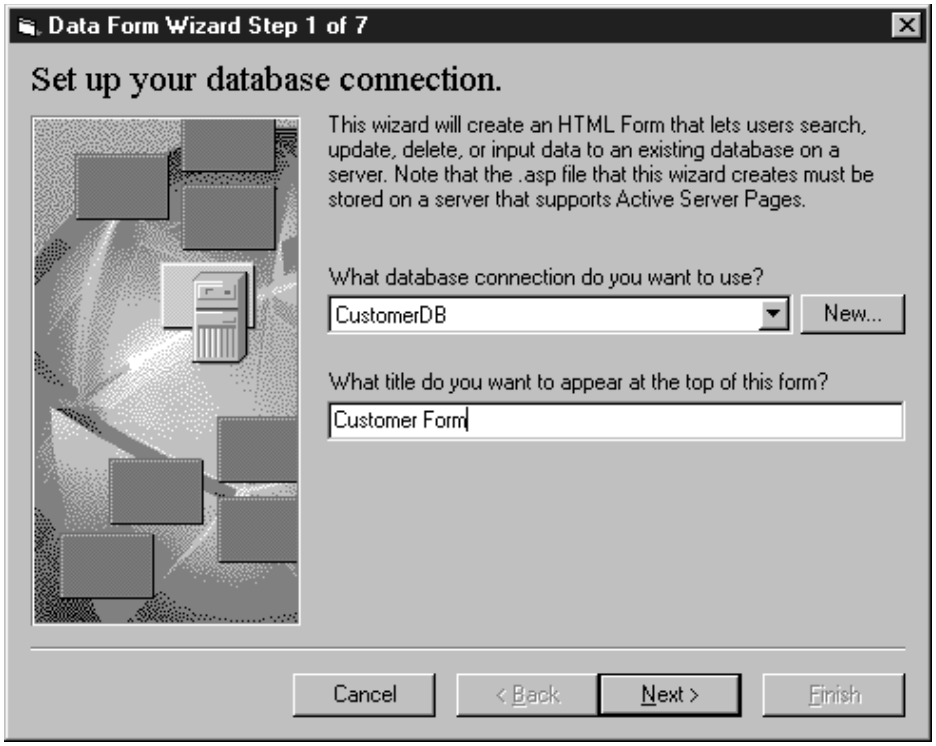
122 Select the **Data Form Wizard**, and enter **Customer** as the **File name.**

123 Click **OK**.

124Verify that **CustomerDB** appears in the database connection box.

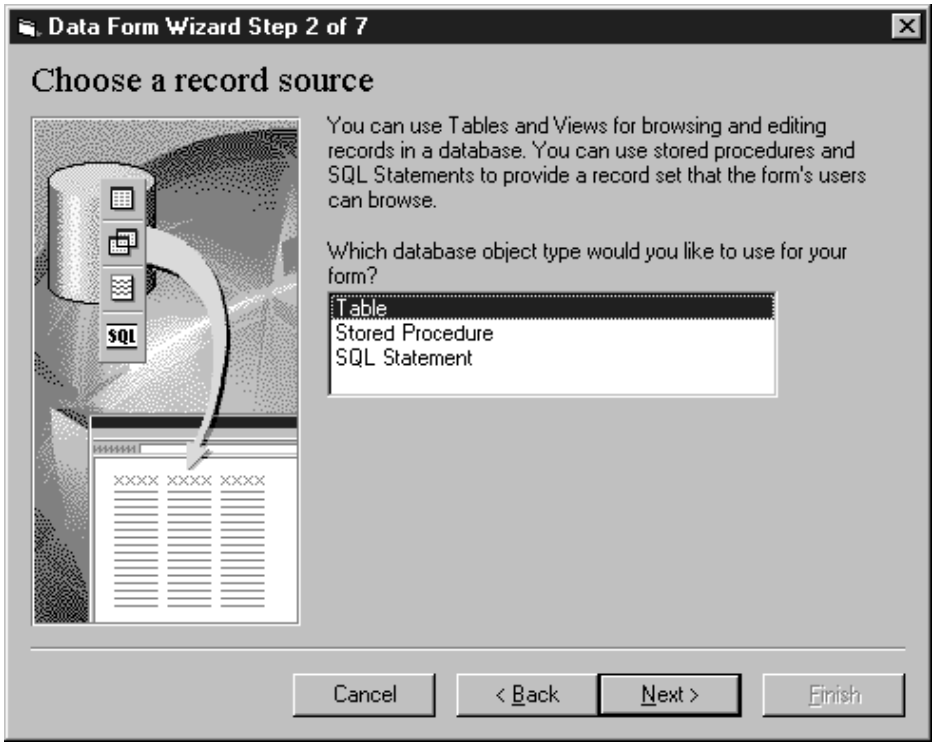
125Enter **Customer Form** for the form title.

37



126Click **Next**.

38

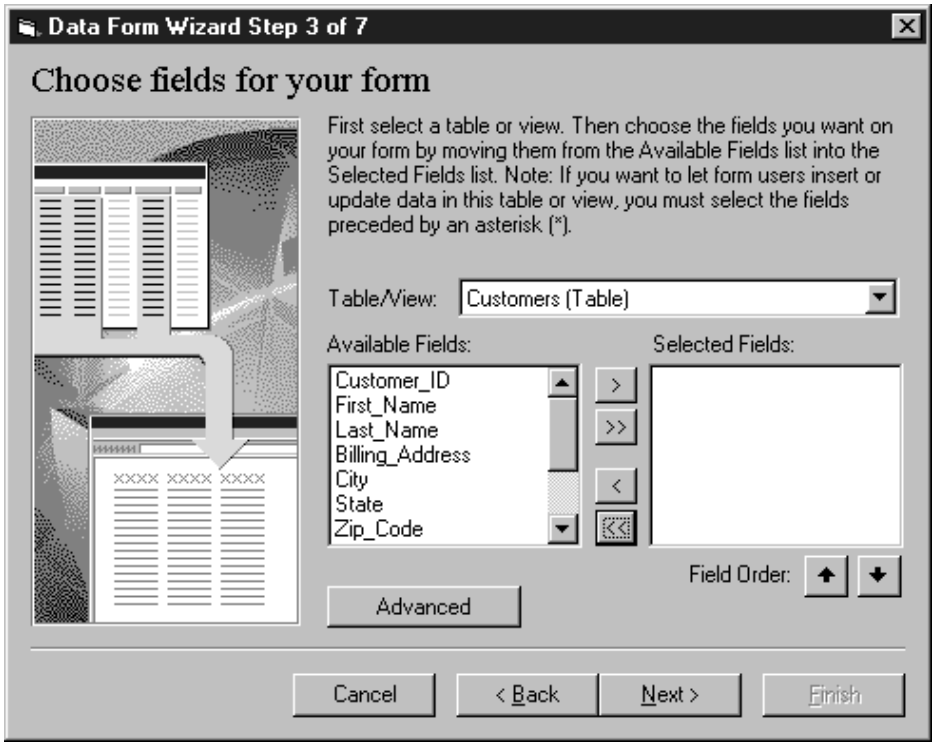


27You need to specify whether a table, a view, or an SQL statement will be used to provide a recordset for a data form.

127Select **Table**, and then click **Next**.

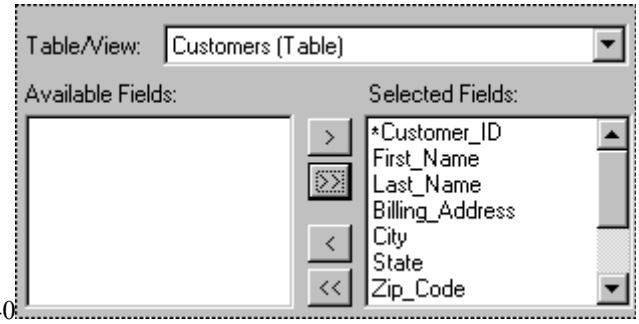
128In the **Table/View** list box, select the **Customers** table.

39



129Click the >> icon to select all available fields.

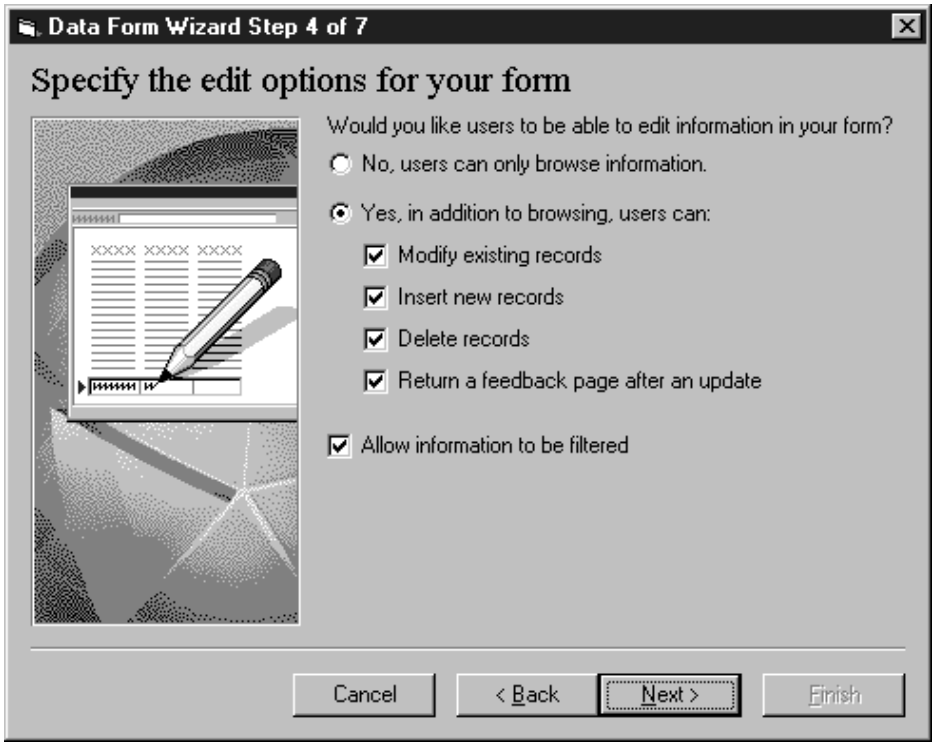
28All the fields in the **Customers** table are moved to the Selected Fields box.



40

130Click **Next**.

41



131 Click **Next** to keep the default values.

29 The **Select Viewing Options** dialog box is displayed.

132 Click **Next** to keep the default viewing options.

30 A dialog box that lets you choose visual themes for your data form is displayed.

133 Click **Next** to keep the default theme.

134 Click **Finish**.

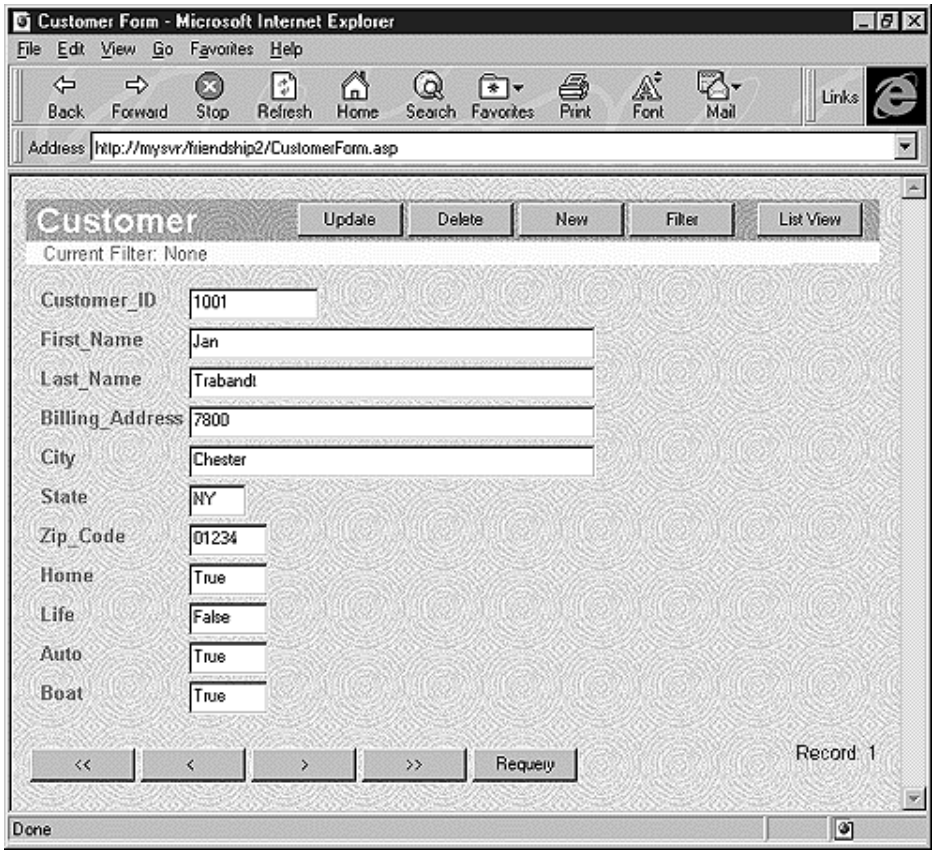
The Data Form Wizard automatically generates three new Active Server Pages for the customized Data Form:

- CustomerForm.asp
- CustomerList.asp
- CustomerAction.asp

You can customize the generated HTML and script in these files as desired.

To view the completed form

- In **FileView,** right-click **CustomerForm.asp**, and then choose **Preview in Browser**.



If you click the **List View** button, you jump to a page displayed in list view. You can view all the records in an HTML table and return to any record in Data Form view. The **Filter** button lets you filter records based on any available field in the form. For example, insurance agents could filter records to view customers in a certain state or with a certain set of insurance policies.

Lesson 7 - Working with Client-Side Components

Visual InterDev provides all the tools you need to build client-side functionality into your Web pages. Both the HTML Source Editor and the FrontPage Editor for Visual InterDev make it easy to insert ActiveX Controls, Java™ applets, and even Netscape plug-ins, while the Script Wizard helps you build client-side scripts in either VBScript or JScript.

In this lesson, you'll use Visual InterDev to create an interactive page that lets Friendship Insurance agents view their appointment schedules via the Web. This scheduler application uses an ActiveX Calendar Control. You'll also use the Script Wizard to add client-side script, which will update the appointments displayed on the page each time the Friendship agent selects a new date using the calendar.

Adding an ActiveX Control

135 In **FileView**, open the **Scheduler** folder, and then double-click **Schedule.htm** to open the file using the default HTML Source Editor.

136 In the Source Editor window, position the pointer on the line below the following HTML comment line:

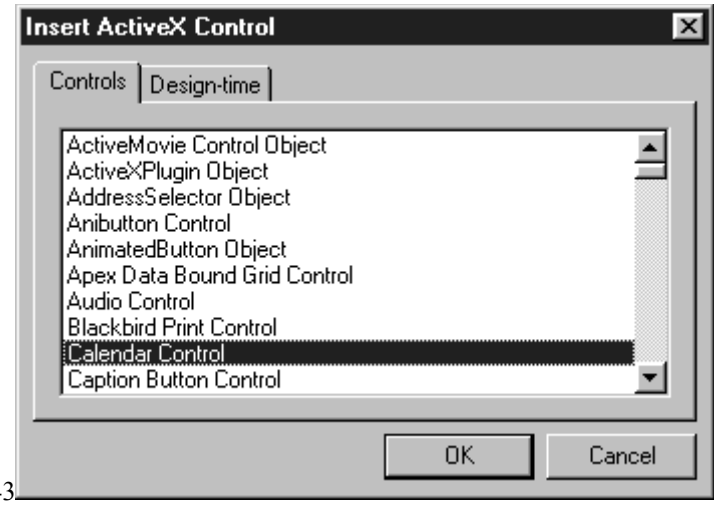
```
<!---Insert Calendar Control Here→
```

35

137 Right-click to display the shortcut menu, and then click **Insert ActiveX Control**.

138 In the list of available ActiveX Controls on the **Controls** tab, click **Calendar Control** and then click **OK**. (If the **Calendar** control isn't on the list, verify that you've followed the instructions in Lesson 1, "Getting Started," to install this developer-licensed control).

36



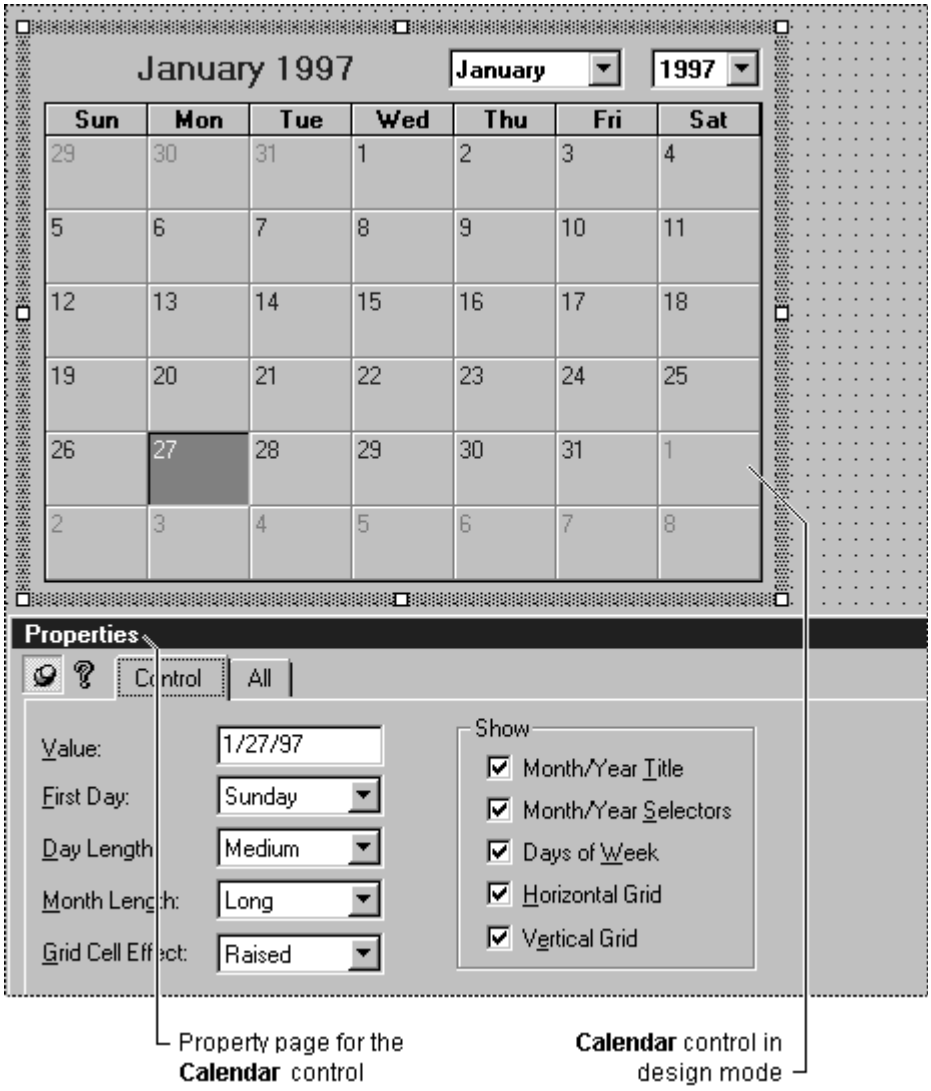
43

37

31The Object Editor opens, letting you specify the control's properties.

38

44



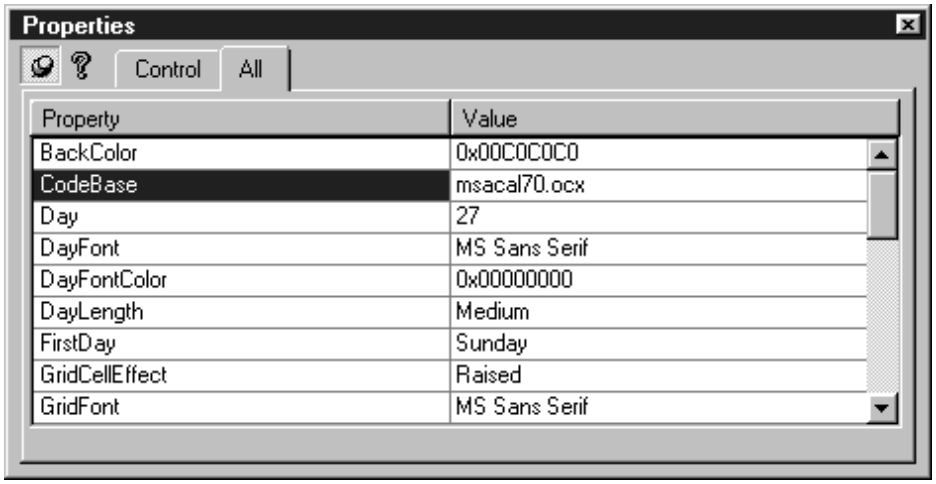
39

139In the **Properties** dialog box, click the **All** tab.

140In the **Value** box for the **CodeBase** property, type “msacal70.ocx.” (This tells the user’s browser where to download the **Calendar** control if not already installed. Microsoft Internet Explorer will install the control automatically as required.)

40

45



41

141To close the Object Editor for the **Calendar** control, double-click the **Close** button at the upper-right corner of the window.

32Visual InterDev automatically generates the HTML necessary to display the **Calendar** control on the Web page.

42

Note If you preview the page before completing the next section, "Using the Script Wizard," clicking the **Calendar** control will not update the schedule information.

7

46

```
content="Microsoft FrontPage (Visual InterDev Edition) 2.0">
<title>Schedule</title>
</HEAD>
<BODY bgcolor="#FFFFFF" link="#0000FF" vlink="#800080">
<BR>
<p align="left"><font color="#0000FF" size="5"><b>Friendship
Appointment Tracker</b></font> <br>
<img src="..\logo.gif"><br>
<br>
Click on a date to view your
appointments. <br>
</p>
<p>

<!---Insert Calendar Control Here -->

<OBJECT ID="Calendar1" WIDTH=372 HEIGHT=279
 CLASSID="CLSID:8E27C92B-1264-101C-8A2F-040224009C02"
 CODEBASE="msacal7.ocx">
    <PARAM NAME="_Version" VALUE="458752">
    <PARAM NAME="_ExtentX" VALUE="9843">
    <PARAM NAME="_ExtentY" VALUE="7382">
    <PARAM NAME="_StockProps" VALUE="1">
    <PARAM NAME="BackColor" VALUE="12632256">
    <PARAM NAME="Year" VALUE="1996">
    <PARAM NAME="Month" VALUE="12">
    <PARAM NAME="Day" VALUE="4">
</OBJECT>
<br>
</p>
    <FORM ACTION="main.asp" METHOD="POST" NAME="FORM1" target="main">
<P><br>

    <b>Date Selected:</b>
        <INPUT TYPE=text SIZE=20 NAME="Date">
</p>
```

HTML source generated by Visual InterDev to display the **Calendar** control

43

Using the Script Wizard

The Script Wizard lets you easily wire together different client-side elements using either VBScript or JScript.

The Script Wizard provides a point-and-click interface for generating client-side script. The Event pane lists all of the scriptable objects currently embedded on an HTML or ASP page, or on an HTML Layout. You associate objects in the Event pane with methods and properties of other objects listed in the Action pane. As you make these associations, the Script Wizard generates the actual script in the Script pane. You can also type custom scripting into the Script pane, or directly modify the generated source.

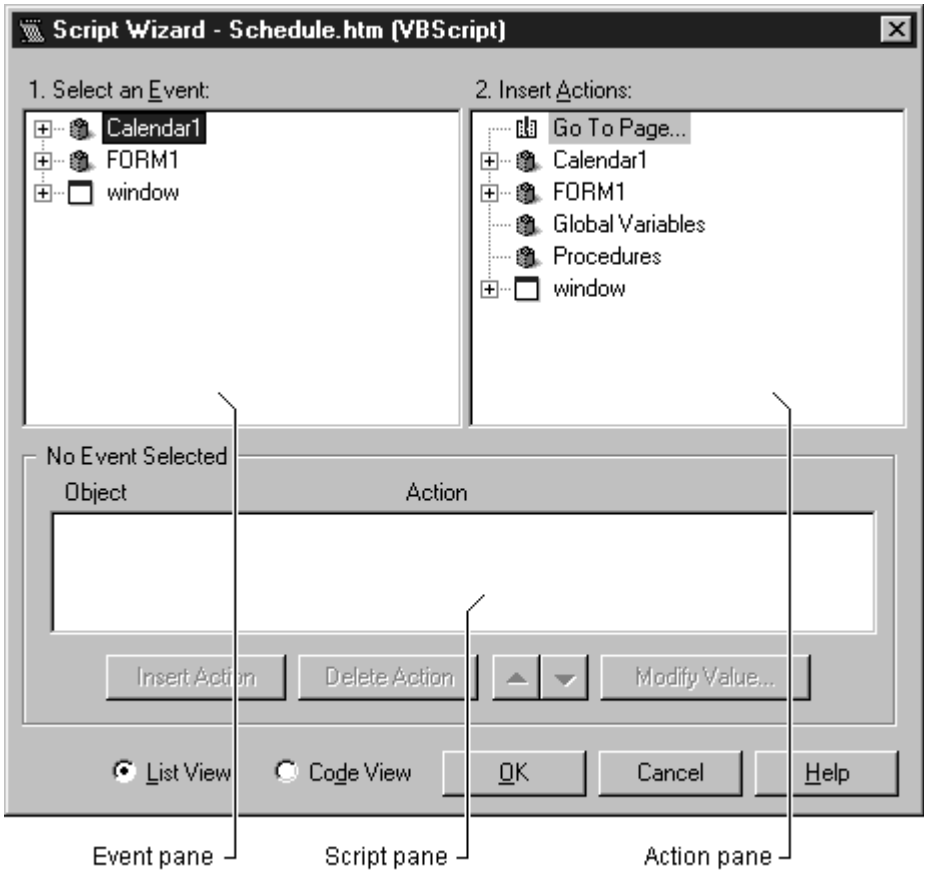
In this section, you'll use the Script Wizard to add client-side script to the appointments page that will update Appointments list each time the Friendship agent clicks a new date on the calendar.

To add script to the Appointments page

142 If you're continuing from the previous section, right-click in the HTML Source Editor window for **Schedule.htm**, and then click **Script Wizard** on the shortcut menu.

44

47



The Script Wizard in List View

45

143 Click **Code View** to change the Script Wizard to Code view.

33 This lets you type or edit script in the Script pane.

144 In the Event pane, click the plus (+) sign next to **Calendar1**.

34 This displays all of the scriptable events for the **Calendar** control.

145In the Action pane, click **AfterUpdate**.

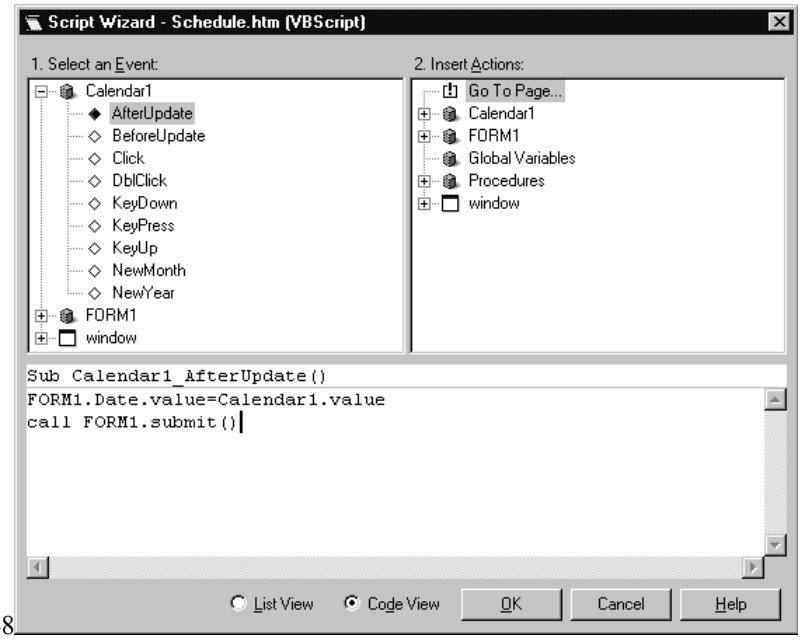
35The **AfterUpdate** event "fires" each time the Friendship agent clicks a new date on the calendar.

146In the Script pane, type the following lines of VBScript to execute on the event:

46

```
FORM1.Date.value=Calendar1.value
Call FORM1.submit()
```

47
48



48

49

36This script updates the HTML **Form1** Date field each time the Friendship agent clicks a new date on the calendar, and then submits the form to the server. The server will then display the appropriate appointment information for the selected day.

To close the Script Wizard and update the form

147Click **OK**.

37The Script Wizard writes the script back to the HTML page, ASP page, or HTML Layout.

148On the **File** menu, click **Save**.

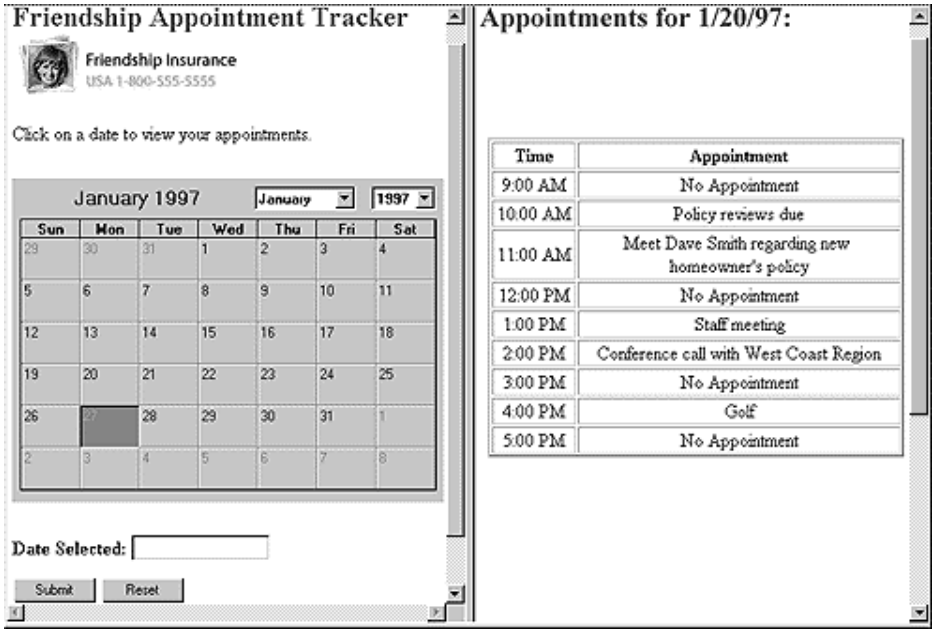
149 Close the HTML Source Editor window for Schedule.htm.

150In **FileView**, right-click **Mainframe.htm**, and then click **Preview in Browser** on the shortcut menu.

38The page uses HTML frames, with **Schedule.htm** displayed in the left frame.

50

49



51

39Try clicking dates on the calendar. Each time you click a date, the client-side script submits the information to the server, which returns new appointment information based on server-side scripting. The server-side scripting runs in the Main.asp page, which is the page processed on the Submit action of the HTML form page, Schedule.htm.

151Click **Return to Home Page** to go back to the Friendship Insurance home page, which contains links to the application elements you've constructed so far.

152Click **Analyze Policy Loss Ratios**.

40You'll notice that this link is broken—the browser can't find the page to display. In the next lesson, you'll use Visual InterDev's site management features to track down and fix this broken link.

52

Lesson 8 - Managing Your Site

Visual InterDev provides integrated site management. You can easily restructure your Web site by dragging and dropping elements, or by adding or modifying folders, in the **FileView** tab. As you rename, move, or delete elements, Visual InterDev can automatically track and update the related hyperlinks. To enable automatic link repair, on the **Tools** menu, click **Options**, and then on the **Web Projects** tab, specify a **Link Repair** option.

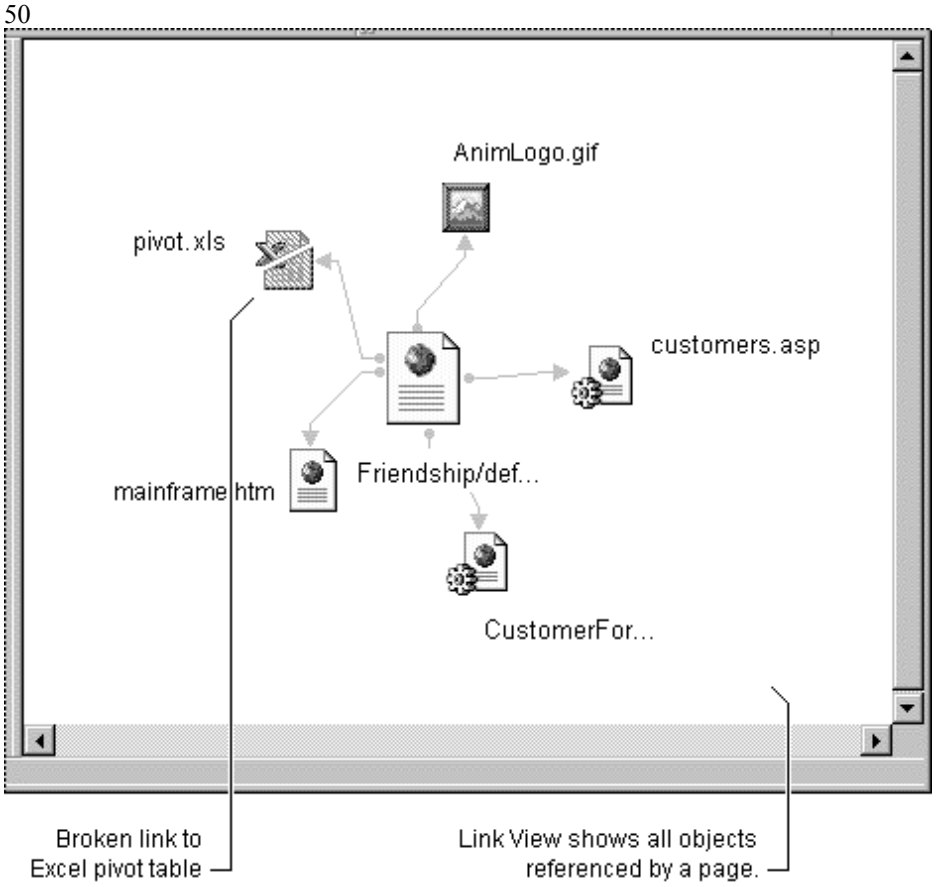
With Link View in Visual InterDev, you can graphically view the hyperlink relationships for any page in your project. Link View shows all of the elements linked to a given page, including other Web pages, images, multimedia files, other document or files, even Java applets and ActiveX controls.

Link View can boost your productivity by quickly depicting the structure of your Web project, and you can filter Link View to show specific file types.

To open Link View in Visual InterDev

153 In **FileView**, right-click **Default.htm**, and then click **View Links** on the shortcut menu.

50



54

41 You'll see a broken link—depicted in red—to the **Pivot.xls**. To fix this link, you'll use drag-and-drop publishing to publish the file to the Web site.

154 In Windows Explorer, open the folder **GuidedTour****Spreadsheet**.

155 Drag **Pivot.xls** from Windows Explorer to the root level of your Web project on the Visual InterDev **FileView** tab.

42 This publishes the spreadsheet over HTTP to your Web server.

156 In the **LinkView** window for **Default.htm,** click anywhere to select this as the active window.

157 On the **View** menu, choose **Refresh**.

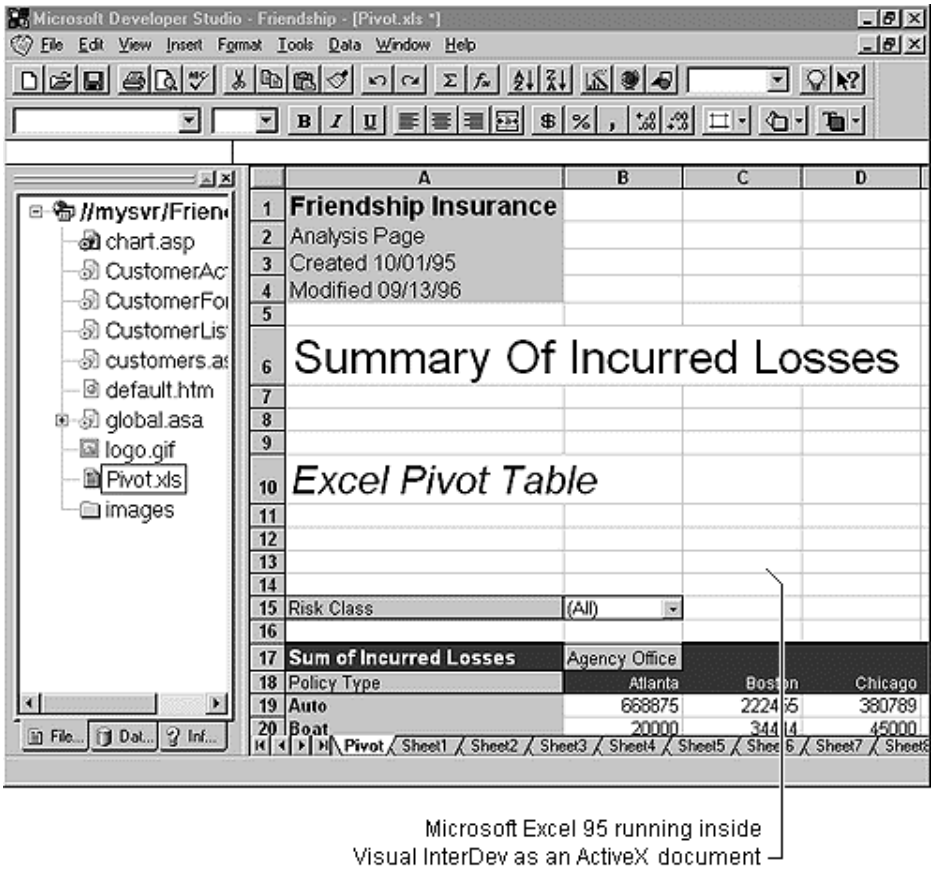
43 The link is now repaired, and is no longer displayed in red.

Previewing and Editing Files in Link View

You can preview or edit any file directly in Link View. For example, to edit the Excel spreadsheet whose link you just repaired, double-click the **Pivot.xls** icon (note that you must have Microsoft Excel installed on your computer to do this). Microsoft Excel launches and opens **Pivot.xls** directly inside the Visual InterDev IDE.

Visual InterDev lets you work on the Microsoft Excel spreadsheet (or any other ActiveX document, such as a Microsoft Word document, a Microsoft PowerPoint presentation, a Visio diagram, and so on) without leaving Visual InterDev. As you change the document, Visual InterDev automatically manages the changes on your Web server and updates the original source file so you don't need to.

55



56

You can also use Link View to plot any site on the Internet or on your corporate intranet. To do so, on the **Tools** menu, click **View Links on WWW**.

Before going on to the next lesson, close the Microsoft Excel window for **Pivot.xls**, and then close the Link View window for **Default.htm**.

Lesson 9 - Working with the Database Designer

The Visual InterDev Database Designer makes it easy to design, set up, and administer Microsoft SQL Server 6.5 databases. In this lesson, you'll use the Database Designer to create a new database design for the Friendship Insurance SQL Server database.

Note This feature doesn't work with databases other than Microsoft SQL 6.5 databases.

In the past, setting up and managing client-server based database engines such as SQL Server, Oracle, and Sybase has been a fairly difficult and time-consuming task. The Database Designer brings the graphical ease-of-use features of Microsoft Access to the process of setting up and administering Microsoft SQL Server databases within the Visual InterDev development environment.

Creating A New Table

To create a new table with the Database Designer

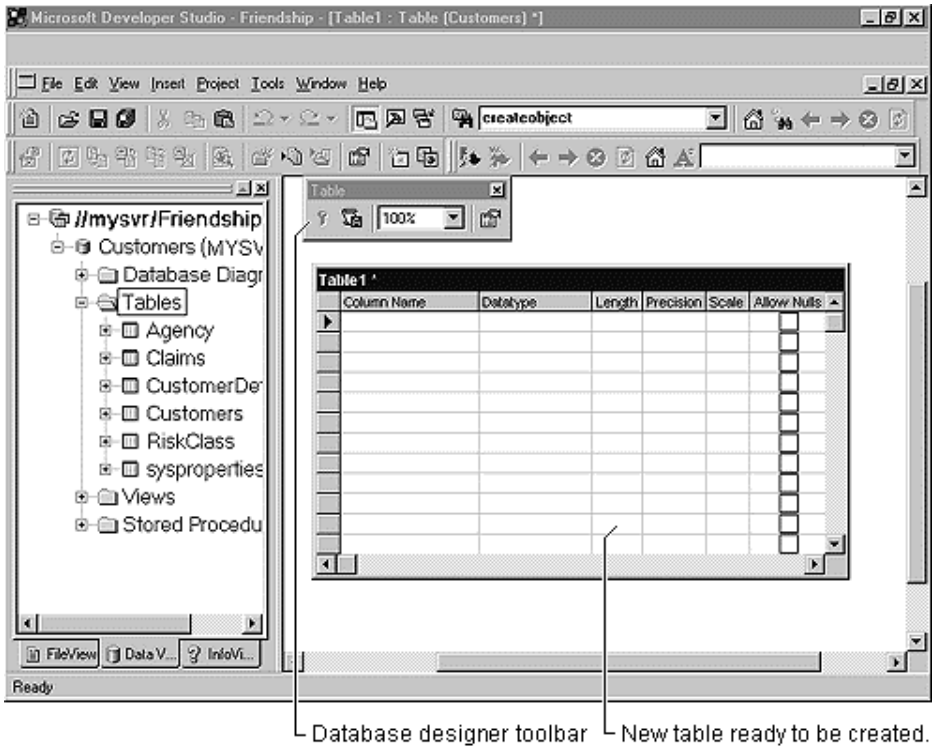
158In **DataView**, open the **Customers** database.

159Right-click the **Tables** folder and then choose **New Table**.

160In the **Choose Name** dialog box, name the table “MyTable.”

44A blank table database diagram opens in the Database Designer.

51



161 Position the cursor in the top left cell of the diagram and type “Column1.”

162 In the Data Type box, choose **int**.

163 Clear the **Allow Nulls** check box.

164 Click the row selector (the area to the left of the row) for the database column you want to define as the primary key.

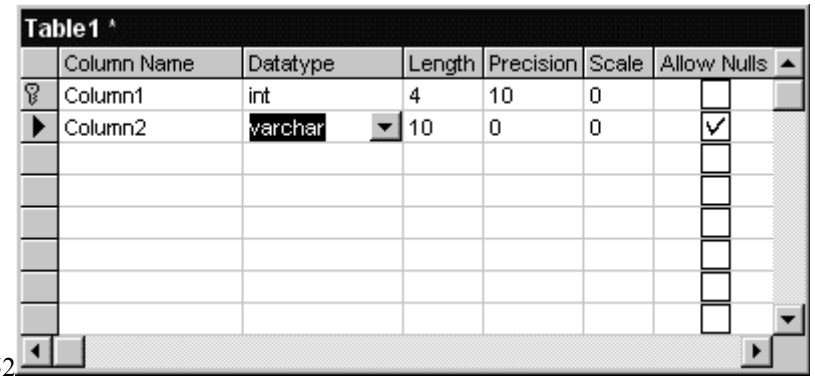
45 This action selects the entire row.

165 Click the **Set Primary Key** icon on the toolbar.

46 A key symbol appears in **Column1** to indicate that the field is a primary key.

166 Move to the second row of the diagram and type “Column2.”

167 In the **Datatype** field, choose **varchar**.

52 

168From the **File** menu, choose **Close**. Choose **Yes** when prompted to save changes.

47In **Data View**, the new table appears in the **Tables** folder.

To edit table properties

169In **Data View**, right-click the table name, and then click **Design**.

170In the **Data Type** field for **Column2**, choose **text** from the list box.

Ordinarily on a DBMS system such as SQL Server, making changes to the database would require many manual commands written in Data Definition Language (DDL). The Database Designer generates the DDL automatically as you work, and no changes are committed until you have completed making them. You can view the SQL change script to see the DDL generated.

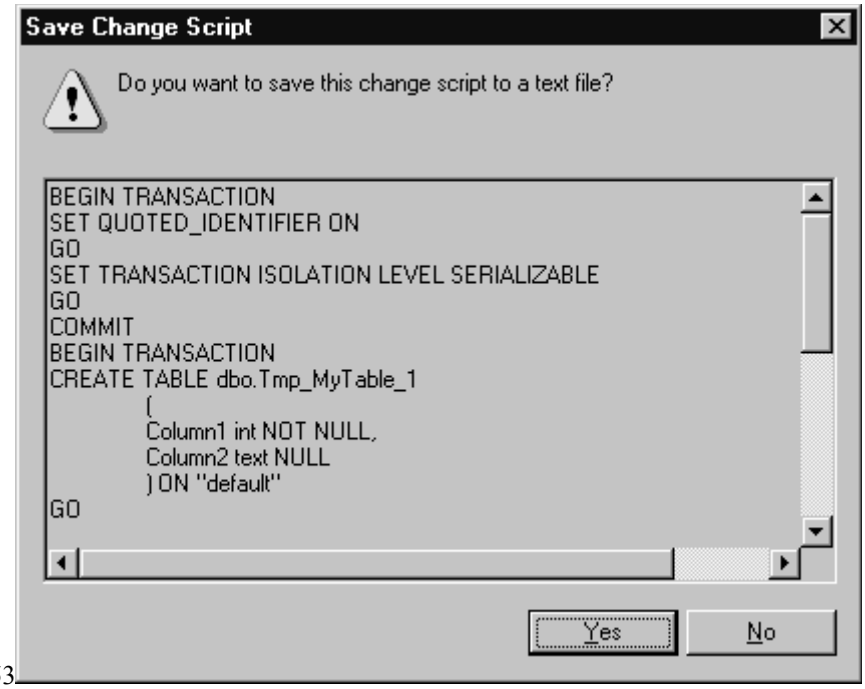
The Database Designer is a great tool for doing “what-if” scenarios on large databases. If developers don't have permission to execute the changes, they can save the change script to a text file (DDL Change Script) and then give the file to an authorized database administrator to review and execute.

To view and save the DDL Change Script

171On the Database Designer toolbar, click the **Save Change Script** button.

172Click **Yes** to save the script to a text file.

Note For this tutorial, you’ll want to update the database directly, so click **No** when prompted.

53 

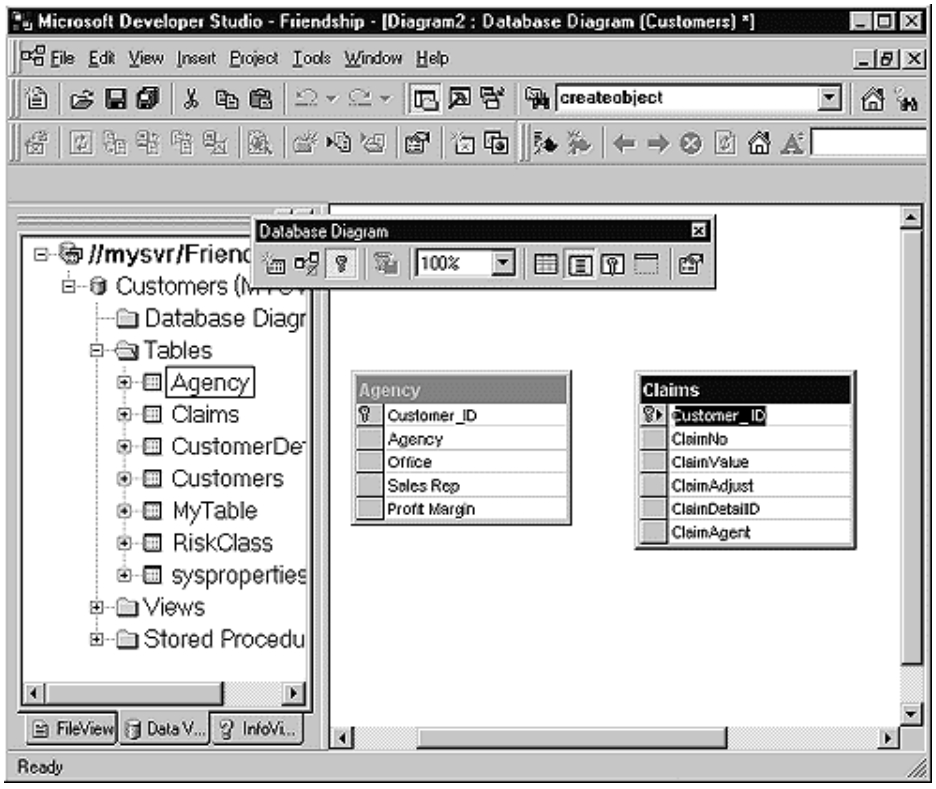
Creating a Database Diagram

Besides creating and designing tables, the Database Designer also enables you to create sophisticated database diagrams. For example, you can create a new diagram to establish relationships between tables.

To create a database diagram

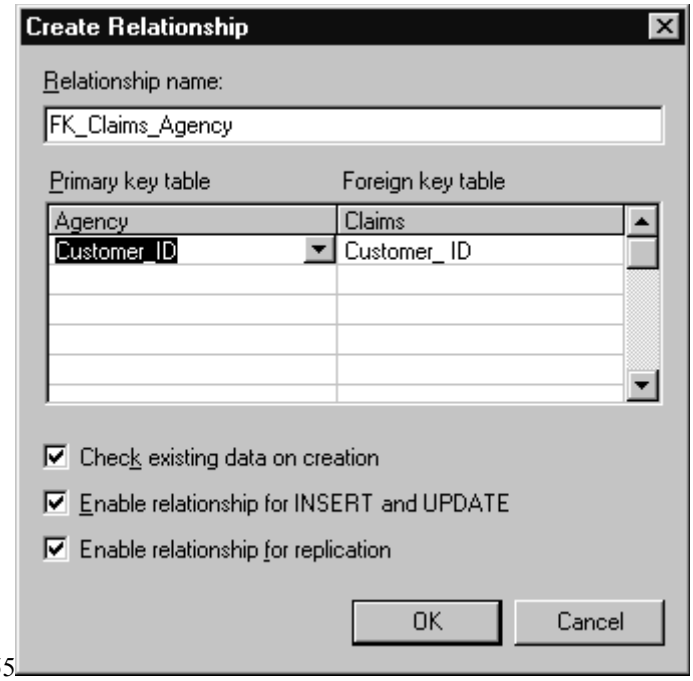
173 In **Data View**, right-click the **Database Diagrams** folder, and then choose **New Diagram**.

174 Drag the **Claims** and **Agency** tables into your database designer.



175 Select the **CustomerID** column in the **Agency** table and drag it onto the **CustomerID** column in the **Claims** table.

48 You'll be prompted for the type of relationship you want to create.

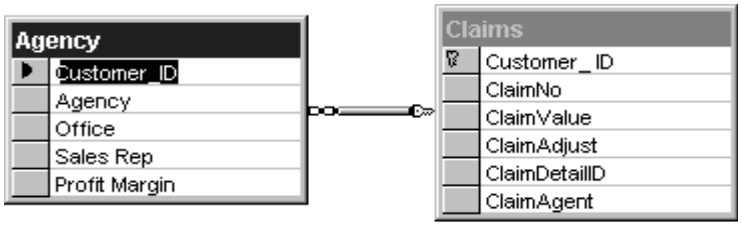


55

176 Click **OK** to keep the default values.

177 Close the diagram. When prompted, answer **Yes** to save your changes and name your diagram "MyDiagram."

178 Choose **Yes** again when prompted with the list of tables to be updated by your new diagram.



56 Notice how a join line appears between the two tables.

Notice that a join line appears between the two tables.

To edit the relationship, click the join line to select it, and then click **Properties** on the Relationship property page.

Lesson 10 - Learning More About Visual InterDev

In this tutorial, we've introduced you to some of the primary features of Visual InterDev, but we've really only scratched the surface. Below are a few suggestions for exploring Visual InterDev's rich feature set on your own.

Team-Based Development Support

Team-based development support means more than just providing a multi-user, distributed project model. It means letting developers and non-programmers work together on the same Web site, each using tools tailored to their specific needs. It also means protecting files from potential editing conflicts, and preserving revisions to roll-back unwanted changes.

Putting a Project Under Source Control

Visual InterDev provides file-locking capabilities and source control through integration with Microsoft® Visual SourceSafe™. To enable source control for your Web project, on the **Project** menu, click **Enable Web Source Control**. (Note that you must have Visual SourceSafe installed on your Web server, and you must have set up the appropriate user IDs using the Visual SourceSafe Administrator program.)

FrontPage Interoperability

Visual InterDev interoperates with Microsoft FrontPage 97. This lets programmers and non-programmers work on the same Web site at the same time.

If you have a Web project that was created in FrontPage 97, you can open the project in Visual InterDev, and vice-versa. FrontPage 97 also supports integration with Visual SourceSafe, so you can coordinate file access between users of both FrontPage 97 and Visual InterDev.

Even without Visual SourceSafe integration, multiple users can work simultaneously on files in shared projects. Once you've retrieved a local working copy of a file, Visual InterDev will detect and notify you of any changes to that file made by another user.

Moving a Web from Staging to Production

To copy a completed Web to a different server, or to copy a Web to a different virtual root name, on the **Project** menu, click **Copy Web**. The Copy Web command works over HTTP so that you can move content through firewalls.

Plotting Your Favorite Web Site in Link View

You can also use Link View to plot any site on the Internet or on your corporate intranet. To do so, on the **Tools** menu, click **View Links on WWW**, and then type the URL of the site you want to view.

Creating an HTML Layout

Visual InterDev lets you design HTML Layouts using the HTML Layout Editor. You can create more advanced page layouts using ActiveX Controls that are exactly positioned on a page. The layout information is saved in the .alx file for the HTML Layout. To create a new HTML Layout, on the **File** menu, click **New**, and then on the **Files** tab, click **HTML Layout.**

To access examples of HTML Layouts, visit the Microsoft Web site at http://www.microsoft.com/workshop/author/layout.

Adding Multimedia to Your Web Project

You can install the following multimedia tools from the Master Setup, Visual InterDev Client:

- Microsoft Image Composer
- Microsoft Media Manager
- Microsoft Music Producer

Using Microsoft Image Composer

Visual InterDev includes a full-featured image editor, MicrosoftImage Composer, which lets developers create and modify Web images. To learn more about Image Composer, install it, and then consult the Guided Tour in the Image Composer Reviewer's Guide. To use Image Composer to edit images in your Visual InterDev Web project, in **FileView**, right-click the image you want to edit, and then click **Open With** on the shortcut menu.

Using Microsoft Music Producer

Microsoft Music Producer makes it easy to create original musical effects for your Web site. To learn more about Music Producer, install it and consult the online documentation.

Using Microsoft Media Manager

Microsoft Media Manager simplifies managing your multimedia files by letting you define "content libraries" and by providing thumbnail views of images and other content. You can drag content from Media Manager directly to the **FileView** tab of your Visual InterDev Web project. To learn more about Media Manager, install it and consult the online documentation.

Using Multiple Projects in a Single Workspace

A workspace is a container for one or more projects. You can add more multiple projects to a single workspace, even if they are on different Web servers. Being in a single workspace means that you can drag and drop files from one project into another project. You can have Microsoft® Visual J++ and Microsoft® Visual C++ projects in the same workspace as Visual InterDev projects.