



# CODECHARGE STUDIO 1.0



# CodeCharge Studio Manual

---

© YesSoftware, Inc.  
2226 A. Westborough Blvd. #405  
South San Francisco, CA 94080  
Phone (650) 754-9810 • (415) 820-7566

**[www.codecharge.com](http://www.codecharge.com)**

## **NOTICE**

**This documentation should be considered a work in progress and will continue to change as CodeCharge Studio evolves to the final version. Some of the instructions, options and screenshots are bound to differ from the actual product as newer beta versions are released.**

# Table of Contents

<b>I.</b>	<b>Introduction .....</b>	<b>7</b>
	Welcome .....	7
	Features .....	7
	IDE (Integrated Development Environment) .....	7
	Application Design .....	7
	Code Generation .....	7
	System and Server Requirements .....	8
	System Requirements .....	8
	Server Requirements .....	8
<b>II.</b>	<b>Getting Started .....</b>	<b>9</b>
	Downloading CodeCharge Studio .....	9
	Installing, Upgrading and Repairing .....	9
	Installing CodeCharge Studio .....	9
	Upgrading, Repairing or Uninstalling CodeCharge Studio .....	13
	Setting up the work environment .....	16
	ASP Setup .....	16
	C# Setup .....	17
	Cold Fusion Setup .....	17
	JSP Setup .....	18
	PHP Setup .....	18
	Perl .....	19
<b>III.</b>	<b>Using CodeCharge Studio .....</b>	<b>20</b>
	The CodeCharge Studio Workspace .....	20
	The Command Area .....	22
	The Project Explorer .....	27
	The Toolbox .....	28
	The Properties Window .....	29
	Customizing the Workspace .....	31
	Project Settings .....	37
	Project .....	37
	Publishing .....	38
	Connections .....	39
	Security .....	40
	Security Groups .....	41
<b>IV.</b>	<b>Databases and Data Sources .....</b>	<b>43</b>
	Connections .....	44
	Creating Connections .....	44
	Design Connection Settings .....	45
	Server Connection Settings .....	46
	Creating a Connection String (DNS-Less Connection) .....	47
	Creating an ODBC Data Source Name (DSN) .....	50
	Form Data Sources .....	53
	Using the Select Query Window .....	61
	Using the Table Parameter Window .....	64
	Using the Stored Procedure Parameter Window .....	66
	Using the SQL Parameter Window .....	67
<b>V.</b>	<b>Controls and HTML Elements .....</b>	<b>69</b>
	HTML Elements .....	69
	Text Box .....	69
	Password .....	69
	Text Area .....	70
	Checkbox .....	70

Radio Button .....	70
Dropdown .....	70
Listbox .....	70
Submit Button .....	71
Reset Button .....	71
Label .....	71
Image Button .....	71
Button .....	72
Horizontal Rule .....	72
Line Break .....	72
Paragraph Break .....	73
Space .....	73
Image .....	73
Hyperlink .....	74
Bookmark .....	74
ImageLink .....	75
Controls .....	76
Label .....	76
Link .....	76
TextBox .....	76
Hidden .....	76
TextArea .....	77
Checkbox .....	77
RadioButton .....	77
ListBox .....	77
Image .....	77
ImageLink .....	78
Include Page .....	78
Button .....	78
<b>VI. Forms .....</b>	<b>79</b>
Grid Form .....	79
Record Form .....	80
Search Form .....	80
Login (Authentication) Form .....	81
Using the Wizards to create forms .....	81
Power Wizard .....	82
Page Wizard .....	98
Grid Form Wizard .....	99
Record Wizard .....	107
Search Wizard .....	110
Authentication Wizard .....	114
Grid and Record Wizard .....	117
<b>VII. Object Properties .....</b>	<b>126</b>
Project Properties .....	126
Form Properties .....	127
Grid Form Properties .....	127
Record, Search and Login Form Properties .....	127
Control Properties .....	128
HTML Properties .....	132
Connection Properties .....	132
<b>VIII. Working with Documents .....</b>	<b>134</b>
Design Mode .....	134
HTML Mode .....	135
Code Mode .....	136
Preview Mode .....	137
Live Page .....	137
Using the Front Page Add-In .....	137

<b>IX.</b>	<b>Client Side Event Actions .....</b>	<b>139</b>
	Set Focus .....	139
	Validate Maximum Length .....	140
	Validate Minimum Length .....	140
	Validate Maximum Value .....	140
	Validate Minimum Value .....	140
	Validate Required Value .....	141
	Regular Expression Validation .....	141
	Validate Entry .....	141
	Validate Form .....	142

# I. Introduction

## **Welcome**

CodeCharge Studio is a powerful Rapid Application Development (RAD) tool used to design, develop and deploy database driven web applications in six different languages (ASP/VBScript, .Net/C#, Cold Fusion, JSP, PHP and Perl) and running off a score of different databases. Time is money and this is particularly true in the fast paced WWW environment. CodeCharge Studio is squarely aimed at saving the developer a considerable amount of time and money by generating robust, extendable and highly configurable code that would otherwise require a significant investment in time and effort if it were to be written from scratch.

With its ability to generate web applications in six different languages, CodeCharge Studio doubles up as a learning tool that can be used to learn about web application development and make quick inroads into the field by learning from code that actually works and is used in real life applications.

## **Features**

### **IDE (Integrated Development Environment)**

- Powerful, professional IDE
- Built-in web editor
- Full integration with Microsoft FrontPage
- Built-in powerful code editor with color code highlighting
- Fast, on-the-fly, two-way code generation
- Extreme extensibility: User interface and components created in an open XML format making them easy to customize and expand

### **Application Design**

- Design web applications using a simple point-and-click process
- Use Wizards to create Data Entry Forms and Grids/Tables
- Design and apply custom Themes to components on the page
- Define Login access to each page by specifying a minimum Security Level or Group
- Define multiple database connections per application
- Include pages as components within other pages
- Create custom Wizards in JavaScript and XML
- “Hook” into generated code by creating Events and Actions
- Create and share custom Actions, which automatically place programming code within Events
- For more flexibility, manually modify generated code, which is fully preserved during subsequent code generations!

### **Code Generation**

- Generates object-oriented applications in ASP, .NET/C#, Java, PHP, Perl and Cold Fusion
- Generated code works with almost all databases via JET, ODBC, ADO, DBI and PHPLib
- Generated code is Windows and Linux compatible

- Helps beginners quickly get familiar with .NET and other web technologies, while rapidly building web applications
- Helps intermediate and advanced developers focus on the system architecture and design, while eliminating tedious coding

## ***System and Server Requirements***

### **System Requirements**

The following are the minimal requirements needed for a system to run the CodeCharge Studio application

- Intel Pentium Processor or equivalent
- 64 MB available RAM
- 20 MB available disk space
- 256-color monitor capable of 800 x 600 resolution
- Windows 95, 98, ME, NT4, 2000 or XP
- Database application (MS Access, MS SQL, MySQL, etc.)
- Microsoft Internet Explorer 5.5 or higher with JavaScript enabled.

### **Server Requirements**

In order to be able to run the generated pages, you need a Web server on the Windows or Unix/Linux platform that supports one of the following languages:

- ASP 2.0 or 3.0
- ASP.Net/C#
- Cold Fusion 4.0 or higher
- JSP 1.1,
- PHP 3.0 / 4.0
- Perl 5.0 or higher

The server should also be capable of accessing the database required by the generated pages. If using a host server in a remote location, the server should have FTP support to enable uploading of the generated pages.



## II. Getting Started

### ***Downloading CodeCharge Studio***

You can acquire a copy of the CodeCharge Studio installation executable from the following address: (<http://www.CodeCharge.com/Studio>).

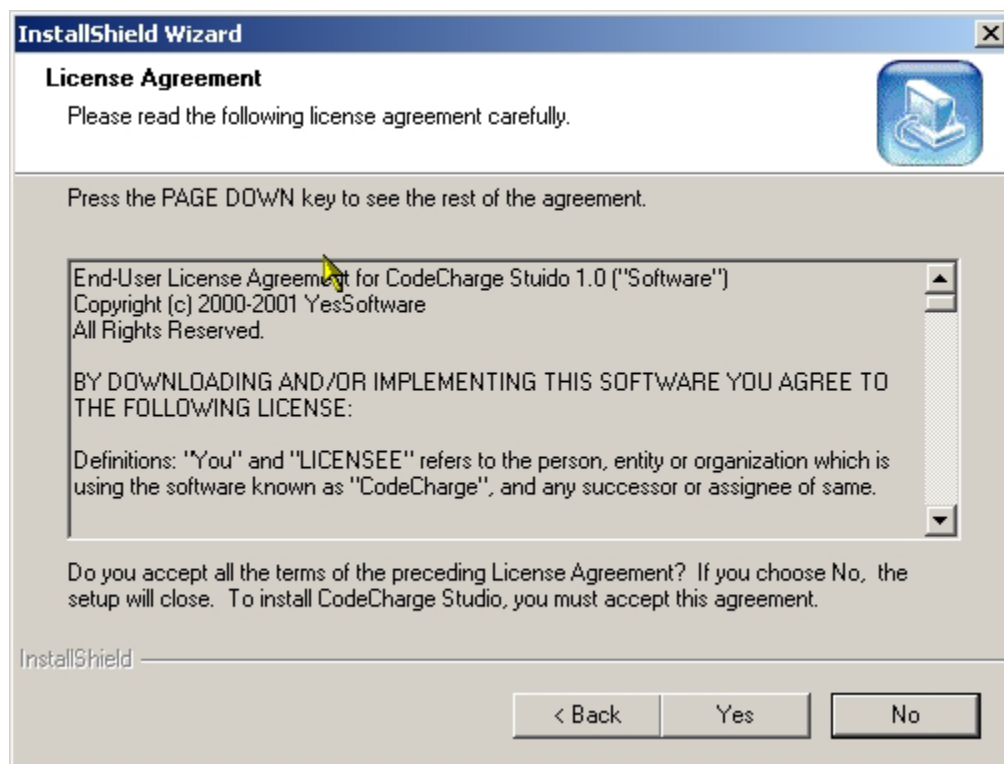
The installation file features a 30-day trial period that allows prospective users to test run the system before deciding on whether to make a purchase. Note that both licensed users and trial users download the same file that is fully functional but time-limited unless activated using a valid serial number.

### ***Installing, Upgrading and Repairing***

#### **Installing CodeCharge Studio**

Before beginning the installation process, you should ensure that your system meets the requirements stipulated above in the [System requirements](#) section.

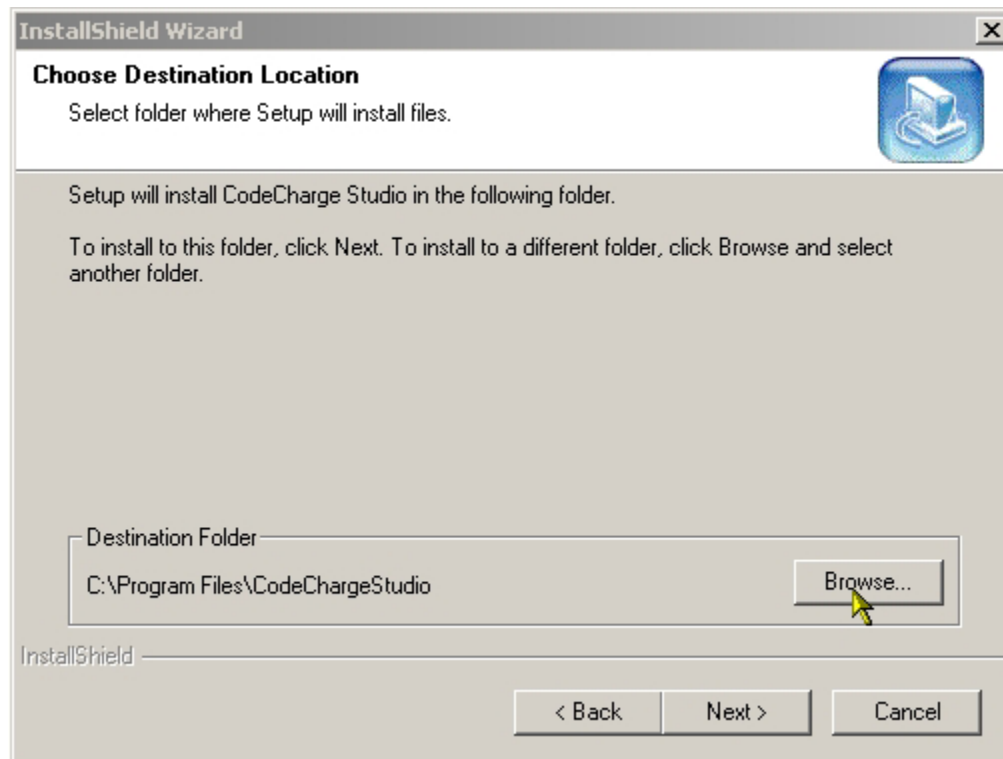
1. To begin the installation process, locate the CodeCharge Studio setup file and double-click on it. Before you can proceed with the installation process, you should read and agree to the License Agreement. Click on the **Yes** button to indicate your acceptance of the License Agreement and proceed to the next step.



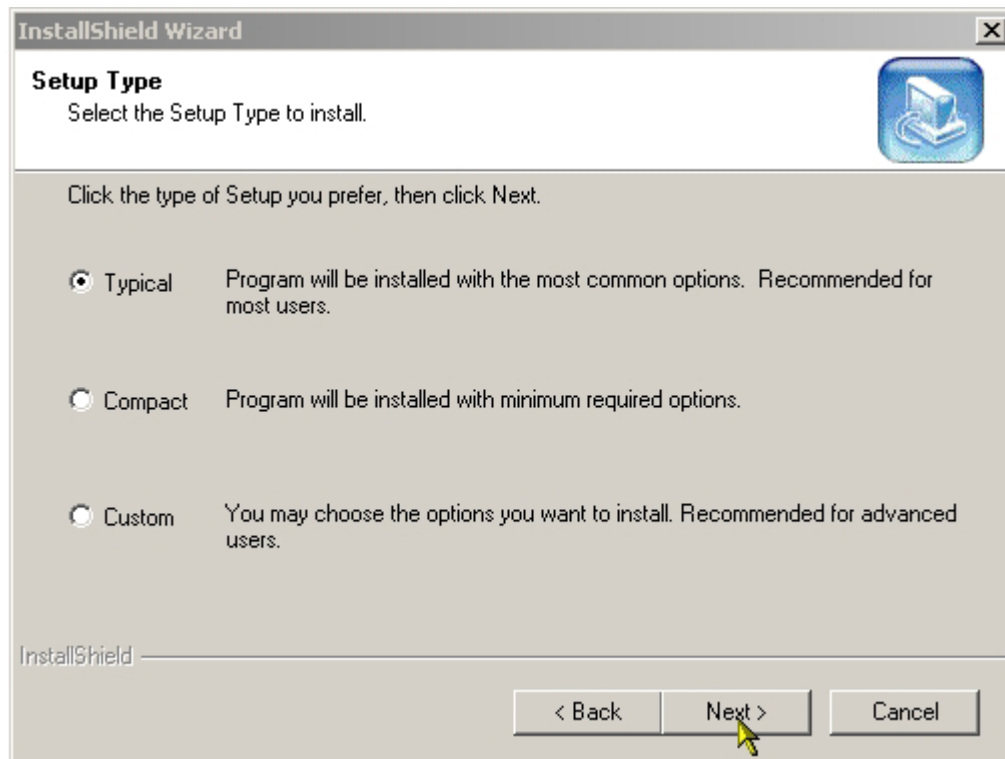
2. In the next window, you are presented with fields to enter your name, company name and Serial Number. If you don't have a serial number and want to make use of the 30 day trial period, simple enter "Trial" in the Serial Number field. Click **N**ext to proceed.

The screenshot shows a Windows-style dialog box titled "InstallShield Wizard". The window has a blue title bar with a close button (X) in the top right corner. Below the title bar, the text "Customer Information" is displayed in bold, followed by the instruction "Please enter your information." in a smaller font. To the right of this text is a blue icon depicting a computer monitor and a mouse. Below this section, a larger text area contains the instruction: "Please enter your name, the name of the company for whom you work and the product serial number." There are three text input fields: the first is labeled "User Name:" and contains the text "Joe Public"; the second is labeled "Company Name:" and contains the text "YesSoftware, Inc."; the third is labeled "Serial Number:" and contains the text "Trial". At the bottom left of the window, the text "InstallShield" is visible. At the bottom right, there are three buttons: "< Back", "Next >", and "Cancel". A yellow mouse cursor is pointing at the top of the text area.

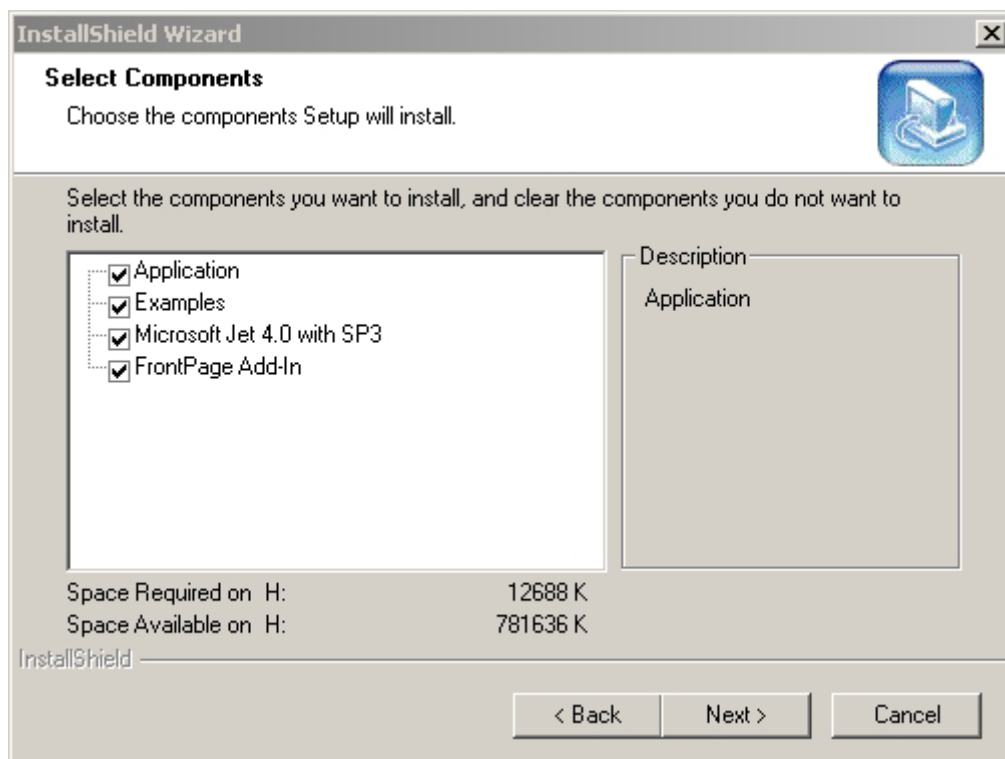
3. In the ensuing window, confirm or specify the destination folder where you want CodeCharge Studio to be installed. The default recommended path for installing CodeCharge Studio is C:\Program Files\CodeCharge Studio. However, if you wish to change this path, click **Browse** and specify an alternate path. Once done, click **Next** to proceed to the next step.



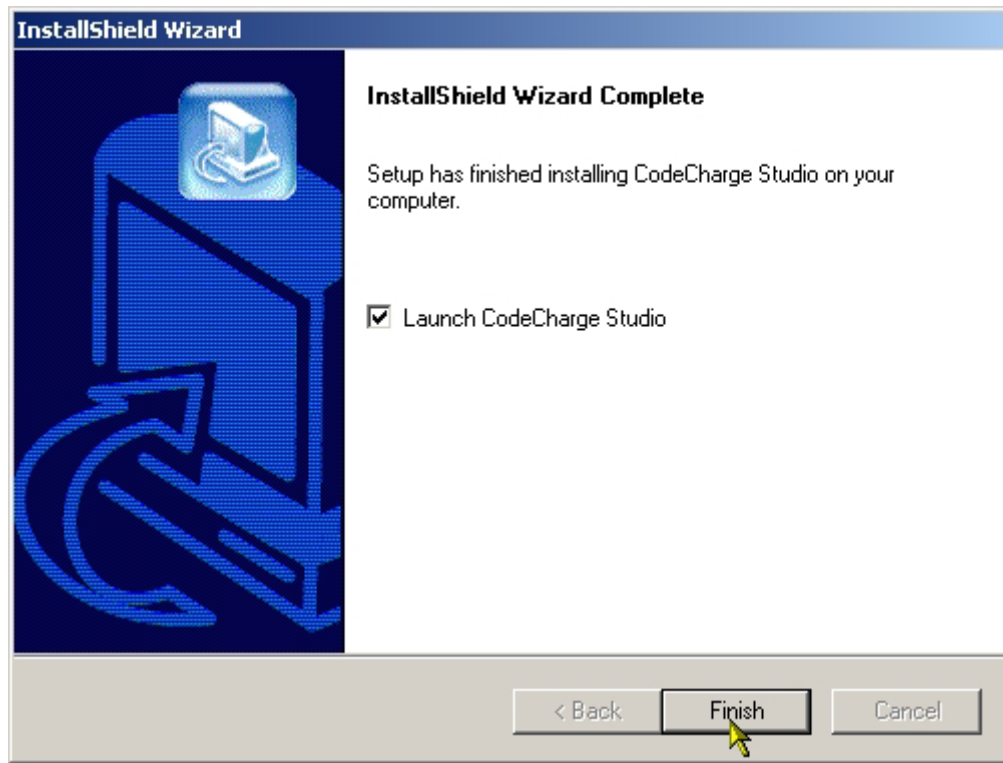
4. In the next window, you are presented with three setup options from which to choose. The Typical option installs all the CodeCharge options as shown in step 5 below. The Compact option installs the CodeCharge application files but does not install the examples or Microsoft Jet 4.0. Finally, the Custom option allows the user to select the components to be installed.



5. If you selected the Typical or Custom options in step 4 above, you are then presented with the following window that shows the components to be installed. You can then specify the components that you want installed.



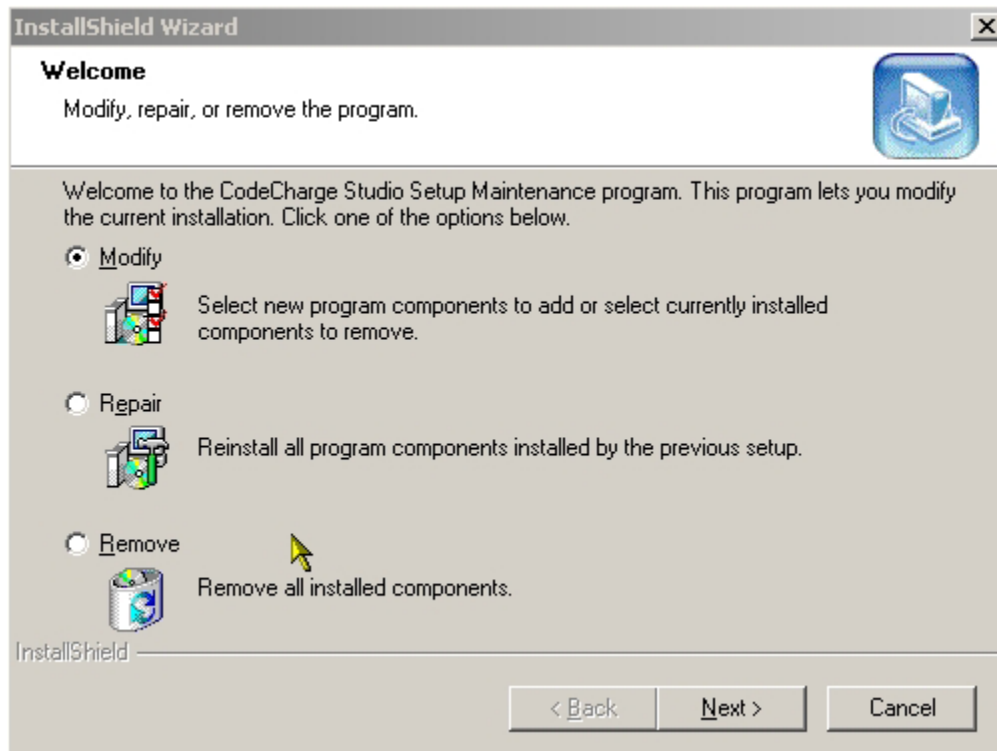
6. After all the necessary files have been installed and configured, you will then be presented with the following window to confirm the success of the installation. At this point, if you want to launch CodeCharge Studio, you can select the “Launch CodeCharge Studio” checkbox then click on the **Finish** button to complete the installation process.



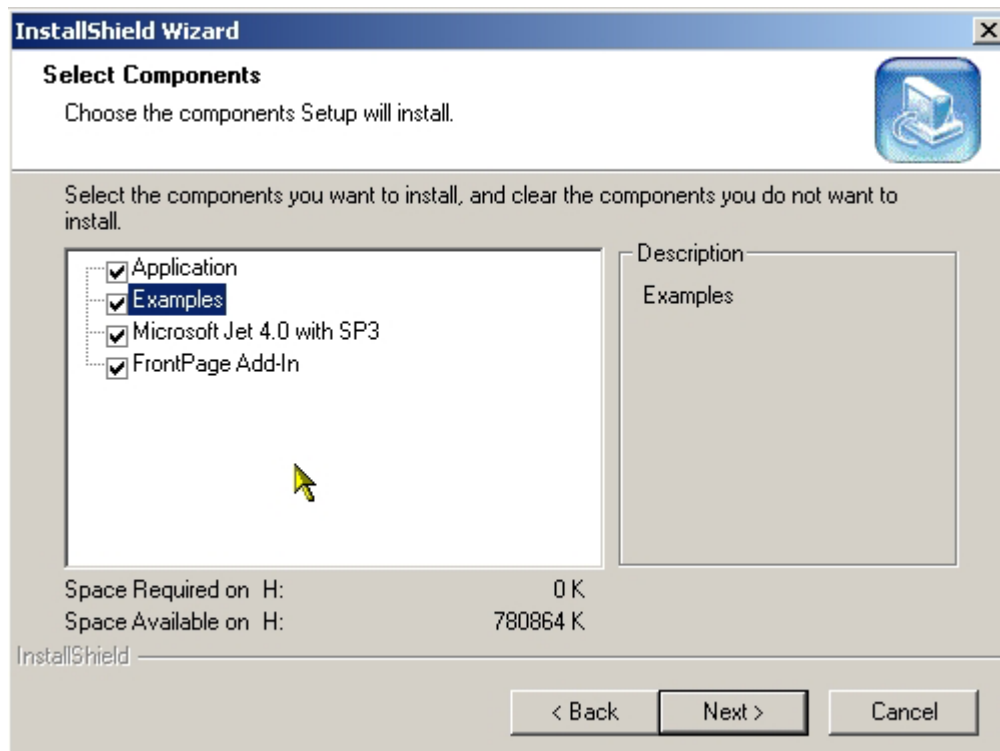
## Upgrading, Repairing or Uninstalling CodeCharge Studio

In the event that a current CodeCharge Studio installation gets corrupted or otherwise damaged, you can run the same setup file to modify or repair the installation or remove it entirely. To do so:

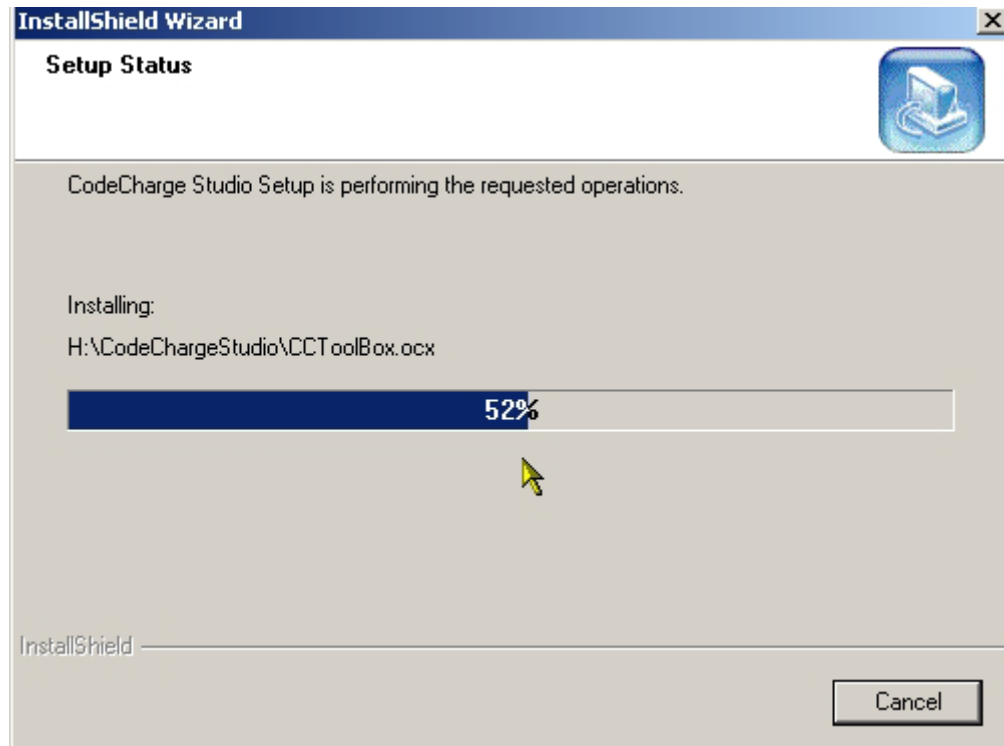
1. Double-click on the setup file to begin the process.
2. In the ensuing window, select whether you want to **Modify**, **Repair** or **Remove** the current CodeCharge installation then click the Next button. If you selected the Repair option, please jump to step 4. For the Remove option, please jump to Step 5.



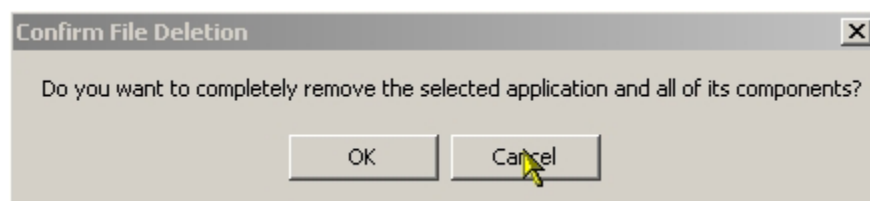
3. For the Modify option, you will be presented with a list of components that you can install or remove. To install a currently unavailable component, activate its respective checkbox. Similarly, to uninstall a component that is already installed, uncheck its checkbox. After making your selections, click the Next button to proceed.



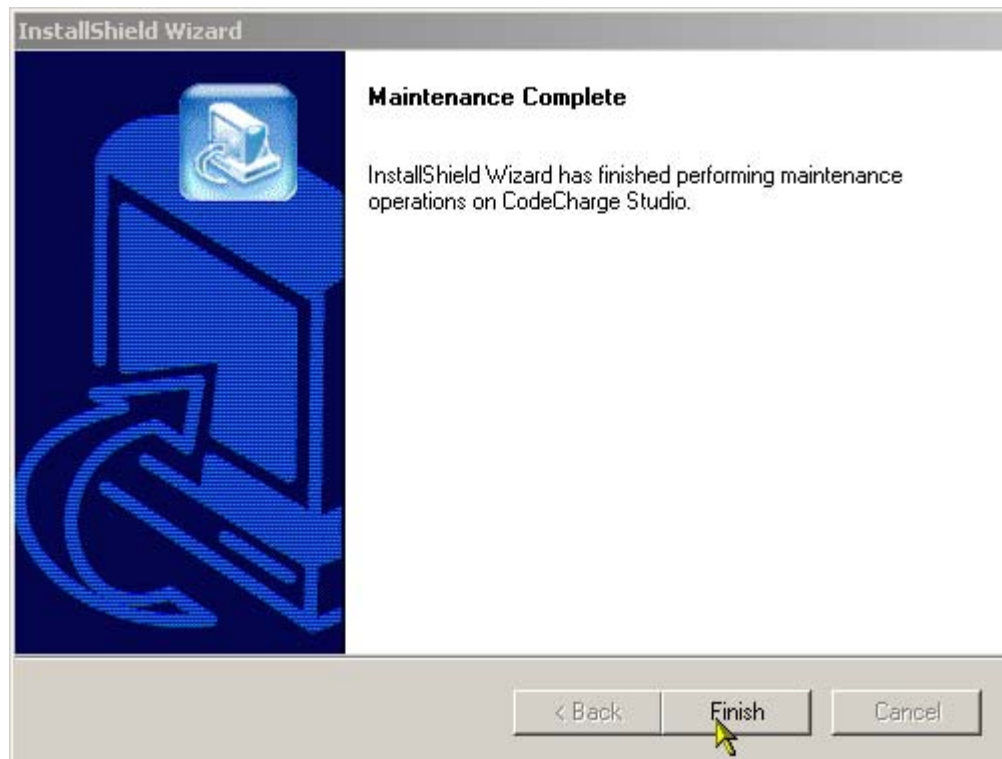
4. If you selected the Repair option, the wizard will now proceed to check the current installation and repair any problems that may be diagnosed while a dialog window shows the progress of the repair operation.



5. If you selected the Remove option, you will be presented with a dialog window asking you to confirm whether you want to proceed with the removal. At this point, you can either cancel and return to the main installation window or proceed with the un-installation.



6. Finally, when all the operations have been completed successfully, a window appears to confirm the completion of the maintenance process. Click on the Finish button to close the dialog box.



## ***Setting up the work environment***

Having setup CodeCharge Studio, you are almost set to start generating and viewing dynamic database driven web pages. However, as you may be well aware by now, CodeCharge Studio works in concert with database and server software so this should be put in place before proceeding to use CodeCharge. While it is possible to generate web pages that aren't based on database content, such an approach is more the exception than the norm and even then, a server would be required to view the pages.

Considering that there are six different languages for generating pages and upwards of a dozen possible databases, there are obviously many configurations that could be used. Each combination of language and database has its own unique merits. In the table below, we examine some of the more typical setups per language. Note that this is not to say that you cannot use setups beyond those stipulated here.

### **ASP Setup**

Operating System	<ul style="list-style-type: none"><li>• Windows 98</li><li>• Windows NT</li><li>• Windows 2000</li><li>• Windows XP</li><li>• Unix/Linux</li></ul>
Web Server	<ul style="list-style-type: none"><li>• Internet Information Server (IIS) 4.0 or 5.0</li><li>• Personal Web Server</li><li>• Apache with Chili!Soft ASP</li></ul>



	<ul style="list-style-type: none"> <li>• iPlanet with Chili!Soft ASP</li> <li>• Zeus with Chili!Soft ASP</li> </ul>
Database Server	<ul style="list-style-type: none"> <li>• Microsoft Access</li> <li>• MS SQL Server</li> <li>• Oracle</li> <li>• MySQL</li> <li>• PostgreSQL</li> <li>• DB2</li> <li>• FoxPro</li> <li>• Excel</li> <li>• Flatfile, CSV, textfile</li> </ul>

## C# Setup

Operating System	<ul style="list-style-type: none"> <li>• Windows 2000</li> <li>• Windows XP</li> <li>• Window .Net Server</li> </ul>
Web Server	Requires installation of the ASP.Net Framework <ul style="list-style-type: none"> <li>• Internet Information Server (IIS) 5.0</li> </ul>
Database Server	<ul style="list-style-type: none"> <li>• Microsoft Access</li> <li>• MS SQL Server</li> <li>• Oracle</li> <li>• MySQL</li> <li>• PostgreSQL</li> <li>• DB2</li> <li>• FoxPro</li> <li>• Excel</li> <li>• Flatfile, CSV, textfile</li> </ul>

## Cold Fusion Setup

Operating System	<ul style="list-style-type: none"> <li>• Windows 98</li> <li>• Windows NT</li> <li>• Windows 2000</li> <li>• Windows XP</li> <li>• Unix/Linux</li> </ul>
Web Server	All servers must have the Cold Fusion server installed. <ul style="list-style-type: none"> <li>• Internet Information Server (IIS) 4.0 or 5.0 with Cold Fusion</li> <li>• Personal Web Server with Cold Fusion</li> <li>• Apache server with Cold Fusion</li> </ul>
Database Server	When using Cold Fusion, the design time connection string has to be made using ODBC.

	<ul style="list-style-type: none"> <li>• Microsoft Access</li> <li>• MS SQL Server</li> <li>• Oracle</li> <li>• MySQL</li> <li>• PostgreSQL</li> <li>• DB2</li> <li>• FoxPro</li> <li>• Excel</li> <li>• Flatfile, CSV, textfile</li> </ul>
--	---

## JSP Setup

Operating System	<ul style="list-style-type: none"> <li>• Windows 98</li> <li>• Windows NT</li> <li>• Windows 2000</li> <li>• Windows XP</li> <li>• Unix/Linux</li> </ul>
Web Server	<ul style="list-style-type: none"> <li>• Internet Information Server (IIS) 4.0 or 5.0</li> <li>• Personal Web Server</li> <li>• Apache</li> </ul> <p>In addition to the web server itself, you need to have a JSP server such as:</p> <ul style="list-style-type: none"> <li>• Tomcat</li> <li>• IBM WebSphere</li> <li>• Macromedia JRun</li> </ul>
Database Server	<ul style="list-style-type: none"> <li>• Microsoft Access</li> <li>• MS SQL Server</li> <li>• Oracle</li> <li>• MySQL</li> <li>• PostgreSQL</li> <li>• DB2</li> <li>• FoxPro</li> <li>• Excel</li> <li>• Flatfile, CSV, textfile</li> </ul>

## PHP Setup

Operating System	<ul style="list-style-type: none"> <li>• Windows 98</li> <li>• Windows NT</li> <li>• Windows 2000</li> <li>• Windows XP</li> <li>• Unix/Linux</li> </ul>
Web Server	All the web servers listed here have to have the PHP engine installed.

	<ul style="list-style-type: none"> <li>• Internet Information Server (IIS) 4.0 or 5.0</li> <li>• Personal Web Server</li> <li>• Apache</li> </ul>
Database Server	<ul style="list-style-type: none"> <li>• Microsoft Access</li> <li>• MS SQL Server</li> <li>• Oracle</li> <li>• MySQL</li> <li>• PostgreSQL</li> <li>• DB2</li> <li>• FoxPro</li> <li>• Excel</li> <li>• Flatfile, CSV, textfile</li> </ul>

## Perl

Operating System	<ul style="list-style-type: none"> <li>• Windows 98</li> <li>• Windows NT</li> <li>• Windows 2000</li> <li>• Windows XP</li> <li>• Unix/Linux</li> </ul>
Web Server	<p>All servers must be Perl enabled with a version of Perl greater than or equal to 5.0. Some popular distributions of Perl include ActiveState Perl, Mod_Perl for Apache and Indigo Perl. Experienced users can also download and compile the Perl code.</p> <ul style="list-style-type: none"> <li>• Internet Information Server (IIS) 4.0 or 5.0</li> <li>• Personal Web Server</li> <li>• Apache server</li> </ul>
Database Server	<p>Perl requires installation of a suitable database interface (DBI) in order to connect to a database. This is a separate package that can be obtained from <a href="http://www.cpan.org">www.cpan.org</a>. e.g. For a MySQL database, the DBD:mysql is required.</p> <ul style="list-style-type: none"> <li>• Microsoft Access</li> <li>• MS SQL Server</li> <li>• Oracle</li> <li>• MySQL</li> <li>• PostgreSQL</li> <li>• DB2</li> <li>• FoxPro</li> <li>• Excel</li> <li>• Flatfile, CSV, textfile</li> </ul>

### III. Using CodeCharge Studio

#### *The CodeCharge Studio Workspace*

This section introduces the CodeCharge Studio workspace, illustrating the main areas of the interface and shows how you can customize the workspace to maximize productivity.

The workspace is the graphic user interface (GUI) with which you interact during the process of using CodeCharge Studio. The following are the main portions of the CCS workspace:

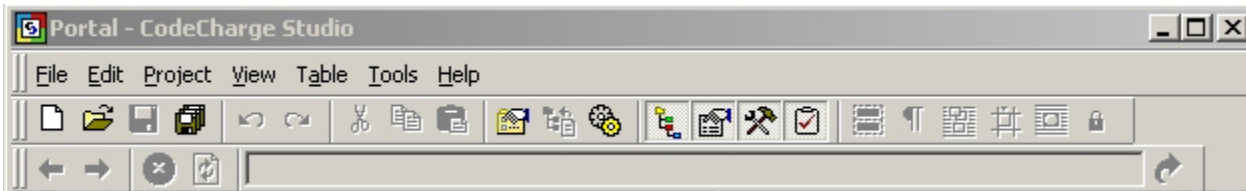
1. **Command Area** – This is the top portion of the workspace that consists of the Title bar, Menu bar and various Toolbars used to provide quick access to various commands.
2. **Project Explorer** – This panel contains a list of the various components that make up a project.
3. **Document window** – This is where most of the action goes on. It is here that project pages are opened edited and viewed. The document window can either be in Design, HTML, Code, Preview or Live Page mode.
4. **Toolbox window** – This panel provides one-click access to various commands and wizards that can be used to add features to project documents.
5. **Properties window** – As the name suggests, the properties window displays relevant information pertaining to the currently selected document element. It can be used to adjust various properties depending on the currently selected element.
6. **Messages window** – This window provides feedback concerning processes that are executed such as publishing of a project.

Apart from the above major components, the workspace also has other minor components such as the Status bar and Tooltips.

The illustration below shows the main parts of the CCS workspace.



## The Command Area



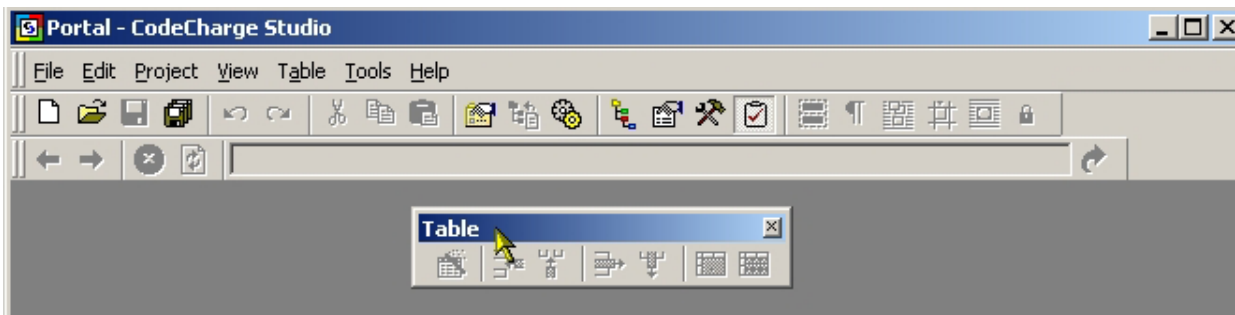
**Title Bar:** The title bar contains the name of the application (CodeCharge Studio) as well as the name of the currently open project. The right end of the title bar also contains buttons that can be used to minimize, maximize or close the application window

**Menu Bar:** The menu bar features seven drop-down menus (File, Edit, Project, View, Table, Tools and Help). Each of these menus contains a series of options that carry out various functions when clicked. Notice that the main menu names as well as some of the sub-menus have an underlined letter. This is indication that that menu can be executed using a short cut key combination by pressing the Ctrl key together with the underlined letter.





















**Toolbar:** The toolbar provides quick access to many of the commonly used commands that are available in the menu bar. If you are concerned about screen space, you can use the View menu in the menu bar to hide some or all of the toolbars. Each of the toolbars can be dragged and docked on the top, left or right sides of the application window.

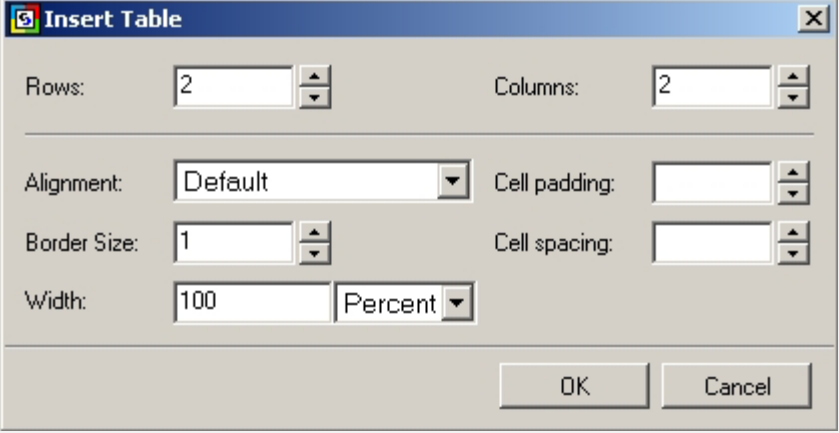






















You can also drag the toolbars away from the sides of the application window and in so doing create a floating toolbar.










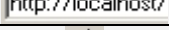



The following is a description of the commands available in the different toolbars.

Command	Description
<b>Standard</b>	
	Create a new page.
	Open an existing page.
	Save the current page.
	Save all changes made in multiple documents.
	Undo the last action.
	Redo the last action that was undone.
	Cut the current selection and place it in the clipboard.
	Copy the current selection to the clipboard.
	Paste the current contents of the clipboard.
	Open the Project Settings window.
	Synchronize code to reflect changes made.
	Publish the project.
	Publish the current page.
	View the Project Explorer window.
	View the Properties window.
	View the Toolbox.
	View the Message window.
	Show borders.
	Reveal paragraph and new-line tags.
<b>Table</b>	
	Open the Insert Table window

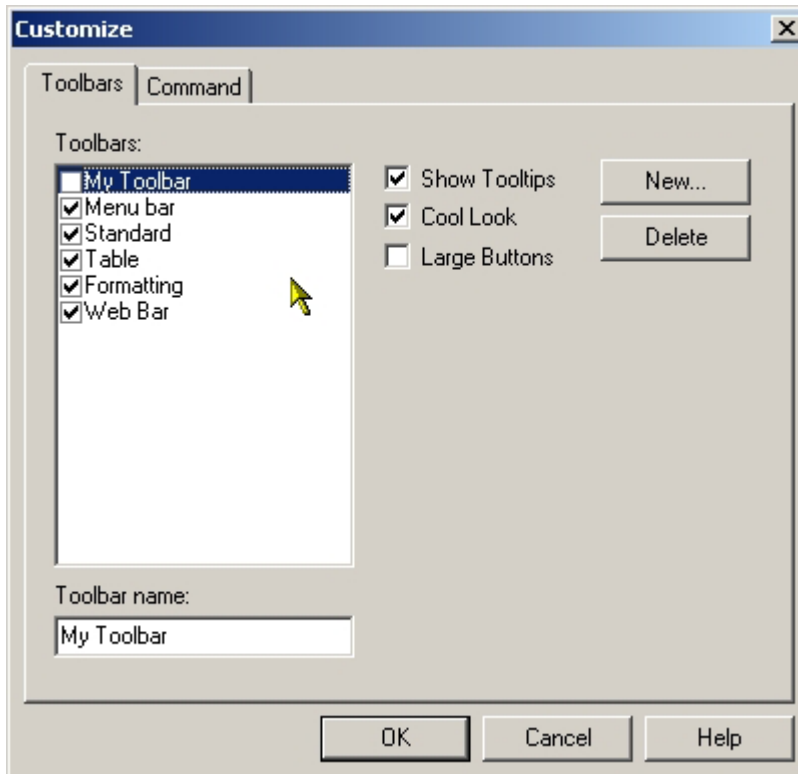
	 <ul style="list-style-type: none"> <li>• Row - Specify the number of rows for the new table.</li> <li>• Columns - Specify the number of columns for the new table.</li> <li>• Alignment - Select an alignment for the table [Default, Left, Center, Right]</li> <li>• Border Size - Specify the width of the table borders.</li> <li>• Cell Padding – Specify the amount of space between the border of the cell and its contents.</li> <li>• Cell Spacing – Specify the amount of space between adjacent cells.</li> <li>• Width – Specify the width of the table in pixels or as a percentage of the available space.</li> </ul>
	Insert a row above the current row.
	Insert a column before the current column.
	Delete the current row.
	Delete the current column
	Merge the currently selected cells.
	Split the current cell.
<b>Formatting</b>	
	Select a format for the currently selected text.
	Select a font.
	Select a font size.
	Convert the selected text to bold typeface.
	Convert the selected text to italics typeface.
	Underline the selected text.
	Insert line numbers.
	Insert bullets.
	Decrease indentation.
	Increase indentation.
	Left align.
	Center.
	Right align.
	Select a font color for the current text.



	<div> <div>Automatic</div> <div>Standard Colors</div> <div> <div>Black</div> <div>Gray</div> <div>Red</div> <div>Green</div> <div>Blue</div> <div>Purple</div> <div>White</div> <div>Light Gray</div> <div>Yellow</div> <div>Cyan</div> <div>Magenta</div> </div> <div>More colors...</div> </div>
	<div> <div>Automatic</div> <div>Standard Colors</div> <div> <div>Black</div> <div>Gray</div> <div>Red</div> <div>Green</div> <div>Blue</div> <div>Purple</div> <div>White</div> <div>Light Gray</div> <div>Yellow</div> <div>Cyan</div> <div>Magenta</div> </div> <div>More colors...</div> </div>
<b>Positioning</b>	
	Align the position of the control to the closest grid line.
	Assign the current control an absolute rather than relative position.
	Fix the position an absolutely positioned control.
<b>Web Bar</b>	
	View the previous page.
	View the preceding page.
	Stop the current operation.
	Refresh the current page.
	The URL of the current page.
	Go to the URL specified in the URL field.

Using the Tools → Customize Toolbar... menu option, you can configure the appearance of the various toolbars as well as add new toolbars.

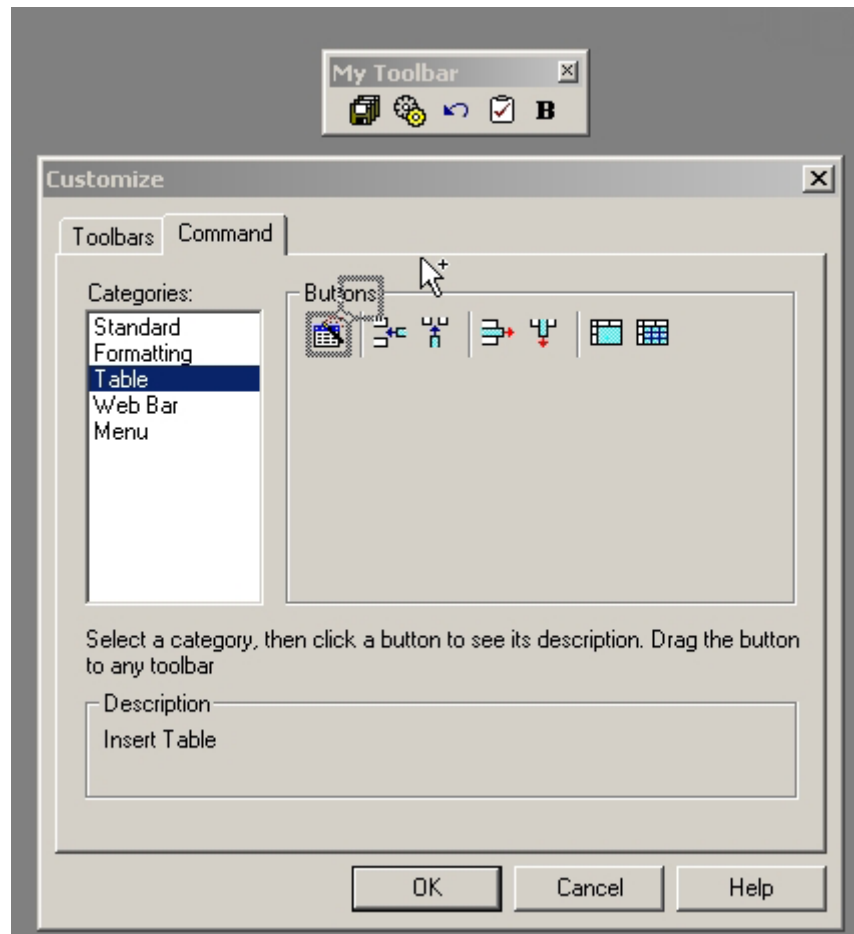
- **Show Tooltips:** When this option is activated, informative Tooltips appear when the mouse hovers over an option in the toolbar.
- **New...:** Clicking on this button adds a new toolbar to which you can later on add buttons using the Command tab.
- **Delete:** Use this button to delete a toolbar you created yourself. The six default toolbars cannot be deleted.
- **Reset:** The Reset button appears in place of the Delete button mentioned above when one of the five default toolbars is selected. The reset button has the effect of returning the toolbars to their factory appearance after they have been customized to look differently.
- **Toolbar name:** If you add a new toolbar, you can use this field to alter its name.



Under the Command tab, you can:

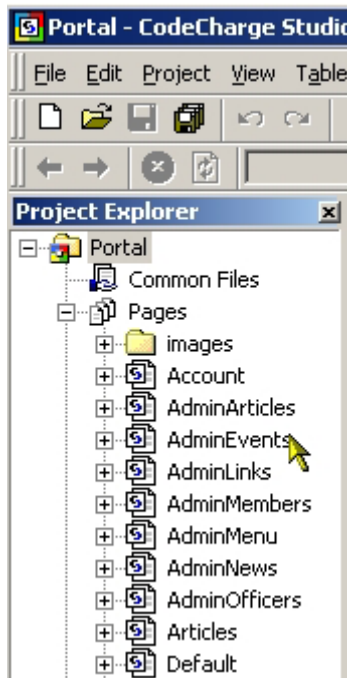
- View the various options available in a particular toolbar as well as see the tooltip text associated with the option.
- Add commands to a new or existing toolbar. To add a command to a toolbar, click and hold down the mouse button as you drag the command icon to the location where you want it to appear on the toolbar. Note that you can also remove command icons from a toolbar by clicking and dragging the icon away from the toolbar.

The illustration below shows the process of adding a command icon to a custom toolbar called My Toolbar.

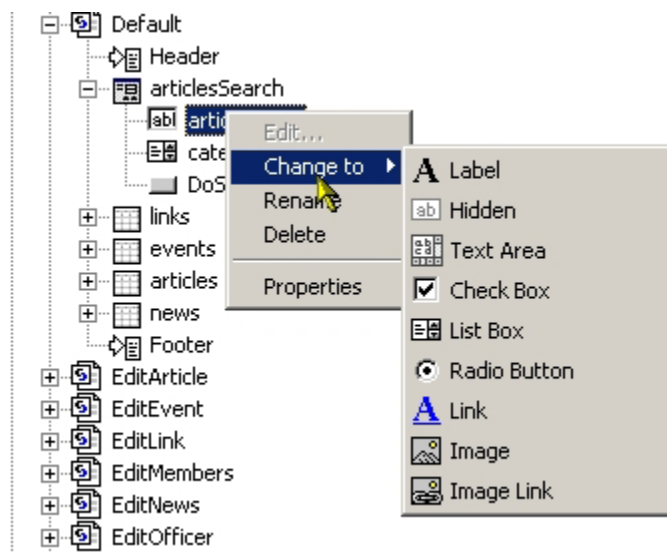


## The Project Explorer

The project explorer is essentially a hierarchical drop-down menu that provides access to the main components that make up a project. If the Project Explorer is not visible, you can use the View menu to make it visible. Like the toolbars and other windows, the Project Explorer window can be docked to the side of the application or against other windows such as the Toolbox. Alternatively, it can be a floating window.



The Project Explorer window works in conjunction with Document window and the Properties window. Double clicking on a page within the Project Explorer window will cause it to be opened in the Document window. At the same time, the Properties window displays the various properties for the currently selected component. Using the Project Explorer, you can drill down to the basic components of a project and execute various commands by right clicking on a component to bring up a popup menu.

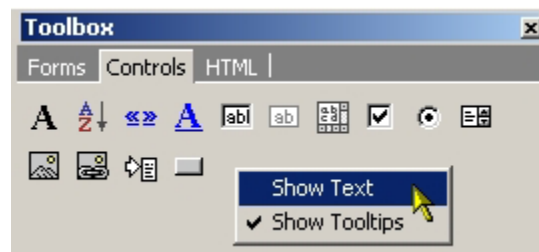


## The Toolbox

The Toolbox consists of three tabbed windows featuring a series of commands for adding various controls to a page or accessing wizards.



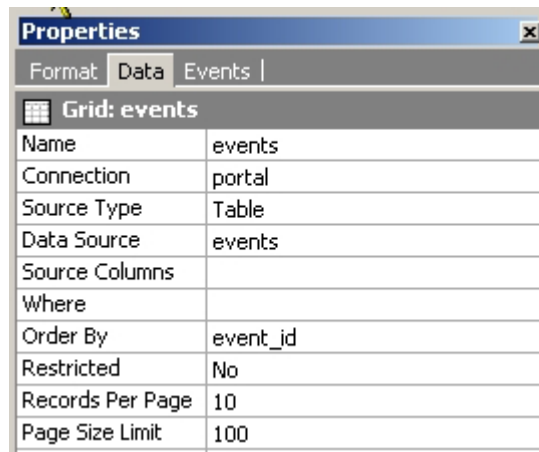
Depending on how familiar you are with the various options on the Toolbox you can opt whether or not to have it display text along with the Icons. You can also enable tooltips to be displayed when the mouse pointer hovers over a particular option. Right clicking within the Toolbox brings up a popup menu where you can toggle these features.



## The Properties Window

As mentioned earlier, the Properties window acts in conjunction with the Project Explorer as well as the Document window. The Properties window consists of three tabs (Format, Data, Events). You will notice that the contents of each of the tabs in the Properties window changes according to the currently selected element in the Project Explorer or Document window.

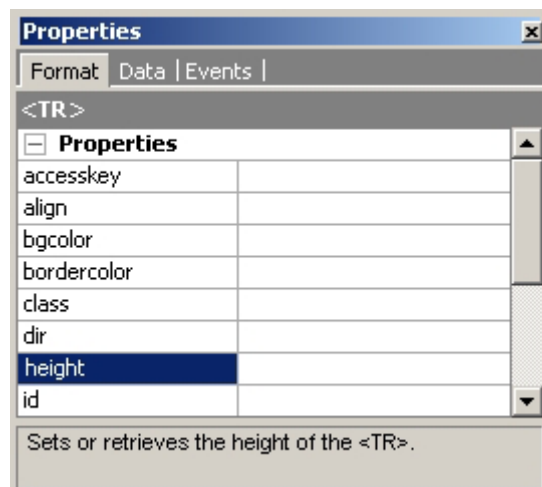
When using the Properties window, a caption that appears between the tabs and the list of properties displays the type and name of the currently selected component. For example, in the illustration below, a Grid form called events is the currently selected component.



It is advisable to make a habit of checking the caption before setting any property or event values so as not to end up making changes to the wrong component.

All the items in the Format tab and the Data tab display properties and events in a Name-Value format. The name of the property or event appears to the left of the window while its attendant value, if any, appears to the right. In some of the properties, drop-down lists are used to specify a series of options from which to pick a value while in others you can simply type in a value. Additionally some properties use popup dialogs of varying complexity and functionality to specify the property's value.

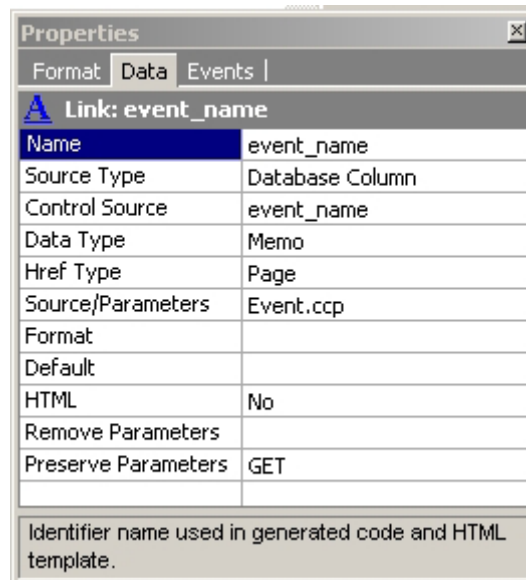
Unlike the other windows that we have seen so far, the Properties window features its own Status bar that is used to display helpful information about the currently selected property. For instance, in the illustration below, the status bar at the bottom of the Properties windows displays information about the height property of the <TR> tag.



**Format Tab:** The Format tab contains properties and events for various HTML/CCS elements. For the most part, the Format tab is mainly used when a page is opened in the Document window in HTML mode. In this case, selecting a HTML tag within the document will cause the Format tab to display the Properties and events applicable to that particular tag.

**Data Tab:** Whereas the Format tab displays HTML/CSS properties and events, the properties displayed under the Data Tab are implemented at the programming language level and can usually be controlled programmatically. As with the Format tab, the properties displayed under the Data tab vary depending on the currently selected control in the Project Explorer or Document window.

The following illustration shows the Data Tab of the Properties window displaying the properties for a Link control called event\_name.



**Events Tab:** Events are designated points that occur during the process of code execution. Using the Events tab, it is possible to add Actions or custom code that would then be executed at the point in the code where the event occurs. Actions are predefined snippets of code that perform common functions such as validating a value such as a phone number or zip code or sending an email message. However, since there cannot be Actions for all the possible operations that you may want to perform in an event, you can also enter your own custom code into an Event. Note that any code added into an Event is language specific. If you change the language in which the project pages are generated, you have to write new event code in the new language.

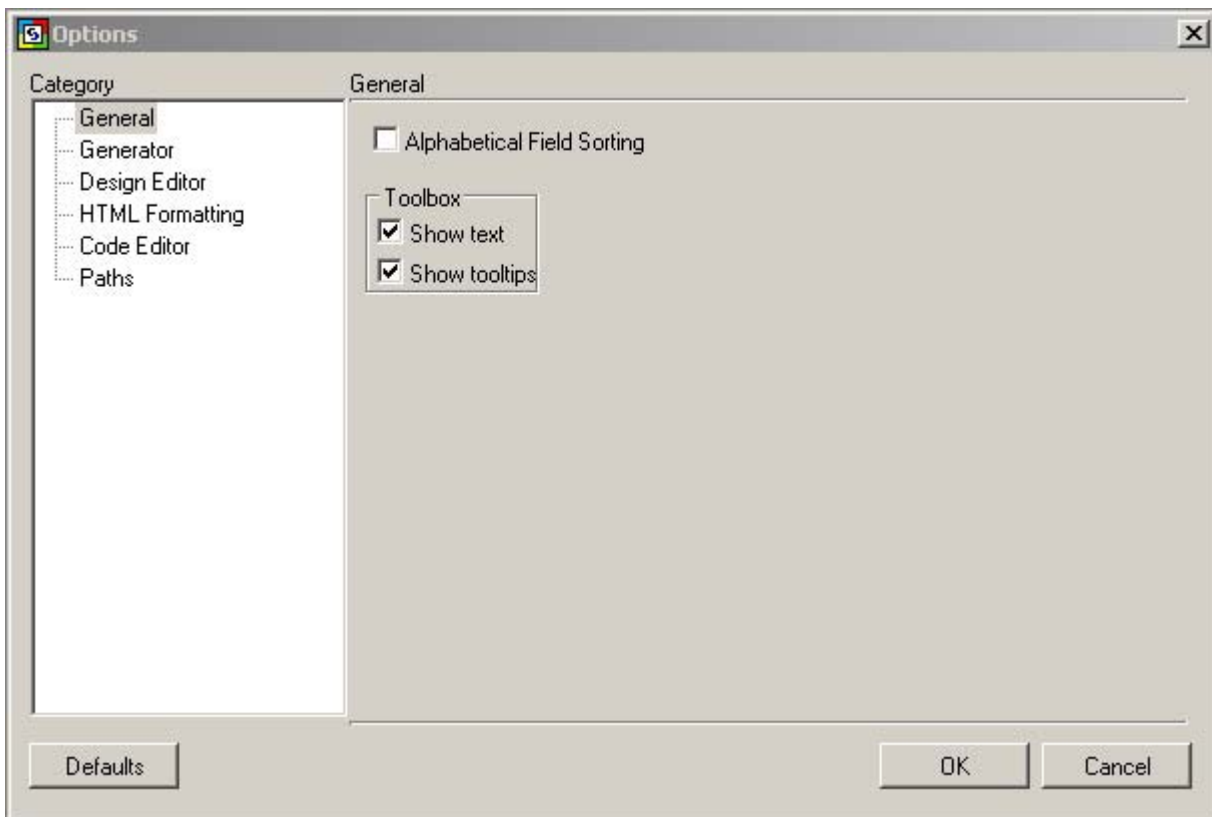
## Customizing the Workspace

By using the **Tools** → **Options...** menu, you can access additional configuration options that further customize the CodeCharge Studio workspace. The Options window has a list of six categories each with a set of configurable options.

### GENERAL

Property	Description
Alphabetical Field Sorting	If selected, database fields listed in the CCS IDE during design time will be ordered alphabetically rather than in the order in which they appear in the database. For instance, when using the wizards to create forms, the fields will be shown alphabetically if this option is selected.
Show text	If selected, caption text will appear next to each of the buttons in the Toolbox

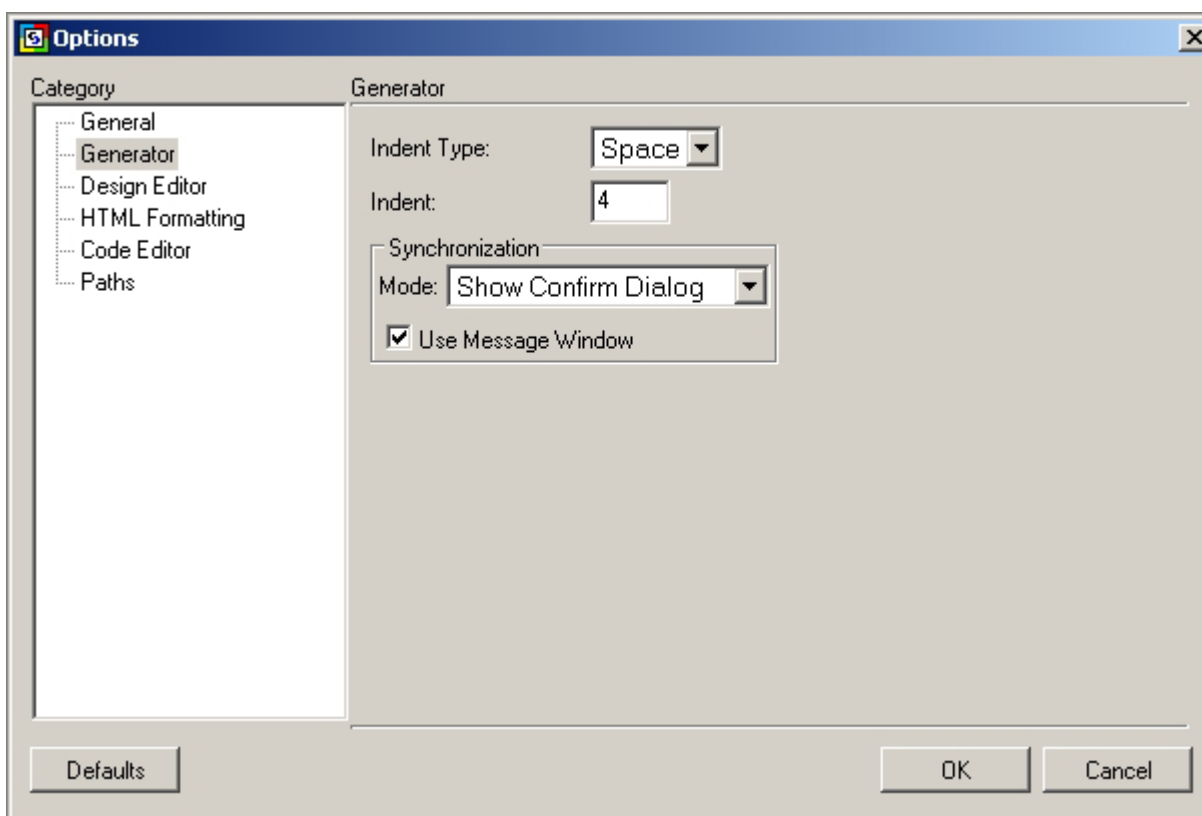
Show tooltips	If selected, caption text will appear as a tooltip when the mouse pointer hovers over each of the buttons in the Toolbox.
---------------	---



## GENERATOR

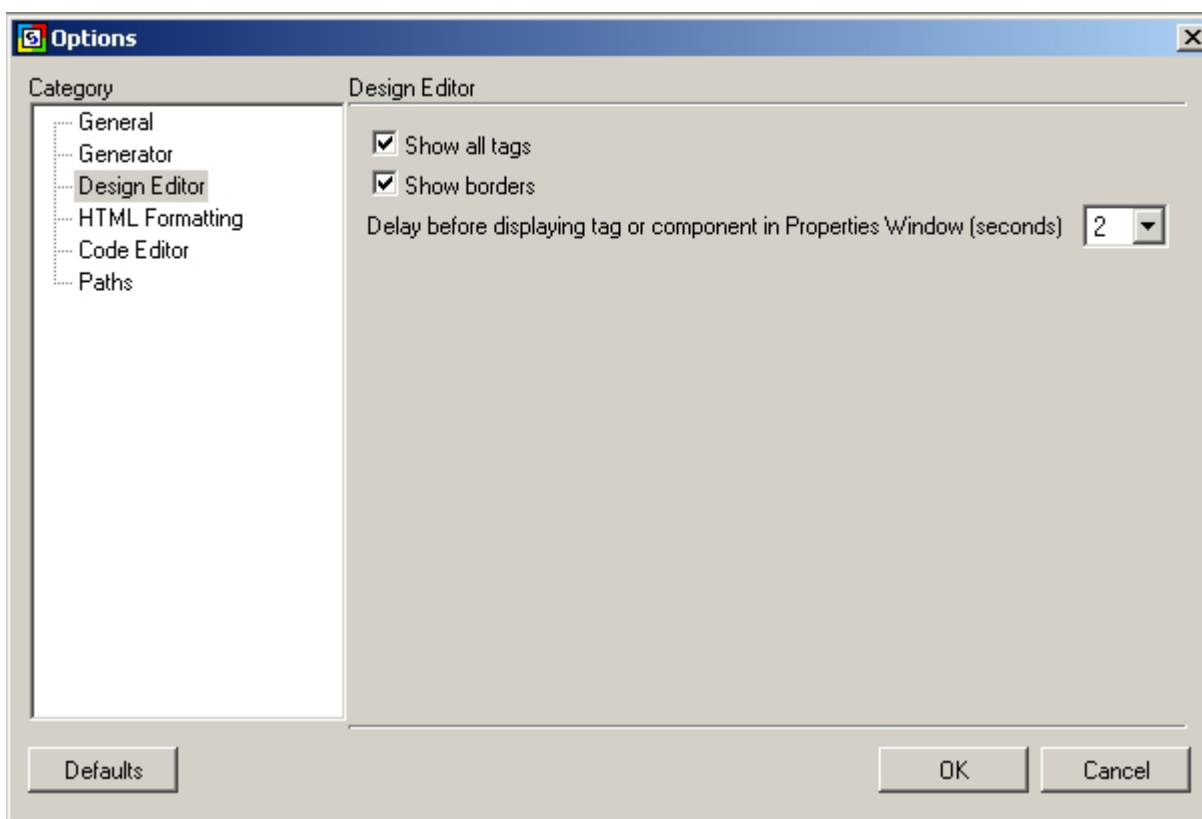
Property	Description
Indent Type	Select whether to uses spaces or tabs to indent the generated code
Indent	Specify the number of spaces to be used for indenting
Synchronization Mode	<p>During the process of creating a page, the user often deletes some components from the page and this is subsequently detected by CCS during the process of synchronization. This option allows you to specify the action to be taken when synchronizing page content:</p> <ul style="list-style-type: none"> <li>• Show Confirm Dialog – A dialog appears asking the user to confirm the deletion of components that were not found on the page.</li> <li>• Delete by default – Any components not found on the page are automatically deleted.</li> <li>• Don't Delete by Default – Components are not deleted from the page even if they are not found in the page code.</li> </ul>
Use Message Window	If selected, the Message window will also be used to display synchronization warning.





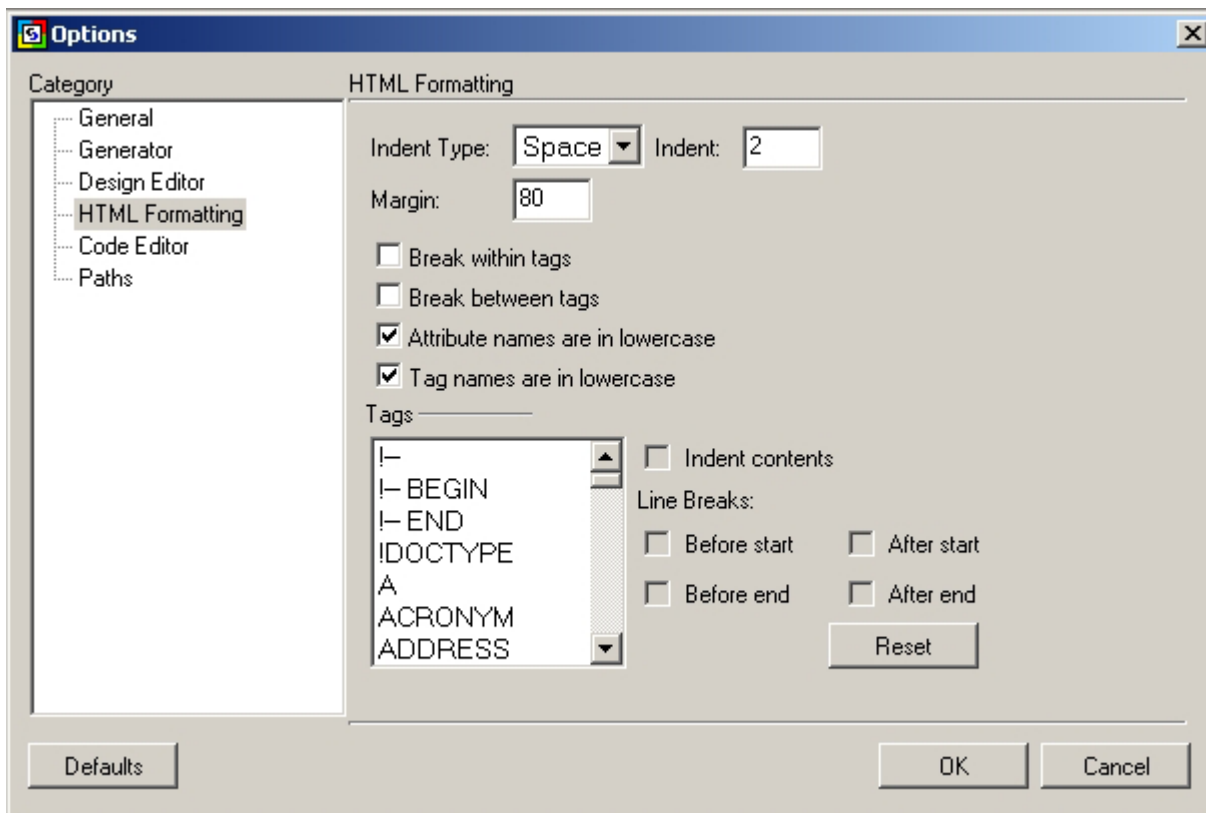
## DESIGN EDITOR

Property	Description
Show All Tags:	If selected, Line Break and Paragraphs Breaks will be shown using icons.
Show Borders:	If selected, a border will be shown around all HTML tables in Design mode.
Delay before displaying tag or component in Properties Window (seconds):	Set the amount of time to wait before switching the Properties window to the currently selected control.



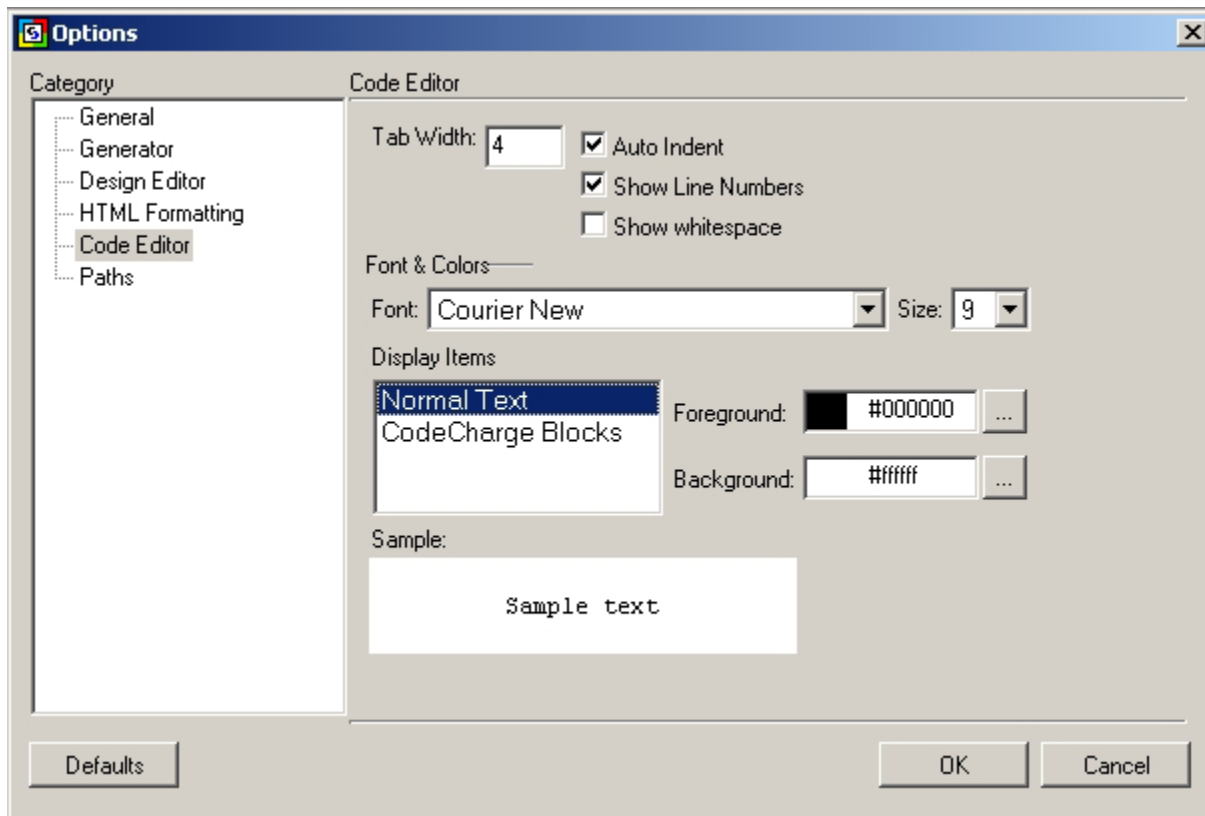
## HTML FORMATTING

Property	Description
Indent Type	Select whether to use space or tab indenting.
Indent	Specify the number of spaces to be used for indenting.
Margin:	The maximum number of characters that a string in a page can span.
Break within tags:	If a tag is longer than the Margin length, a line break will be inserted even if the tag is not closed.
Break between tags:	Line breaks for strings longer than the Margin length will only be inserted in between tags.
Attribute names are in lowercase:	Attribute names for tags will be in lowercase.
Tag names are in lowercase:	Tag names will be in lowercase.
<b>Tags</b>	
Indent Contents	Specify whether a specific tag will be indented.
<b>Line Breaks</b>	
Before Start:	Insert a Line break before the start of the tag.
After Start:	Insert a Line break after the start of the tag.
Before End:	Insert a Line break before the end of the tag.
After End:	Insert a Line break after the end the tag.
Reset:	Return to the previous settings.



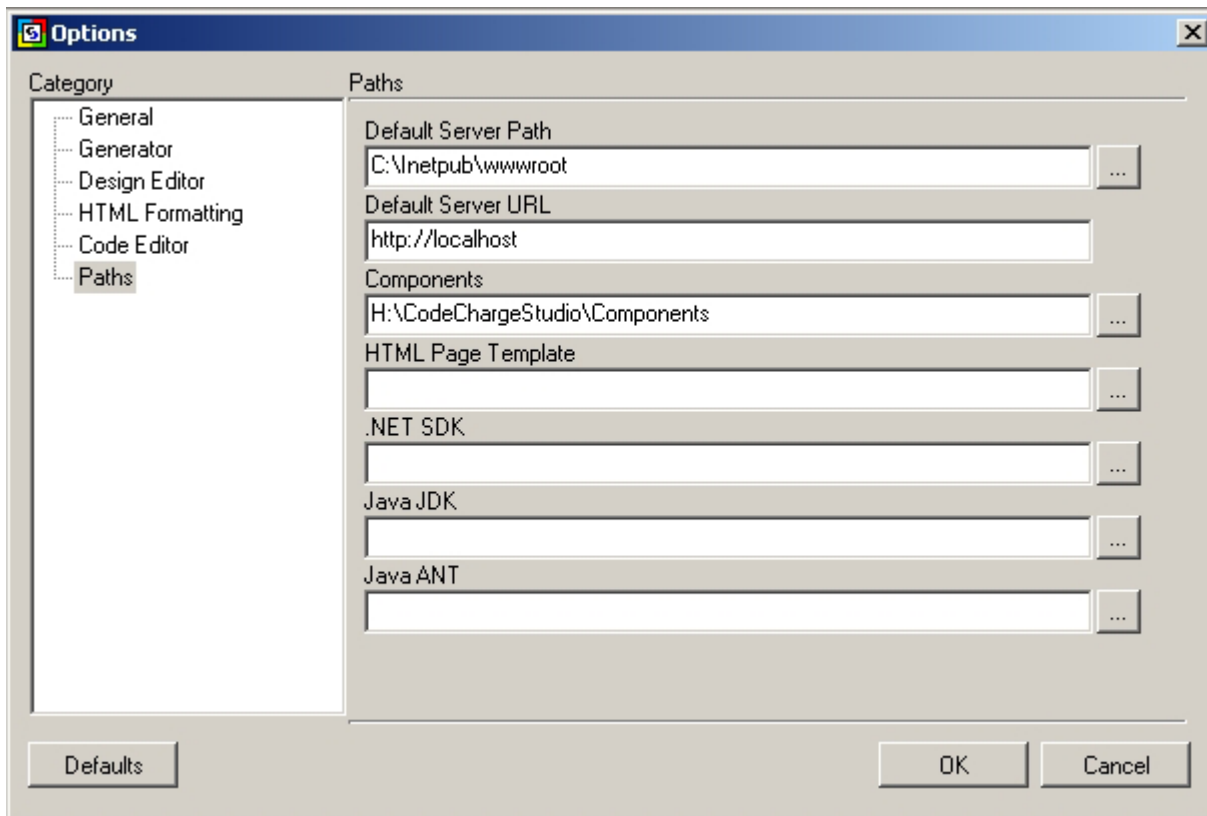
## CODE EDITOR

Property	Description
Tab Width	Specify the number of spaces to be used to represent a tab.
Auto Indent	If selected, code will be indented automatically.
Show Line Numbers	If selected, line numbers are shown on the left margin of the Code editor
Show whitespace	If selected, white spaces are visually represented by periods (.) and tabs are represented using the symbol >>.
<b>Font &amp; Colors</b>	
Font	Select the font in which the code will be displayed
Size	Select the size of the display font
Display Items	Select either Normal Text or CodeCharge Blocks to configure their display options
Foreground	Select the foreground color of the selected item
Background	Select the background color of the selected item.
Sample	This shows a preview of the item according to the selections made



## PATHS

Property	Description
Default Server Path:	Specifies the default path to the web server to which pages are generated.
Default Server URL:	Specifies the HTTP URL that maps to the default server path mentioned above.
Components:	Specifies the path where CCS components are located.
HTML Page Template:	Specifies the path to a HTML template page that will be used as the base for all newly created pages.
.Net SDK:	Specifies the location of the .Net framework SDK if available.
Java JDK:	Specifies the location of the Java JDK if available.
Java ANT:	Specifies the location of the Java ANT if available.

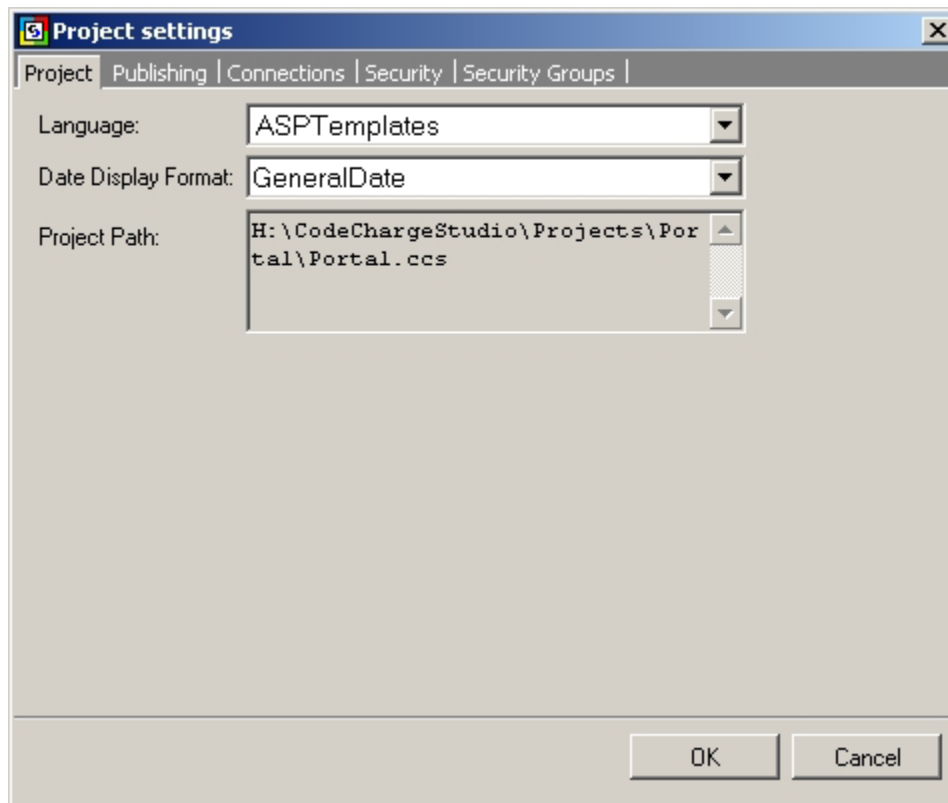


## Project Settings

When working with a project, there are certain settings that apply globally to the project and these can be accessed using the **Project → Settings...** menu. The **Project Settings** window contains a series of five tabs under which the various settings are categorized.

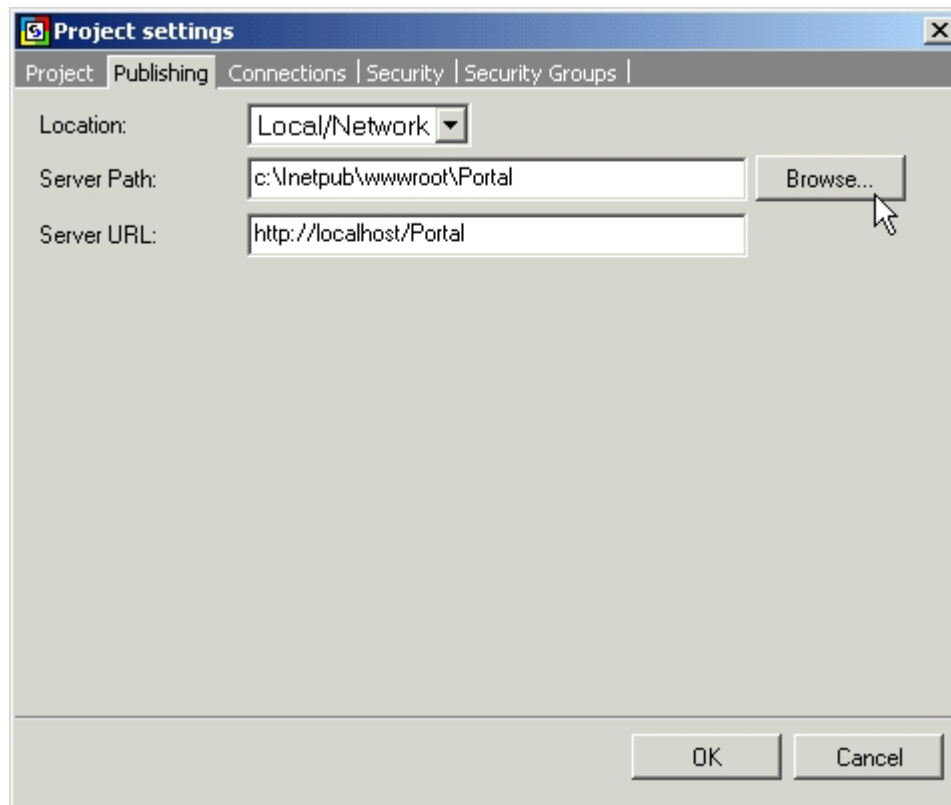
## Project

Property	Description
Language	Select the programming language in which the code will be generated
Date Display Format	Select the format in which you want date values to be displayed in the project. If no selection is made, date values will be displayed in the default database format
Project Path	This is the path to the CCS file for the project



## Publishing

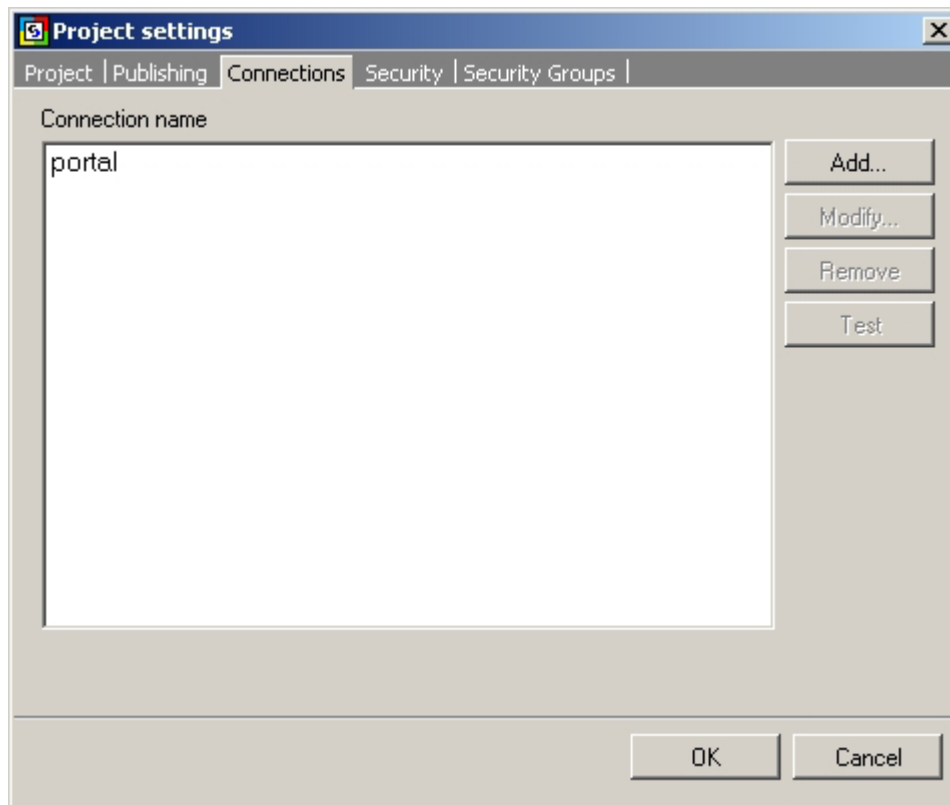
Property	Description
Location:	Specify whether the generated pages will be published to a local computer or network or whether they will be published to a remote computer via FTP
Server Path:	The filing system path to where the pages will be generated
Server URL:	The HTTP URL that maps to the path specified above



## Connections

Under this tab is a list of all the database connections configured for the project. A project can have multiple database connections each connecting to a different type of database.

Property	Description
Connection name	The unique name used to identify a particular connection
Add...	Click this button to initiate the process of adding a connection
Modify...	Select an existing connection then click this button to modify it
Remove	Click this button to remove an existing connection
Test	Click this button to test whether a connection is working properly.



## Security

Property	Description
Security Method:	<ul style="list-style-type: none"> <li>• None: No site security is implemented</li> <li>• Table: Security will be based on information contained in a database table</li> <li>• Database: The inbuilt database security will be used for the site</li> </ul>
Login Page:	Select the project page that will be used to enter authentication details by the users
Store User ID:	Select the method to be used to store the User ID after a user logs in (Session, URL, Cookies)
Advanced:	<ul style="list-style-type: none"> <li>• User ID Variable: Enter a name to be used for the User ID variable</li> <li>• Group ID Variable: Enter a name to be used for the Group ID variable</li> </ul>
Connection:	Select the name of the connection to be used for security information
Table:	Select the database table containing user security information
User ID Field:	Select the database table field containing the unique ID that identifies the users
Login Field:	Select the database table field containing the login usernames of the users
Password Field:	Select the database table field containing the login passwords of the users
Groups Field:	If applicable, select the database table field containing the security level of the users



**Project settings**

Project | Publishing | Connections | **Security** | Security Groups

Security Method: ☐ None ☒ Table ☐ Database

Login Page:  ...

Store User ID:  Advanced...

Connection:

Table:

User ID Field:

Login Field:

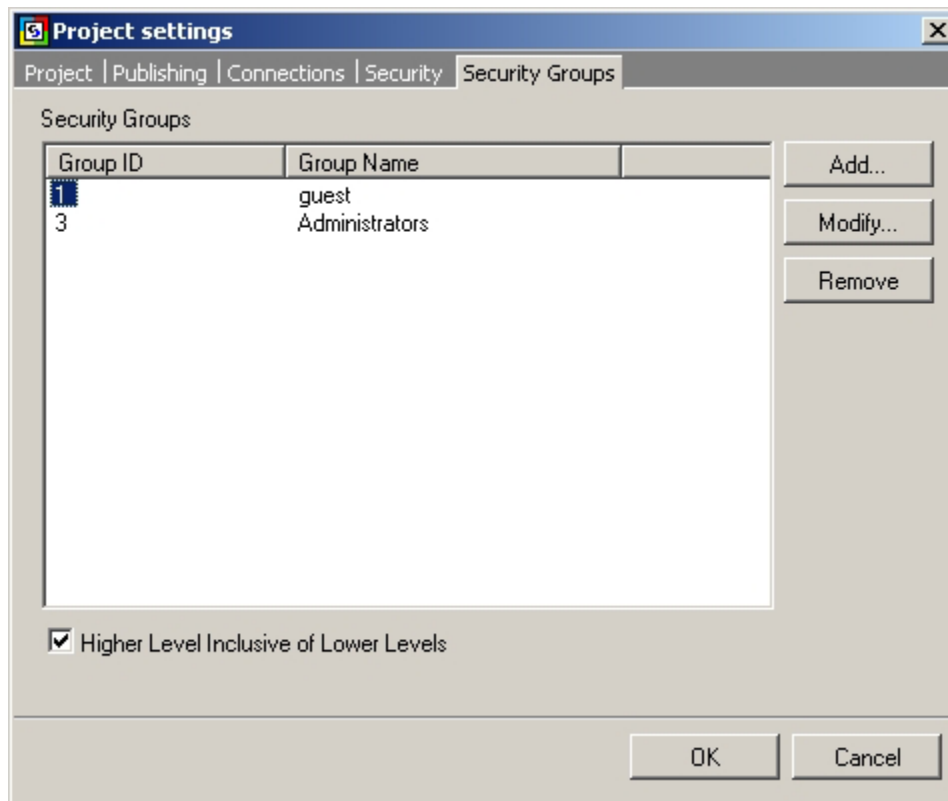
Password Field:

Groups Field:

OK Cancel

## Security Groups

Property	Description
Security Groups:	<ul style="list-style-type: none"> <li>Group ID: The unique value used to identify the group</li> <li>Group Name: A descriptive name for the group</li> </ul>
Add...	Click this button to add a new security group
Modify...	Click this button to modify an existing security group
Remove	Click this button to remove an existing security group
Higher Level Inclusive of Lower Levels	If this option is selected, users with a higher security level will be able to access pages designated with a lower security level. Otherwise, an exact match will be needed in order to access any page.



## IV. Databases and Data Sources

One of the main features that makes CodeCharge Studio stand out from other web development tools is its ability to put database content on the web. Needless to say, some interaction with databases will inevitably occur during the process of working with a CodeCharge Studio project.

As far as databases go, there is a wide range of applicable options each with its own pro's and con's. It would be inconceivable to accommodate all or the vast majority of the different options if each were unique. Thankfully, this is not the case since the Standardized Query Language (SQL) provides a uniform way of interacting with databases and the content contained therein. Broadly speaking, four main components make up the database experience:

1. The database software itself, otherwise known as archive or repository (e.g. Access, Oracle, MySQL, etc)
2. The driver/provider used to connect to the database. Essentially, this acts as a bridge between the database and the language being used. (e.g. ODBC, OLE-DB, Perl DBI)
3. The programming language being used. The programming language includes programming constructs that enable it to establish connections to the database and execute SQL commands against it. (e.g. ASP, PHP, Perl, etc)
4. The SQL language that is used to command the database. It is important to make the distinction that it is SQL and not the programming language that actually commands the database. The programming language is used to establish a connection to the database and then it sends SQL commands to be executed over the opened connection. The SQL transmitted is largely standard but different products do have proprietary SQL commands above and beyond the standard commands.

Various factors influence the choice of database used. These include but are not limited to:

- The amount of information being stored
- The speed and versatility required of the resultant application
- Security, backup and recovery concerns
- Price as well as operating cost of the database
- Compatibility with the intended programming language
- Availability of database drivers to connect to the database using the chosen programming language
- Availability of enhanced functionality such as stored procedures, procedural programming, replication, etc.
- Availability of a web host who supports the database
- Ease of use

Having selected the database to use and ensured that all the necessary components are in place, the next step is to make the connection within CodeCharge Studio. Note that you are not limited to a single database but can use multiple databases within the same project.

## Connections

Within CodeCharge Studio, a Connection is a link to a database for the purpose of creating pages based on the structure and contents of the database. The majority of the forms and field controls used within CodeCharge Studio draw their content from database fields.

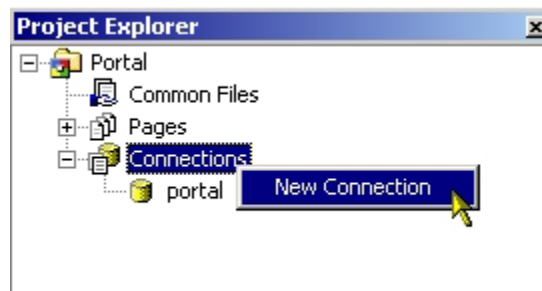
Any connection used within CodeCharge Studio has two sides to it. For designing pages within CodeCharge Studio, a **Design** Connection is used. This is the connection that CodeCharge Studio uses to communicate with the database. On the other side of this is the **Server-Side** connection that is used by the generated pages to communicate with the database.

Often times, the Design connection is the same as the Server-side connection. However, there arises occasions that warrant the use a Server connection that is different from the Design connection. This is usually the case when the generated pages have to be uploaded to a third party host machine where remote database access is not allowed or would be too slow.

We shall now examine the process of creating connections within CodeCharge Studio.

## Creating Connections

There are two ways of creating a Connection to be used in a particular project. The first method involves using the **Connections** option within the Project Explorer window. Right-click on the **Connections** option and select **New Connection** as shown below.

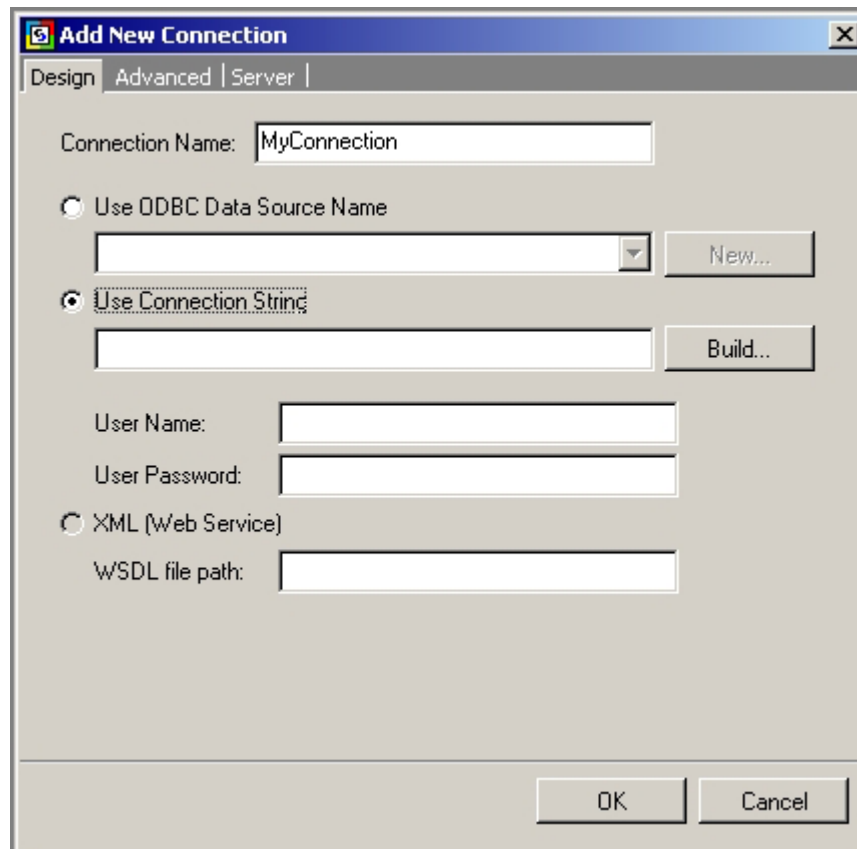


You are then presented with the **Add New Connection** window where you specify the various properties of the Connection. Notice that the **Add New Connection** window has three tabs whose functions are as follows:

**Design:** This is used to configure the Design part of the connection as will be used within CodeCharge Studio.

**Advanced:** This tab has advanced options that are supported by some databases.

**Server:** This is used to configure the Server-side Connection if it is different from the Design connection.



## Design Connection Settings

**Connection Name:** This unique user selected name identifies the connection. No two connections within the same project can have identical names.

**Use ODBC Data Source Name:** This option should be selected if the connection to the database will be done via ODBC. Using the listbox field, you can then select one of the ODBC DSN's that is currently registered on the machine. Alternatively, you can [create a new ODBC DSN](#) by clicking on the **New...** button.

**Use Connection String:** This option is used to specify a connection string that contains the details necessary to make a connection to the database. Some of the details included in the connection string are:

- Database driver to be used
- Location of the database file
- Username and Password needed to connect to the database

If you are an experienced user, you can simply type in the proper connection string into the provided field. However, most people would find it necessary to click the **Build** button and use the **Data Link Properties** windows to [create the connection string](#).

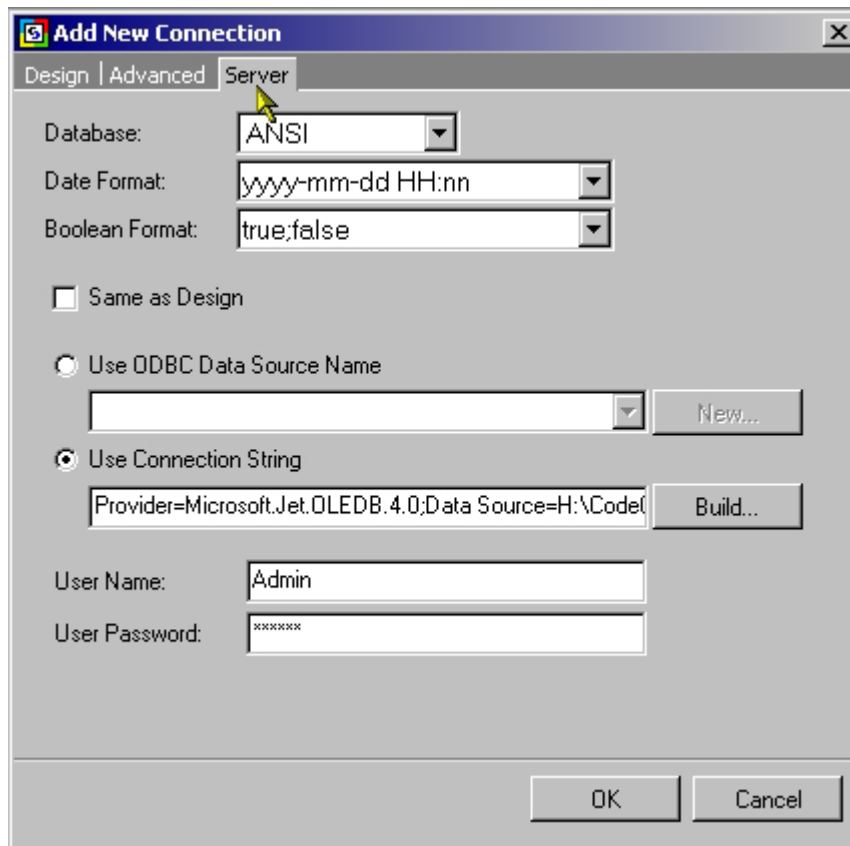
**User Name:** This is an optional field used to specify the username required to login to the database.

**User Password:** This is an optional field used to specify the password required to login to the database.

## Server Connection Settings

The **Server** tab of the **Add New Connection** window is used to specify additional properties for the database connection that is used in the generated pages.

In situations where the generated pages are deployed to a web server located on the development machine, it usually is the case that the Design connection is the same as the Server connection. This means that the connection used within CodeCharge Studio is the same as that used in by the generated pages. However, if the generated pages are to be deployed to a remote web server, it is usually necessary to configure a Server connection that is different from the Design connection.



The first three fields under the Server tab (**Database**, **Date Format**, and **Boolean Format**) apply regardless of whether the Server Connection is the same as the Design connection or not.

- **Database:** The selection made in this field influences the nature of the SQL used in the generated pages. The default option for this value is ANSI, which refers to the ANSI SQL92 standard. ANSI SQL is standard SQL that does not implement any database specific extensions or syntax. You should use this option if your database is not listed in the drop-down field. Otherwise, you can select your particular database from the drop-down list so that the SQL generated is specifically geared for that database.

Bear in mind that if a particular database is specified other than ANSI, the SQL generated would most likely not work if a different database was used. As such, if you would like the generated

pages to be able to function if used in conjunction with different databases, you should select the ANSI option.

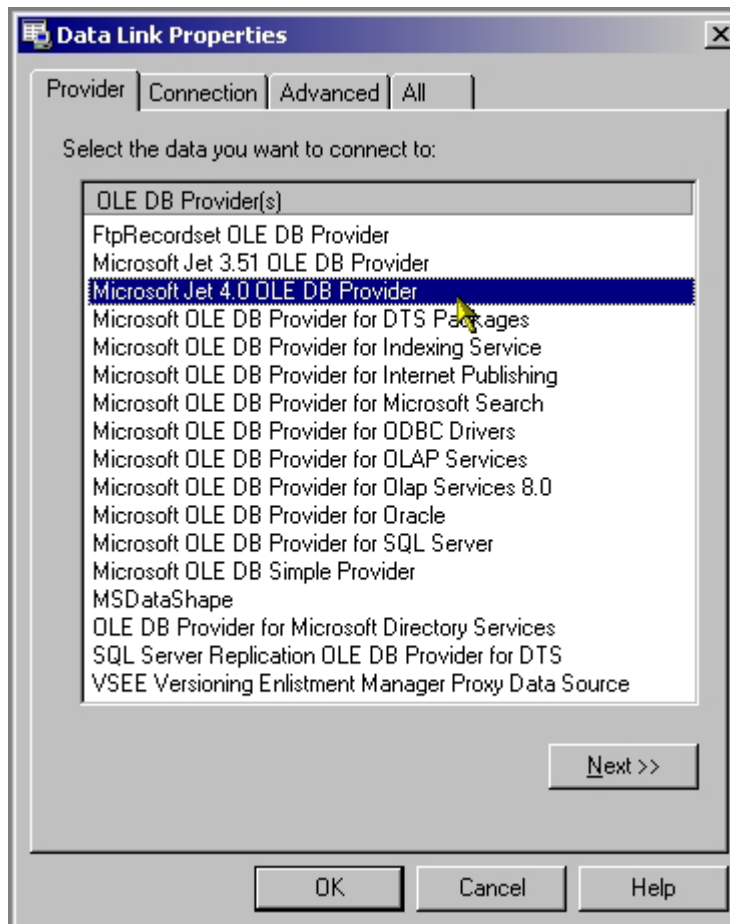
- **Date Format:** Specify the format to be used for date values that are entered into the database. For instance, if you specify yyyy-mm-dd: HH:nn a date such as 11-29-2001 would be entered into the database as 2001-11-29: 00:00.
- **Boolean Format:** Specify the format in which Boolean values will be entered into the database.

If the **Same as Design** checkbox is not activated, you have to specify an ODBC DSN or Connection string to be used for the Server Connection. The process of doing so is essentially similar to that used for the Design connection except that the database referenced should be the one that will be used on the deployment server by the generated pages.

### ***Creating a Connection String (DNS-Less Connection)***

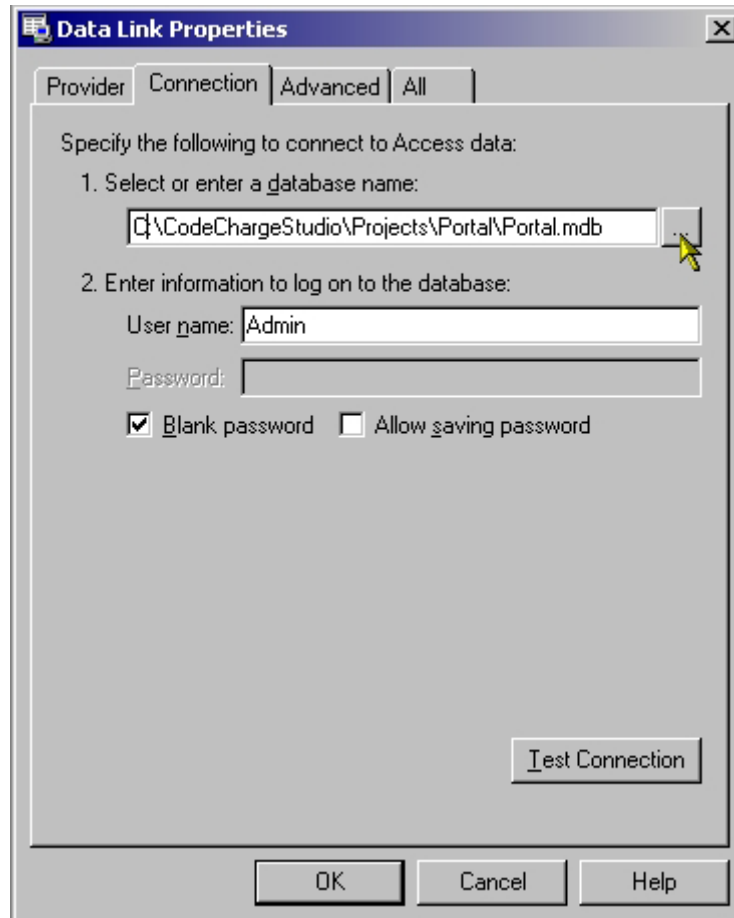
An alternative to using a DSN connection is using a DNS-Less connection that is in essence a connection string containing the details necessary to establish the connection to the database. The makeup of a connection string varies depending on the database being used as well as the provider/driver. While it is possible to simply type out a proper connection string, the better alternative would be to use the **Data Link Properties** window to construct one for you.

The **Data Link Properties** window can be accessed from the **Add New Connection** or **Modify Connection** windows by clicking on the **Build** button.

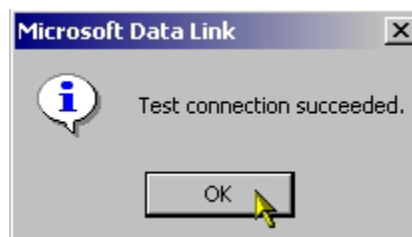


1. The first tab in the Data Link Properties window is the **Provider** tab that is used to specify the database provider/driver to be used. Under this tab will be listed all the OLE DB providers currently installed on the machine. Select the appropriate provider for the database you want to connect to then click on the **Next** button to move to the **Connection** tab. In the illustration above, the Microsoft Jet 4.0 OLE DB Provider has been selected in order to create a connection string to an Access database.
2. The fields displayed in the Connection tab will vary depending on the Provider selected in the Provider tab. In the case where the Microsoft Jet 4.0 OLE DB Provider was selected, the Connection tab will look like the illustration below. In this case, it is necessary to enter a path to the Access database file in the **Select or enter a database name** field. If the Access database file is password secured, then you would also need to enter the necessary authentication details in the **User name** and **Password** field. Otherwise, as is most frequently the case, the **Blank password** checkbox is activated to indicate that no authentication is required to access the database.





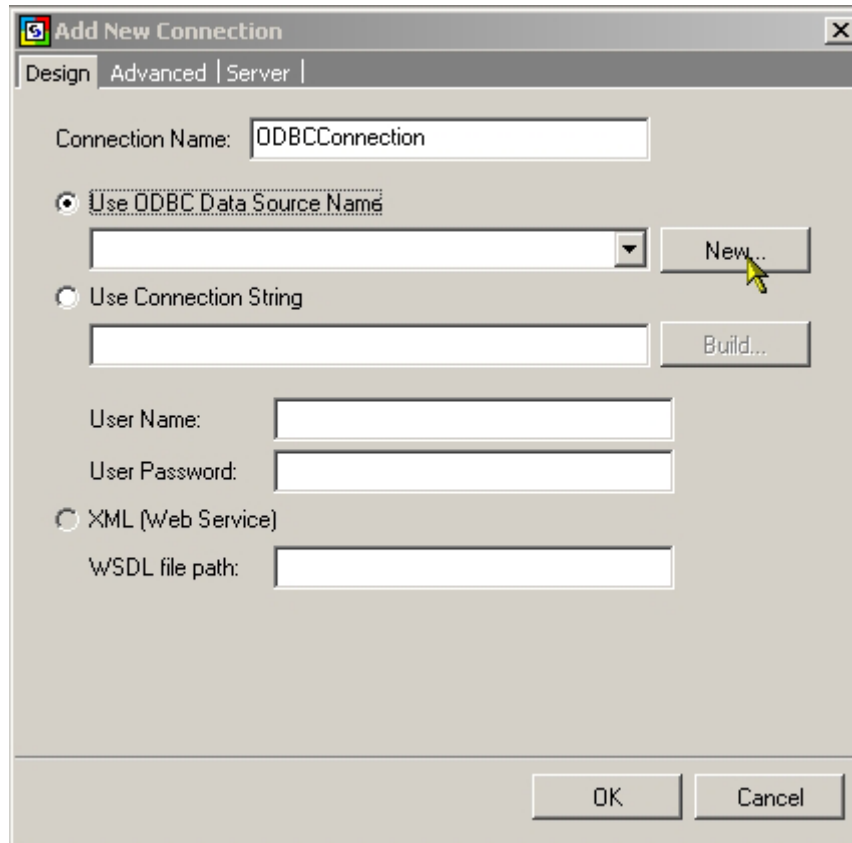
3. After entering the necessary details, it is a good idea to click on the Test Connection button to make sure that the configuration works correctly. If all is well, a dialog box appears to confirm the soundness of the connection otherwise, an error message is displayed depending on what the problem is.



## Creating an ODBC Data Source Name (DSN)

If you need to create a New ODBC DSN, you can do so from either within CodeCharge Studio or by using the **Data Sources (ODBC)** option available in the system Control Panel. In either case, the steps involved are essentially identical so we shall demonstrate how the process is accomplished from within CodeCharge Studio.

The first step is to open the **Add New Connection** window. This can be done by right clicking on the **Connections** option in the Project Explorer window and selecting **New Connection**.

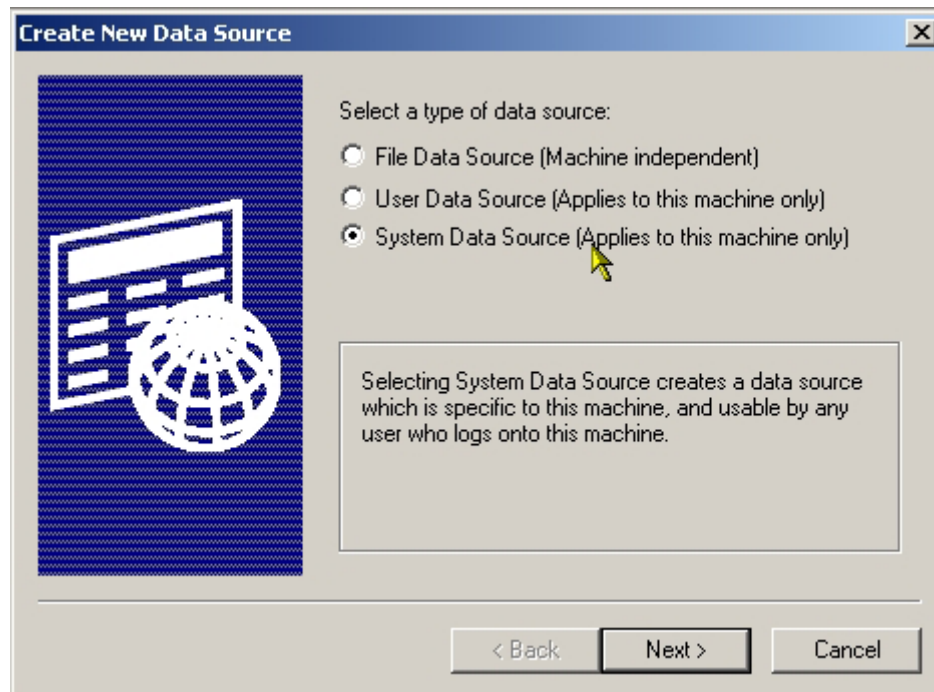


Select the **Use ODBC Data Source Name** radio button to indicate your intention to use an ODBC DSN then click on the **New...** button to begin the process of creating a new DSN.

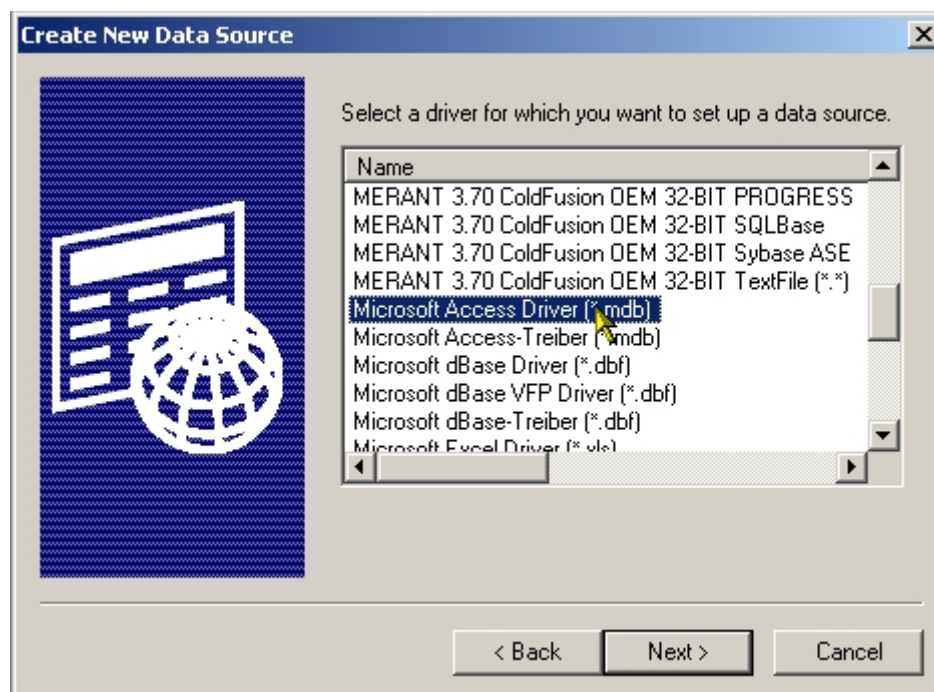
You then have to select the type of DSN to create. There are three choices to select from as described below:

- **File DSN:** This is a data source that can be shared among all users who have the same drivers installed. A file DSN need not be user-dedicated or local to a computer
- **User DSN:** A User DSN is local to a computer and accessible only by the current user.
- **System DSN:** This is a DSN that is local to a computer but not user-dedicated; any user with privileges can access a System DSN.

Unless you have a specific reason for doing otherwise, you should opt to create a System DSN that will be accessible to all users including the account under which the web server process runs.

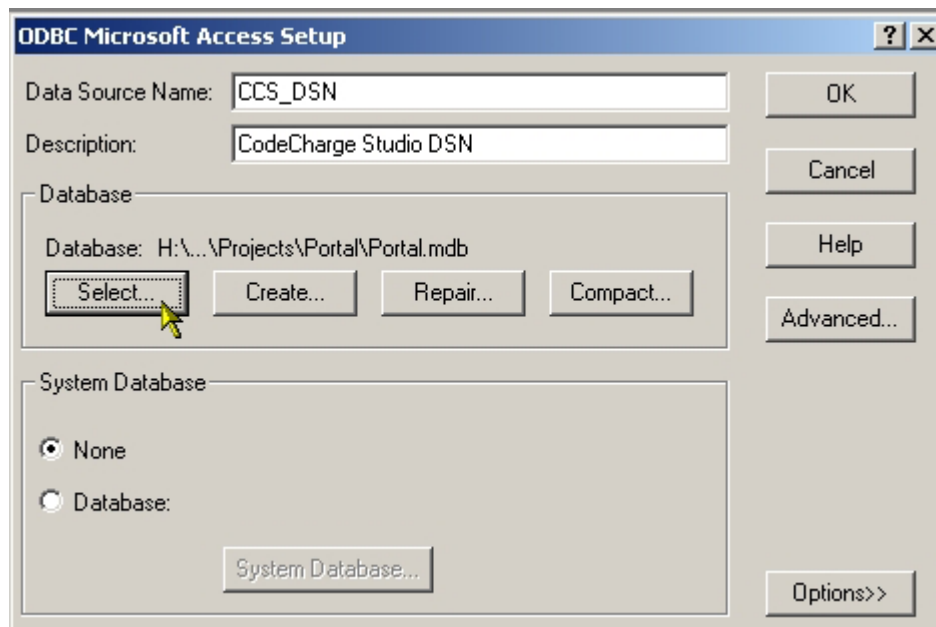


Once you have selected the DSN type, click on the **Next** button to proceed to the next step. You are then presented with a list of ODBC drivers that are currently installed on the Local machine. Scroll through the list to locate the driver for the database you intend to use then select it.

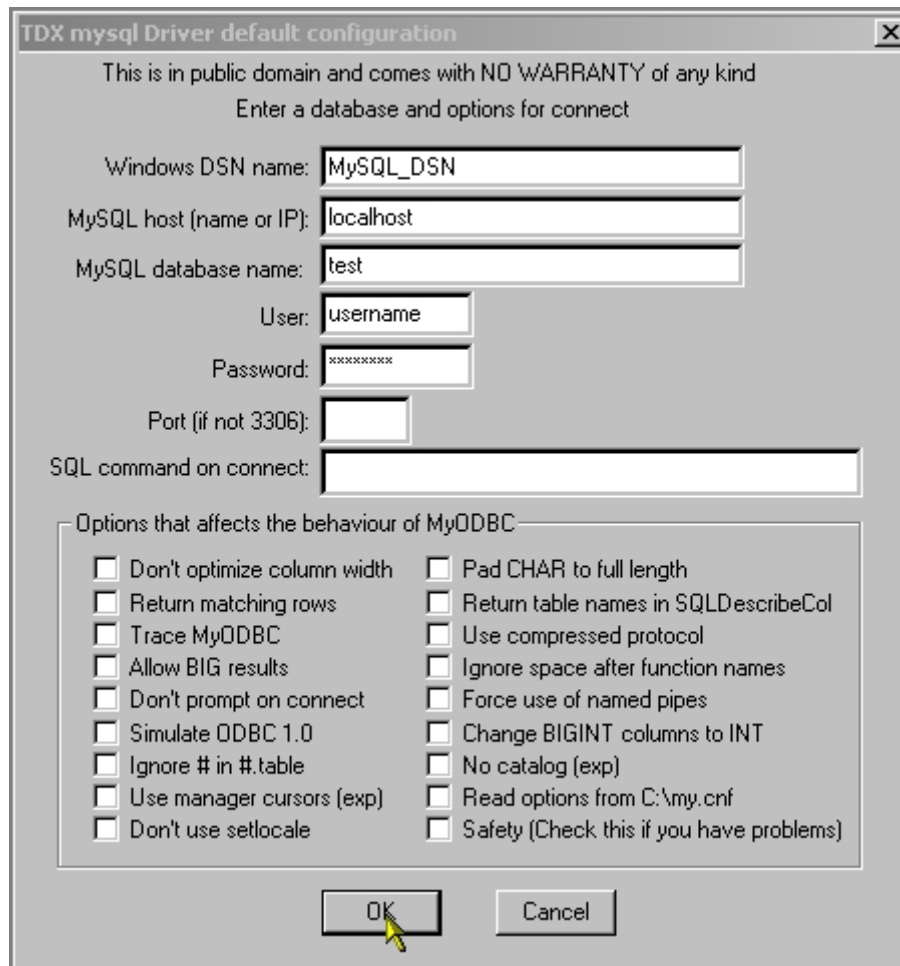


Click **Next** to proceed to the next step, then in the ensuing window, click on the **Finish** button to create the data source.

Depending on the driver you selected, you may or may not be required to enter additional information. For an Access database driver, you are presented with the window below to specify the name of the DSN, an optional description as well as locate Access the file to be associated with the DSN.



Alternatively, if you opted to create a DSN for a MySQL database, you would be presented with a window similar to that shown below. Notice that in this case, instead of specifying a file for the database, you are required to identify the host machine on which the MySQL server resides. You also have to enter a username and password to be used to connect to the database since this is mandatory in the case of a MySQL database.



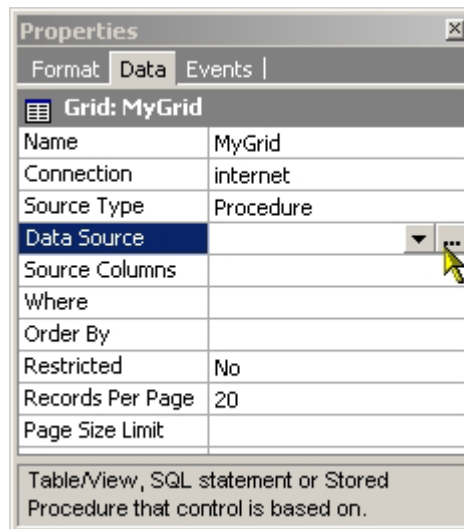
Creating an ODBC DSN for other databases such as MSSQL, Oracle, Excel, etc is inherently similar to the above steps except for the last step that varies depending on the database driver. However, once you have created a DSN, you can use and re-use it in multiple projects by simply selecting it from among the list of available DSN's.

## Form Data Sources

In CCS, dynamic data from databases is displayed using fields that are in turn contained within different types of forms. The data that could be displayed in a form field depends on the data source configured for the form itself. Put in another way, a forms data sources determines the selection of database fields that a field in that form could possibly display.

In this section, we shall examine how form data sources are configured. The main emphasis here is on how to configure a form data source from scratch as opposed to using a [form wizard](#) to automate the process of building the form as well as its data source.

A form data source is configured using the **Data Source** window. The **DataSource** window can be accessed using the Data Source property of a form in the **Data** tab of the **Properties** window.



In addition to facilitating the configuration of a data source for a form, the **DataSource** window is also used setup input parameters. Input parameters are values that are received by the form and are used in the process of filtering the data that is displayed in the form. Input parameters can be received from other forms, via the URL, cookies or session variables. For instance, a search form working in conjunction with a grid form makes uses of input parameters. The search form submits the input parameters and the grid form receives then and uses them to filter the records that are retrieved and displayed on the form.

The following are the options available in the **DataSource** window:

Property	Description
Connection	The name of the connection where the data will be retrieved.
Refresh	If you make changes to the database during the process of using the DataSource window, click on this button to update the window and reflect the current database status.
Data Source Type	Select the type of data source to be used to obtain the data.
Test	This button is only functional when the data source type is a procedure or SQL query. Click this button to execute the procedure or SQL query and confirm its validity.
Review Data	Depending on the data source type selected, clicking this button will retrieve and display a sample of the database content.

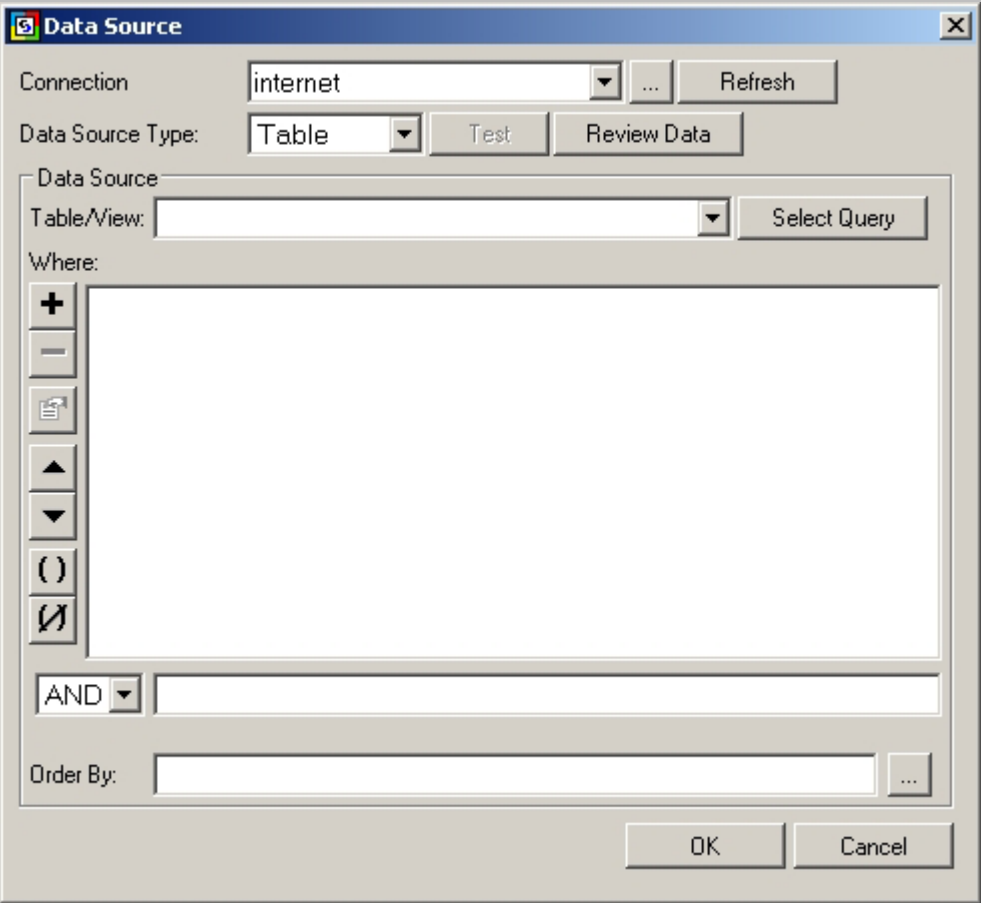
Review Data

Row Count:  Refresh

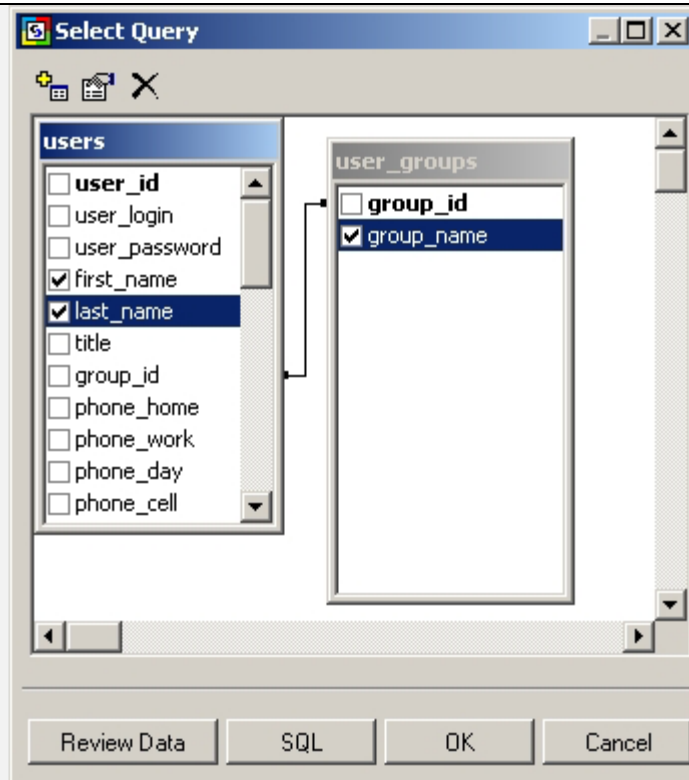
country_id	country_name
1	United States
2	Canada
3	United Kingdom
4	Afghanistan
5	Albania
6	Algeria
7	American Sam...
8	Andorra
9	Angola
10	Anguilla

Close

You can specify the number of records returned by entering a value in the Row Count field then clicking on the Refresh button to retrieve the records.

Data Source	<p>The layout and options in the Data Source portion of the window vary depending on the value of the Data Source Type field. We shall first examine the case where a Table is being used as the data source.</p> 
Table/View	Select the database table or view from where the data will be retrieved.
Select Query	If the form requires data that resides in two or more tables, it is necessary to construct a SQL statement that picks the required fields from the concerned tables. Clicking on this button will open the <a href="#">Select Query window</a> shown below where you can construct the SQL query.

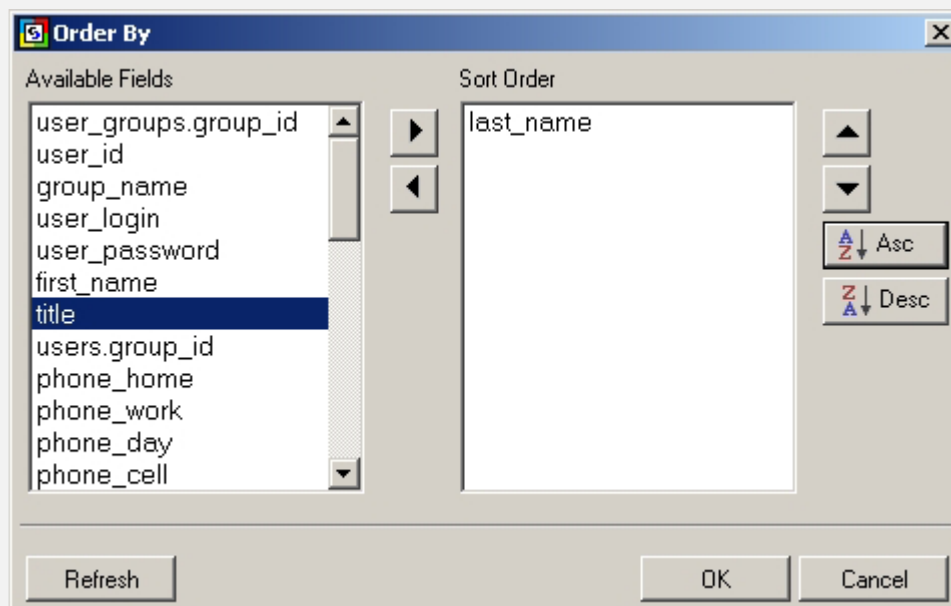



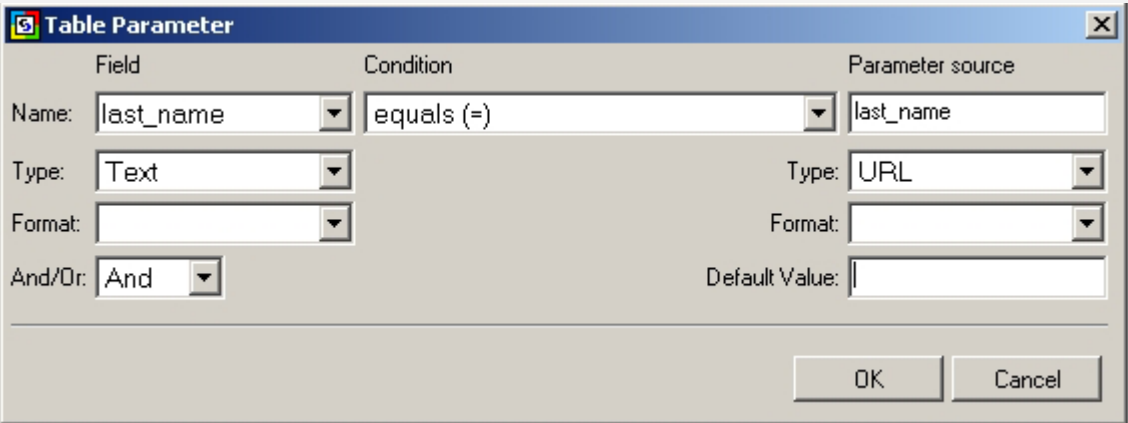








For more information on how to use the **Select Query** window, please refer to the section entitled [Using the Select Query](#) window.

**AND/OR** With either the AND or OR operator selected, you can add a condition that would form part of a valid where clause. For instance, this could involve the use of a function or a condition with a static value.

**Order By** Use the adjacent button to select fields by which the data should be ordered. You can specify whether the field should be sorted in ascending or descending order.



Where	The options in this portion of the window are used to construct the where clause of the SQL statement.
	<p>Click this button to add a parameter to the where clause. Please refer to the section on <a href="#">Using the Table Parameter window</a> for more information.</p> 
	Click this button to remove a parameter from the where clause.
	Click this button to view the configuration of an existing where clause parameter.
	Click this button to move an existing parameter upwards in the order or appearance.
	Click this button to move an existing parameter downwards in the order or appearance.
	Click this button to enclose a parameter definition within brackets. You can group together multiple parameters by holding down the <i>ctrl</i> key while clicking on them then click on the button to place a bracket around all the selected parameters.
	Click this button to remove existing brackets around a parameter definition or multiple parameters.
Data Source	We now turn to the options available in the <b>DataSource</b> window when 'Procedure' is selected in the Data Source Type Field.



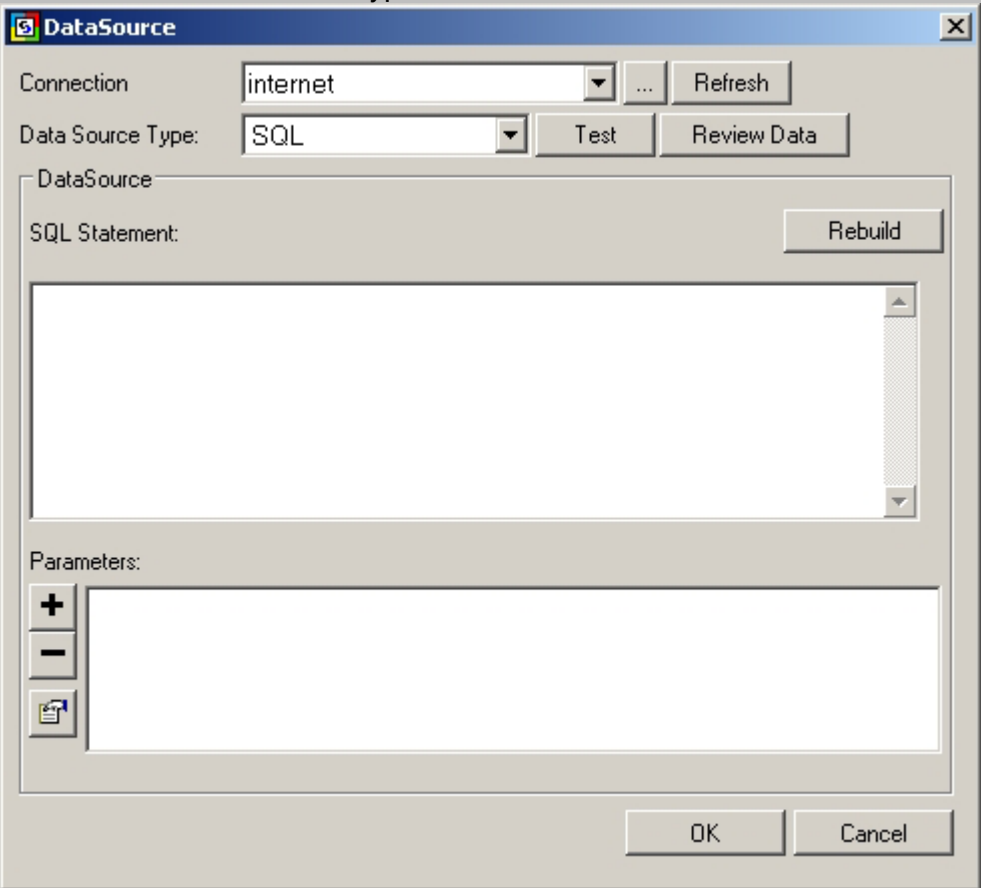

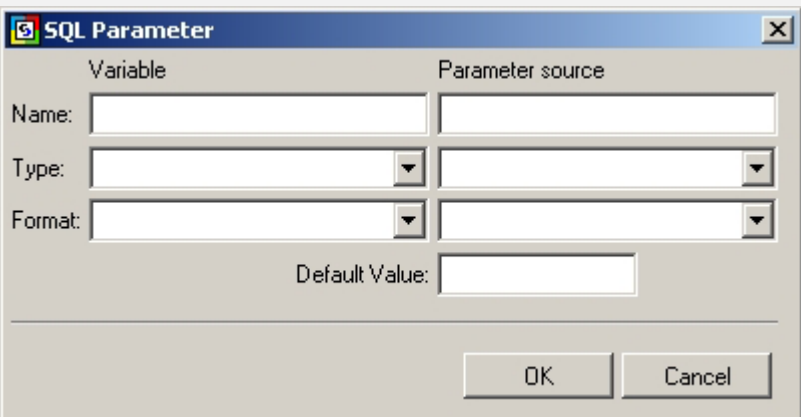
Procedure Select the stored procedure to be used as the data source.



Get Parameters

**+** Click this button to add a stored procedure parameter. Please refer to the section on [Using the Stored Procedure Parameter window](#) for more information.

**-** Click this button to remove an existing stored procedure parameter.

Click this button to view the configuration of an existing stored procedure parameter.

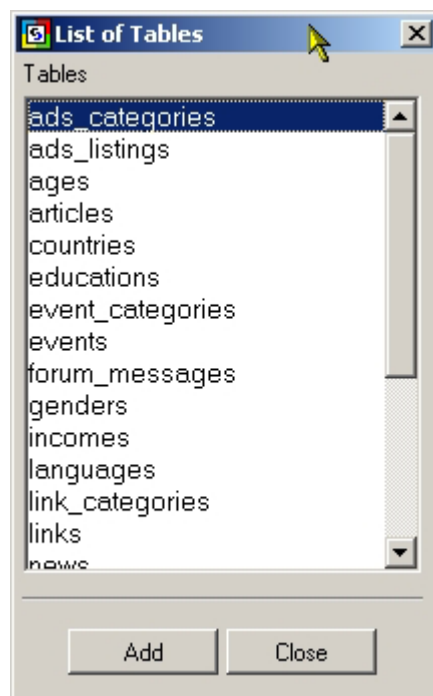
	Click this button to move a stored procedure parameter upwards in the order of appearance.
	Click this button to move a stored procedure parameter downwards in the order of appearance.
Data Source	<p>Finally, the following are the options available in the <b>DataSource</b> window when 'SQL' is selected in the Data Source Type Field.</p> 
SQL Statement	The SQL statement to be used as the data source for the form.
Rebuild	
	<p>Click this button to add a SQL parameter. Please refer to the section on <a href="#">Using the SQL Parameter Window</a> for more information.</p> 

	Click this button to remove an existing SQL parameter.
	Click this button to view the configuration of an existing SQL parameter.

## Using the Select Query Window

When using the wizards to create a grid form or during the process of configuring the Data Source property for a grid form, you might need to use the **Select Query** window to create a SQL query based in multiple tables. The options available in this window are designed to be relatively intuitive and easy to use compared to the task to writing a join SQL query from scratch. This section explains how to use the **Select Query** window to build SQL statement without the hassle.



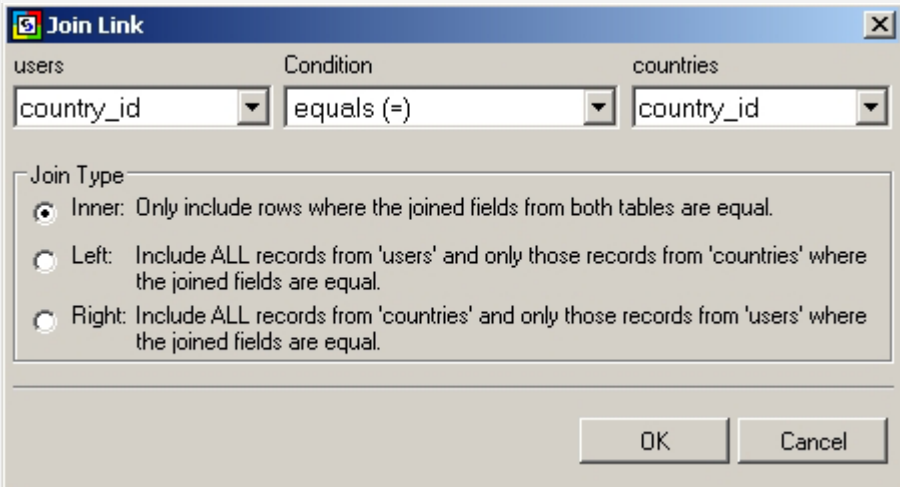

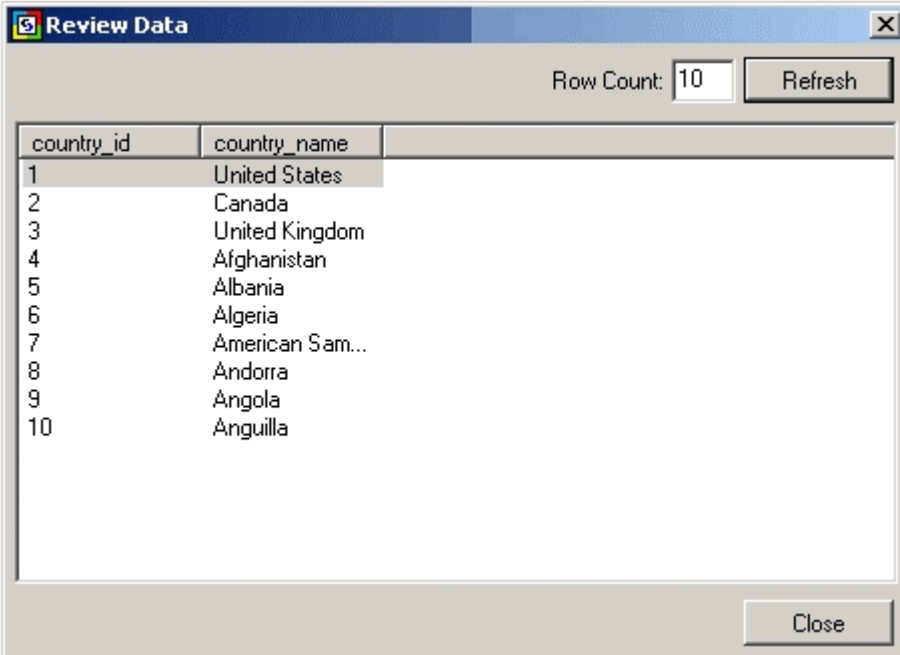
When the **Select Query** window first appears, you are also presented with another smaller window that contains a list of database tables available in the current connection.

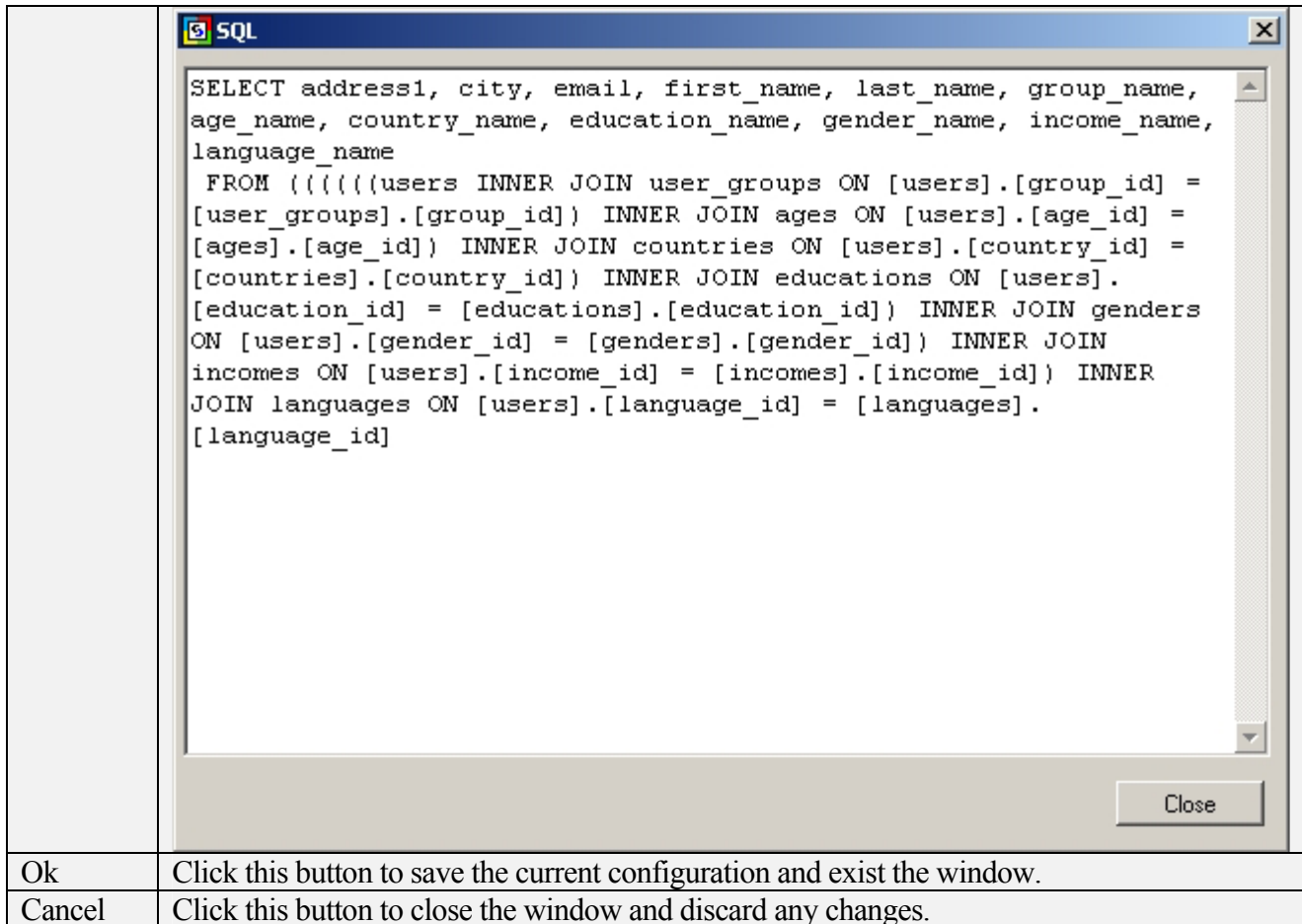


From this list of tables, you should select the tables you want to use in building the SQL query and click on the **Add** button. You can select multiple tables by holding down the *ctrl* key while clicking on the table names. Once done, click on the **Close** button to close the window.

The tables you selected appear within the **Select Query** window, each in a small window containing a list of the fields in the table. Depending on how the fields in the different tables are named, you might notice that some tables are joined together by lines. These lines represent relationships between the tables. The **Select Query** window automatically creates these lines if it encounters two tables with fields that are potential primary and foreign key fields based on their name and data type.

There are also a number of buttons in the **Select Query** window whose functions are described below:

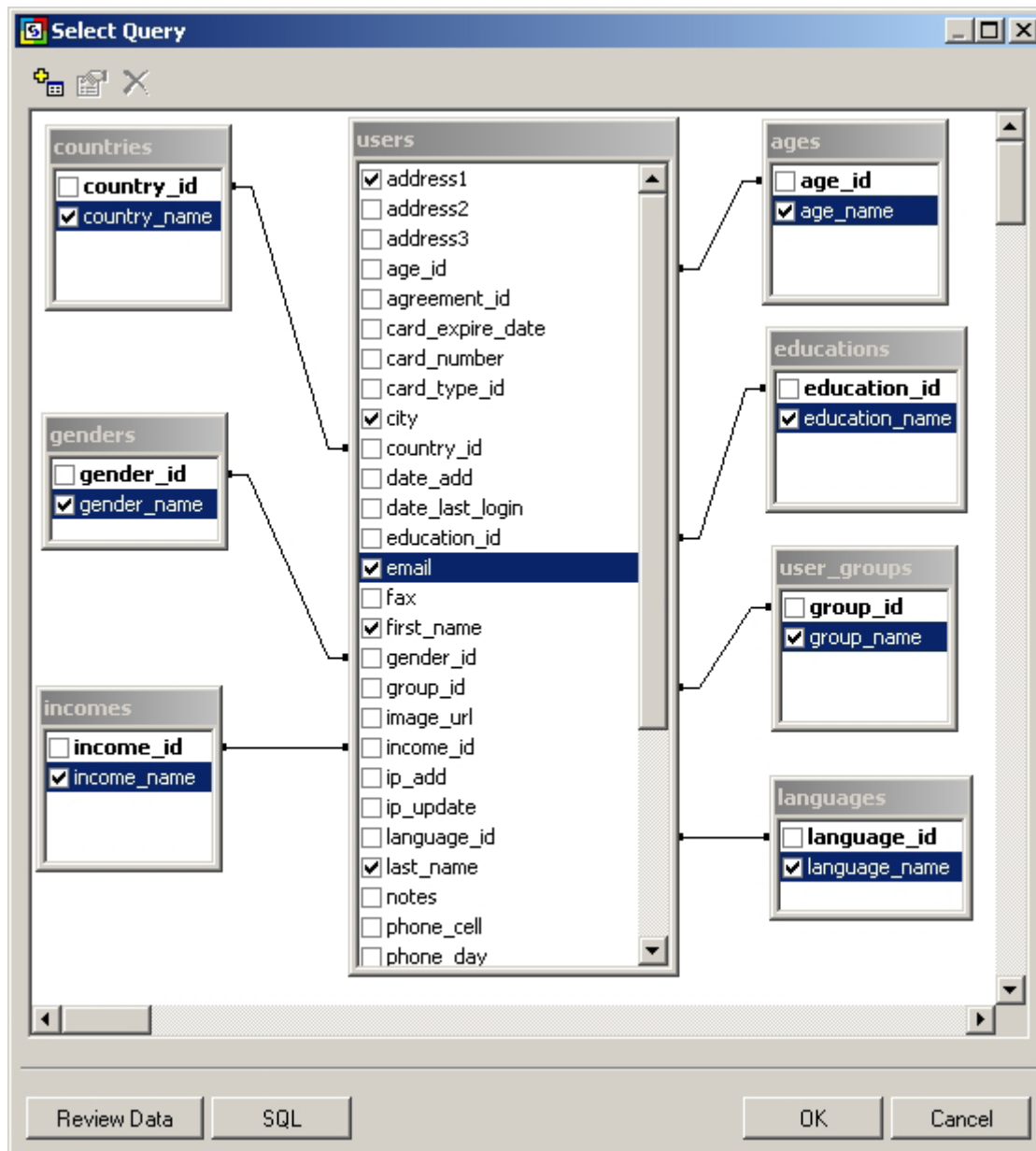
Property	Description
	This buttons displays the list of tables in case you need to add a table.
	<p>Click this button to display the <b>Join Link</b> window that contains the properties of a join. Joins are the lines that appear between two tables to represent a relationship between the tables.</p> 
	With a table or join selected, click this button to delete the table or join.
Review Data	<p>Clicking this button displays the <b>Review Data</b> window that shows sample data based on the SQL query configured in the <b>Select Query</b> window. You can determine the amount of data displayed by entering a value in the Row Count field then clicking the Refresh button.</p> 
SQL	This button displays the SQL statement corresponding to the table and join setup in the <b>Select Query</b> window.



After adding tables to the **Select Query** window, you then proceed to create join links between the different tables.

- Some join links are automatically created based on matching field names and data type. If these relationships are wrong or you don't need them, you can delete them by right clicking on them and selecting the **Delete Link** option.
- You can create new join links by dragging a field from one table to another field in a different table. Once the join link appears between the tables, you can use the properties button to confirm the joined fields as well as set the join type.
- Notice that all the fields in the tables have checkboxes appearing before them. If the corresponding checkbox for a field is activated, then the field will be one of the fields included in the select statement. As such, any fields that you need to use in the form must be checked. You can use the **Review Data** window to confirm that you have selected all the required fields.

The illustration below shows the **Select Query** window with a number of tables with joins links between them. You can maximize the **Select Query** window as well as move the individual tables to achieve a better visualization of the relationships.



### ***Using the Table Parameter Window***

The **Table Parameter** window is used to define input parameters when a table is being used as the data source. Input parameters are used in the where clause of the SQL statement to determine the records to be retrieved from the table. For instance, the values submitted by a search form are received as input parameters by the corresponding grid or record form and subsequently used to retrieve records based on the search criteria.



**Table Parameter**

Field	Condition	Parameter source
Name: last_name	equals (=)	last_name
Type: Text		Type: URL
Format:		Format:
And/Or: And		Default Value:

OK Cancel

Property	Description
Name	Select the name of the database table field that will be matched against the input parameter.
Type	Select the data type of the field: <ul style="list-style-type: none"> <li>Text</li> <li>Integer</li> <li>Float</li> <li>Date</li> <li>Boolean</li> <li>Memo</li> </ul>
Format	Specify the format of the field if applicable e.g. a Boolean field could have the format 0;1 or Y;N
And/Or	Specify whether the field should be included in the where clause using the 'AND' operator or the 'OR' operator.
Condition	Specify the condition to be used to match the field value and the parameter value: <ul style="list-style-type: none"> <li>equals (=)</li> <li>does not equal (&lt;&gt;)</li> <li>less than (&lt;)</li> <li>less than or equal (&lt;=)</li> <li>greater than (&gt;)</li> <li>greater than or equal (&gt;=)</li> <li>begin with (like '...')</li> <li>does not begin with (not like '...%')</li> <li>end with ('%...')</li> <li>does not end with (not like '%....')</li> <li>contains (like '%;..%')</li> <li>does not contain (not like '%...%')</li> <li>is null (is null)</li> <li>not null (is not null)</li> </ul>
Parameter Source	Specify the name of the input parameter.
Type	Select the source of the parameter: <ul style="list-style-type: none"> <li>Control</li> </ul>

	<ul style="list-style-type: none"> <li>• Expression</li> <li>• URL</li> <li>• Form</li> <li>• Session</li> <li>• Application</li> <li>• Cookie</li> <li>• DataField</li> </ul>
Format	Specify the format of the parameter value if applicable e.g. a Boolean field could have the format True/False or Yes/No.
Default Value	Specify a default value to be used incase no value is received from the parameter.

### ***Using the Stored Procedure Parameter Window***

The **Stored Procedure Parameter** window is used to define input parameters when a stored procedure is being used as the data source.

Property	Description
Parameter name	Specify the name of the stored procedure parameter
Data Type	Select the data type of the stored procedure parameter
Data Size	If applicable, specify the data size of the stored procedure parameter
Direction	
Format	If applicable, specify the format of the stored procedure parameter e.g. a Boolean parameter could have the format true;false.
Parameter source	The name of the input parameter.
Parameter Type	Select the type of the input parameter. <ul style="list-style-type: none"> <li>• Expression</li> <li>• URL</li> <li>• Form</li> <li>• Session</li> </ul>

	<ul style="list-style-type: none"> <li>• Application</li> <li>• Cookie</li> </ul>
Format	If applicable, specify the format of the input parameter value e.g. a Date could have the format mm/dd/yyyy.
Default Value	Specify a default value to be used incase no value is received from the input parameter source.

## Using the SQL Parameter Window

The **SQL Parameter** window is used to define input parameters when a SQL query is being used as the data source.

Property	Description
Variable	
Name	Specify the name of the SQL variable/field.
Type	Select the data type of the variable/field: <ul style="list-style-type: none"> <li>• Text</li> <li>• Integer</li> <li>• Float</li> <li>• Date</li> <li>• Boolean</li> <li>• Memo</li> </ul>
Format	If applicable specify the format of the variable/field e.g. a Boolean parameter could have the format true;false.
Parameter Source	
Name	Specify the name of the input parameter source.
Type	Select the type of the input parameter. <ul style="list-style-type: none"> <li>• Expression</li> <li>• URL</li> <li>• Form</li> <li>• Session</li> <li>• Application</li> <li>• Cookie</li> </ul>

Format	If applicable, specify the format of the input parameter value e.g. a Boolean field could have the format True/False or Yes/No.
Default Value	Specify a default value to be used incase no value is received from the input parameter source.

## V. Controls and HTML Elements

In the process of creating a page, various controls and HTML elements are added to the page. Some of the controls are used to expose dynamic data that is retrieved from the database while others are purely functional such as buttons.

The CCS Toolbox provides a convenient way to add controls and HTML elements to a page. Before delving into the details of these components, it is important to understand the difference between **Controls** and **HTML Elements**.

When viewed using a browser, Controls are no different from HTML elements. The distinction between the two exists only within CCS. Of the two, HTML elements are simpler than Controls and are made up entirely of HTML markup. When you add a HTML element to a page, the necessary HTML code is added to the page but you don't see the element listed in the Project Explorer window.


Controls are more advanced than HTML elements and some such as the Navigator and Sorter controls are composed of different types of HTML elements. Controls can be programmed dynamically using the programming language and they have properties that can be changed programmatically to alter their behavior. When you add a control to a page, the name of the control appears in the Project Explorer window.

### HTML Elements


The HTML elements are listed under the HTML tab of the CCS Toolbox. During the process of designing your pages, you can add HTML elements to the page as needed. After adding a HTML element, you can further customize it using the options available under the **Format** tab of the **Properties window**. You can configure a number of HTML properties or specify actions to be performed when certain events occur.

The following are the available HTML elements.

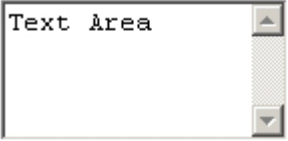
#### Text Box

<b>HTML</b>	<code>&lt;INPUT TYPE="text" ID="TextBox2" NAME="TextBox2"&gt;</code>
<b>Image</b>	
<b>Description</b>	A text box is used to provide single line input. It is usually used when a small amount of data is to be entered.


#### Password

<b>HTML</b>	<code>&lt;INPUT TYPE="password" ID="Password1" NAME="Password1"&gt;</code>
<b>Image</b>	
<b>Description</b>	This is very similar to the Text Box except that all characters entered are masked using asterisks. Most login forms will contain a password field where the user is expected to enter the password.


## Text Area

HTML	<code>&lt;TEXTAREA ID="TextArea1" NAME="TextArea1" ROWS="4" COLS="15"&gt;Text Area&lt;/TEXTAREA&gt;</code>
Image	
Description	A Text Area field is used when a large amount of data needs to be entered. It is possible to set the number of rows as well as columns to be displayed. The control also has scrollbars to allow adding text beyond the specified rows and columns.


## Checkbox

HTML	<code>&lt;INPUT type="checkbox" id="Checkbox1" name="Checkbox1"&gt;</code>
Image	
Description	A checkbox is used to toggle the selection of an option. It can otherwise be viewed as a Boolean or On/Off control.

## Radio Button


HTML	<code>&lt;INPUT TYPE="radio" ID="Radio1" NAME="Radio1"&gt;</code>
Image	
Description	Radio button controls are used for the selection of mutually exclusive options. A number of radio buttons with the same name are presented and out of these, the user can select only one.

## Dropdown

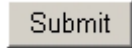
HTML	<code>&lt;SELECT ID="Select1" NAME="Select1"&gt;&lt;OPTION&gt;Select1&lt;/OPTION&gt;&lt;/SELECT&gt;</code>
Image	
Description	A Dropdown is used to present a list of options from which the user can select one. The user clicks on the dropdown arrow to view all the options available.

## Listbox

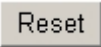
HTML	<code>&lt;SELECT ID="Select1" NAME="Select1" SIZE="3"&gt;&lt;OPTION&gt;Select1&lt;/OPTION&gt;&lt;/SELECT&gt;</code>
------	---

<b>Image</b>	
<b>Description</b>	A Listbox is essentially the same as a dropdown with the distinction of being able to display multiple options. With a listbox, it is also possible for the user to select more than one option by specifying the Multiple property.

## Submit Button

<b>HTML</b>	<code>&lt;INPUT TYPE="submit" ID="Submit1" NAME="Submit1" VALUE="Submit"&gt;</code>
<b>Image</b>	
<b>Description</b>	When placed within a HTML form, a submit button serves the purpose of submitting all the fields within the form to the designated action page.


## Reset Button

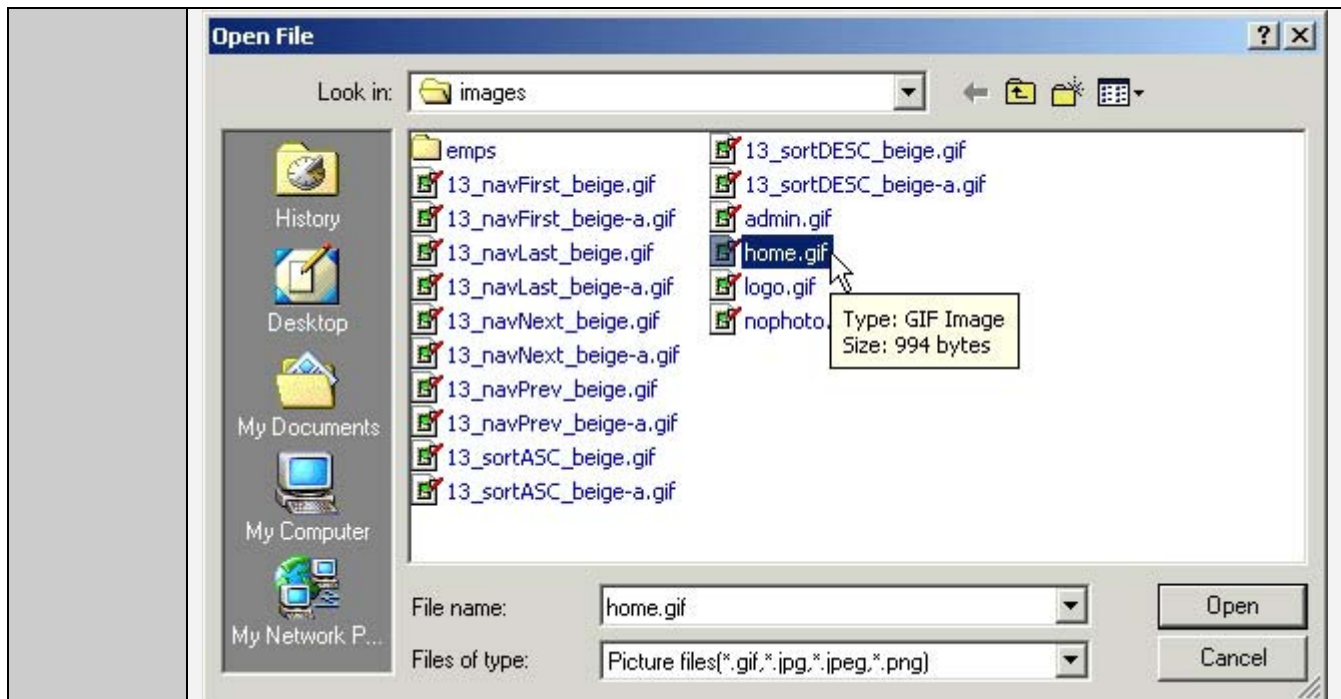
<b>HTML</b>	<code>&lt;INPUT TYPE="reset" ID="Reset1" NAME="Reset1" VALUE="Reset"&gt;</code>
<b>Image</b>	
<b>Description</b>	A Reset button is used to return the fields in a form to the state they originally were in when the page loaded. For instance, if a field was empty then a value was entered, clicking on the Reset button would clear the value.

## Label

<b>HTML</b>	<code>&lt;label&gt;&lt;/label&gt;</code>
<b>Description</b>	As the name suggests, Labels are used to provide text that identifies other elements within the page. They can otherwise be termed as captions.

## Image Button

<b>HTML</b>	<code>&lt;input type="image" id="ImageButton1" name="ImageButton1" src="images\home.gif"&gt;</code>
<b>Image</b>	
<b>Description</b>	This is the graphical equivalent of the Submit Button. When you click on the Image Button option in the <b>Toolbox</b> , a dialog box appears where you can select the image to be used to represent the button.



## Button

HTML	<code>&lt;INPUT TYPE="button" ID="Button1" NAME="Button1" VALUE="Button1"&gt;</code>
Image	
Description	A Button is a control that performs a specified action when clicked. Usually, the OnClick event of the button is assigned some script code that executes the desired function.

## Horizontal Rule


HTML	<code>&lt;HR&gt;</code>
Image	
Description	This is simply a customizable line often used to demarcate sections of a page to make their difference more discernable.

## Line Break

HTML	<code>&lt;BR&gt;</code>
Image	(only visible in Design Mode)
Description	A Line Break forcibly ends the current line of text and continues the flow of content on the next line. A Line Break is defined by a carriage return or a line feed. When in <b>Design</b> mode, you can enter a Line Break by clicking on the <i>shift+Enter</i> keys together.




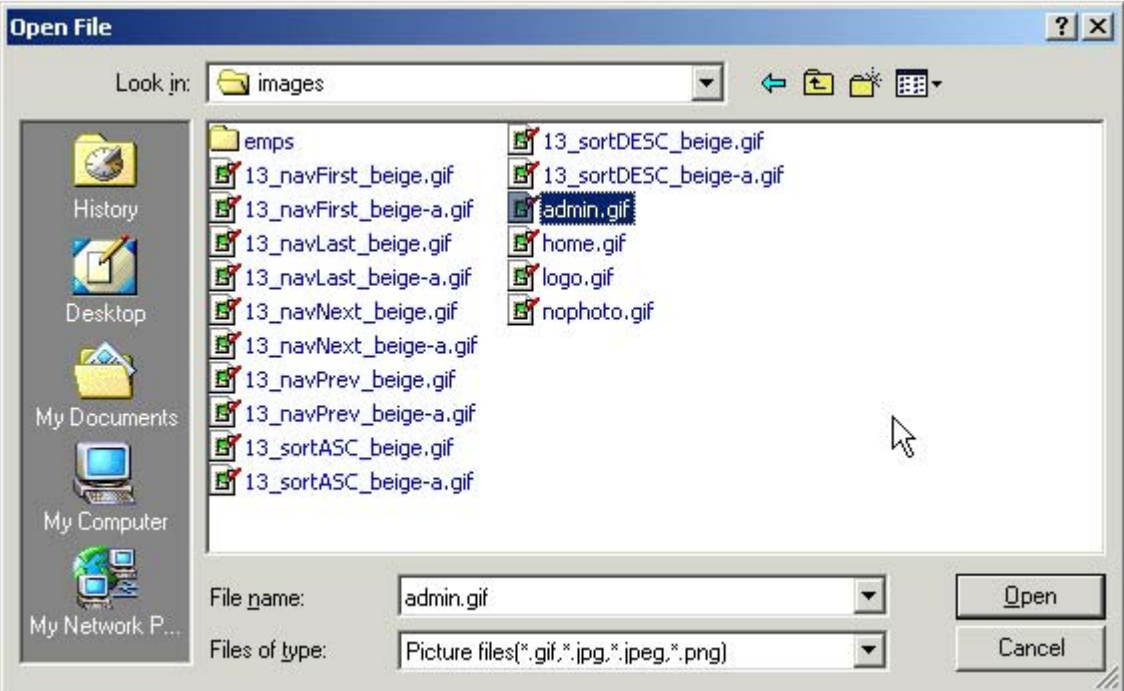
## Paragraph Break

HTML	<P></P>
Image	 (only visible in Design Mode)
Description	A Paragraph Break represents a paragraph and has the effect of placing an empty space between two adjacent paragraphs.

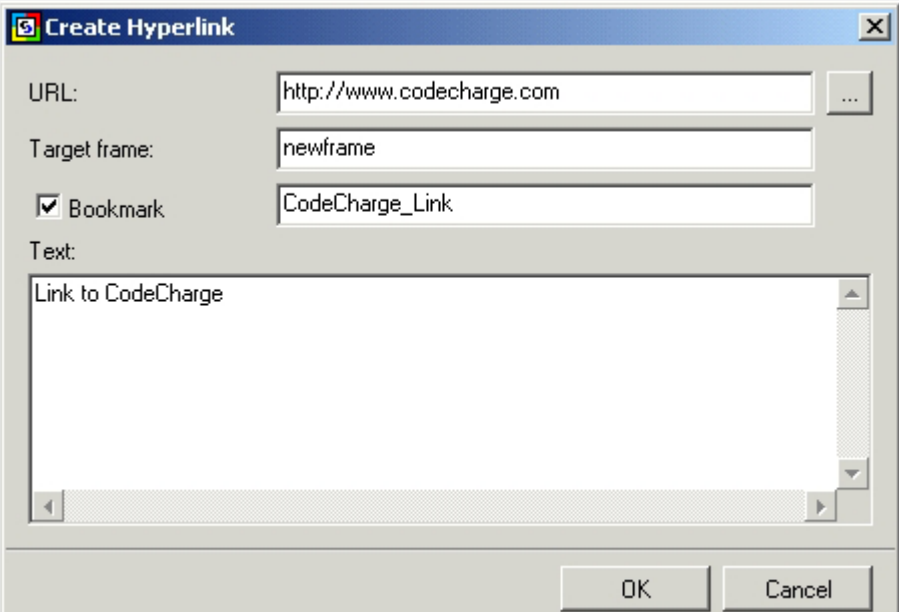
## Space

HTML	&nbsp;
Description	This is a non-breaking space meaning that a line break cannot occur in its place. It is used for instance when you don't want a line break to occur between two words.

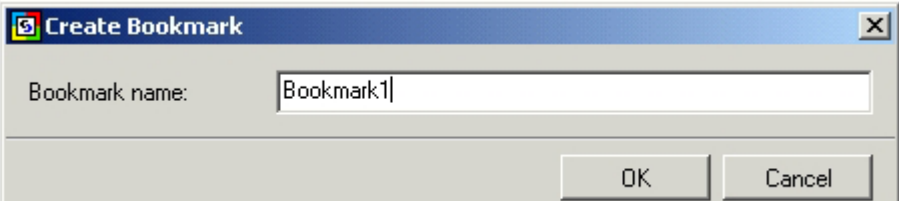
## Image

HTML	<IMG ID="Image1" NAME="Image1" SRC="images\admin.gif">
Image	
Description	<p>This is used to embed an image in the current page. When you click on the image option in the <b>Toolbox</b>, a dialog box appears where you can navigate and select the image file to be used.</p> 


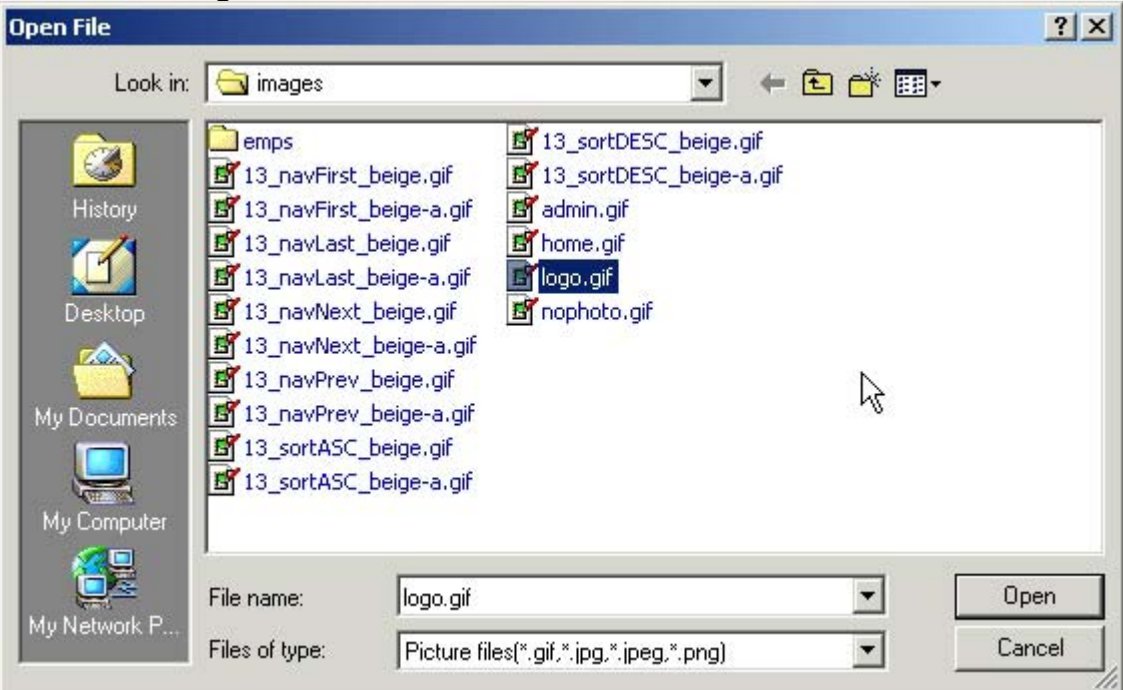
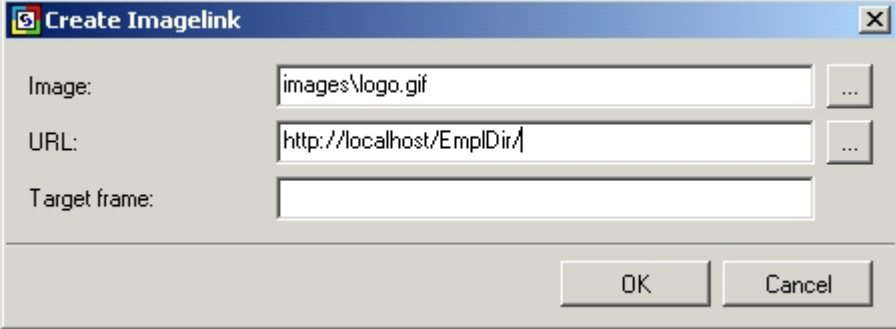
## Hyperlink

HTML	<code>&lt;a name="CodeCharge_Link" href="http://www.codecharge.com" target="newframe"&gt;Link to CodeCharge&lt;/a&gt;</code>
Image	<a href="http://www.codecharge.com">Link to CodeCharge</a>
Description	<p>This is a link to another resource:</p> <ul style="list-style-type: none"><li>• <b>URL:</b> Specify the location of the resource to be linked to</li><li>• <b>Target Frame:</b> If using frames, enter the name of the frame where you want the linked resource to be opened.</li><li>• <b>Bookmark:</b> If want the link to also serve as a named anchor, enter a unique name to identify it.</li><li>• <b>Text:</b> This is the text that the user will see and click on.</li></ul> 

## Bookmark

HTML	<code>&lt;a name="Bookmark1"&gt;&lt;/a&gt;</code>
Description	<p>A Bookmark is essentially a named anchor that can be the destination of another link. Anchors are often used in long pages so that the user can conveniently jump to designated positions within the document by using links to the anchors.</p> 

## ImageLink

<b>HTML</b>	<code>&lt;a href="http://www.codecharge.com"&gt;&lt;img src="images\logo.gif" border="0"&gt;&lt;/a&gt;</code>
<b>Image</b>	 <p>The image shows a logo for 'Employee Directory'. It features a yellow folder icon with a magnifying glass on the left, followed by the word 'Employee' in a large, bold, white font, and 'Directory' in a smaller, bold, white font, all on a yellow rectangular background.</p>
<b>Description</b>	<p>This is very similar to creating a HyperLink except that instead of having text to represent the link, an image is used.</p> <p>When you click on the ImageLink option in the Toolbox, a dialog box appears where you can select the image to be used.</p>  <p>The 'Open File' dialog box shows the 'images' folder selected in the 'Look in:' field. The file list contains various GIF files, with 'logo.gif' highlighted. The 'File name' field at the bottom shows 'logo.gif' and the 'Files of type' dropdown is set to 'Picture files (*.gif;*.jpg;*.jpeg;*.png)'. 'Open' and 'Cancel' buttons are at the bottom right.</p> <p>After selecting an image, you then have to specify the URL of the resource to be linked to. If applicable, you can also enter the name of a frame where the link will be opened.</p>  <p>The 'Create Imagelink' dialog box has three input fields: 'Image:' with 'images\logo.gif', 'URL:' with 'http://localhost/EmpDir/', and 'Target frame:'. 'OK' and 'Cancel' buttons are at the bottom.</p>

## Controls

Controls are dynamic elements placed on a page and are often used to display database content. Controls play a central role in any given form that deals with database content. Controls and their values can also be controlled programmatically.

In their stripped down form, Controls are pretty much HTML elements albeit with more functionality due to the server side code that interacts with them. In this section, the various Controls available in the Toolbox will be introduced. For more information on how to program Controls using actions and custom code as well as a look at the code that runs the Controls, please refer to the Reference Manual.

### Label

<b>HTML</b>	{Label1}
<b>Image</b>	{Label1}
<b>Description</b>	<p>A label field is probably the simplest of the Controls although its flexibility allows it to be used to represent a wide variety of content. It is important to understand that Label fields are not HTML controls such as Text Box or Checkbox fields. A Label is simply plain text and therefore cannot be used to submit a value in a form.</p> <p>Due to their read-only nature, Labels are used to present content that should not be altered. As such, grid forms are mainly composed of Label fields.</p>

### Link

<b>HTML</b>	<a href="{Link1_Src}">{Link1}</a>
<b>Image</b>	<a href="#">{Link1}</a>
<b>Description</b>	This creates a hyperlink to a resource whose URL comes from a database column or expression.

### TextBox

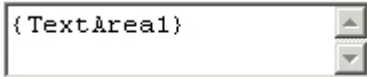
<b>HTML</b>	<input value="{TextBox1}" name="TextBox1">
<b>Image</b>	<input type="text" value="{TextBox1}"/>
<b>Description</b>	A text box is used to provide single line input. It is usually used when a small amount of data is to be entered.

### Hidden


<b>HTML</b>	<input type="hidden" value="{Hidden1}" name="Hidden1">
<b>Image</b>	<input type="text" value="{Hidden1}"/>

<b>Description</b>	A Hidden field is the equivalent of a Textbox except that a textbox has no visual representation on the rendered page. Unless the user views the source HTML for a page, there is no visual indication of the existence of a Hidden field.
--------------------	--

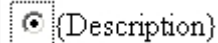
## TextArea

<b>HTML</b>	<code>&lt;textarea name="TextArea1"&gt;{TextArea1}&lt;/textarea&gt;</code>
<b>Image</b>	
<b>Description</b>	A Text Area field is used to when a large amount of data needs to be entered. It is possible to set the number of rows as well as columns to be displayed. The control also has scrollbars to allow adding text beyond the specified rows and columns.

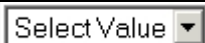
## Checkbox

<b>HTML</b>	<code>&lt;input type="checkbox" name="Checkbox1" value="1" {Checkbox1}&gt;</code>
<b>Image</b>	
<b>Description</b>	A checkbox is used to toggle the selection of an option. It can otherwise be viewed as a Boolean or On/Off control.

## RadioButton

<b>HTML</b>	<code>&lt;!-- BEGIN RadioButton RadioButton1 --&gt;&lt;input {Check} type="radio" name="RadioButton1" value="{Value}"&gt;{Description}&lt;!-- END RadioButton RadioButton1 --&gt;</code>
<b>Image</b>	
<b>Description</b>	Radio button controls are used for the selection of mutually exclusive options. A number of radio buttons with the same name are presented and out of these, the user can select only one.

## ListBox


<b>HTML</b>	<code>&lt;select name="ListBox1"&gt;&lt;option value=""&gt;Select Value&lt;/option&gt;{ListBox1_Options}&lt;/select&gt;</code>
<b>Image</b>	
<b>Description</b>	A Listbox is essentially the same as a dropdown with the distinction of being able to display multiple options. With a listbox, it is also possible for the user to select more than one option by specifying the Multiple property.

## Image

<b>HTML</b>	<code>&lt;img src="{Image1}"&gt;</code>
-------------	---

<b>Image</b>	
<b>Description</b>	This is used to embed an image in the current page.


## ImageLink

<b>HTML</b>	<code>&lt;a href="{ImageLink1}"&gt;&lt;img src="{ImageLink1_Src}" border="0"&gt;&lt;/a&gt;</code>
<b>Image</b>	
<b>Description</b>	This is similar to an ordinary hyperlink but instead of text being used to represent the link, an image is used.

## Include Page

<b>HTML</b>	<code>{IncludePage1}</code>
<b>Image</b>	<code>{IncludePage1}</code>
<b>Description</b>	Certain pages can be created specifically for being included in other pages. The Include Page control is used to include such pages, a good example of which are the header and footer pages that appear in many sites.

## Button

<b>HTML</b>	<code>&lt;!-- BEGIN Button Button1 --&gt;&lt;input type="submit" value="Button1" name="Button1"&gt;&lt;!-- END Button Button1 --&gt;</code>
<b>Image</b>	
<b>Description</b>	When placed within a HTML form, a submit button serves the purpose of submitting all the fields within the form to the designated action page.

## VI. Forms

Within CodeCharge Studio, forms are the main building blocks of a page. Although possible, it is highly unlikely that you will have a page without any form on it. It is therefore important to understand the mechanics of the various forms used within CodeCharge.

At this early point, it is important to clarify what is meant by the term “Form” in the CodeCharge Studio context. This is mostly to avoid confusion with the HTML Form element that is represented by the tags `<Form></Form>`. In CodeCharge Studio, a form refers to a section of a page containing normal content, markup, labels and optionally may contain a HTML form element with form controls in it. A CodeCharge Studio form is visually demarcated using one or multiple HTML tables with various content appearing in the table rows and columns.

The following are the different types of forms you can build within CodeCharge Studio:

- Grid Form
- Record Form
- Search Form
- Login Form (Authentication)

All of the above forms can be constructed manually by putting together various Controls and HTML elements or alternatively you can use one of the wizards to automate the process.

### **Grid Form**

As the name suggests, grid forms are used to display multiple records of data. In addition to displaying data, a grid form can be enhanced to allow the user to browse through multiple records as well as change the order in which the records are displayed.

Of all the forms that can be created within CodeCharge Studio, the grid form is the only form that does not have a HTML form element by default. This is because grid forms are intended to display data as opposed to altering it. However, it is possible to manually add a Form element to a grid form by editing the HTML code.

The illustration below shows a basic grid form. This form is created when the **Grid** option in the **Forms** tab of the **Toolbox** is clicked. The form has a title, a row for the field captions and another row where the database content is displayed.

NewGrid1

Label1	Label2
{Label1}	{Label2}

## Record Form

Record forms are used to add, edit or delete database content. By default, a record form has a HTML Form element, a number of controls that can submit values as well as four buttons (Insert, Update, Delete and Cancel).

The display of the four buttons on a record form is controlled programmatically such that only the applicable buttons appear at the right time when the page is viewed live. For instance, if no record data is retrieved from the database, only the Insert and Cancel buttons will appear since you can't update or delete a non existent record. On the other hand, if a record is retrieved and displayed in the form controls, the Update, Delete and Cancel buttons are shown but not the Insert button.

The illustration below shows a basic record form. This form is created when the **Record** option in the **Forms** tab of the **Toolbox** is clicked. The form has a title, a column for the field captions and another column with controls where the database content is displayed and can be altered. Notice too that unlike the grid form, a record form has a HTML form element.

The screenshot shows a web form titled "NewRecord1" enclosed in a yellow "form" tag. Inside the form, there is an error message placeholder "{Error}" at the top. Below it are two text input fields: "TextBox1" with a data-binding expression "{TextBox1}" and "TextBox2" with a data-binding expression "{TextBox2}". At the bottom of the form are four buttons: "Insert", "Update", "Delete", and "Cancel". The form is closed with a yellow "/form" tag.

## Search Form

Search forms are used to enter values into controls such that when the values are submitted, they are used to search and retrieve records from the database. Search forms are used in conjunction with grid or record forms whereby the search form is used to submit the search criteria and the grid or record form displays the retrieved records that match the search criteria.

In their simplest form, search forms simply submit search criteria. However, you can use the Search wizard to create complex search forms that also determine the number of records to be displayed per page, the order in which the records are displayed as well as the SQL operator to be used to perform the search.

The illustration below shows a basic search form. This form is created when the **Search** option in the **Forms** tab of the **Toolbox** is clicked. The form has a title, a column for the field captions and another column with controls where search criteria is entered.



```

form NewSearch1
{Error}
TextBox1 {TextBox1}
DoSearch
/form

```

## ***Login (Authentication) Form***

A login or authentication form is used to input authentication details (Login and Password) that are required to grant access to users in sites that implement security.

The illustration below shows a basic Login form that is created using the Authentication wizard in the **Forms** tab of the **Toolbox**.

```

form Login
{Error}
Login {login}
Password *****
Login
/form

```

## ***Using the Wizards to create forms***

In the interest of saving time and effort, the process of creating forms can be greatly automated by using wizards. A wizard is essentially a series of dialogs spanning the process of creating a page or form. The user configures the various options available in the series of dialogs, at the end of which the wizard generates forms based on the selections made.

Generally speaking, you will find it much easier to create forms using the wizards then later on edit the forms as desired, rather than create the forms from scratch manually. This is especially the case if you would like to maintain a consistent look since the wizards offer the option of creating forms based on preset style and image themes.

The following are the different wizards that can be used to create forms:

- Power Wizard
- Page Wizard
- Grid Wizard
- Record Wizard
- Search Wizard
- Authentication Wizard
- Grid and Record Wizard

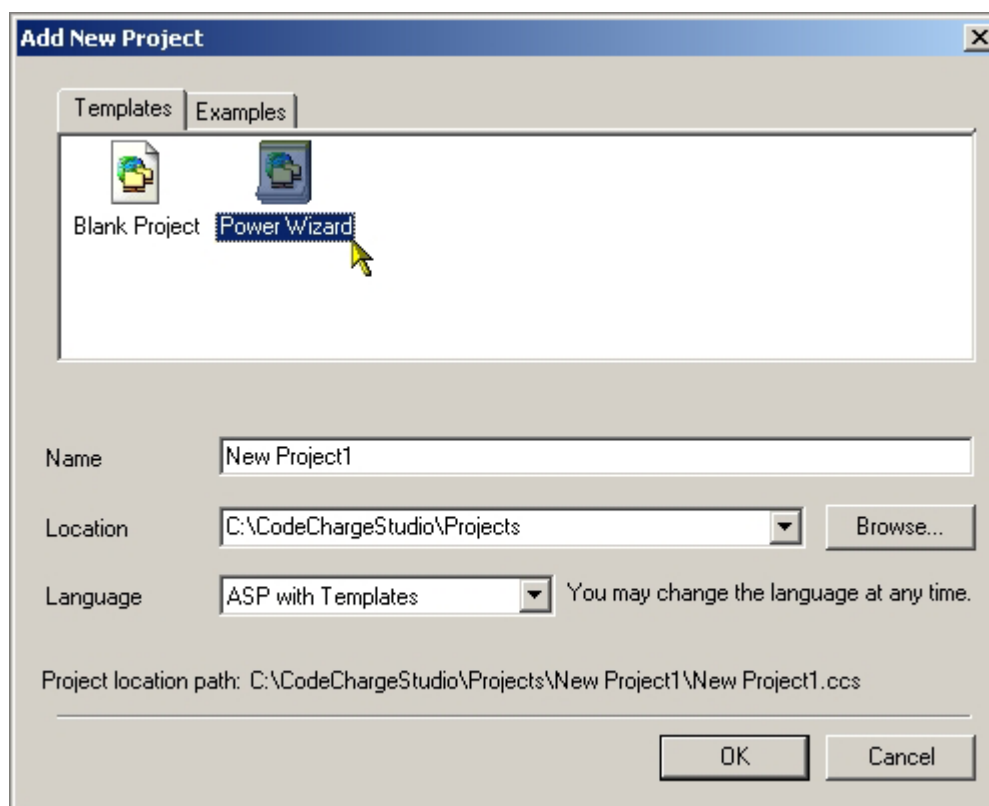
In this section, we shall go through the process of creating different forms using each of the wizards.

## Power Wizard

The Power Wizard can be used to create multiple project pages based on the tables or queries in a designated database connection. In some instances, the Power Wizard can be used to create all the pages needed for a project.

The steps below show how to create multiple project pages using the Power Wizard.

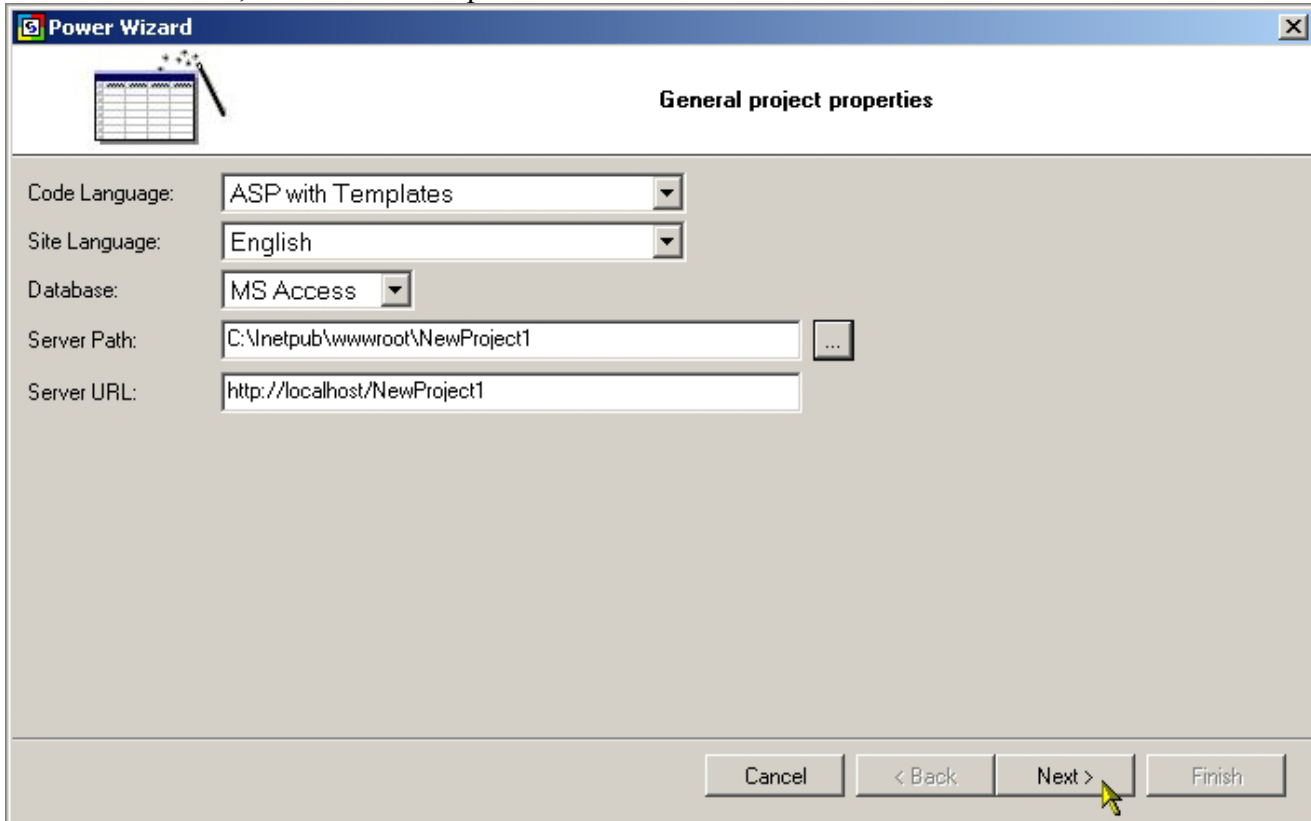
1. To access the Power Wizard, click on the **File** menu and select the **New → Project...** option. The **Add New Project** window pops up where you should select the **Power Wizard** option then click **OK**.



2. In the ensuing window, various project settings need to be set.

Property	Description
Code Language	Select the programming language in which the pages will be generated.
Site Language	Select the spoken language for the project pages
Database	Select the database type to be used. If your intended database type is not listed, select the ANSI SQL-92 option that is a database independent SQL standard.
Server Path	Specify the server path where the pages will be published.
URL Path	Specify the URL that maps to the Server Path specified above. This is the URL you would type into a browser in order to access the pages.

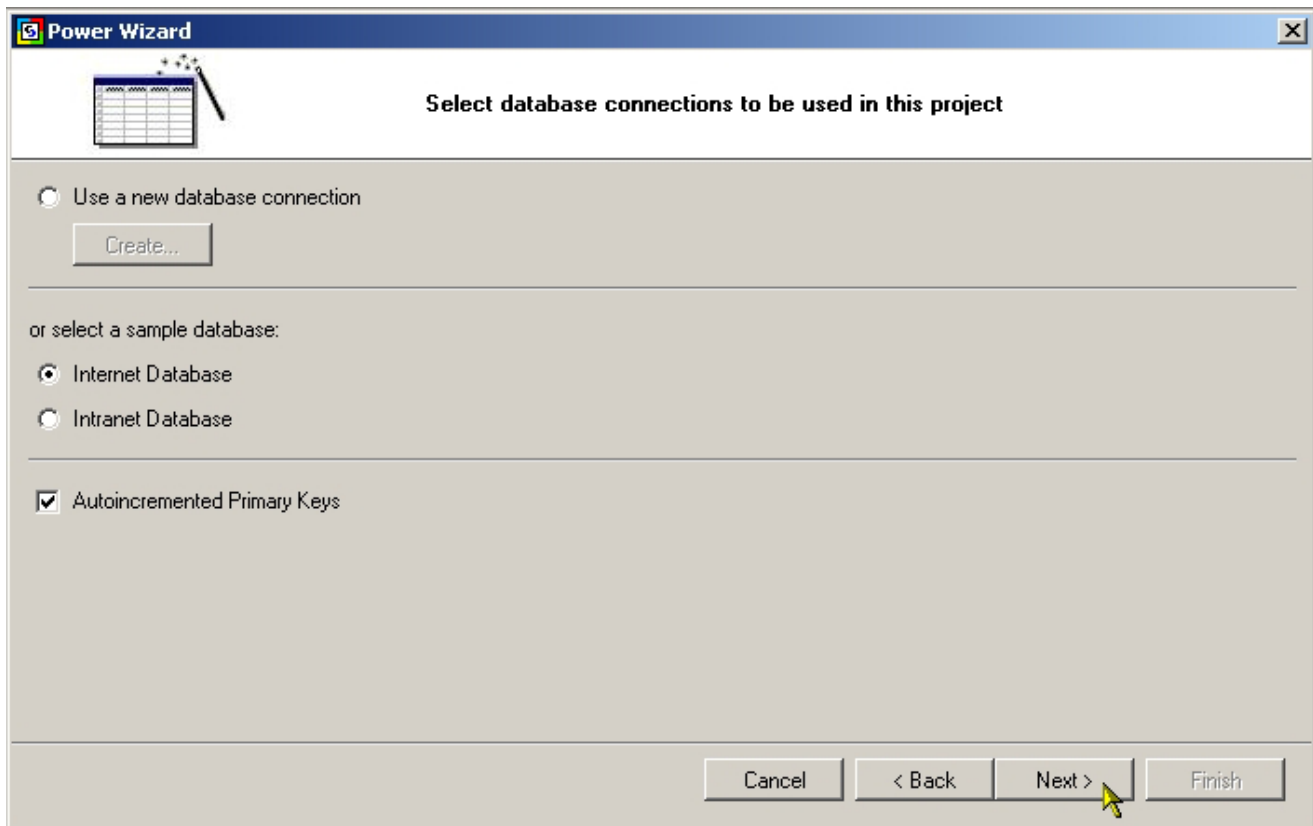
Once done, click on **Next** to proceed.



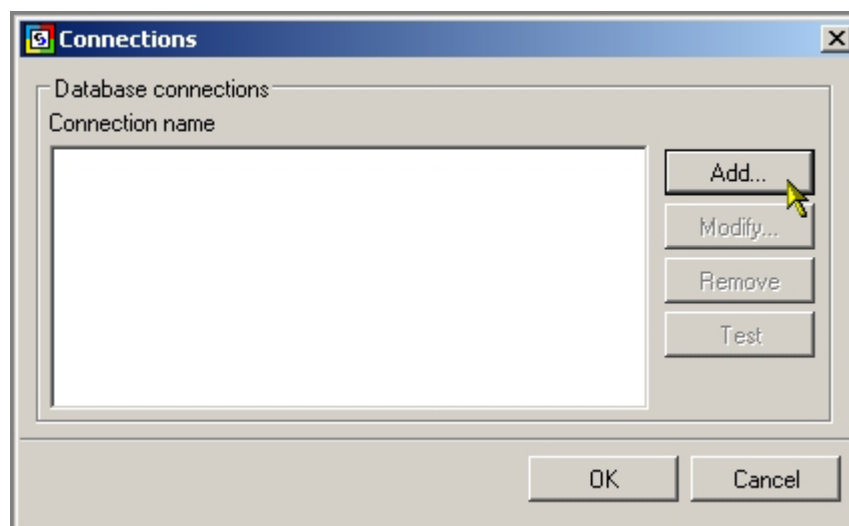
The screenshot shows the 'Power Wizard' window with the title 'General project properties'. It contains several configuration fields: 'Code Language' set to 'ASP with Templates', 'Site Language' set to 'English', 'Database' set to 'MS Access', 'Server Path' set to 'C:\inetpub\wwwroot\NewProject1', and 'Server URL' set to 'http://localhost/NewProject1'. At the bottom right, there are four buttons: 'Cancel', '< Back', 'Next >', and 'Finish'. A yellow mouse cursor is pointing at the 'Next >' button.

3. You now have to configure a new database connection for the project or select one of the sample databases.

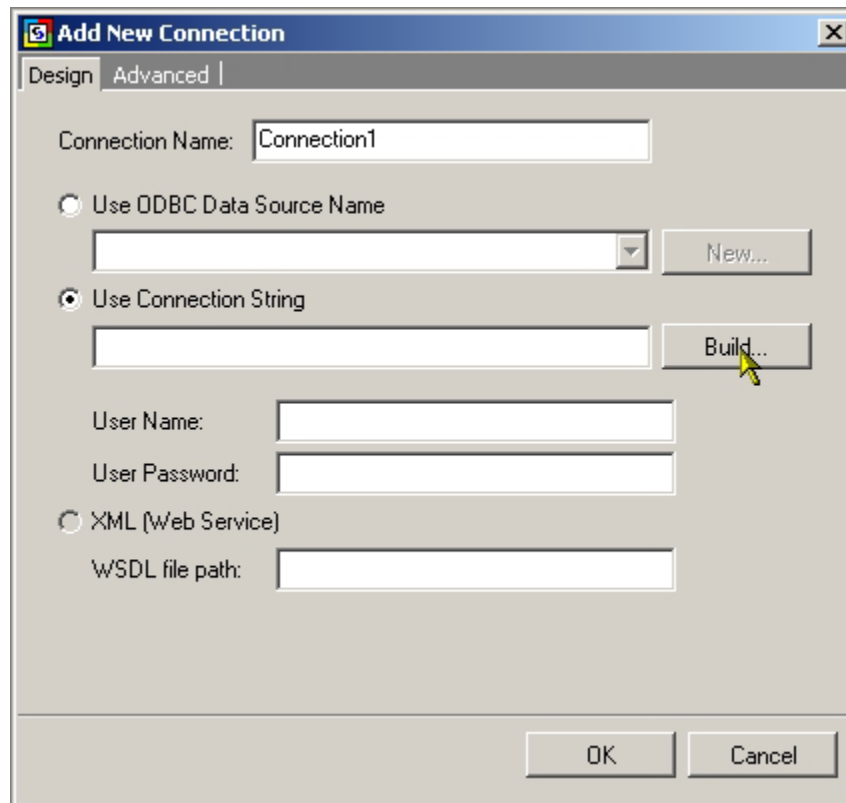
Property	Description
Use a new database connection	Select this option if you want to configure a new database connection. You can then click on the <b>Create...</b> button to create the connection.
Select a sample database connection	Select this option is you want to use one of the sample database connections.
Autoincremented Primary Keys	This option should be checked if the tables in your database have autoincremented primary keys. With autoincremented primary keys, new records are automatically assigned unique primary key values by the database. Note that both of the sample databases have autoincremented primary keys.



If you opted to create a new database connection, the following window will appear upon clicking the **Create...** button. Since there are no database connections already configured, click on the **Add...** button to create a new connection.



The **Add New Connection** window will appear where you can select an [ODBC DSN](#) or [Connection string](#) as the basis of the connection.



Once you have selected or configured a database connection, click **Next** to proceed.

4. The next step involves setting up the security mechanism of the project.

Property	Description
Use authorization	Select this option if you want your pages to have security authorization.
User Table	Select the table where user authorization information is contained.
User ID Field	Select the field with the ID of the users.
Login Field	Select the field with the Login name of the users.
Password Field	Select the field with the user passwords.
Level/Group Field	Select the field with the user security levels.
Security Groups	This table contains a list of all the security groups that registered users can be members of.
Group ID	The ID for the security group. This also serves as the level of the group.
Group Name	The name for the security group.
Higher Level Inclusive of Lower Levels	If this option is selected, users will higher level security groups will be able to access pages with lower security levels e.g. a user in security group 3 will be able to access pages with security levels 2 or 1.
Add...	Click this button to add a new security group.
Modify...	Click this button to modify an existing security group.
Remove	Click this button to remove an existing security group

**Power Wizard**

**Group Access authorization usage**

☒ Use authorization

User Table:

User ID Field:

Login Field:

Password Field:

Level/Group Field:







Security Groups

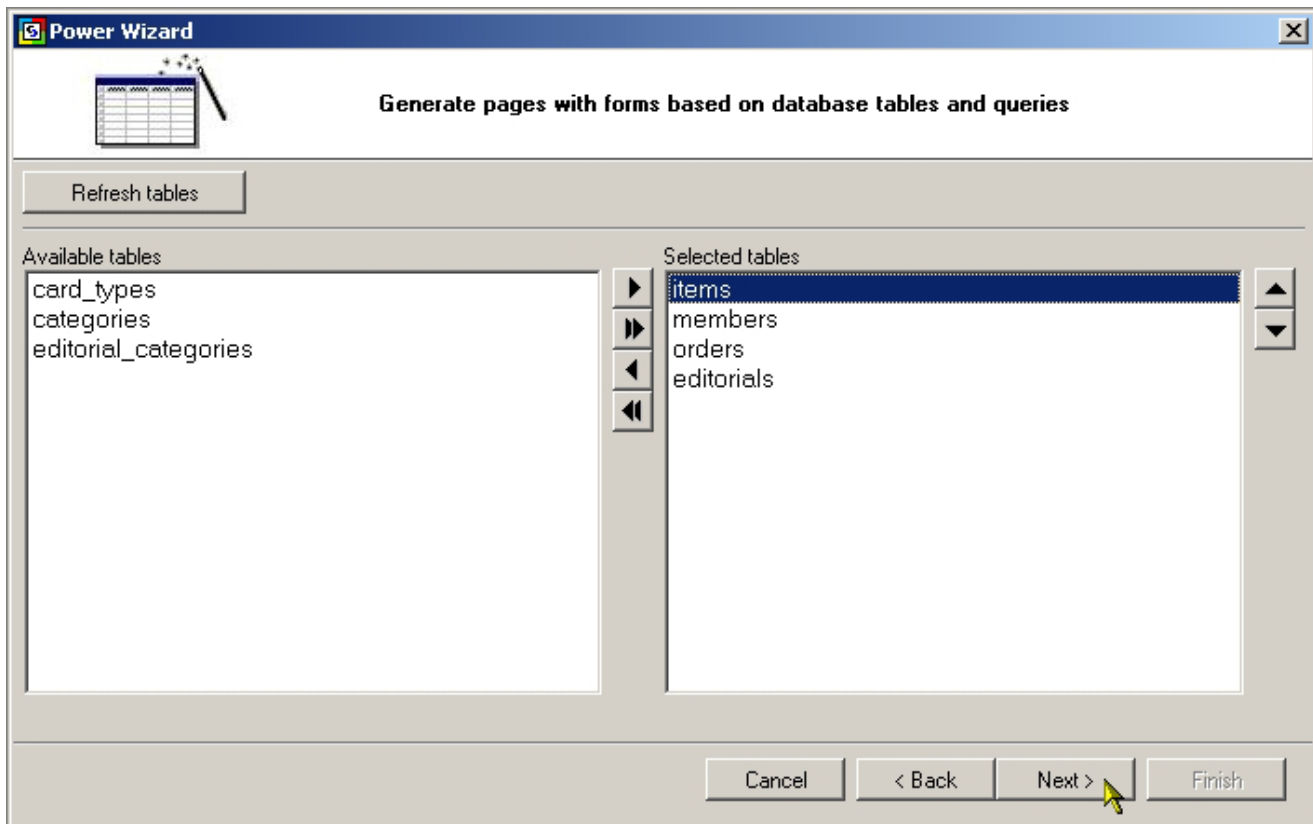
Group ID	Group Name
1	User
2	Admin

☒ Higher Level Inclusive of Lower Levels

Once you are done configuring the security options, click **Next** to proceed.

- In this step, you are presented with a list of the tables in the database and out of these, you select those for which you want to create pages.

Property	Description
Refresh tables	If you make changes to the database during the process of using the wizard, click this button to refresh the connection and retrieve the latest state of the tables.
Available tables	This is a list of the tables available in the database.
Selected table	This is a list of the tables you have selected and based on which you will be able to create pages with forms.
	Select an Available table then click this button to add it to the list of selected tables
	Click this button to add ALL Available tables to the Selected tables list.
	Select a Selected table then click this button to remove it from the list.
	Click this button to remove all Selected tables from the list.
	Select a table then click this button to move it upwards in the order of appearance.
	Select a table then click this button to move it downwards in the order of appearance.



Once done, click **Next** to proceed.

6. This is the pivotal step in the process and involves selecting the types of forms to be created for each of the tables you selected in the previous step.

Property	Description
Tables	A list of the tables you selected. Click on a particular table to access more configuration details.
Grid Pages	
Grid	Activate this option if you want to have a grid form for the table.
Search	Activate this option if you want to have a search form for the table.
Security Level	Specify the security level required to access the form.
Record View	
Record	Activate this option if you want to have a record form for the table.
Updateable	Activate this option if you want the record form to be able to update records.
Security Level	Specify the security level required to access the form.

**Power Wizard**

Generate pages with forms based on database tables and queries







Table	Grid Pages			Record View		
(Click for details)	Grid	Search	Security Level	Record	Updateable	Security Level
<a href="#">items</a>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	User
<a href="#">members</a>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Admin
<a href="#">orders</a>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	User	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	User
<a href="#">editorials</a>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Admin

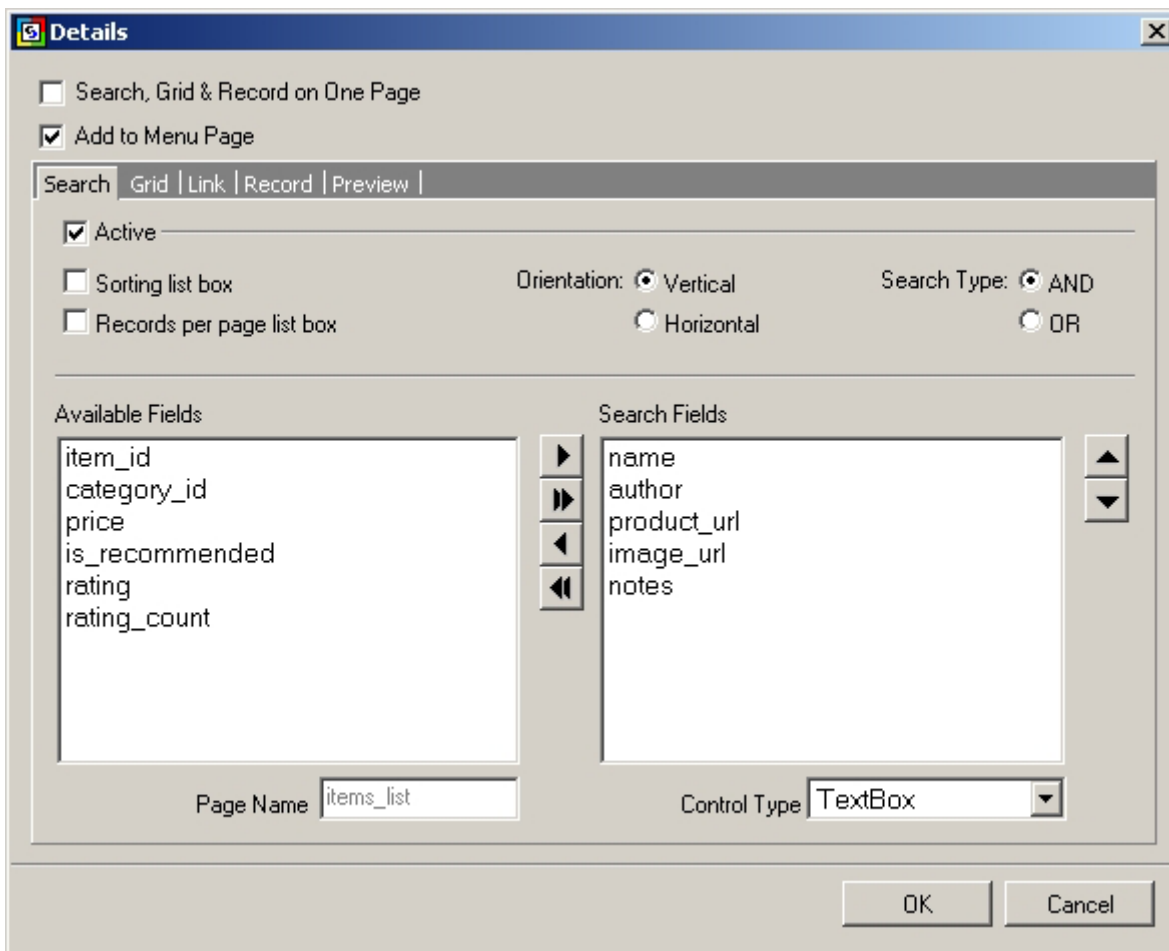
Cancel < Back Next > Finish

If you click on a particular table name under the **Table** column, the **Details** window shown below appears using which you can configure more details for the table. The first tab in the **Details** window is the **Search** tab where you can configure an optional search form.

Property	Description
Search, Grid & Record form on One Page	Select this option if you want the Search, Grid and Record forms to appear on the same page.
Add to Menu Page	If selected, a link to this page will be added in the menu page.
Active	Check this option to indicate your intention to create a search form.
Sorting list box	If selected, the search form will have two listboxes for sorting. The first one will contain a list of all the available fields so that the user can select the field to sort by. The user will also be able to indicate the desired sort order (ASC, DESC) using the second listbox.
Records per page list box	If selected the search form will have a listbox that can be used to specify the number of records to be returned per page after the search is conducted
Orientation	Specify whether the search form should be vertically or horizontally oriented
Search Type	Specify the SQL operator to be used for the search. If AND is selected, the search form will have an input form control for all the fields. Otherwise, if OR is selected, the search form will have only one input form control. Any value entered into the single control will be used to search against all the fields in the form. You can use the preview button to see how the form looks like with either option selected.
Available Fields	This listbox shows all the fields available in the table or view.



Search Fields	These are the fields to be shown in the search form. They are selected from among the Available Fields.
	Select an Available Field then click this button to add it to the form.
	Click this button to add ALL Available Fields to the form
	Select a form field then click this button to remove it from the form
	Click this button to remove all fields from the form.
	Select a form field then click this button to move it upwards in the order of appearance.
	Select a form field then click this button to move it downwards in the order of appearance.
Control Type	Specify the control type to be used for each of the fields in the form.



**Details**

☐ Search, Grid & Record on One Page  
☒ Add to Menu Page

Search | Grid | Link | Record | Preview |

☒ Active

☐ Sorting list box      Orientation: ☒ Vertical      Search Type: ☒ AND  
☐ Records per page list box      ☐ Horizontal      ☐ OR

Available Fields: item\_id, category\_id, price, is\_recommended, rating, rating\_count



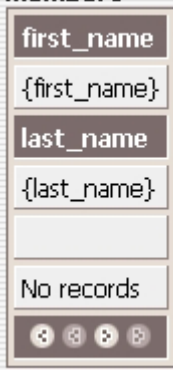






Search Fields: name, author, product\_url, image\_url, notes

Page Name: items\_list      Control Type: TextBox

OK Cancel

Further options for the grid form are available under the **Grid** tab of the **Details** window.

Property	Description
Active	Check this option to indicate your intention to create a grid form.
Grid Layout	Select the desired layout of the grid form: <ul style="list-style-type: none"> <li>▪ Tabular</li> </ul>

	<p><b>members</b></p>  <ul style="list-style-type: none"> <li>▪ Columnar</li> </ul> <p><b>members</b></p>  <ul style="list-style-type: none"> <li>▪ Justified</li> </ul> <p><b>members</b></p> 
Sorting	Select the type of sorting controls you want to have on the grid form.
Page Navigator	Select the type of navigation controls you want to have on the grid form.
Available Fields	This listbox shows all the fields available in the table or view
Fields for the Grid	These are the fields to be shown in the grid form. They are selected from among the Available Fields.
	Select an Available Field then click this button to add it to the form.
	Click this button to add ALL Available Fields to the form.
	Select a form field then click this button to remove it from the form.
	Click this button to remove all fields from the form.
	Select a form field then click this button to move it upwards in the order of appearance.
	Select a form field then click this button to move it downwards in the order of appearance.
Page Name	Specify the name of the page where the grid form will appear.
Order by	Select a field by which the records will be ordered by when initially displayed.

**Details**

☐ Search, Grid & Record on One Page  
☒ Add to Menu Page

Search | **Grid** | Link | Record | Preview |

☒ Active

Grid layout: Tabular  
Sorting: Single Direction Indicator  
Page Navigator: |< << 5 of 10 >> >|

Available Fields  
notes  
rating\_count  
rating  
is\_recommended

Fields for the Grid  
item\_id  
category\_id  
name  
author  
price  
product\_url  
image\_url







Page Name: items\_list  
Order by: item\_id

OK Cancel

In the **Links** tab of the **Details** window, you can specify how the records in the grid form will be linked to the records in the record form.

Property	Description
Database column	Select this option if you want to use one of the fields in the database table as the link field. You can then specify the field to be used from among the available fields.
New Grid column	You can also create a new column in the Grid form that will specifically be used to link to the Record form. You can enter some text of your choice to be used for this new grid column.

After configuring the link to the record form, you now proceed to configure options for the record form itself. This is done under the **Record** tab of the **Details** window.

Property	Description
Active	Activate this option to indicate your intention to create a record form.
Primary Key	Select the primary key field in the table
Available Fields	This listbox shows all the fields available in the table
Fields for the Form	These are the fields to be shown in the record form. They are selected from among the Available Fields.
	Select an Available Field then click this button to add it to the form
	Click this button to add ALL Available Fields to the form
	Select a form field then click this button to remove it from the form
	Click this button to remove all fields from the form
	Select a form field then click this button to move it upwards in the order of appearance
	Select a form field then click this button to move it downwards in the order of appearance
Control Type	Specify the control type to be used for each of the fields in the form
Page Name	Specify the name of the page where the record form will appear.

**Details**

☐ Search, Grid & Record on One Page

☒ Add to Menu Page

Search | Grid | Link | **Record** | Preview

☒ Active

Primary Key: item\_id

Available Fields

Fields for the Record Form

- item\_id
- category\_id
- name
- author
- price
- product\_url
- image\_url
- notes
- is\_recommended
- rating

Page Name: items\_list

Control Type: TextBox

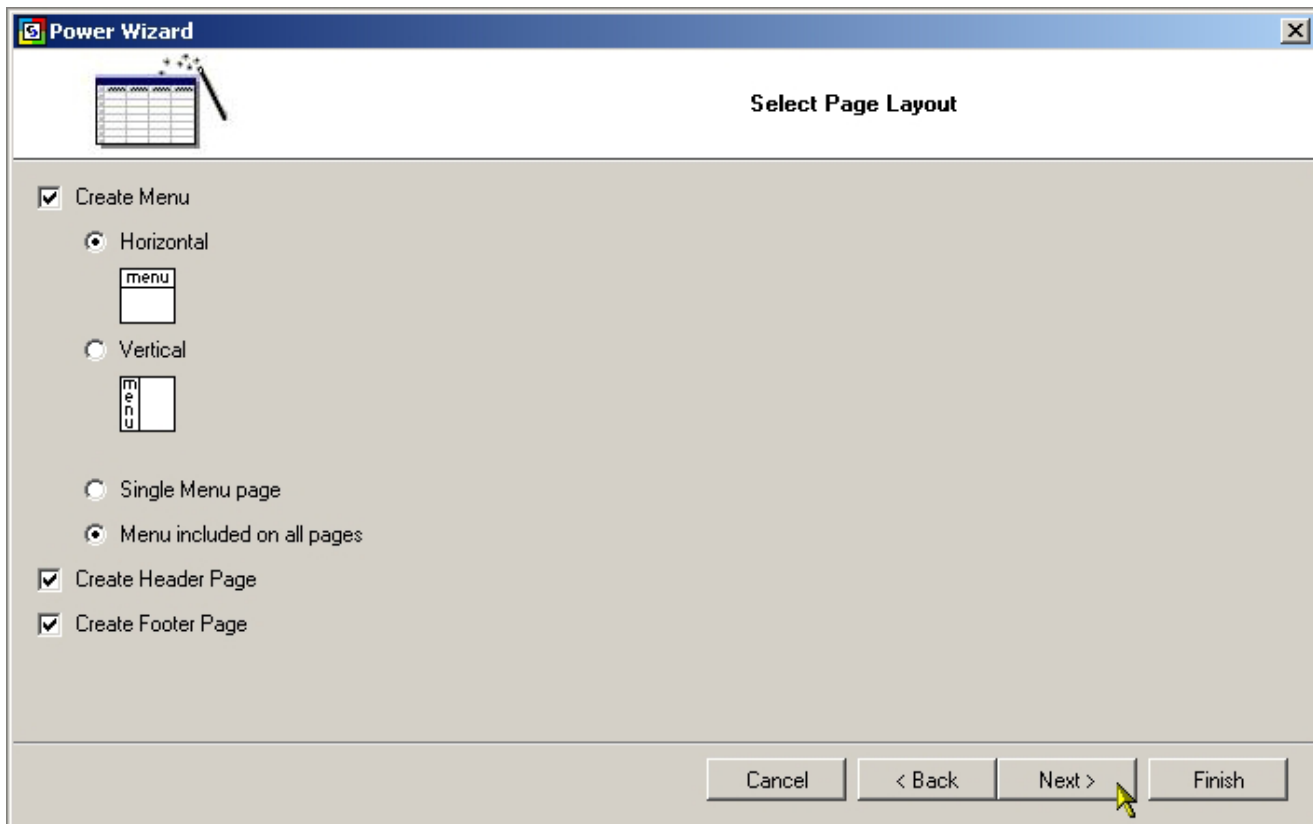
OK Cancel

Finally, after configuring all the forms, you can preview how they will look like when generated by using the **Preview** tab of the **Details** window. You will have a chance to select a style at a later stage but at the moment, you can see the layout of the three forms based on the selections you made.

Once you are satisfied with the appearance of the forms, click on the **OK** button to close the **Details** window and return to the **Power Wizard** window. If need be, you can click on another table name and access the **Details** window for that table. Otherwise, click on the **Next** button in the **Power Wizard** to proceed.

7. You now have options to customize the page layout.

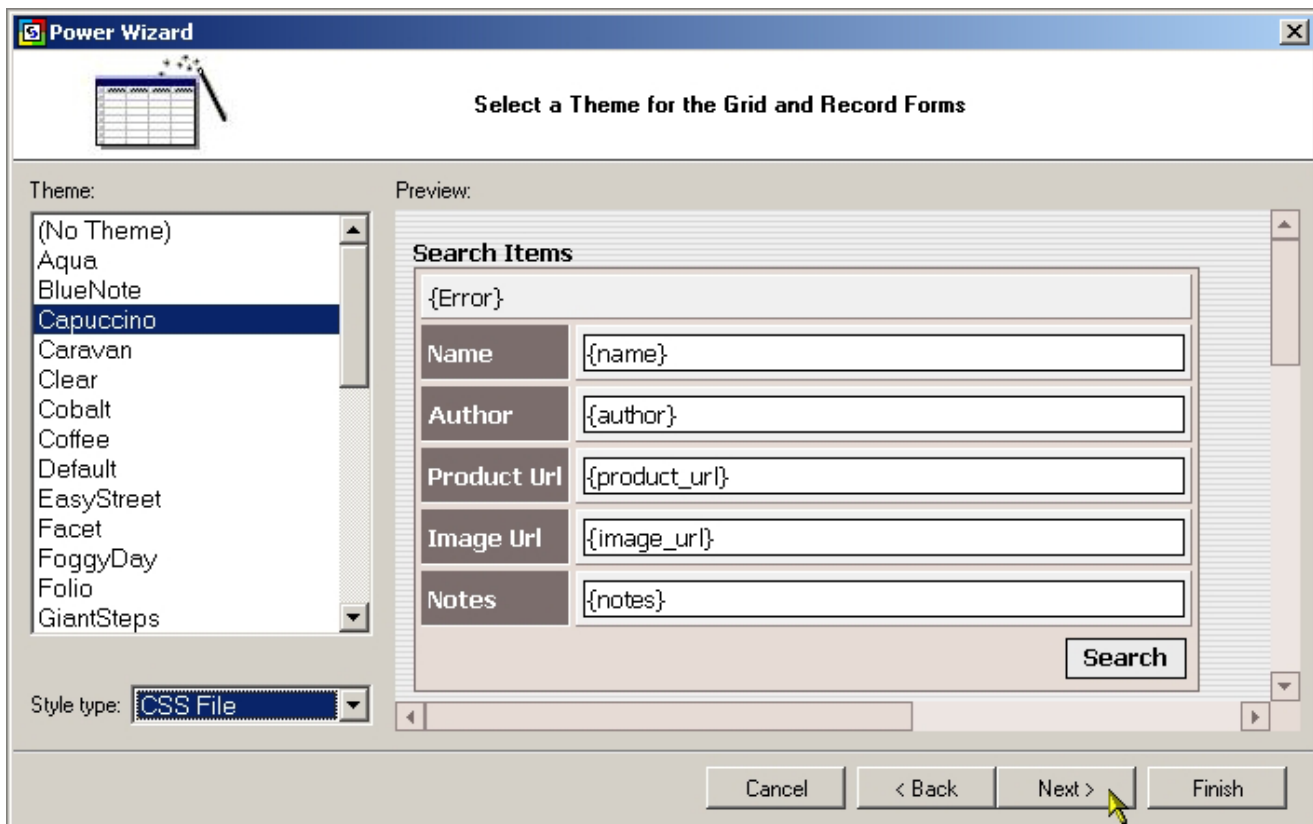
Property	Description
Create Menu	Activate this option to indicate that you want to have a menu for navigating between pages.
Single Menu Page	If this option is selected, the menu will be placed in the Header page that will in turn be included in all other pages.
Menu included on all pages.	With this option, the menu is placed in each of the pages generated rather than in a header page that is included in the other pages.
Create Header Page	Select this option if you want all pages to have a header page.
Create Footer Page	Select this option if you want all pages to have a footer page.



After configuring the page layout, click **Next** to proceed.

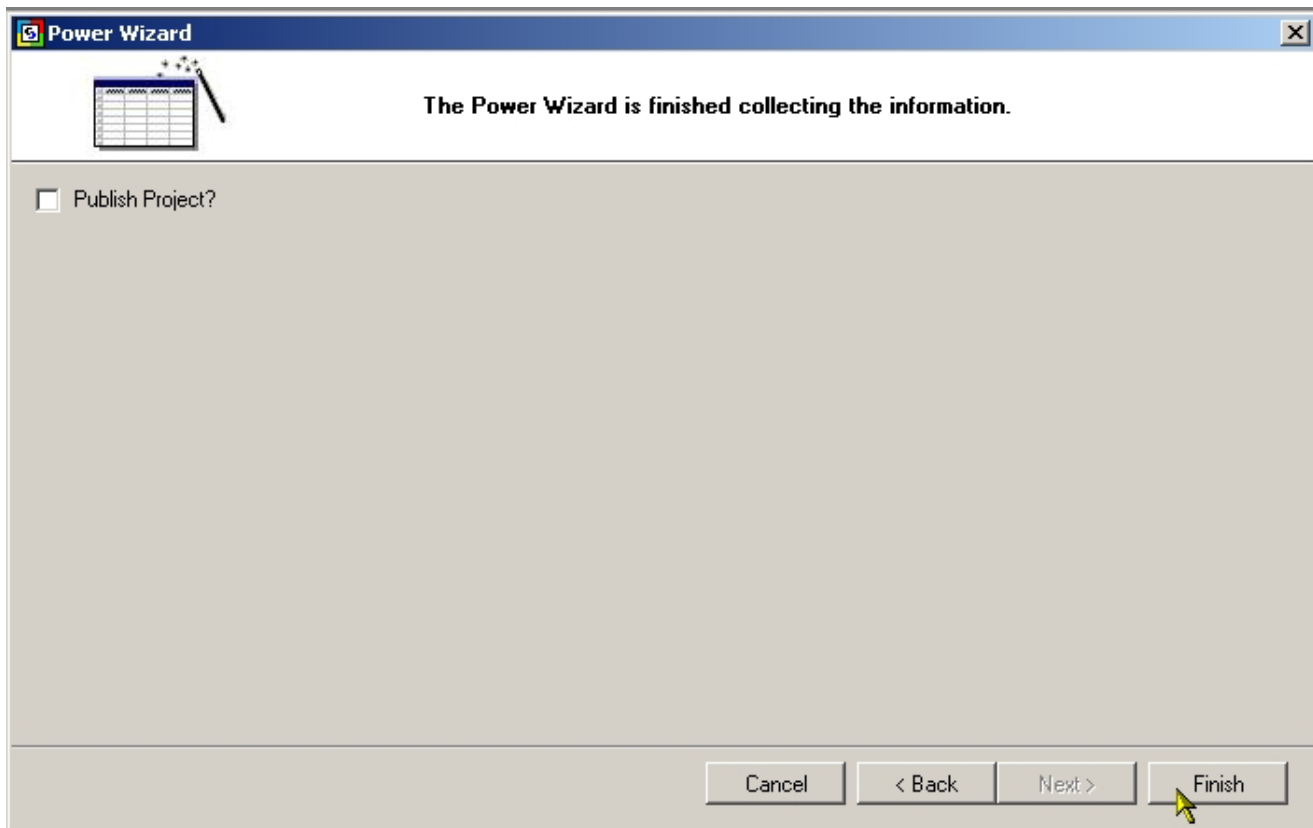
8. With all the other configurations in place, you now have the chance to select a theme to be used for the forms

Property	Description
Theme	This is a list of available themes. Select the (Not Theme) option if you don't want to use a theme.
Preview	This window shows how the forms would look like based on the selected theme.
Style Type	Select how the style will be applied: <ul style="list-style-type: none"> <li>▪ HTML – Styles are defined using HTML attributes and properties.</li> <li>▪ CCS File – Cascading Style Sheet definitions are placed at the top of the HTML pages in the &lt;HEAD&gt; section.</li> <li>▪ CCS Page - Cascading Style Sheet definitions are placed in a separate file that is linked to the HTML pages.</li> <li>▪ CCS Inline - Cascading Style Sheet definitions are placed within the HTML tags.</li> </ul>

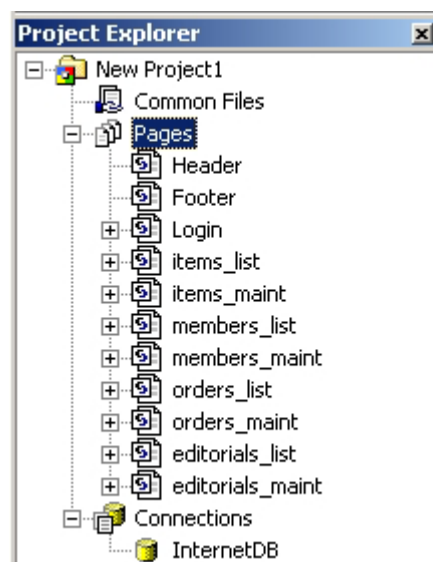


9. After deciding on a theme to use, click **Next** to proceed to the last step. Here, you can indicate whether you want the project pages to be published after they have been generated. The pages will be published to the Server Path you specified earlier on.



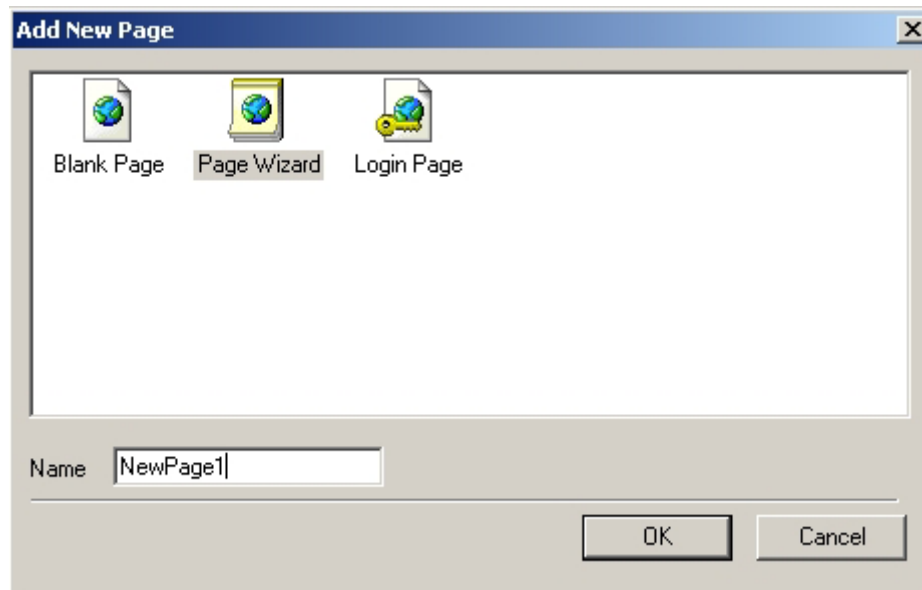


Upon clicking the Finish button, all the necessary pages will be generated and optionally published. The Project Explorer window will show all the pages for the project and you can then open them and make any adjustments you want.

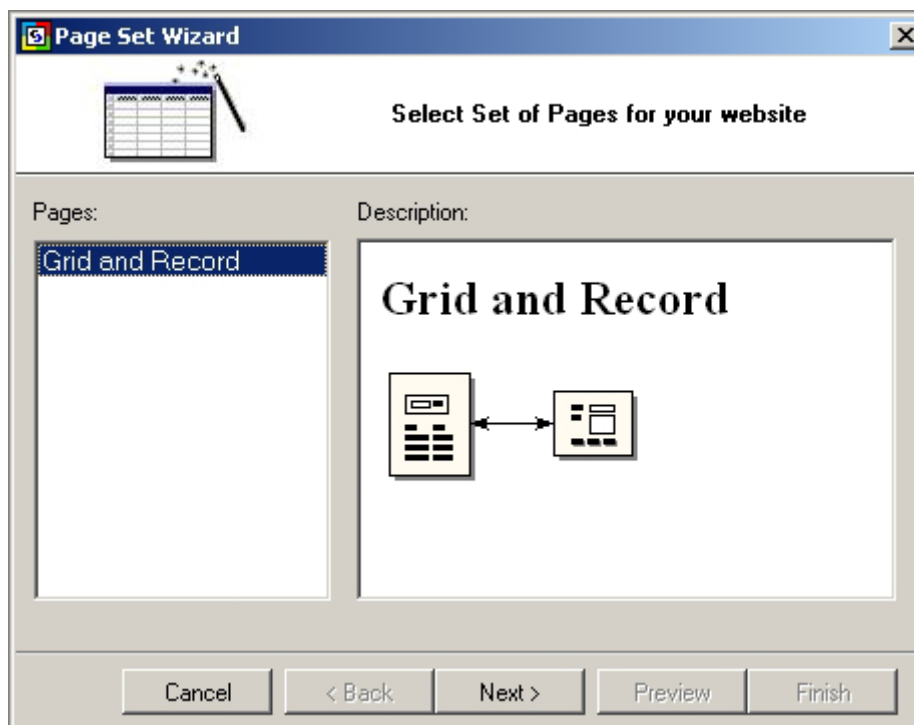


## Page Wizard

The Page wizard is used to create new pages with forms in them. To get to the page wizard, select the **New → Page...** option under the **File** Menu then in the **Add New Page** window, select the **Page Wizard** option. Click **OK** to begin the process of creating the page.



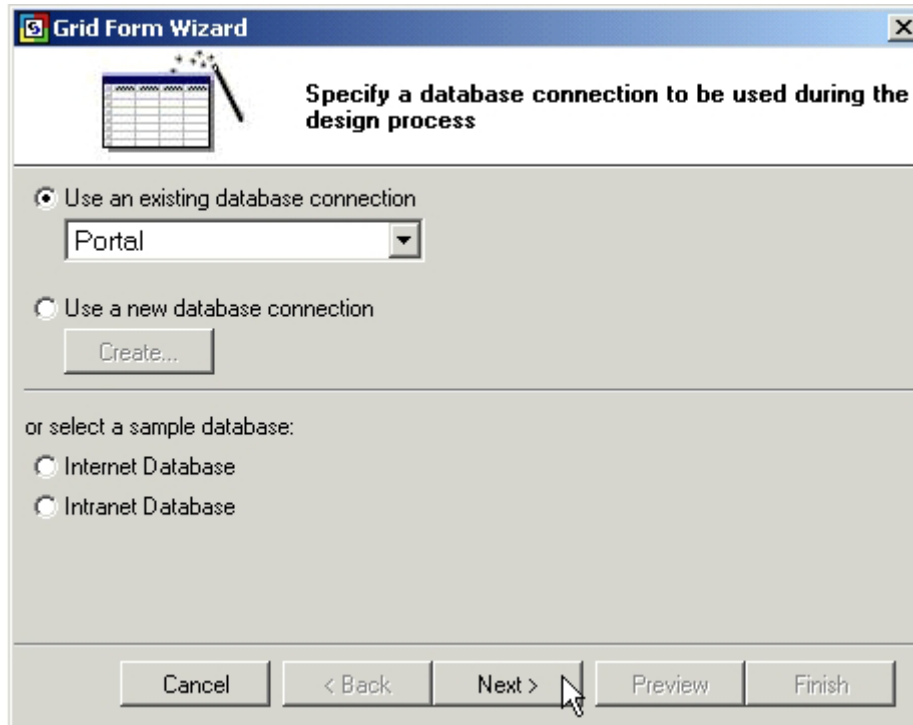
The steps involved are similar to those used for the [Grid and Record wizard](#) on the Toolbox so it will suffice to refer to the instructions there.







## Grid Form Wizard



Use the grid form wizard to create grid forms that can optionally have a related search form. As will become evident as we proceed through the process of creating the grid forms, the wizard provides many options that would otherwise require a considerable amount of time to add manually.

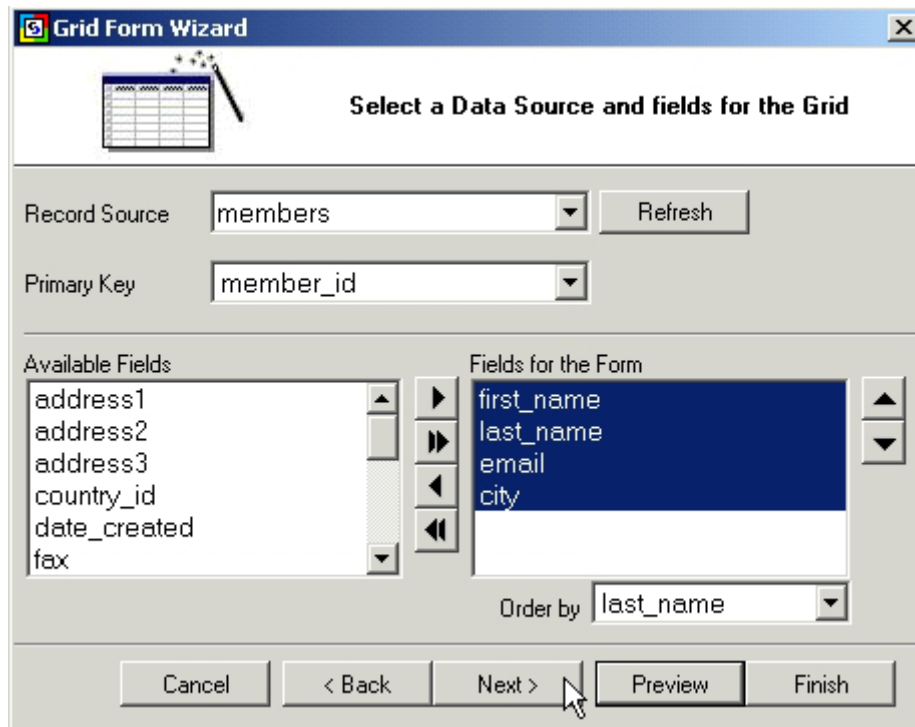
1. To begin the process, click on the Grid wizard option in the Toolbox.
2. The first step involves selecting or creating a database connection from where the data shown in the grid will be retrieved. You can select one of the existing database connections, create a new connection or select one of the sample databases. Once done click **Next** to proceed.



3. Having selected a connection, you now have to select a data source or table as well as the fields to be included in the grid.

Property	Description
Record Source	Select the table or view from where the data will come from
Refresh	Click this button to refresh the connection if you make changes to the database after having began using the wizard
Primary Key	Select the primary key field in the table or view
Available Fields	This listbox shows all the fields available in the table or view
Fields for the Form	These are the fields to be shown in the grid form. They are selected from among the Available Fields.
	Select an Available Field then click this button to add it to the form
	Click this button to add ALL Available Fields to the form
	Select a form field then click this button to remove it from the form
	Click this button to remove all fields from the form

	Select a form field then click this button to move it upwards in the order of appearance
	Select a form field then click this button to move it downwards in the order of appearance
Order by	Select a field by which the records will be ordered by when initially displayed









After having added fields to the form and optionally set the Order by field, you can click on the **Preview** button to see how the form looks like at this point. For the above setup, the illustration below shows how the form looks like in its current state of construction.

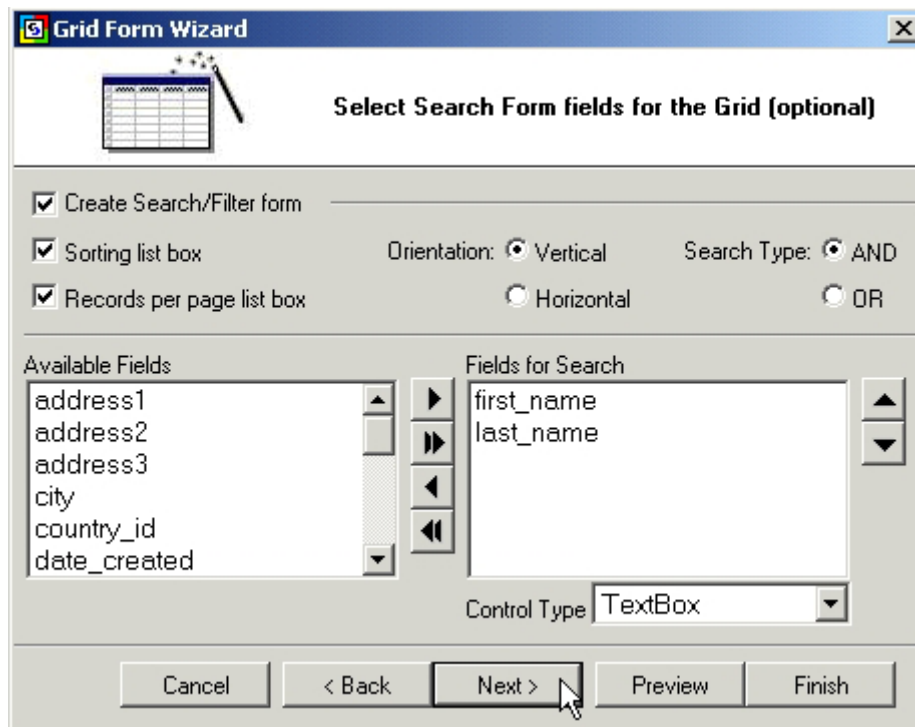
```
members
first_name  last_name  email  city
{first_name} {last_name} {email} {city}
```

After previewing the form, close the preview window and click on the **Next** button to proceed. Note that if you are satisfied with the form at this point, you can click on the **Finish** button to close the wizard and generate the form.

- The next step involves adding a search form that can be used to filter the records displayed in the grid form. This step is optional and if you don't want a corresponding search form for the grid, you can simply click the **Next** button to proceed to the next step.

Property	Description
----------	-------------

Create Search/Filter Form	Check this option to indicate your intention to create a search form
Sorting list box	If selected, the search form will have two listboxes for sorting. The first one will contain a list of all the available fields so that the user can select the field to sort by. The user will also be able to indicate the desired sort order (ASC, DESC) using the second listbox.
Records per page list box	If selected the search form will have a listbox that can be used to specify the number of records to be returned per page after the search is conducted
Orientation	Specify whether the search form should be vertically or horizontally oriented
Search Type	Specify the SQL operator to be used for the search. If AND is selected, the search form will have a form control for all the fields. Otherwise, if OR is selected, the search form will have only one form control. Any value entered in the single control will be used to search against all the fields in the form. You can use the preview button to see how the form looks like with either option selected.
Available Fields	This listbox shows all the fields available in the table or view
Fields for Search	These are the fields to be shown in the search form. They are selected from among the Available Fields.
	Select an Available Field then click this button to add it to the form
	Click this button to add ALL Available Fields to the form
	Select a form field then click this button to remove it from the form
	Click this button to remove all fields from the form
	Select a form field then click this button to move it upwards in the order of appearance
	Select a form field then click this button to move it downwards in the order of appearance
Control Type	Specify the control type to be used for each of the fields in the form



Once again, during the process of creating the search form, you should use the Preview button to see how the form looks with the selected options. You can therefore try out various options and preview them before deciding on the appropriate one.

```

membersSearch
{Error}
first_name {first_name}
last_name {last_name}
Sort by [v] [v]
Records per page [v]
[Search]

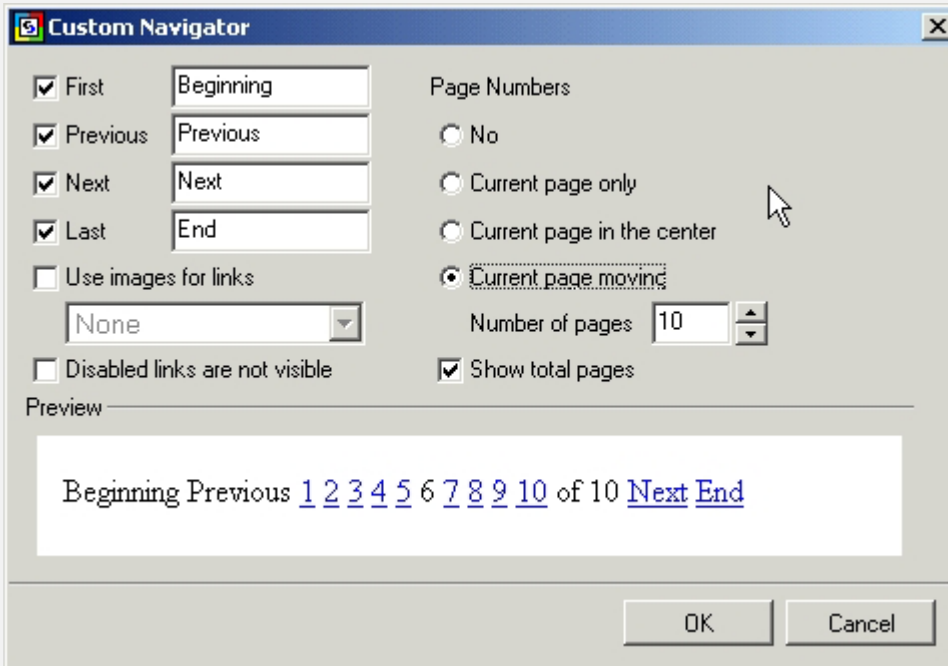
members
first_name last_name email city
{first_name} {last_name} {email} {city}

```

After configuring the form satisfactorily, click on the **Next** button to proceed.

- At this juncture, you can further customize the grid form by selecting the caption to be used for sorting as well as the navigational controls.

Property	Description
----------	-------------

Sorting	Select the type of sorting controls that you want to have in the grid.																				
Page Navigator	Select the type of Navigation controls that you want to have in the grid.																				
Modify	<p>If you want to have custom navigation controls on the form, click on this button to open the <b>Custom Navigator</b> window where you can customize the controls to suit your needs.</p>  <table border="1"> <thead> <tr> <th>Property</th><th>Description</th></tr> </thead> <tbody> <tr> <td>First</td><td>Text for the link to the first record</td></tr> <tr> <td>Previous</td><td>Text for the link to the previous record</td></tr> <tr> <td>Next</td><td>Text for the link to the next record</td></tr> <tr> <td>Last</td><td>Text for the link to the last record</td></tr> <tr> <td>Use images for links</td><td>Check this option if you want to have images instead of text for the navigation links</td></tr> <tr> <td>Disabled links are not visible</td><td>If this option is selected, links to pages that cannot logically be browsed are not shown e.g. if you are on page 1, you cannot browse using the first or previous links.</td></tr> <tr> <td>Page Numbers</td><td>Apart from having the four links mentioned above, specify whether you also want to have links to the individual pages and how they should appear.</td></tr> <tr> <td>Number of Pages</td><td>Specify how many page links should be displayed</td></tr> <tr> <td>Show total pages</td><td>Check this option if you want to display the total number of pages that can be browsed</td></tr> </tbody> </table>	Property	Description	First	Text for the link to the first record	Previous	Text for the link to the previous record	Next	Text for the link to the next record	Last	Text for the link to the last record	Use images for links	Check this option if you want to have images instead of text for the navigation links	Disabled links are not visible	If this option is selected, links to pages that cannot logically be browsed are not shown e.g. if you are on page 1, you cannot browse using the first or previous links.	Page Numbers	Apart from having the four links mentioned above, specify whether you also want to have links to the individual pages and how they should appear.	Number of Pages	Specify how many page links should be displayed	Show total pages	Check this option if you want to display the total number of pages that can be browsed
Property	Description																				
First	Text for the link to the first record																				
Previous	Text for the link to the previous record																				
Next	Text for the link to the next record																				
Last	Text for the link to the last record																				
Use images for links	Check this option if you want to have images instead of text for the navigation links																				
Disabled links are not visible	If this option is selected, links to pages that cannot logically be browsed are not shown e.g. if you are on page 1, you cannot browse using the first or previous links.																				
Page Numbers	Apart from having the four links mentioned above, specify whether you also want to have links to the individual pages and how they should appear.																				
Number of Pages	Specify how many page links should be displayed																				
Show total pages	Check this option if you want to display the total number of pages that can be browsed																				
Records per page	Specify the default number of records to be shown per page																				
No records found message	Specify the message that will be displayed in the grid form if there are no records to be shown																				

**Grid Form Wizard**

**Define Grid Form Options**

**Sorting**

☐ No sorting

☐ Caption

☐ Caption ▼

☒ Caption ▼ ▲

**Page Navigator**

☐ No Page Navigator

☐ Prev [3] Next

☒ ◀ Prev Next ▶

☐ < << 5 of 10 >> >

☐ First Prev 1 2 3 4 5 of 10 Next Last

☐ Custom Modify...

Records per Page: 10

No records found message: No records were returned

Cancel < Back Next > Preview Finish

Once you have made all the selections, you can use the preview button to see how the form looks like and if necessary, make adjustments.

membersSearch

{Error}

first\_name

last\_name

Sort by ▼ ▼

Records per page ▼

Search

members

first\_name ▲▲▼▼ last\_name ▲▲▼▼ email ▲▲▼▼ city ▲▲▼▼

{first\_name} {last\_name} {email} {city}

No records were returned

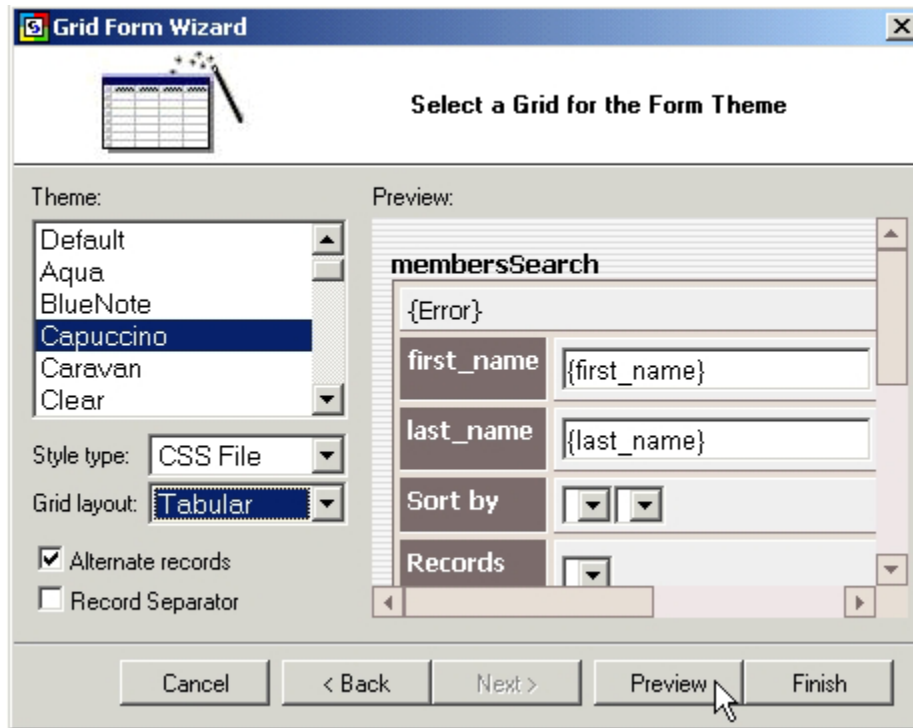
◀ Prev ◀ Prev Next ▶ Next ▶

- The final step is optional and involves selecting a theme for the form.

Property	Description
----------	-------------



Theme	This listbox contains a list of themes from which you can select from																	
Style Type	<ul style="list-style-type: none"><li>HTML – Styles are defined using HTML attributes and properties.</li><li>CCS File – Cascading Style Sheet definitions are placed at the top of the HTML pages in the &lt;HEAD&gt; section.</li><li>CCS Page - Cascading Style Sheet definitions are placed in a separate file that is linked to the HTML pages.</li><li>CCS Inline - Cascading Style Sheet definitions are placed within the HTML tags.</li></ul>																	
Grid Layout	<p>Specify the layout of the grid form</p> <ul style="list-style-type: none"><li>Tabular</li></ul> <div><p><b>members</b></p><table><tr><td>first_name</td><td>last_name</td></tr><tr><td>{first_name}</td><td>{last_name}</td></tr><tr><td colspan="2">No records</td></tr></table></div> <ul style="list-style-type: none"><li>Columnar</li></ul> <div><p><b>members</b></p><table><tr><td>first_name</td><td>{first_name}</td></tr><tr><td>last_name</td><td>{last_name}</td></tr><tr><td colspan="2">No records</td></tr></table></div> <ul style="list-style-type: none"><li>Justified</li></ul> <div><p><b>members</b></p><table><tr><td>first_name</td></tr><tr><td>{first_name}</td></tr><tr><td>last_name</td></tr><tr><td>{last_name}</td></tr><tr><td>No records</td></tr></table></div>	first_name	last_name	{first_name}	{last_name}	No records		first_name	{first_name}	last_name	{last_name}	No records		first_name	{first_name}	last_name	{last_name}	No records
first_name	last_name																	
{first_name}	{last_name}																	
No records																		
first_name	{first_name}																	
last_name	{last_name}																	
No records																		
first_name																		
{first_name}																		
last_name																		
{last_name}																		
No records																		
Alternate records	If this option is selected, every other record in the grid will appear in a row with a different background color than the preceding record.																	
Record Separator	With this option selected, each row where a record appears is followed by an empty row that separates it from the next row with a record.																	



Clicking the preview button now will show the final form as it will look when generated. The search form appears at the top with the grid right below it.

**membersSearch**

{Error}

first\_name

last\_name

Sort by

Records per page

**members**

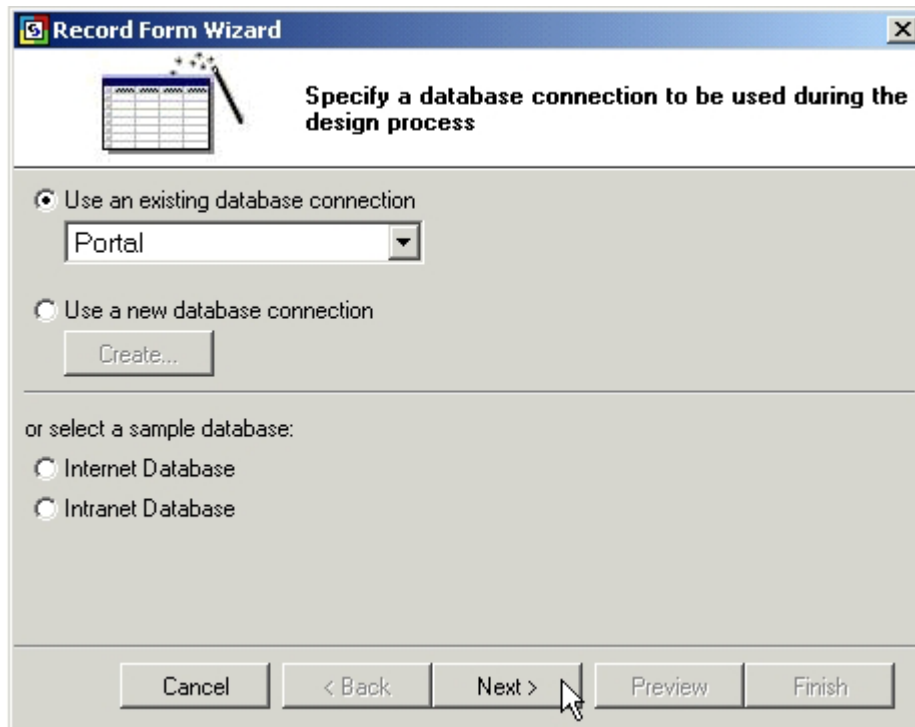
first_name 🗕🗕🗕🗕	last_name 🗕🗕🗕🗕	email 🗕🗕🗕🗕	city 🗕🗕🗕🗕
{first_name}	{last_name}	{email}	{city}
{Alt_first_name}	{Alt_last_name}	{Alt_email}	{Alt_city}

No records were returned

## Record Wizard







Like grid forms, record forms can similarly be created using the Record form wizard.

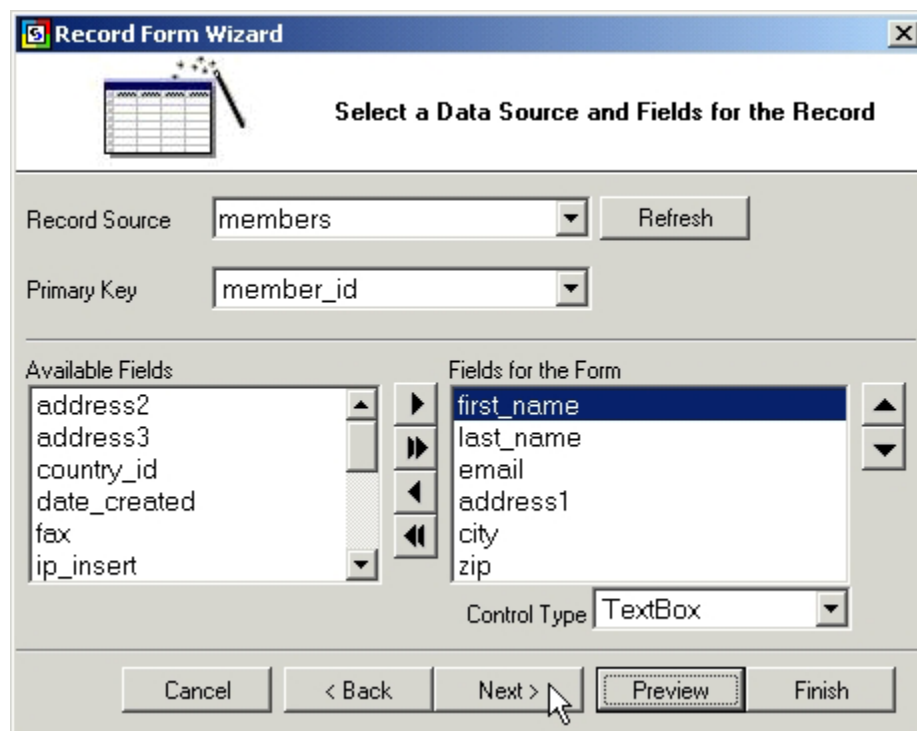
1. To begin the process, click on the **Record wizard** option in the **Toolbox**.
2. The first step involves selecting or creating a database connection from where the data to be edited or added using the record form is stored. You can select one of the existing database connections, create a new connection or select one of the sample databases. Once done click **Next** to proceed.



3. Having selected a connection, you now have to select a data source or table as well as the fields to be included in the record form

Property	Description
Record Source	Select the table from where the data will come from
Refresh	Click this button to refresh the connection if you make changes to the database after having began using the wizard
Primary Key	Select the primary key field in the table
Available Fields	This listbox shows all the fields available in the table
Fields for the Form	These are the fields to be shown in the record form. They are selected from among the Available Fields.

	Select an Available Field then click this button to add it to the form
	Click this button to add ALL Available Fields to the form
	Select a form field then click this button to remove it from the form
	Click this button to remove all fields from the form
	Select a form field then click this button to move it upwards in the order of appearance
	Select a form field then click this button to move it downwards in the order of appearance
Control Type	Specify the control type to be used for each of the fields in the form



The image shows a 'Record Form Wizard' dialog box with the title 'Select a Data Source and Fields for the Record'. It contains the following elements:

- Record Source:** A dropdown menu showing 'members' and a 'Refresh' button.
- Primary Key:** A dropdown menu showing 'member\_id'.
- Available Fields:** A list box containing 'address2', 'address3', 'country\_id', 'date\_created', 'fax', and 'ip\_insert'.
- Fields for the Form:** A list box containing 'first\_name', 'last\_name', 'email', 'address1', 'city', and 'zip'. 'first\_name' is currently selected.
- Control Type:** A dropdown menu showing 'TextBox'.
- Navigation Buttons:** 'Cancel', '< Back', 'Next >', 'Preview' (highlighted with a dashed border), and 'Finish'.

Once you have made the selections, you can click on the preview button to see how the form looks like and maybe make some adjustments. Once done, click **Next** to proceed

members

{Error}

first\_name {first\_name}

last\_name {last\_name}

email {email}

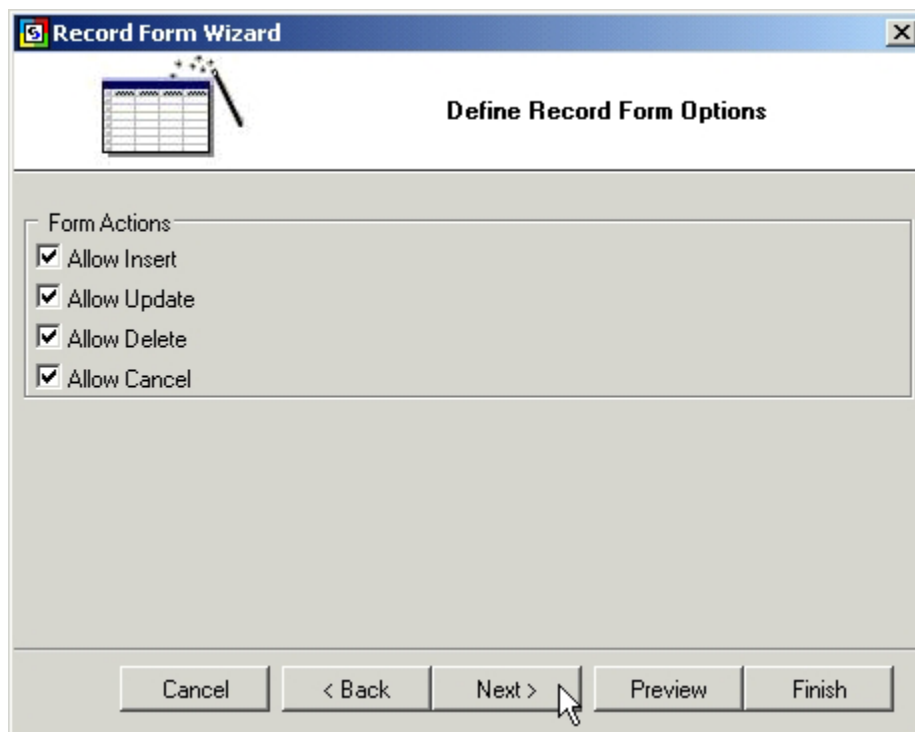
address1 {address1}

city {city}

zip {zip}

Insert Update Delete Cancel

- You now have to decide the kind of functionality that will be available in the record form. You can uncheck any of the options if you don't want the form to have the corresponding functionality.



**Record Form Wizard**

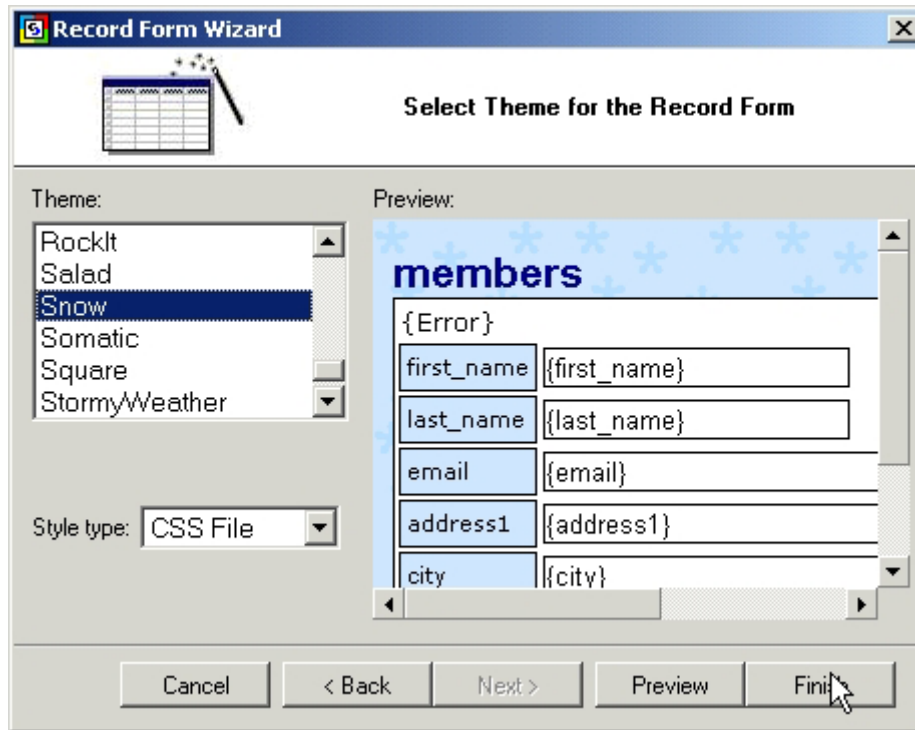
**Define Record Form Options**

Form Actions

- ☒ Allow Insert
- ☒ Allow Update
- ☒ Allow Delete
- ☒ Allow Cancel

Cancel < Back Next > Preview Finish

- In the final step, you optionally select a theme for the record form. The Preview portion of the window shows how the form would look like with the currently selected theme.

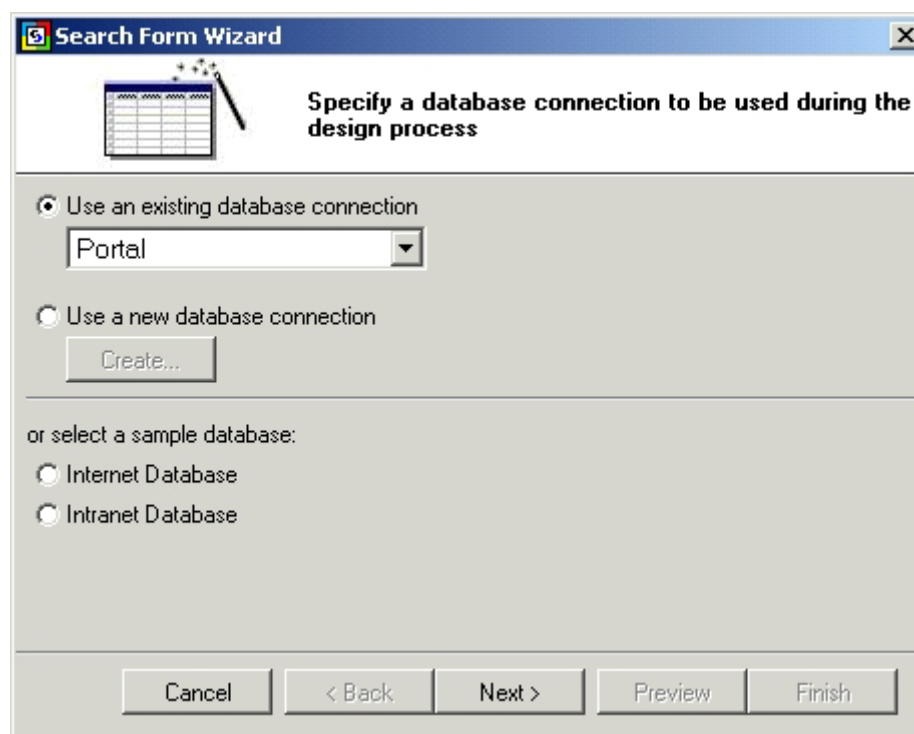


Once done, click on the **Finish** button to close the wizard and generate the form. The illustration below shows the final form based on the configurations made above.

## Search Wizard







Use the search wizard to create search forms to work in conjunction with other forms that already exist or will be created later.

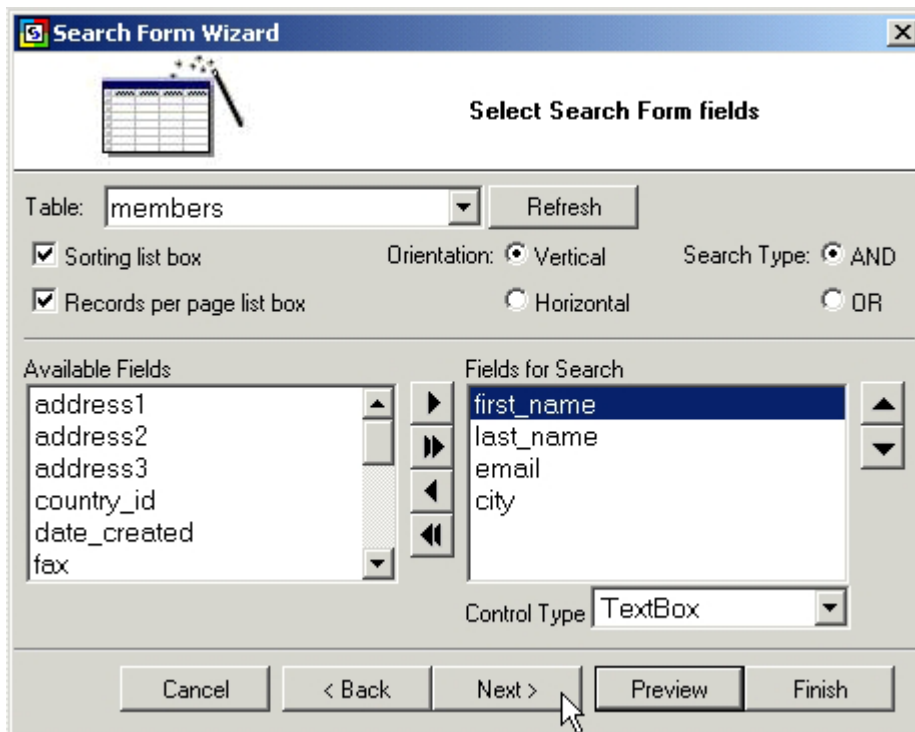
1. To begin the process, click on the **Search wizard** option in the **Toolbox**.
4. Select or create the database connection whose fields will be used in the search. You can select one of the existing database connections, create a new connection or select one of the sample databases. Once done click **Next** to proceed.



5. This is the main step in creating the search form and involves selecting fields to be shown in the form as well selecting other display options.

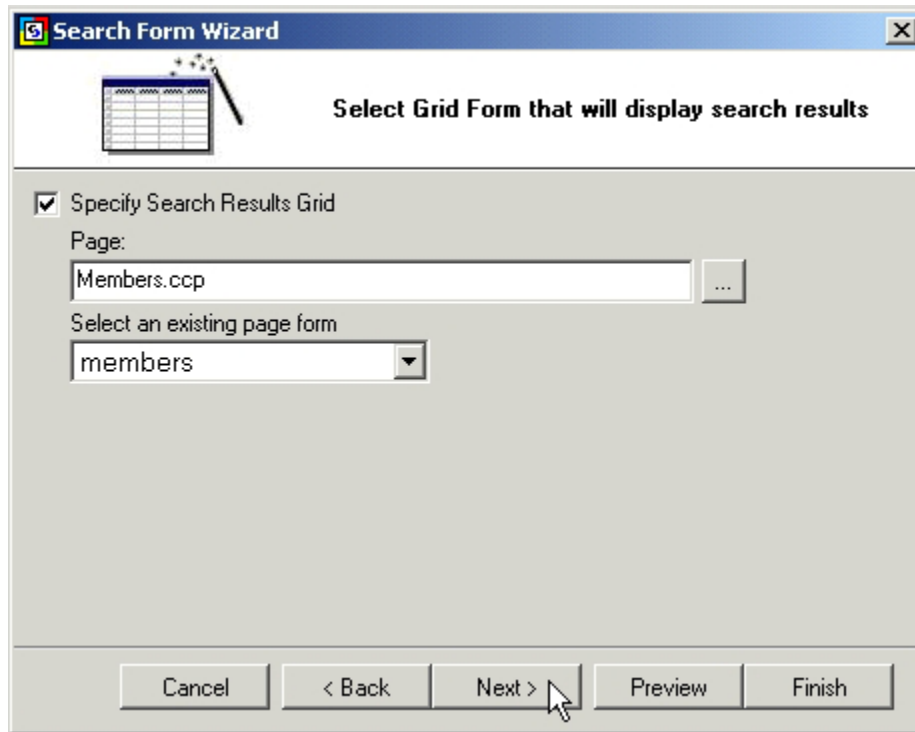
Property	Description
Table	Select the database tables whose fields will be used in the search form
Sorting list box	If selected, the search form will have two listboxes for sorting. The first one will contain a list of all the available fields so that the user can select the field to sort by. The user will also be able to indicate the desired sort order (ASC, DESC) using the second listbox.
Records per page list box	If selected the search form will have a listbox that can be used to specify the number of records to be returned per page after the search is conducted
Orientation	Specify whether the search form should be vertically or horizontally oriented
Search Type	Specify the SQL operator to be used for the search. If AND is selected, the search form will have a form control for all the fields. Otherwise, if OR is selected, the search form will have only one form control. Any value entered in the single control will be used to search against all the fields in the form. You can use the preview button to see how the form looks like with either option selected.
Available Fields	This listbox shows all the fields available in the table or view
Fields for Search	These are the fields to be shown in the search form. They are selected from

	among the Available Fields.
	Select an Available Field then click this button to add it to the form
	Click this button to add ALL Available Fields to the form
	Select a form field then click this button to remove it from the form
	Click this button to remove all fields from the form
	Select a form field then click this button to move it upwards in the order of appearance
	Select a form field then click this button to move it downwards in the order of appearance
Control Type	Specify the control type to be used for each of the fields in the form

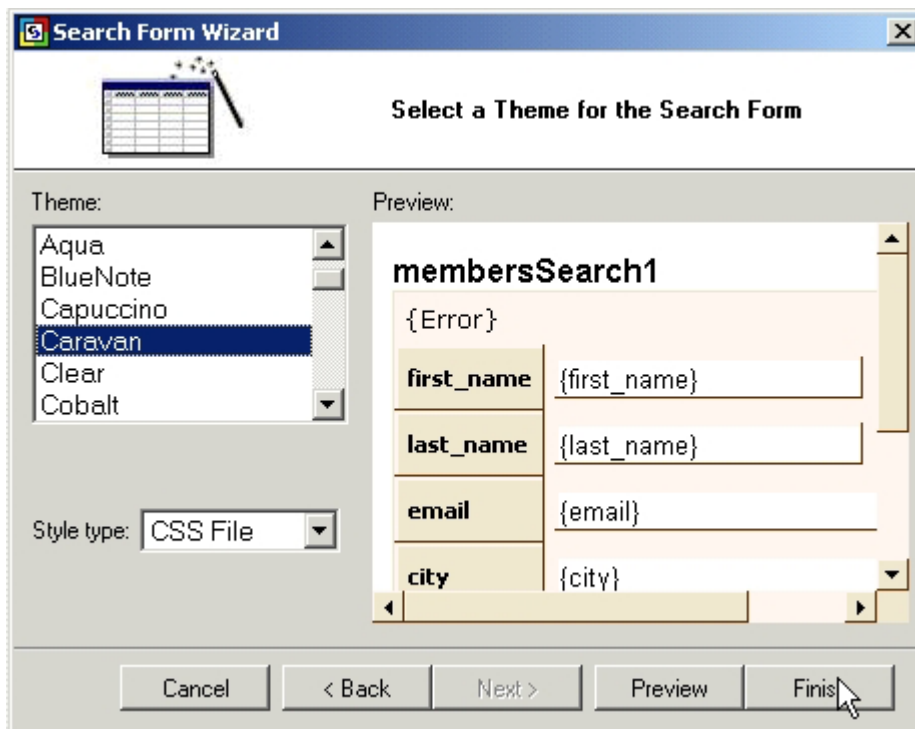


- Next you can optionally specify a page and a grid form to which the search will apply. You can jump this step if the form is not yet available or you intend to use the search form for a different purpose.





7. In the final step, you can optionally select a theme to be used to display the form.



Once done, click the **Finish** button to close the wizard and generate the search form.

### membersSearch1

{Error}

first_name	<input data-bbox="695 296 1000 338" type="text" value="{first_name}"/>
last_name	<input data-bbox="695 363 1000 405" type="text" value="{last_name}"/>
email	<input data-bbox="695 430 1118 472" type="text" value="{email}"/>
city	<input data-bbox="695 497 1118 539" type="text" value="{city}"/>
Sort by	<div><div>▼</div><div>▼</div></div>
Records per page	<div>▼</div>

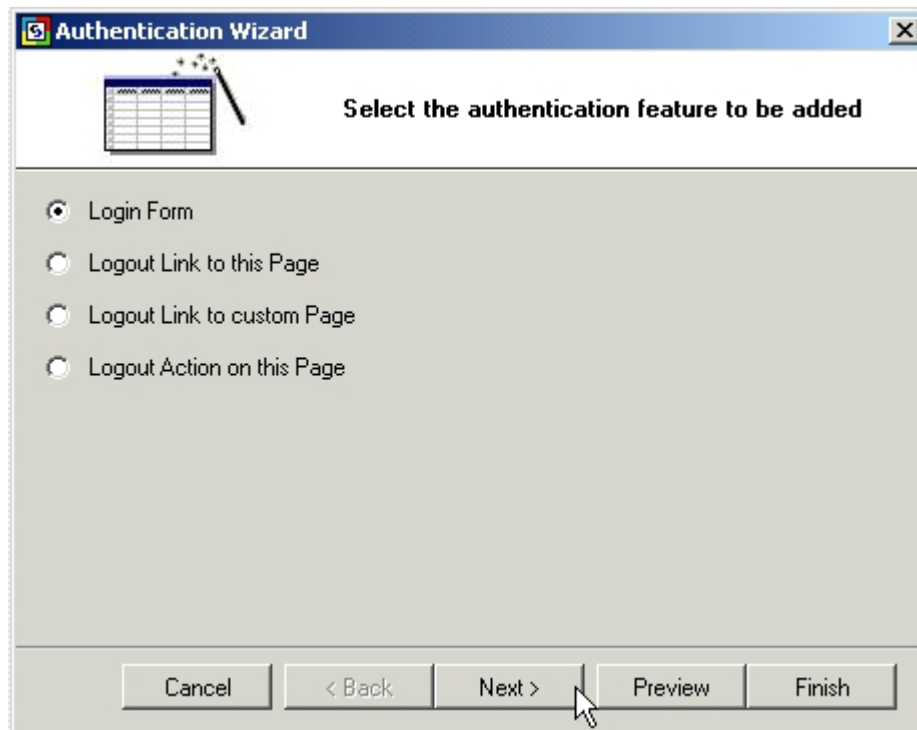
Search

## Authentication Wizard

The Authentication wizard is used to add login/logout functionality to an existing page. Click on the **Authentication wizard** option in the **Toolbox** to begin the process.

1. In the first step, you have to select the type of functionality you want to add to the page.

Option	Description
Login Form	Select this option to add a login form to the current page
Logout Link to this Page	Select this option to add a link to the logout page. The link added points to the current page.
Logout Link to custom Page	Select this option to add a link to the logout page. The link added points to the page you select in the next step.
Logout action on this Page	Select this option to add a logout action to the current page. The action is added in the After Initialize event of the page.



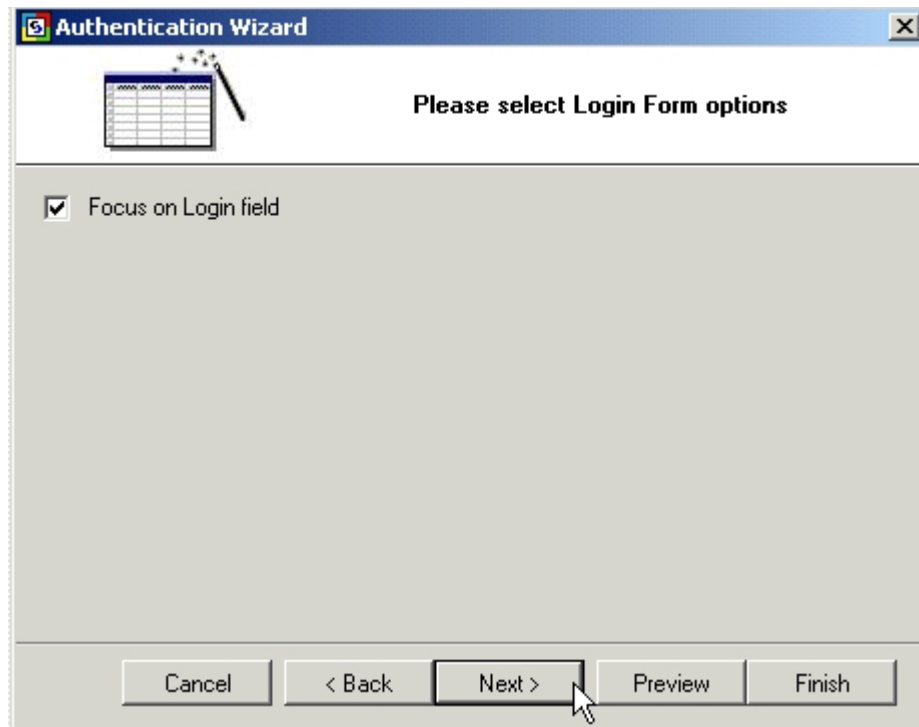
If you opt to add a Login form, clicking the preview button will show a basic form similar to that shown below.

Login  
{Error}

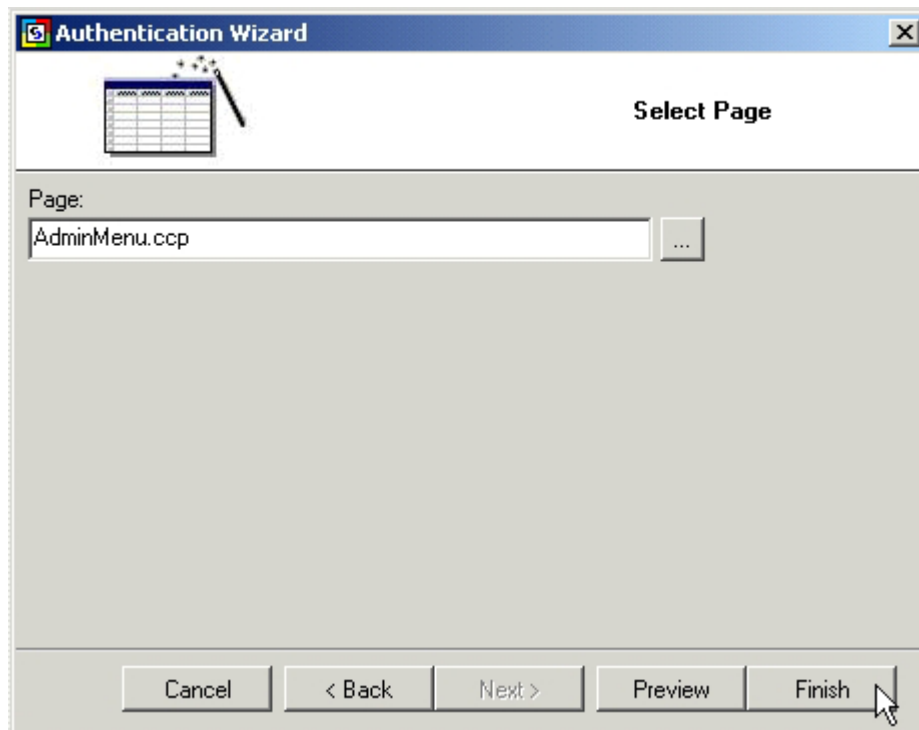
Login

Password

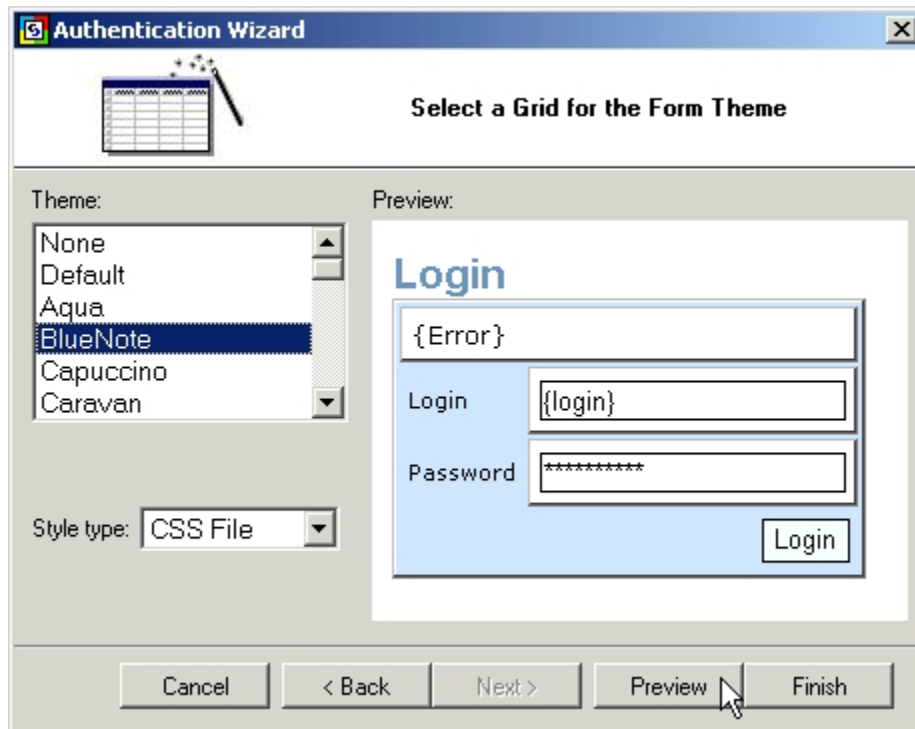
2. If in the first step you opted to add a Login form, the next step allows you to specify whether you want the login fields in the login form to have focus when the page loads. If the field is in focus, the user can begin typing into it immediately without having to first click on it.



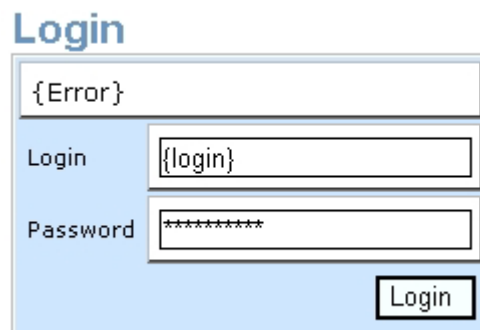
Alternatively, if in the first step you opted to add a Logout Link to a Custom Page, you now have to select the custom page where the link will refer.



3. The final step that applies only if you are building a login form involves the selection of a theme for the form.



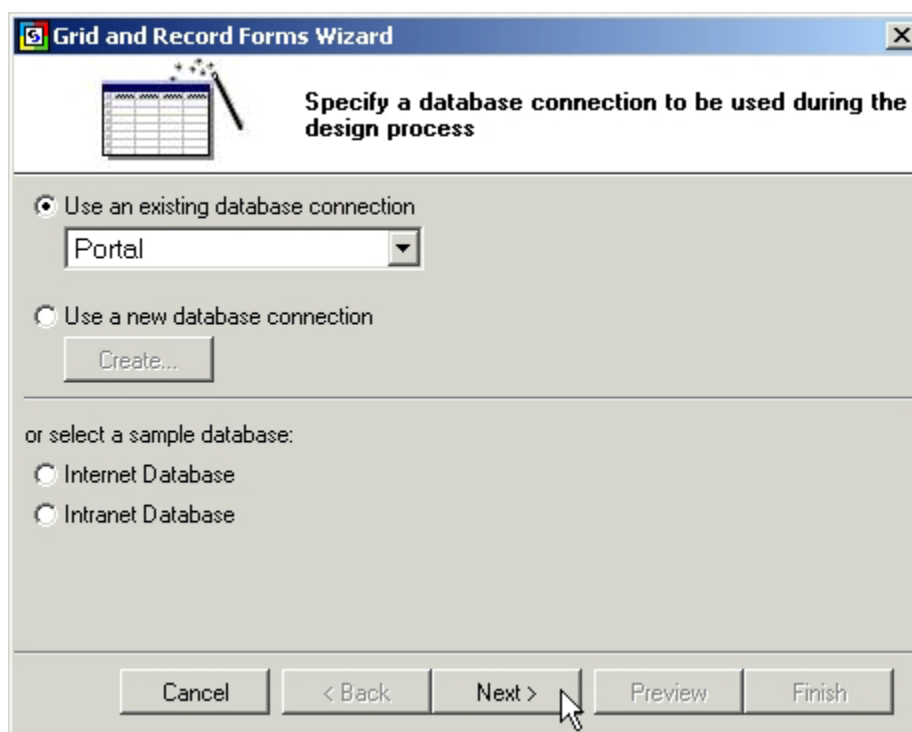
As usual, you can use the Preview button to see how the form looks like. Click the Finish to close the wizard and generate the form.









## Grid and Record Wizard

Often times a grid form is used in conjunction with a record form such that the grid form is used to display multiple records and a link is provided from the grid form records to the record form where the details of the record can be viewed and edited. For this purpose, the Grid and Record wizard can be used to produce a grid form with an attendant record form.





1. Click on the **Grid and Record Wizard** on the **Toolbox** to begin the process.
2. Select or create the database connection whose fields will be used in the two forms. You can select one of the existing database connections, create a new connection or select one of the sample databases. Once done click **Next** to proceed.





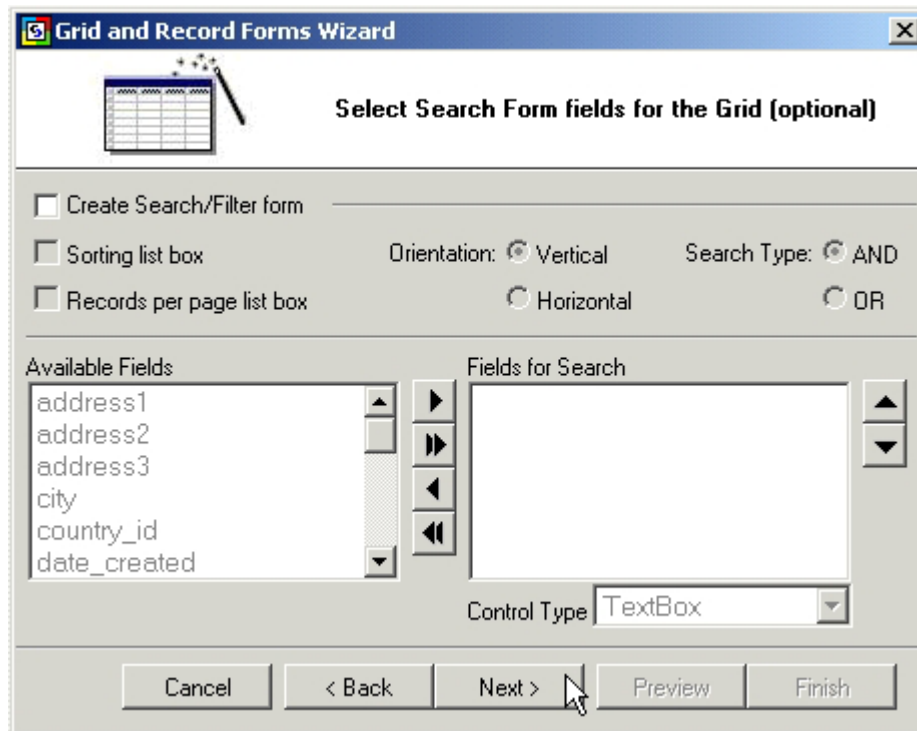
3. Now select the fields to be shown in the grid form.

Property	Description
Record Source	Select the table from where the data will come from
Refresh	Click this button to refresh the connection if you make changes to the database after having began using the wizard
Primary Key	Select the primary key field in the table
Available Fields	This listbox shows all the fields available in the table
Fields for the Form	These are the fields to be shown in the grid form. They are selected from among the Available Fields.
	Select an Available Field then click this button to add it to the form
	Click this button to add ALL Available Fields to the form
	Select a form field then click this button to remove it from the form
	Click this button to remove all fields from the form
	Select a form field then click this button to move it upwards in the order of appearance
	Select a form field then click this button to move it downwards in the order of appearance
Control Type	Specify the control type to be used for each of the fields in the form

4. The next step is optional and should only be used if you want to have a search form that can be used to filter the records displayed in the grid. If a search form is not needed, simply click the **Next** button to proceed.

Property	Description
Table	Select the database tables whose fields will be used in the search form
Sorting list box	If selected, the search form will have two listboxes for sorting. The first one will contain a list of all the available fields so that the user can select the field to sort by. The user will also be able to indicate the desired sort order (ASC, DESC) using the second listbox.
Records per page list box	If selected the search form will have a listbox that can be used to specify the number of records to be returned per page after the search is conducted
Orientation	Specify whether the search form should be vertically or horizontally oriented
Search Type	Specify the SQL operator to be used for the search. If AND is selected, the search form will have a form control for all the fields. Otherwise, if OR is selected, the search form will have only one form control. Any value entered in the single control will be used to search against all the fields in the form. You can use the preview button to see how the form looks like with either option selected.
Available Fields	This listbox shows all the fields available in the table or view
Fields for Search	These are the fields to be shown in the search form. They are selected from among the Available Fields.
	Select an Available Field then click this button to add it to the form
	Click this button to add ALL Available Fields to the form
	Select a form field then click this button to remove it from the form
	Click this button to remove all fields from the form

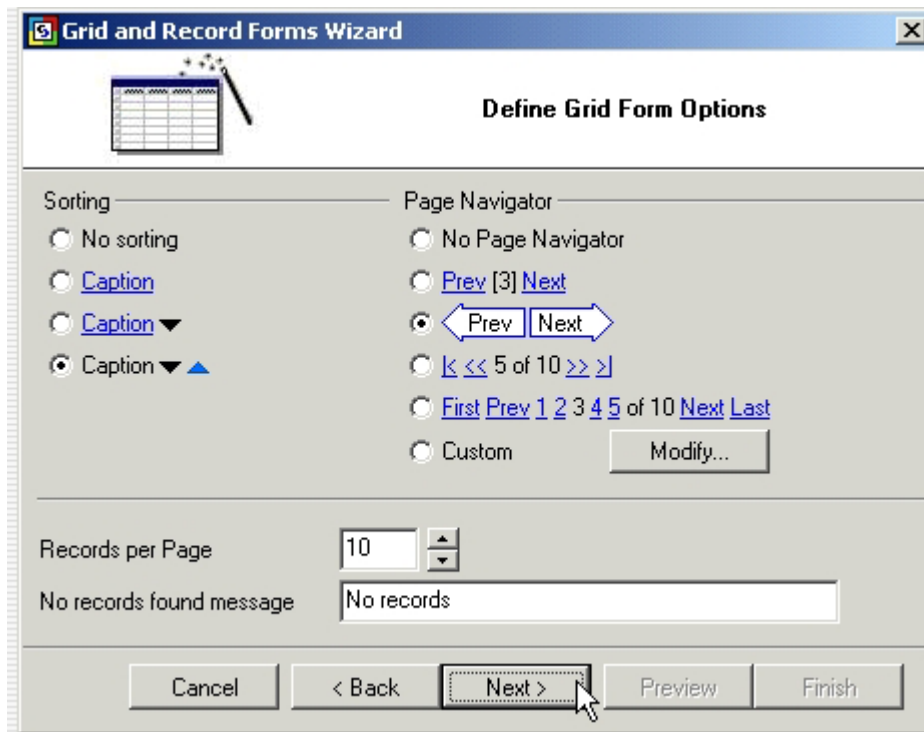
	Select a form field then click this button to move it upwards in the order of appearance
	Select a form field then click this button to move it downwards in the order of appearance
Control Type	Specify the control type to be used for each of the fields in the form



5. You can now specify the type of sorting and navigation controls to appear in the grid form.

Property	Description
Sorting	Select the type of sorting controls that you want to have in the grid.
Page Navigator	Select the type of Navigation controls that you want to have in the grid.
Modify	If you want to have custom navigation controls on the form, click on this button to open the <b>Custom Navigator</b> window where you can customize the controls to suit your needs.
Records per page	Specify the default number of records to be shown per page
No records found message	Specify the message that will be displayed in the grid form if there are no records to be shown





**Grid and Record Forms Wizard**

**Define Grid Form Options**

**Sorting**

- ☐ No sorting
- ☐ Caption
- ☐ Caption ▼
- ☒ Caption ▼ ▲

**Page Navigator**

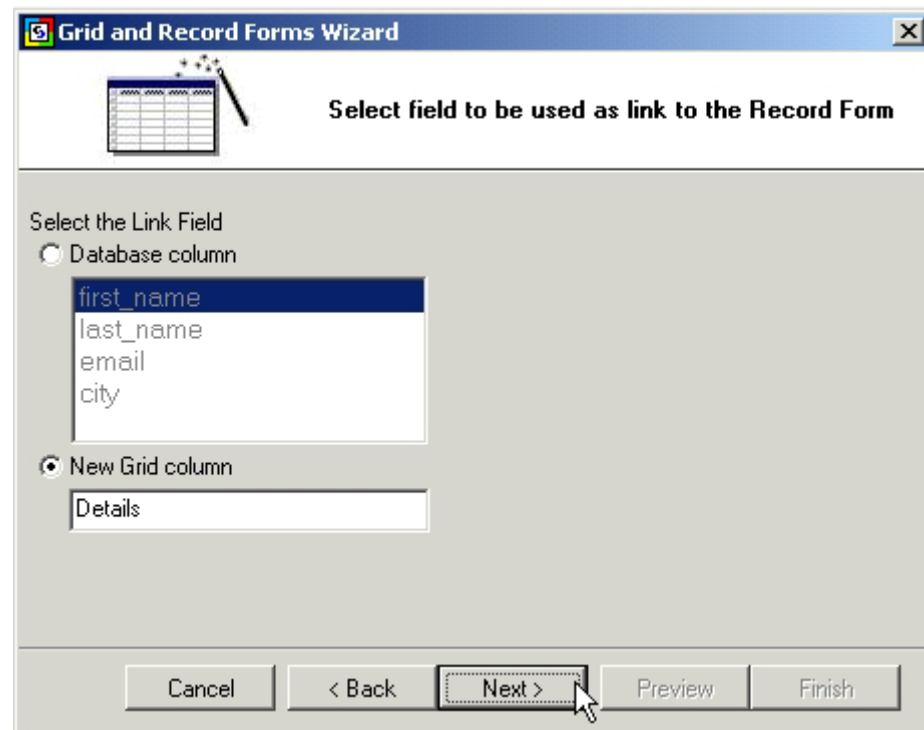
- ☐ No Page Navigator
- ☐ Prev [3] Next
- ☒ ◀ Prev Next ▶
- ☐ < << 5 of 10 >> >
- ☐ First Prev 1 2 3 4 5 of 10 Next Last
- ☐ Custom Modify...

Records per Page: 10

No records found message: No records

Cancel < Back Next > Preview Finish

- The last step in configuring the grid is selecting the field to be used to link the records in the grid form to the details in the record form. You can select any one of the fields that are already in the grid form or alternatively, you can specify a new column whose sole purpose will be to act as the link to the record form.



**Grid and Record Forms Wizard**

**Select field to be used as link to the Record Form**

**Select the Link Field**

- ☐ Database column
  - first\_name
  - last\_name
  - email
  - city
- ☒ New Grid column
  - Details

Cancel < Back Next > Preview Finish

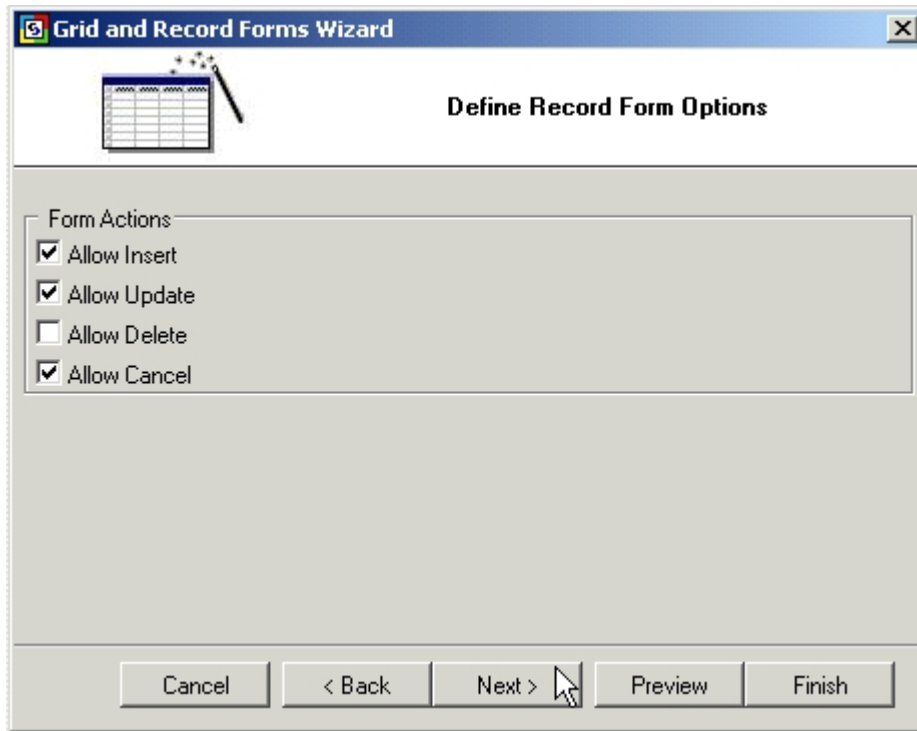
- The next set of windows pertains to the configuration of the record form. The record form will be based on the same table as was the grid form but you have the option of selecting different fields to appear in the record form. After adding the fields, you should also select the Control Type to be used for each of the fields.

At this point, you can use the Preview button to see how the forms look.


members  
 Detail ▲▲▼▼ first\_name ▲▲▼▼ last\_name ▲▲▼▼ email ▲▲▼▼ city ▲▲▼▼  
[Details](#) {first\_name} {last\_name} {email} {city}  
 No records  
[Insert](#) < Prev Prev Next > Next


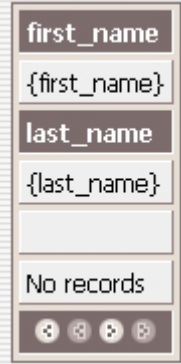
members  
 {Error}  
 first\_name {first\_name}  
 last\_name {last\_name}  
 email {email}  
 city Select Value ▼  
 country\_id {country\_id}  
 Insert Update Delete Cancel

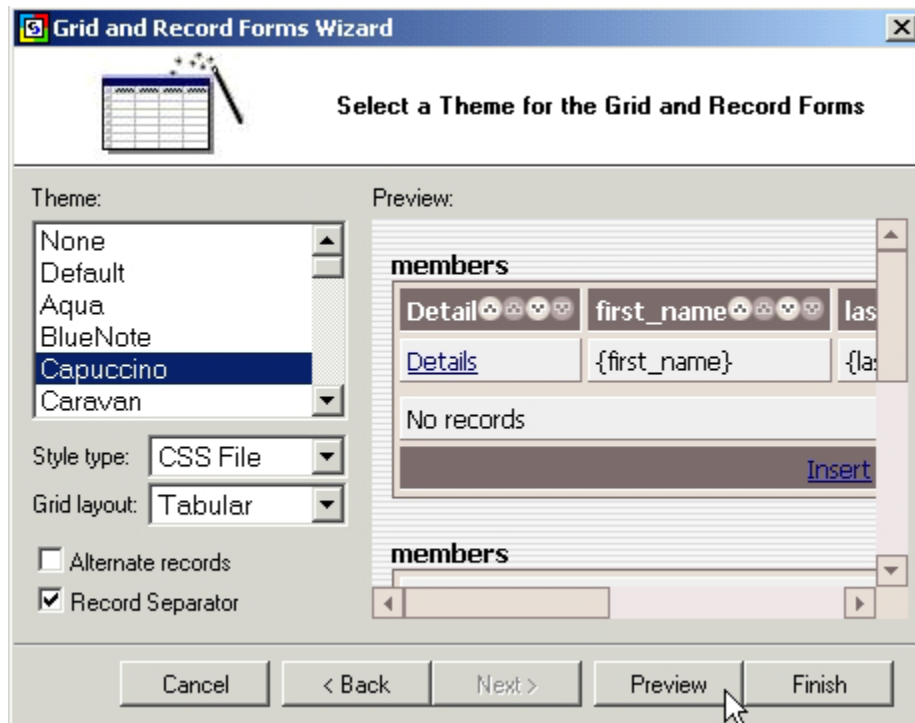
8. Next, select the type of operations you want the user to be able to perform using the record form. In the illustration below, the **Allow Delete** option is disabled meaning that the record form will not allow the user to delete any existing records.



9. In the last step, select a theme to be used to display both forms.

Property	Description
Theme	This listbox contains a list of themes from which you can select from
Style Type	<ul style="list-style-type: none"> <li>▪ HTML</li> <li>▪ CCS File</li> <li>▪ CCS Page</li> <li>▪ CCS Inline</li> </ul>
Grid Layout	<p>Specify the layout of the grid form</p> <ul style="list-style-type: none"> <li>▪ Tabular</li> </ul> <p><b>members</b></p>  <ul style="list-style-type: none"> <li>▪ Columnar</li> </ul>

	<p><b>members</b></p>  <p>▪ Justified</p> <p><b>members</b></p> 
Alternate records	If this option is selected, every other record in the grid will appear in a row with a different background color than the preceding record.
Record Separator	With this option selected, each row where a record appears is followed by an empty row that separates it from the next row with a record.



You can use the **Preview** button to see how the forms will look like then click on the **Finish** button to close the wizard and generate the forms.

The image displays two sequential screenshots of a database wizard interface for a table named "members".

The top screenshot shows the "Preview" stage. It features a table structure with columns: "Detail", "first\_name", "last\_name", "email", and "city". Each column has a small icon to its right. Below the column headers, there is a row of placeholder text: "{first\_name}", "{last\_name}", "{email}", and "{city}". A "Details" link is visible on the left. Below the table structure, a message "No records" is displayed. At the bottom, there is an "Insert" button with a small icon to its right.

The bottom screenshot shows the "Finish" stage. It displays a form with the following fields:

- first\_name**: A text input field containing the placeholder "{first\_name}".
- last\_name**: A text input field containing the placeholder "{last\_name}".
- email**: A text input field containing the placeholder "{email}".
- city**: A dropdown menu with the text "Select Value" and a downward arrow.
- country\_id**: A text input field containing the placeholder "{country\_id}".

At the bottom of the form, there are three buttons: "Insert", "Update", and "Cancel".

## VII. Object Properties

This section deals with the various properties exposed by the **Properties** Window. A good understanding of these properties is essential since the process of creating and customizing pages revolves around configuring different settings for the properties.

Many of the properties encountered are applicable to multiple objects. For instance, a Text Box control shares many similar properties with a Hidden Field control. On the other hand, a Navigator control bears very few similarities with any other object.

### ***Project Properties***

Clicking on the project name within the **Project Explorer** window will expose the project properties under the **Data** tab of the **Properties** window. Note that a project does not have any properties under the **Format** or **Event** tabs. In addition, most of the project properties can be accessed using the **Project → Settings...** menu option that opens the [Project Settings](#) window.

Property	Description
Code Language	Select the programming language in which the code will be generated.
Site Language	Specify the spoken language for the site.
Server URL:	The HTTP URL to be used to access the published pages.
Secured Server URL	Specify the SSL HTTP URL to be used to access the published pages. This option is only used if you want to access the pages over a SSL connection and the server support SSL.
Server Path	The filing system path to where the pages will be published.
Location:	Specify whether the generated pages will be published to a local computer or network or whether they will be published to a remote computer via FTP.
Default Date Format	Select the format in which you want date values to be displayed in the project. If no selection is made, date values will be displayed in the default database format
Security Type	<ul style="list-style-type: none"><li>• None: No site security is implemented</li><li>• Table: Security will be based on information contained in a database table</li><li>• Database: The inbuilt database security will be used for the site</li></ul>
Login Page	Select the project page that will be used to enter authentication details by the users.
Convert URL To	
User ID Stored As	Select the method to be used to store the User ID after a user logs in (Session, URL, Cookies).
Security Connection	Select the connection from where security information will be retrieved.
User Table	Select the database table containing user security information.
User ID Field	Select the database table field containing the unique ID that identifies the users.
User Login Field	Select the database table field containing the login usernames of the users.
User Password Field	Select the database table field containing the login passwords of the users.
Level/Group Field	If applicable, select the database table field containing the security level of the users.
User ID Variable	Enter a name to be used for the user ID variable.

User Login Variable	Enter a name to be used for the user login name variable.
Group ID Variable	Enter a name to be used for the group ID variable.
Level Inclusion	If this option is selected, users with a higher security level will be able to access pages designated with a lower security level. Otherwise, an exact match will be needed in order to access any page.
Character Set CodePage	Set the value for the code page used for setting the body of the HTTP request. This can be necessary when sending data in the body using a character set other than the system default.
Geographic Local LCID	The LCID is a locale identifier that identifies the language in use.

## Form Properties

The different types of forms can be divided into two main categories. Those that submit value (i.e. have a HTML <Form> element) and those that don't. Record, Search and Login forms belong in the first category while the second is composed solely of grid forms. Record, Search and Login forms therefore have similar properties and events while grid forms have their own distinct properties.

## Grid Form Properties

Property	Description
Name	The name of the form.
Connection	The database connection from where information will be retrieved.
Source Type	The type of data source or the method used to retrieve the data.
Data Source	The database column, SQL Query or other expression from where the data will be retrieved.
Source Columns	The columns in the data source that will be available for usage in the form.
Where	An optional 'Where' clause portion of an SQL statement. The 'where' clause could be used to filter the data to be retrieved.
Order By	An optional 'Order By' clause of an SQL statement. The clause could be used to specify the order in which the data is retrieved and displayed.
Restricted	Specify where users must be logged in before accessing the form.
Records Per Page	The numbers of records to be displayed per page by default.
Page Size Limit	The maximum number of records a user may opt to display per page if using a search form that allows control over the number of records displayed.

## Record, Search and Login Form Properties

Property	Description
Name	The name of the form.
Connection	The database connection from which information will be retrieved.
Source Type	The type of data source or the method used to retrieve the data.
Data Source	The database column, SQL Query or other expression from where the data will be retrieved.
Source Columns	The columns in the data source that will be available for usage in the form.
Where	An optional 'Where' clause portion of an SQL statement. The 'where' clause could be used to filter the data to be retrieved.

Order By	An optional 'Order By' clause of an SQL statement. The clause could be used to specify the order in which the data is retrieved and displayed.
Return Page	The page to which the user is directed to after the form has been submitted successfully.
Convert URL To	
Restricted	Specify where users must be logged in before accessing the form.
Custom Insert Type	If you wish to override the default insert mechanism, select the type of method to be used to perform the insert operation.
Custom Insert	Specify the procedure to be used to insert data.
Custom Update Type	If you wish to override the default update mechanism, select the type of method to be used to perform the update operation.
Custom Update	Specify the procedure to be used to update data.
Custom Delete Type	If you wish to override the default delete mechanism, select the type of method to be used to perform the delete operation.
Custom Delete	Specify the procedure to be used to delete data.
Allow Insert	Specify where the form can be used to insert new data.
Allow Update	Specify where the form can be used to update existing data.
Allow Delete	Specify where the form can be used to delete existing data.
Preserve Parameters	Specify whether Get or Post parameters should be preserved.

## Control Properties

This section examines the properties for the various controls found under the **Control** tab of the **Toolbox**. Many of the Controls share similar properties and the table below presents all the possible properties in alphabetic order. The third column indicates the controls that possess the property.

Property	Description	Applicable Controls
Bound Column	Each option in a ListBox or RadioButton control is represented by a pair of values whereby one value is displayed to the user and another is submitted. Bound Column refers to the data source for the values that are submitted.	ListBox, RadioButton
Caption	The name used to refer to the control in an error message.	TextBox, TextArea, ListBox, RadioButton, Hidden
Checked Value	The value submitted by a Checkbox Control when it is checked.	Checkbox
Column	For a Sorter control, this specifies the Grid column that the sorter applies.	Sorter
Connection	The database connection from where information will be retrieved.	ListBox, RadioButton



Control Source	Specify the source of data for the control e.g. the name of a database column	TextBox, TextArea, ListBox, Checkbox, RadioButton, Label, Image, Hidden, Link, ImageLink
Convert URL To	Specify whether the URL should be converted to an absolute URL or a secure (https://) URL.	Button, ImageLink, Link
Data Type	The type of data that will be contained in the control: <ul style="list-style-type: none"> <li>• Text</li> <li>• Integer</li> <li>• Float</li> <li>• Date Boolean</li> <li>• Memo</li> </ul>	TextBox, TextArea, Checkbox, RadioButton, ListBox, Label, Hidden, Link, ImageLink
DBFormat	Specify the format in which you want the control value to be entered into the database e.g. the user may enter a date value in one format but you want it to be entered into the database in a different format.	TextBox, TextArea, Label, Hidden
Default Button	[Yes/No] If you select 'Yes' the button will be the default button on the form so that clicking on the Enter key will be equivalent to clicking on the button.	Button
Default [Value]	Specify a default value to be used for the control if no value is otherwise specified.	TextBox, TextArea, Checkbox, RadioButton, ListBox, Label, Image, Hidden, Link, ImageLink
Enable Validation	[Yes/No] Specify whether validation should be enabled when the button is clicked.	Button
Error Control	Specify the name of a control that will be used to display any error messages pertaining to the	TextBox, TextArea, Checkbox, RadioButton, ListBox, Hidden

Format	Depending on the Data Type property, select the format of the data to be contained in the control.	TextBox, TextArea, Label, Hidden, Link, ImageLink
Href Source	The page path, database column or other expression containing the URL of the hyperlink.	ImageLink
Href Type	The type of source where the URL value will come from.	ImageLink, Link
HTML	[Yes/No] If 'Yes' is selected, the control value will not be converted to text before being displayed. As such, if the value is HTML markup, it will be appear as regular HTML markup on the page.	TextArea, TextArea, Label, Link, ImageLink
Input Verification	Select a regular expression to be used to check the validity of the data entered in the control.	TextBox, Hidden
List Data Source	Specify the actual source of data for the list of values depending on the List Source Type.	ListBox, RadioButton
List Source Type	Specify the type of source of data for the list of values to be shown in the control e.g. the name of a database column.	ListBox, RadioButton
Name	The name of the control.	ALL
Operation	The type of operation performed when the button is clicked [Insert, Update, Delete, Cancel or Search]	Button
Order By	Specify an Order By clause for the SQL statement used to retrieve values for the control.	RadioButton, ListBox
Page Links	Specify the number of page links to be shown on the Navigator. The default is 10.	Navigator
Preserve Parameters	Specify whether GET or Post parameters should be preserved.	ImageLink, Link
Remove Parameters	Specify a semicolon-separated list of parameters that should be removed from the hyperlink.	ImageLink, Link
Required	[Yes/No] Select 'Yes' if a value must be entered for the control in order for the form to be submitted successfully.	TextBox, TextArea, RadioButton, ListBox
Return Page	Specify the page where the user should be directed after the button operation has been performed.	Button

Source Type	<p>The type of data source that will provide data for the control:</p> <ul style="list-style-type: none"> <li>• Database Column</li> <li>• Database Expression</li> <li>• Code Expression</li> </ul>	TextBox, TextArea, ListBox, Checkbox, RadioButton, Label, Image, Hidden, Link, ImageLink
Source/Parameters	This opens the <b>Href Source</b> window where you can specify the page to be linked to as well as any parameters that should be transmitted via the URL.	Link
Text Column	Each option in a ListBox or RadioButton control is represented by a pair of values whereby one value is displayed to the user and another is submitted. Text Column refers to the data source for the values that are displayed.	ListBox, RadioButton
Type	Select the type of page links to be shown in the Navigator control.	Navigator
Unchecked Value	The value submitted by a Checkbox Control when it is unchecked.	Checkbox
Unique	[Yes/No] Select 'Yes' if you want a check to be made to ensure that any value entered in the control does not already exist in the database.	TextBox, TextArea, RadioButton, ListBox, Hidden
Validation Rule	Specify an expression that will be used to check the validity of the data entered in the control.	TextBox, TextArea, Checkbox, RadioButton, ListBox, Hidden
Validation Text	Specify the text that will be displayed if the data entered in the control does not meet the validation rule.	TextBox, TextArea, Checkbox, RadioButton, ListBox, Hidden
Where	Specify a Where clause for the SQL statement used to retrieve values for the control.	RadioButton, ListBox

## HTML Properties

The properties that appear under the **Data** tab of the **Properties** window pertain to objects that are created using the **Control** tab of the **Toolbox**. However, the **Format** tab of the **Properties** windows also exposes many other properties that apply to HTML elements in general. When the **Document** window is in **HTML** mode, the **Format** tab of the **Properties** window displays HTML properties for the HTML element where the cursor is currently located.

Different HTML elements have different properties while some properties are common to some elements. You will also notice that along with the properties, there are JavaScript Events located at the bottom of the **Properties** window **Format** tab. The events vary according to the selected element and can be used to execute JavaScript code in response to certain actions.

Both the properties and events under the **Format** tab are standard HTML properties and events and as such, you can refer to suitable HTML documentation for their meaning and usage.

## Connection Properties

Selecting a project connection in the **Project Explorer** window will expose its properties under the **Data** tab of the **Properties** window. Connections have no Events not **Format** properties. The following are the available connection properties.

Property	Description
Name	The name of the connection. Has to be unique.
Database Type	The type of database being accessed. If your database type is not listed, select the ANSI SQL-92 option.
Connection Type	The type of connection. ODBC or Connection String.
Connection String	Depending on the type of connection, this can either be an ODBC DSN or a connection string.
Login	The user login name needed to access the database.
Password	The user password needed to access the database.
Date Format	The format in which date values should be entered into the database.
Boolean Format	The format in which Boolean values should be entered into the database.
Catalog	Specify a database catalog from which to display objects during design time.
Schema	Specify a database schema from which to display objects during design time.
Views	[Yes/No] Select 'Yes' to enable the display of database views during design time.
Synonyms	[Yes/No] Select 'Yes' to enable the display of database Synonyms during design time.
System Tables	[Yes/No] Select 'Yes' to enable the display of database System Tables during design time.
System Views	[Yes/No] Select 'Yes' to enable the display of database System Views during design time.
Aliases	[Yes/No] Select 'Yes' to enable the display of database aliases during design time.
Server Same As Design	[Yes/No] Select 'Yes' if the same connection will be used for design time as well as on the server.
Server Connection	If the server connection is different from the design connection, specify whether

Type	the server connection will use an ODBC DSN or connection string.
Server Connection String	If the server connection is different from the design connection, specify the ODBC DSN or connection string to be used.
Server Login	If the server connection is different from the design connection, enter the user login name needed to access the server database.
Server Password	If the server connection is different from the design connection, enter the user password needed to access the server database.

## VIII. Working with Documents

A good deal of the time spent using CCS will involve the use of the document window. The document window serves multiple purposes depending on the mode in which it is currently set to. The five different modes in which the document window can be set are:

- **Design:** This is a visual representation of the HTML content of a page as it would appear if rendered by a web browser. The user is able to add, modify and remove different components of the page. The Design mode is also known as ‘What You See Is What You Get’ (WYSIWYG).
- **HTML:** This shows the actual HTML code for a page as well as any style and Javascript code that might be on the page. If you are familiar with HTML, you can add your own HTML code or adjust the existing code as desired. Switching to Design mode will show the appearance of the HTML code when rendered in a browser.
- **Code:** This shows the language specific code for a page. For instance, if using the ASP & Templates language, this mode will display the ASP/VBScript code for a page.
- **Preview:** This mode is used to preview a page without actually running it off the server. The page is shown without the special tags or borders that are visible in Design mode. The user cannot make any changes to the page while in Preview mode.
- **Live Page:** This involves actually executing the page from the server. Any database content, include pages and styles are displayed in their entirety. The page is fully functional and can be used to navigate, post or adjust database content or perform any other function that the page is designed to have. Error messages are displayed if problems are encountered with any of the page code.

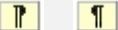





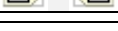
### Design Mode






Design mode allows the user to construct a page while being able to immediately see the appearance of the different components on the page. As the name implies, Design mode is more concerned with the appearance of the page as opposed to its functionality.

While in design mode, you can use the Toolbox to add various components to the page such as form controls, plain text and images. Tables and cascading style sheets can be used to format the layout of the page.

The **Properties** window displays the pertinent properties based on the component that is selected in the document window. You can then set or adjust these properties as you wish.

While in design mode, a number of icons are used to give a visual representation of otherwise non-visual elements. The following table shows the meaning of these icons.

Icon	Represent
	Beginning and end paragraph tags <p> </p>.
	Line break tag  .
	Beginning and end form tags <form> </form>.
	Submit button <input type=“submit”>.
	An Include page.
	Mark the beginning and end of a Grid form .
	Mark the beginning and end of a Record or Search form. Both these forms

	submit values and have a HTML form <form></form> within them.
 	A Label control based on database content.
 	A Link control based on database content.
	Represents an ImageLink or Image control. The actual image will be visible then the page is viewed in Live Page mode.

While in design mode, you can toggle the display of the Paragraph and Line break icons by pressing the *ctrl*+/*/* shortcut key or using the View → Reveal Tags menu option.

You will also notice that some of the text displayed will be enclosed within curly braces (e.g. {article}). The content in curly braces are termed as **Blocks** and are indication that the text will be substituted for database content when the page is viewed live. Care should be observed when handling any blocks since if they are inadvertently changed, then the corresponding database content would not be displayed. As a rule, you should always use the **Properties** window or the **Project Explorer** window to change the properties of this content or if you need to rename or remove it. Note that you can change the position of the blocks relative to the other page contents without causing any harm. What could be potentially harmful is changing the block itself i.e. spelling, or deleting a block.

## HTML Mode

This displays the raw HTML code for a page. Along with HTML content, there could be Javascript code that was added manually or using the events tab of the Properties window. For pages that make use of Cascading Style Sheets, style content will also be visible here depending on the method used to apply the style.

Just as was the case in **Design** mode, you will see content within the HTML code that is enclosed within curly braces. As usual, caution should be observed when editing this content and this should only be done using the **Properties** window or **Project Explorer** window. In HTML mode, there is another form of block that is not evident in Design mode. This occurs in the form of HTML comments and is used to mark out the beginning and end of content that can be viewed as a logical unit.

```
<!--BEGIN blockname-->

Block contents go here

<!-- END blockname -->
```

There are numerous types of logical units that are represented by blocks. The common denominator is that there is a HTML comment to mark their beginning and another to mark their end. Additionally, it is possible to have nested blocks such that one block is within another. In such scenarios, the naming of the blocks makes it clear where each of the blocks begins and ends. It is therefore important to make sure that none of the block comments are altered or removed otherwise the page would not function as correctly.

Any client side events that are added using the **Events** tab of the **Properties** window are also visible in HTML mode. The script code appears at the top of the page in the <head> </head> section. The illustration below shows the code that is added for an On Load event of a page to set focus to a field in a form.

```

<script language="JavaScript">
//Begin CCS script
function page_OnLoad() { //page_OnLoad @1-1C723708

//Set Focus @5-84B91275
    document.Form1.TextBox1.focus();
//End Set Focus

} //Close page_OnLoad @1-FCB6E20C

//bind_events @1-F4D605D7
function bind_events() {
    page_OnLoad();
}
//End bind_events

window.onload = bind_events; //Assign bind_events @1-19F7B649

//End CCS script
</script>

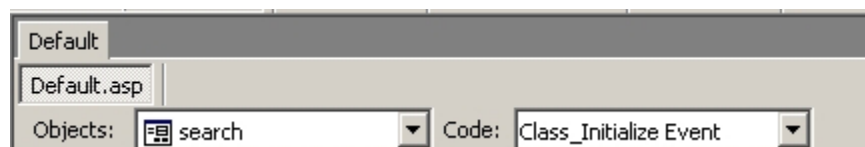
```

Notice that event code is displayed with a different background color compared to the rest of the HTML content. Usually the background color is light gray and signifies that the code was generated by CCS and has not been altered. If you modify or add your own code anywhere within the gray background, the background will change to default color of the rest of the document. This is an indication that the code is different from what CCS generated.

## Code Mode

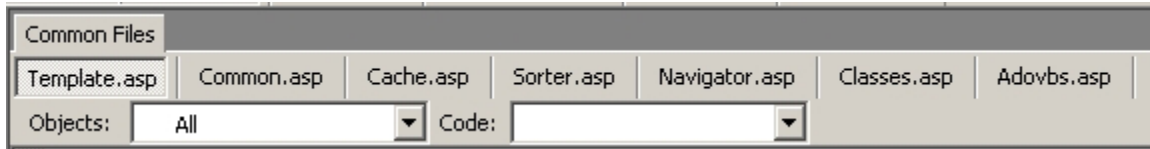
Code Mode displays the programming code for a page. By default, the code in a page is displayed against a gray background to indicate that it has not been altered from its original state. Making changes to the code will cause the background to change to white by default. In addition to this, all the code will be color coded in order to distinguish various parts of the code. For instance, commented code will appear in a different color compared to Strings or even reserved words.

It is not unusual to have pages with upwards of a thousand lines of code. To assist in navigating large amounts of code, the document window makes it possible to jump to specific points in the code by selecting an object then a member of the object to bring into focus. Whenever appropriate, the two listbox fields at the top left of the document window will display the Objects in the page as well as the members of that object if any. Selecting a particular object then it's member in the Code field will cause the window to scroll to the location where the relevant code is located.





In each project, regardless of coding language, there is always a set of **Common Files**. These files contain functionality that is needed often in multiple pages and hence it is placed in a central files. In order to access these files, click on the **Common Files** Icon in the **Project Explorer**. The files will be opened in the document window and you will be able to access a specific file by clicking on the tab with its name.



The number and names of the Common Files will vary depending on the coding language being used. The illustration above shows the common files for the ASP language.

## ***Preview Mode***

Preview mode shows how the static HTML part of a page would look like when viewed in a browser. None of the programming code is executed or published to the server. Preview mode is intended to show the layout of the page without the editing features that are visible in HTML mode. It is not possible to make changes to a page while in Preview mode.

## ***Live Page***

In Live Page Mode, the page is published to the server and executed such that dynamic content is made available for viewing. Before attempting to view a page in Live Page mode, you should make sure that the Project Publishing settings have been properly configured so that the pages are generated to the correct location.

In this mode, you can navigate to other pages and manipulate database content as you would normally do if the page were viewed using a browser window.

If problems are encountered during the process of generating or publishing the page, the **Messages** window will show appropriate error messages. Additionally, if the generated pages themselves are not properly created, the document window will show the error message returned from the server.

## ***Using the Front Page Add-In***

If you installed the FrontPage Add-in, you can take advantage of the robust HTML and code editing functionality provided by FrontPage to enhance the development process. The Add-In has the effect of adding the main CCS functionality to FrontPage so that you can create entire project using the combined power of CCS and FrontPage.

When you open FrontPage, you will notice that the menu bar has a menu for CodeCharge as shown in the illustration below.



In order to bring up the other CCS toolboxes and windows, you need to open an existing project or create a new one. Once you open or create a project, the **Project Explorer** and **Properties** windows will become visible as well as other CCS toolbars. Using the CodeCharge → Windows menu option, you can also make the **Message** and **Toolbox** windows visible.

With the exception of two commands that are unique to the FrontPage add-in toolbar, the rest of the commands in the toolbar shown below are similar to the commands within CCS.



Command	Description
	Use this button to open a window with the programming code for a particular page.
	Use this button to generate the current page and view it in live page mode.

The add-in also features a toolbar with CCS controls. These same controls are available under the **Controls** tab of the **Toolbox**.



There also is a toolbar that can be used to create various form types. You can hover over a particular option to view the tooltip text for the option. Again, the same options are available under the **Forms** tab of the **Toolbox**.



Note that FrontPage workspace is very customizable and you can arrange the various windows as well as toolbars in the manner that best accommodates your working requirements.

While using the FrontPage add-in confers considerable leverage to performing various design tasks that would otherwise be tedious within CCS, caution should be exercised so as not to alter the blocks of code that have to remain intact in order for the pages to function properly. Any content within curly braces {example}, should not be directly edited and neither should the HTML nor the code comments.

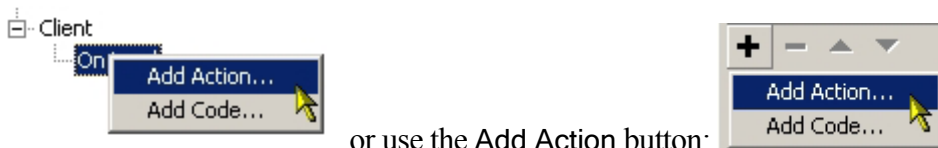
## IX. Client Side Event Actions

The **Properties** window features two categories of events. There are the **Server** events that are implemented in programming language and there are **Client** events that are implemented using browser-scripting languages. The former are executed entirely on the server before a page is sent back to the requesting browser while the latter are executed by the browser of the client machine that requested the page.

By their nature, client side events are used mainly to provide interactivity without having to post feedback to the server. A client side script can respond to a user action and continuously provide feedback entirely on the client side. This is made possible by the event-driven nature of HTML objects that provide a mechanism to respond to particular actions using event handlers. In fact, all the client events are event handlers for a corresponding browser document object. The event names are not unique to CCS but rather are the same names used in the document object model (DOM) for a browser document.

There are many common tasks often performed in response to events and for this reason, CCS provides a number of **Actions** that are essentially snippets of code that implement often used functionality. Instead of having to hand code this functionality each time it is needed, you can simply add an Action and customize it for the particular situation.

Actions are added via the events listed under the Events tab of the **Properties** window. You can either right-click on a client event then select the Add Action... option:



or use the Add Action button:

Where an action is placed will affect its execution or failure to execute. When placing an action, you should bear in mind the event that you anticipate to trigger the action. If the event does not occur, then the action will not be executed.

### Set Focus

This action is used to set focus to a particular object in the window. For an input field, having focus means that anything typed by the user will be entered into the field. For a button, having focus means that pressing the enter key is equivalent to clicking on the button.

Use this action to set focus to a particular field for the convenience of the user or in cases where you want input to begin at a particular field.

When you add this action, there are two properties you have to provide.

Property	Description
Form	The name of the form in which the control to be focused on is located.
Control	The name of the control to receive focus.

## ***Validate Maximum Length***

This action is usually used in conjunction with input fields and serves to ensure that the value entered in a field does not exceed a specified number of characters. Among other places, this action could be placed in the On Click event of a button in the form or the On Change event of the field being validated.

The following are the properties that need to be configured for the action.

<b>Property</b>	<b>Description</b>
Control Name	The name control to be validated.
Maximum Length	The maximum allowable length of input in the control.
Error Message	The error message to be displayed if the maximum length is exceeded.

## ***Validate Minimum Length***

This action is usually used in conjunction with input fields and serves to ensure that the value entered in a field is at least a specified number of characters long. Among other places, this action could be placed in the On Click event of a button in the form or the On Change event of the field being validated.

The following are the properties that need to be configured for the action.

<b>Property</b>	<b>Description</b>
Control Name	The name control to be validated.
Minimum Length	The minimum allowable length of input in the control.
Error Message	The error message to be displayed if the input is less than the minimum length.

## ***Validate Maximum Value***

This action is usually used in conjunction with input fields that receive numeric values. The action ensures that the value entered is less than a specified maximum value. Among other places, this action could be placed in the On Click event of a button in the form or the On Change event of the field being validated.

The following are the properties that need to be configured for the action.

<b>Property</b>	<b>Description</b>
Control Name	The name control to be validated.
Maximum Value	The maximum allowable input value.
Error Message	The error message to be displayed if the input value exceeds the maximum value.

## ***Validate Minimum Value***

This action is usually used in conjunction with input fields that receive numeric values. The action ensures that the value entered is more than a specified minimum value. Among other places, this

action could be placed in the On Click event of a button in the form or the On Change event of the field being validated.

The following are the properties that need to be configured for the action.

Property	Description
Control Name	The name control to be validated.
Minimum Value	The minimum allowable input value.
Error Message	The error message to be displayed if the input value exceeds the maximum value.

### ***Validate Required Value***

This action is usually used for any control that accepts input and ensures that the control has a value. The action can be placed in the On Click event of a button among other places.

The following are the properties that need to be configured for the action.

Property	Description
Control Name	The name control to be validated.
Error Message	The error message to be displayed if the control has no value.

### ***Regular Expression Validation***

This action is used to validate the input in a control based on a specified regular expression.

The following are the properties that need to be configured for the action.

Property	Description
Control Name	The name control to be validated.
Regular Expression	The regular expression to be used for validation.
Error Message	The error message to be displayed if the control has no value.

### ***Validate Entry***

Property	Description
Required	
Minimum Length	
Maximum Length	
Input Mask	
Regular Expression	
Error Message	
Validation Function	

### ***Validate Form***

Property	Description