

[http://msdn.microsoft.com/en-us/library/879kf95c\(VS.80\).aspx](http://msdn.microsoft.com/en-us/library/879kf95c(VS.80).aspx)

Walkthrough: Creating a Web Site with Membership and User Login (Visual Studio)

Creating the Web Site

If you have already created a Web site in Microsoft Visual Studio (for example, by working with the topic [Walkthrough: Creating a Basic Web Page in Visual Web Developer](#)), you can use that Web site and skip to "Configuring Membership" later in this walkthrough. Otherwise, create a new Web site and page by following these steps.


To create a local IIS Web site

1. Open Visual Studio.
2. On the **File** menu, click **New Web Site**.

The **New Web Site** dialog box appears.

3. Under **Visual Studio installed templates**, select **ASP.NET Web Site**.
4. In the **Location** list box, select **HTTP**. Click **Browse**.

The **Choose Location** dialog box appears.

5. Select **Local IIS**.
6. Open **Local Web Servers**.
7. Select **Default Web Site**.
8. Click the **Create New Web Application** icon () above the list of Web sites and then name the new Web site **membership**.
9. Click **Open**.

The **Choose Location** dialog box closes.

10. In the **Languages** box, click the programming language you prefer to work in.

The programming language you choose will be the default for your Web site, but you can set the programming languages for each page individually.

11. Click **OK** in the **New Web Site** dialog box.

Visual Web Developer creates the Web site and a new page named Default.aspx.

Configuring Membership

Later in this walkthrough you will put pages into a subdirectory that is protected. You must create the subdirectory now so that you can configure security for it later in the walkthrough.

To add a new folder to the Web site

1. In **Solution Explorer**, right-click the name of your Web site and click **New Folder**.
2. Name the folder **MemberPages**.

Before you work with ASP.NET membership, you must configure your application to enable membership and to set up users. You can use the Web Site Administration tool, which provides a wizard-like interface for making configuration settings.

For this walkthrough, you will define a single user.

To create a membership user

1. On the **Website** menu, click **ASP.NET Configuration**.
2. Select the **Security** tab, click the link to **Use the security Setup Wizard to configure security step by step**, and then click **Next**.
3. Proceed to Step 2 of the wizard and select the **From the Internet** option.

The wizard displays a page where you can select the authentication method that your Web site will use. This option specifies that your application will use Forms authentication, where users will log in to the application using a login page that you will create later in this walkthrough.

4. Click **Next**.

The wizard displays a message stating that user information will be stored using **Advanced provider settings**. By default, membership information is stored in a Microsoft SQL Server Express database file in the **App_Data** folder of your Web site.

5. Click **Next**.

The wizard displays an option to create roles. You will perform this step separately later in the walkthrough. Therefore, do not select the **Enable roles for this web site** check box.

6. Click **Next**.

The wizard displays a page where you can create new users.

7. Enter information that defines a user of your application. Use the following values as guidelines (you can use any values that you like, but be sure to note your entries for later in the walkthrough):
 - **User Name** Your name (with no spaces), or a sample name.
 - **Password** A password. A strong password is required (one that includes uppercase and lowercase letters, punctuation, and that is at least eight characters long).
 - **E-mail** Your personal e-mail address. Later in the walkthrough you will send yourself an e-mail message, so you need a legitimate e-mail address.
 - **Security Question and Security Answer** Type a question and answer that can be used later if you need to recover your password.
8. Click **Create User**.

The wizard displays a confirmation page.

Note

Leave the Web Site Administration tool open.

Earlier in the walkthrough you created a folder named **MemberPages**. In this part of the walkthrough, you will create a rule that makes sure that only logged-in users can access pages in that folder.

To set up access rules for the MemberPages subdirectory

1. In the wizard, click **Next**.

The wizard displays a page that allows you to create access rules.

2. In the **Add New Access Rule** box, expand the node for your Web site.
3. Select **MemberPages**, the folder you created earlier.
4. Under **Rule applies to**, select **Anonymous users**.
5. Under **Permission**, select **Deny**.

The rule you are creating denies access to anonymous users — that is, users who have not logged in.

6. Click **Add This Rule**.

The new rule is displayed in the grid below. When users request a page from the **MemberPages** subdirectory, the rules are checked to determine whether the user is allowed access to the page.

7. Click **Finish**.

You are now done with the wizard. The wizard closes and you are returned to the **Security** tab of the Web Site Administration tool.

Configuring the Application for E-Mail

For part of this walkthrough, the application needs to be able to send e-mail messages. To send messages, your application must have access to a Simple Mail Transport Protocol (SMTP) server, which forwards e-mail messages from your application to an e-mail recipient.

IIS includes the Default SMTP virtual server as an optional component, which is suitable for this walkthrough. For more information about configuring this server, see [How to: Install and Configure SMTP Virtual Servers in IIS](#). If you are working on a local area network, check with your network administrator for information about access to an e-mail server.

After you have set up or determined how to access an SMTP server, you must configure your application to route e-mail messages to that server. You can do so by making an entry in your Web site's Web.config file, which contains a series of settings that determine how your application runs.

To configure the application to use a specific SMTP server

1. In the Web Site Administration tool, click the **Application** tab.
2. Under SMTP Settings, click **Configure SMTP e-mail settings**.

The tool displays a page where you can configure e-mail.

3. If you are using the SMTP virtual server that is on your computer, enter **localhost** as the **Server Name**; otherwise, enter the appropriate server name.

Include information for the port number and for authentication according to the requirements of your SMTP server. See your administrator for more information on how to configure these settings.

4. In the **From** box, type a valid e-mail address.
5. Click **Save**, and in the confirmation page, click **OK**.

The Web Site Administration tool creates a Web.config file (if one did not already exist) with the settings you have made.

Note

The Web.config file will not appear in Solution Explorer until you refresh the view.

6. Close the Web Site Administration tool.

Logging the User In

As part of your application, you need to establish the user's identity so that the application can perform actions — such as showing or hiding information — based on who the user is. To get the user's identity, you have the user log in.

In this walkthrough, you will add a link on the home page that takes users to a login page, and then you will create the login page.

To create a home page with a login button

1. Open or switch to the **Default.aspx** page of your site. (If you do not have a Default.aspx page, you can add one or use a different page.)
2. Switch to **Design** view.
3. Type static text such as **Welcome to our site** and, in the **Formatting** toolbar, use the **Block Format** drop-down list to format the text as Heading 1.
4. From the **Login** group of the **Toolbox**, drag a **LoginStatus** control onto the page.

By default, the **LoginStatus** control is rendered as a link. When users click it, the application displays a login page. You can now create the login page.

To create a login page

1. In **Solution Explorer**, right-click your Web application and select **Add New Item**. Add a **Web Form** named **Login.aspx** to your site.

Note

For this walkthrough, the page must be named **Login.aspx**. By default, forms authentication is configured to work with a page with this name. Although you will not do so in this walkthrough, you can change the default login page — the page to which users are redirected — in the Web.config file.

2. In the Login.aspx page, switch to **Design** view.
3. From the **Login** group of the **Toolbox**, drag a **Login** control onto the page.

The **Login** control is a single control that will prompt the user for credentials and validate them.

Displaying Login Errors

The **Login** control includes validation to help users enter correct information. For example, if a user skips the password, a validator control displays an asterisk (*) next to the **Password** box. You can provide more complete information for login errors by adding a **ValidationSummary** control to the page.

To display detailed login errors

1. From the **Validation** group of the **Toolbox**, drag a **ValidationSummary** control onto the page.
2. In the **Properties** window for the **ValidationSummary** control, set the **ValidationGroup** property to **Login1**, which is the default ID of the **Login** control you added previously.

Displaying Information for Logged-In Users

You will now modify the home page to customize the display depending on whether the user is logged in. Anonymous users will see a generic message inviting them to log in. Logged-in users will see a message that welcomes them by their logged-in name.

To customize the display for logged-in users

1. Switch to or open the Default.aspx page.
2. From the **Login** group of the **Toolbox**, drag a **LoginView** control onto the page.

The **LoginView** control is displayed with its **AnonymousTemplate** template open. This template allows you to define the content that users will see before they have logged in.

3. Click the edit area of the **LoginView** control to activate editing.
4. In the edit area of the **LoginView** control's **AnonymousTemplate** template, type **You are not logged in. Click the Login link to sign in.**
5. On the **LoginView Tasks** panel, in the **Views** list, click **LoggedInTemplate**. If you do not see the **LoginView Tasks** panel, right-click the heading of the **LoginView** control and select **Show Smart Tag**.

You are now defining the content that will be displayed to users who have already logged in.

6. Click the edit area of the **LoginView** control to activate editing, and then type **You are logged in. Welcome.**
7. From the **Login** group of the **Toolbox**, drag a **LoginName** control into the template after the text.

Testing Login

You can now test the login capability of your application.

To test login

1. In **Solution Explorer**, right-click **Default.aspx** and click **Set As Start Page**.

This configures the Web site so that when you run the site, the Default.aspx page appears first.

2. Press CTRL+F5 to run the Web site.

The home page (Default.aspx) appears in the browser, showing the **Login** link and the generic message.

3. Click the **Login** link.

The login page you created is displayed.

4. Type the login name of the user you created earlier in the walkthrough, and then click **Log In**. (Do not yet enter a password.)

An asterisk (*) is displayed next to the **Password** box, and an error message is displayed in the **ValidationSummary** control.

5. Type both a user name and password and then click **Log In**.

If you entered correct credentials, you are returned to the home page. The page now displays a **Logout** link, your user name, and the welcome message that you defined for the logged-in user.

6. Close the browser.

Limiting Access for Members-Only Pages

A typical task in many Web sites is to configure pages so that only logged-in users can view the pages. Earlier in the walkthrough, you created the **MemberPages** subdirectory and created a rule that limits access to pages in the subdirectory. In this section of the walkthrough, you will add a page to the protected subdirectory and test the access rule.

To create the members-only page

1. In **Solution Explorer**, right-click the **MemberPages** folder, click **Add New Item**, and add a new **Web Form** named **Members.aspx**.

Note

Be sure to create the page in the **MemberPages** folder.

2. In **Design** view, add text to the page, such as **Welcome, members!** The exact text does not matter, as long as you will be able to recognize this page when you see it in the browser.

You can now add a link to the members-only page from the home page. In a real application, you would probably put the members-only page link in the logged-in template of the **LoginView** control. That way, visitors to your site would not see the link until they were logged in. For this walkthrough, however, you will make the link available to all users so that you can see the effect of trying to view a members-only page without first logging in.

To add a link to the members-only page

1. Switch to or open the Default.aspx page.
2. From the **Standard** group of the **Toolbox**, drag a [HyperLink](#) control onto the page.
3. In the **Properties** window for the **HyperLink** control, set the **Text** property to **Members page** and the **NavigateUrl** property to `~/MemberPages/Members.aspx` to point to the page that you created previously.

Testing the Members-Only Page

You can test the members-only page by accessing it both as an anonymous user and a logged-in user.

To test the members-only page

1. Press CTRL+F5 to run the Web site.
2. When the Default.aspx page appears in the browser, do not log in. Instead, click the **Members page** link.

You are redirected to the Login.aspx page because access to the members page is denied for anonymous users.

3. In the login page, type the user name and password that you used earlier in the walkthrough to log in.

You are redirected to the Members.aspx page because the user name you are logged in as has been authorized to access the page.

4. Close the browser window.

Creating New Users

In the first part of the walkthrough, you created a user with the Web Site Administration tool. That strategy is useful if you are working with a small, defined list of users; for example, if you are creating users for a small team. In many Web sites, however, users are allowed to register themselves. ASP.NET includes the [CreateUserWizard](#) control that performs the same task you performed earlier using the Web Site Administration tool.

In this part of the walkthrough, you will add a facility that allows users to register on your Web site. To start, you will create a registration page.

To create a registration page

1. In **Solution Explorer**, right-click the name of your Web site, click **Add New Item**, and add a new **Web Form** named **Register.aspx**.

Note

Be sure to create the page in the root of the Web site, not in the **MemberPages** folder.

2. In the Register.aspx page, switch to **Design** view and type static text such as **Register** into the page. In the **Formatting** toolbar, use the **Block Format** drop-down list to format the text as Heading 1.
3. From the **Login** group of the **Toolbox**, drag a **CreateUserWizard** control onto the page.

4. In the **Properties** window for the **CreateUserWizard** control, set the [ContinueDestinationPageUrl](#) property to `~/Default.aspx`.

This configures the control so that when users click **Continue** after creating a user, the control returns to the home page.

5. From the **Standard** group of the **Toolbox**, drag a **HyperLink** control onto the page. In the **Properties** window for the **HyperLink** control, set the **Text** property to **Home** and the **NavigateUrl** property to `~/Default.aspx`.

You can now add a link to the home page that displays the registration page. For this walkthrough, assume that you want to display the registration link only to users who are not logged in.

To create a registration link on the home page

1. Switch to or open the Default.aspx page.
2. Right-click the **LoginView** control added previously, and select **Show Smart Tag**. In the **LoginView Tasks** panel, select **AnonymousTemplate** from the **Views** list box to activate editing in the anonymous template.
3. From the **Standard** group of the **Toolbox**, drag a **HyperLink** control into the anonymous template. In the **Properties** window for the **HyperLink** control, set the **Text** property to **Register** and the **NavigateUrl** property to **Register.aspx**. The **Register** link will be displayed only to users who are not logged in.

You can now test the registration process.

To test registration

1. Press CTRL+F5 to run the Web site and display the Default.aspx page.

Because you are not logged in, the page containing the **Register** link is displayed.

2. Click the **Register** link.

The registration page is displayed.

3. In the text boxes, enter a new user name, a strong password, an e-mail address, and a security question and answer. (All five pieces of information are required.)
4. Click **Create User**.

A confirmation message is displayed.

5. Click the **Continue** button.

You are returned to the home page as a logged-in user. Note that the **Login** link has changed to **Logout** and that the information displayed in the **Login** control is from the [LoggedInTemplate](#) property, not from the [AnonymousTemplate](#) property.

6. Click the **Logout** link.

The page changes to display the information for anonymous users.

7. Click the **Login** link.

8. Enter the credentials for the user you just created.

You are logged in as the new user.

9. Close the browser window.

Changing Passwords

Users sometimes might want to change their passwords, and it is often impractical to perform this task by hand. You can therefore use another ASP.NET control to allow users to change passwords on their own. To change a password, users must know their existing password.

In this walkthrough, you will add a page where logged-in users can change their password.

To create a password-change page

1. In **Solution Explorer**, right-click the **MemberPages** folder, click **Add New Item**, and add a new **Web Form** named **ChangePassword.aspx**.

Note

Be sure to create the page in the **MemberPages** folder.

2. You are putting the page in the members-only folder because only logged-in users can change their passwords.
3. In the **ChangePassword.aspx** page, switch to **Design** view and type static text such as **Change Password**. In the **Formatting** toolbar, use the **Block Format** drop-down list to format the text as **Heading 1**.
4. From the **Login** group of the **Toolbox**, drag a **ChangePassword** control onto the page.
5. In the **Properties** window for the **ChangePassword** control, set **ContinueDestinationPageUrl** property to **~/Default.aspx**.
6. This configures the control so that when users click **Continue** after changing a password, the control returns to the home page.

You can now add a link to the home page that displays the password-change page. You will make the link available only to users who are logged in.

To create a password-change link on the home page

1. Switch to or open the **Default.aspx** page.
2. Right-click the **LoginView** control and then click **Show Smart Tag**. In the **LoginView Tasks** menu, in the **Views** list, click **LoggedInTemplate**.

This switches the **LoginView** control to edit mode for the content that will appear to users who are logged in.

3. From the **Standard** group of the **Toolbox**, drag a **HyperLink** control into the editing region. In the **Properties** window for the **HyperLink** control, set the **Text** property to **Change password** and the **NavigateUrl** property to **~/MemberPages/ChangePassword.aspx**.

The **Change password** link will be displayed only to users who are logged in, which is the opposite of the **Register** link you created earlier.

You can now test the password-change process.

To test password change

1. Press CTRL+F5 to run the Web site.
2. In the Default.aspx, page, click the **Login** link and log in as one of the users you have created.

When you are finished, you are returned to the home page as a logged-in user.

3. Click the **Change password** link.
4. In the password-change page, enter the old password and a new password, and then click **Change Password**.
5. Click **Continue**.
6. On the home page, click **Logout**.
7. Click the **Login** link.
8. Enter the new password.

You are logged in with the new password.

9. Close the browser window.

Recovering a Password

Users will occasionally forget their passwords. You can add a password recovery page to your Web site so that they can once again log in to your site. Password recovery can take two forms:

- You can send users the password they selected (or that you created for them when you set up the site). This option requires that the site store the password using reversible encryption.
- You can send users a new password, which they can change using the Change Password page you created earlier. This option is useful if the Web site stores passwords using a non-reversible encryption scheme such as hashing.

Note

Returning a password in clear text using e-mail is not recommended for sites that require a high level of security. For high-security sites, it is recommended that you return passwords using encryption, such as Secure Sockets Layer (SSL).

By default, the ASP.NET membership system secures passwords by hashing them, meaning that the passwords cannot be recovered. Therefore, for this part of the walkthrough, your Web site will send users a new password.

Note

Password recovery requires that your Web site can send e-mail messages. If you are not able to configure your Web site to send e-mail (as explained under "Configuring the Application for E-Mail" earlier in this

walkthrough), you will not be able to add password recovery to your site.

To add password recovery

1. In **Solution Explorer**, right-click the name of your Web site, click **Add New Item**, and add a new **Web Form** named **RecoverPassword.aspx**.

Note

Be sure to create the page in the root of the Web site, not in the **MemberPages** folder.

2. In the **RecoverPassword.aspx** page, switch to **Design** view and type static text such as **Reset my password to a new value**. In the **Formatting** toolbar, use the **Block Format** drop-down list to format the text as **Heading 1**.
3. From the **Login** group of the **Toolbox**, drag a **PasswordRecovery** control onto the page.
4. From the **Standard** group of the **Toolbox**, drag a **HyperLink** control onto the page. In the **Properties** window for the **HyperLink** control, set the **Text** property to **Home** and the **NavigateUrl** property to **~/Default.aspx**.
5. Switch to the **Default.aspx** page.
6. Right-click the **LoginView** control and then click **Show Smart Tag**. In the **LoginView Tasks** menu, in the **Views** list, click **AnonymousTemplate**.

This switches the **LoginView** control to edit mode for the content that will appear to users who are not logged in.

7. From the **Standard** group of the **Toolbox**, drag a **HyperLink** control into the template. In the **Properties** window for the **HyperLink** control, set the **Text** property to **Forgot your password?** and the **NavigateUrl** property to **~/RecoverPassword.aspx**.

Now you can test password recovery.

To test password recovery

1. Press **CTRL+F5** to run the Web site.
2. By default, you are not logged in, so you see the anonymous template of the **LoginView** control.
3. Click the **Forgot your password?** link.

The **RecoverPassword.aspx** page appears.

4. Type your user name and click **Submit**.

The security question is displayed and you are prompted to type the security answer.

5. Type the answer and click **Submit**.

If you entered a correct answer, the Web site resets your password and sends you an e-mail message with the new password.