.NET Framework 4 - ASP.NET

**ASP.NET Login Controls Overview**

ASP.NET provides robust login (authentication) functionality for ASP.NET Web applications without requiring programming. The default Visual Studio project templates for Web applications and for Web sites include prebuilt pages that let users register a new account, log in, and change their passwords For information about how to use the built-in login page templates, see [Walkthrough: Creating an ASP.NET Web Site with Basic User Login](http://msdn.microsoft.com/en-us/library/ff184050.aspx).

You can also create your own pages that you can add ASP.NET login controls to in order to add login functionality. To use the login controls, you create a Web pages and then add the login controls to them from the **Toolbox**.

Typically, you restrict access to ASP.NET pages by putting them into a protected folder. You then configure the folder to deny access to anonymous users (users who are not logged in) and to grant access to authenticated (logged-in) users. (The default project template for Web projects includes a folder named Accounts that is already configured to allow access only to logged-in users.) Optionally, you can assign users to roles and then authorize users to access specific Web pages by role.

By default, login controls integrate with ASP.NET membership and ASP.NET forms authentication to help automate user authentication for a Web site. For information about how to use ASP.NET membership with forms authentication, see [Introduction to Membership](http://msdn.microsoft.com/en-us/library/yh26yfzy.aspx).

By default, the ASP.NET login controls work in plain text over HTTP. If you are concerned about security, use HTTPS with SSL encryption. For more information about SSL, see [Configuring SSL on a Web Server or a Web Site](http://go.microsoft.com/fwlink/?LinkId=38553) in the IIS documentation.

|  |
| --- |
| **Description: NoteNote** |
| Login controls might not function correctly if the [Method](http://msdn.microsoft.com/en-us/library/system.web.ui.htmlcontrols.htmlform.method.aspx) of the ASP.NET Web page is changed from POST (the default) to GET. |

This topic describes each ASP.NET login control and provides links to the control's reference documentation.

This topic contains the following sections:

* [Built-in Login Pages](http://msdn.microsoft.com/en-us/library/ms178329.aspx#builtin_login_pages)
* [The Login Control](http://msdn.microsoft.com/en-us/library/ms178329.aspx#the_login_control)
* [The LoginView Control](http://msdn.microsoft.com/en-us/library/ms178329.aspx#the_loginview_control)
* [The LoginStatus Control](http://msdn.microsoft.com/en-us/library/ms178329.aspx#the_loginstatus_control)
* [The LoginName Control](http://msdn.microsoft.com/en-us/library/ms178329.aspx#the_loginname_control)
* [The PasswordRecovery Control](http://msdn.microsoft.com/en-us/library/ms178329.aspx#the_passwordrecovery_control)
* [The CreateUserWizard Control](http://msdn.microsoft.com/en-us/library/ms178329.aspx#the_createuserwizard_control)
* [The ChangePassword Control](http://msdn.microsoft.com/en-us/library/ms178329.aspx#the_changepassword_control)

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifBuilt-in Login Pages

When you use the default Visual Studio template to create a Web site or Web application, pages that support login functionality are created in the Account folder. By default, pages in the Account folder are not accessible to anonymous users, except the registration page (Register.aspx) and the login page (Login.aspx). The settings that define access to pages in the Account folder are configured in the Web.config file in that folder. The settings that define access the Login page are configured in the root Web.config file.

The following illustration shows the contents of the Account folder.

**Account Folder**

The following table lists the contents of the Account folder:

|  |  |
| --- | --- |
| **Page** | **Description** |
| Login.aspx | Enables users who have an account to log in by providing a user name and password. This page includes a link to the registration page. The Login.aspx page is accessible to anonymous users.  Contains the [Login](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.login.aspx) control.  The following illustration shows the login page in the browser.  Description: Login Page Template |
| Register.aspx | Enables users to register and create a new account. This page is accessible to anonymous users and to authenticated users.  Contains the [CreateUserWizard](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.createuserwizard.aspx) control.  The following illustration shows the create new account (registration) page in the browser.  Description: Create New Account Page Template |
| ChangePassword.aspx | Enables logged-in users to change their password. This page is accessible only to authenticated users.  Contains the [ChangePassword](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.changepassword.aspx) control.  The following illustration shows the change password page in the browser.  Description: Change Password Page Template |
| ChangePasswordSuccess.aspx | Displays a message that indicates that the password was changed successfully. This page is accessible to authenticated users only.  This page does not use any ASP.NET login controls. |
| Web.config | Contains the settings that define access to pages in the Account folder |

**Login Functionality in the Master Page**

When you use the default Visual Studio template to create a Web site or Web application, the default master page (Site.master) contains the following controls that provide login functionality.

* [LoginView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginview.aspx)
* [LoginStatus](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginstatus.aspx)

The Site.master page also includes a hyperlink to the login page, which is accessible to all users.

**Storing the Login Information**

Login (membership) information is stored in a database. By default, this is a local database in the ASPNETDB.mdf file in the App\_Data folder of the Web application. The database can be created in the following ways:

* You can use the Web Site Administration Tool to configure membership and roles manually, which automatically creates the database. For more information, see [Walkthrough: Creating a Web Site with Membership and User Login](http://msdn.microsoft.com/en-us/library/879kf95c.aspx).
* When the first user registers and creates an account, and if the database does not already exist, ASP.NET creates the database automatically. This option does not provide a way to specify roles for a user.

**Modifying the Membership Attributes**

The settings for ASP.NET membership are in the **membership** section of the root Web.config file. In the **providers** section you can change attributes such as the number of invalid login attempts to allow, password length, and so on. For more information, see [Introduction to Membership](http://msdn.microsoft.com/en-us/library/yh26yfzy.aspx).

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifThe Login Control

The [Login](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.login.aspx) control displays a user interface for user authentication. The [Login](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.login.aspx) control contains text boxes for the user name and password and a check box that allows users to indicate whether they want the server to store their identity using ASP.NET membership and automatically be authenticated the next time they visit the site.

The [Login](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.login.aspx) control has properties for customized display, for customized messages, and for links to other pages where users can change their password or recover a forgotten password. The [Login](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.login.aspx) control can be used as a standalone control on a main or home page, or you can use it on a dedicated login page.

If you use the [Login](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.login.aspx) control with ASP.NET membership, you do not need to write code to perform authentication. However, if you want to create your own authentication logic, you can handle the [Login](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.login.aspx) control's [Authenticate](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.login.authenticate.aspx) event and add custom authentication code.

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifThe LoginView Control

The [LoginView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginview.aspx) control allows you to display different information to anonymous and logged-in users. The control displays one of two templates: the [AnonymousTemplate](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginview.anonymoustemplate.aspx) or the [LoggedInTemplate](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginview.loggedintemplate.aspx). In the templates, you can add markup and controls that display information appropriate for anonymous users and authenticated users, respectively.

The [LoginView](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginview.aspx) control also includes events for [ViewChanging](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginview.viewchanging.aspx) and [ViewChanged](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginview.viewchanged.aspx), which allow you to write handlers for when the user logs in and changes status.

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifThe LoginStatus Control

The [LoginStatus](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginstatus.aspx) control displays a login link for users who are not authenticated and a logout link for users who are authenticated. The login link takes the user to a login page. The logout link resets the current user's identity to be an anonymous user.

You can customize the appearance of the [LoginStatus](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginstatus.aspx) control by setting the [LoginText](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginstatus.logintext.aspx) and [LoginImageUrl](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginstatus.loginimageurl.aspx) properties.

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifThe LoginName Control

The [LoginName](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.loginname.aspx) control displays a user's login name if the user has logged in using ASP.NET membership. Alternatively, if your site uses integrated Windows authentication, the control displays the user's Windows account name.

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifThe PasswordRecovery Control

The [PasswordRecovery](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.passwordrecovery.aspx) control allows user passwords to be retrieved based on the e-mail address that was used when the account was created. The [PasswordRecovery](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.passwordrecovery.aspx) control sends an e-mail message containing a password to the user.

You can configure ASP.NET membership to store passwords using non-reversible encryption. In that case, the [PasswordRecovery](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.passwordrecovery.aspx) control generates a new password instead of sending the original password to the user.

You can also configure membership to include a security question that the user must answer to recover a password. If you do, the [PasswordRecovery](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.passwordrecovery.aspx) control asks the question and checks the answer before recovering the password.

The [PasswordRecovery](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.passwordrecovery.aspx) control requires that your application can forward e-mail message to a Simple Mail Transfer Protocol (SMTP) server. You can customize the text and format of the e-mail message sent to the user by setting the [MailDefinition](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.passwordrecovery.maildefinition.aspx) property.

|  |
| --- |
| **Description: NoteNote** |
| Password information sent in an e-mail message is sent as clear text. |

The following example shows a [PasswordRecovery](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.passwordrecovery.aspx) control declared in an ASP.NET page with [MailDefinition](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.passwordrecovery.maildefinition.aspx) property settings to customize the e-mail message.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl77_ctl00_ctl09_code');" \o "Copy Code)

<asp:PasswordRecovery ID="PasswordRecovery1" Runat="server"

SubmitButtonText="Get Password" SubmitButtonType="Link">

<MailDefinition From="administrator@Contoso.com"

Subject="Your new password"

BodyFileName="PasswordMail.txt" />

</asp:PasswordRecovery>

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifThe CreateUserWizard Control

The [CreateUserWizard](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.createuserwizard.aspx) control collects information from potential users. By default, the [CreateUserWizard](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.createuserwizard.aspx) control adds the new user to the ASP.NET membership system.

The [CreateUserWizard](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.createuserwizard.aspx) control gathers the following user information:

* User name
* Password
* Confirmation of password
* E-mail address
* Security question
* Security answer

This information is used to authenticate users and recover user passwords, if necessary.

|  |
| --- |
| **Description: NoteNote** |
| The [CreateUserWizard](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.createuserwizard.aspx) control is inherited from the [Wizard](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.wizard.aspx) control. |

The following example shows a typical ASP.NET declaration for the [CreateUserWizard](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.createuserwizard.aspx) control:

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl78_ctl00_ctl07_code');" \o "Copy Code)

<asp:CreateUserWizard ID="CreateUserWizard1" Runat="server"

ContinueDestinationPageUrl="~/Default.aspx">

<WizardSteps>

<asp:CreateUserWizardStep Runat="server"

Title="Sign Up for Your New Account">

</asp:CreateUserWizardStep>

<asp:CompleteWizardStep Runat="server"

Title="Complete">

</asp:CompleteWizardStep>

</WizardSteps>

</asp:CreateUserWizard>

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifThe ChangePassword Control

The [ChangePassword](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.changepassword.aspx) control allows users to change their password. The user must first supply the original password and then create and confirm the new password. If the original password is correct, the user password is changed to the new password. The control also includes support for sending an e-mail message about the new password.

The [ChangePassword](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.changepassword.aspx) control includes two templated views that are displayed to the user. The first is the [ChangePasswordTemplate](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.changepassword.changepasswordtemplate.aspx), which displays the user interface used to gather the data required to change the user password. The second template is the [SuccessTemplate](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.changepassword.successtemplate.aspx), which defines the user interface that is displayed after a user password has been successfully changed.

The [ChangePassword](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.changepassword.aspx) control works with authenticated and non-authenticated users. If a user has not been authenticated, the control prompts the user for a login name. If the user is authenticated, the control populates the text box with the user's login name.