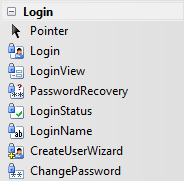
**LOGIN CONTROLS**

In Visual Studio, these controls are located in the toolbox under the Login tab.



These seven controls are as follows:

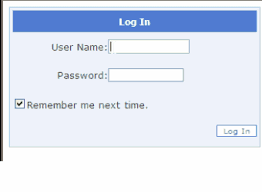
1. **Login :** Allows the user to log in by entering a user name and password.
2. **CreateUserWizard :** Allows the user to create a new user account.
3. **PasswordRecovery :** Allows the user to retrieve a forgotten password.
4. **ChangePassword :** Allows the user to change his or her password.
5. **LoginView :** This displays the contents of a template based on the user’s login status. (It allows user to customize its view for both anonymous user and logged in user. Internally,)
6. **LoginStatus :** If the user is logged in, displays a Logout link for the user to log out. If the user isn’t logged out, displays a Login link that leads to the application’s login page.
7. **LoginName:** This displays the user’s login name if the user is logged in.

**1. Login Control**

* Login control provides you with a more convenient way to let the users of your application log in to your application.
* The Login control should be placed in a page called “Login.aspx” unless you change the default <authentication> in the web.config file for the login page.
* The Login control in its simplest form will look like the following:

<asp:Login id="Login1" runat="server" />

* The Login control displays text boxes that let the user enter a user name and password.
* If the fields are filled in, the Login control uses the membership provider to look up the user name and password in the membership database.
* If the user name and password are valid, the user is logged in and the page requested is displayed else if it is not valid, an error message is displayed and the user will not be logged in to show the requested page.

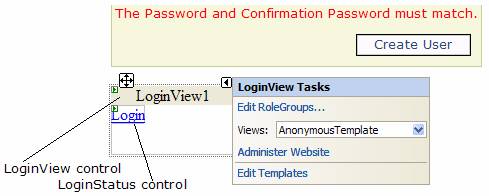


**Properties of the Login Control**

|  |  |
| --- | --- |
| **Name** | **Description** |
| TitleText | Indicates the text to be displayed in the heading of the control. |
| InstructionText | Indicates the text that appears below the heading of the control. |
| UserNameLabelText | Indicates the label text of the username text box. |
| PasswordLabelText | Indicates the label text of the password text box. |
| FailureText | Indicates the text that is displayed after failure of login attempt. |
| UserName | Indicates the initial value in the username text box. |
| LoginButtonText | Indicates the text of the Login button. |
| LoginButtonType | Button/Link/Image. Indicates the type of login button. |
| DestinationPageUrl | Indicates the URL to be sent after login attempt successful. |
| DisplayRememberMe | true/false. Indicates whether to show Remember Me checkbox or not. |
| VisibleWhenLoggedIn | true/false. If false, the control is not displayed on the page when the user is logged in. |
| CreateUserUrl | Indicates the url of the create user page. |
| CreateUserText | Indicates the text of the create user link. |
| PasswordRecoveryUrl | Indicates the url of the password recovery page. |
| PasswordRecoveryText | Indicates the text of the password recovery link. |

**LoginView Control**

* LoginView control is very simple yet very powerful and customizable.
* It allows user to customize its view for both anonymous user and logged in user.
* This control is a template control.
* This template control can display the contents of its templates according to the login status of the user.
* This gives you the ability to customize your content of your web application for the needs of different users.
* Internally, When it is rendered on the page, it is implemented through <table></table> HTML tag.
* **For example:** The User authentication application should use a LoginView control to display a link to the administrator’s page because this page should be only visible to the users with the login status of an “Admin”.
* The LoginView control does not have any special attributes to customize its appearance or behaviour but you can customize the LoginView control by using the three types of templates of which each can be coded in as a child element.
* The three templates are:
* **Anonymous template:** Displayed if the user isn’t logged in.
* **LoggedIn template:** Displayed if the user is logged in.
* **RoleGroup template:** Displayed if the user is logged in and is a member of a particular role group.



**LoginName**

* LoginName control is used to display the user’s username which the user logged in to the web application from.
* If a user is not currently logged into the web application, the LoginName control does not display anything.
* The LoginName control in its simplest form will look like the following:

<asp:LoginName ID="LoginName1" runat="server" />

**LoginStatus**

* LoginStatus control will display a link for the user to log into or log out of the web application depending on whether the user is logged in or out.
* For example: If the user is logged in, a link is displayed to Logout. If the user is not logged in, a link is displayed to Login
* The LoginStatus control in its simplest form will look like the following:

<asp:LoginStatus ID="LoginStatus1" runat="server" />

**Properties of LoginStatus Control**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| id | The ID for the LoginStatus control |
| runat | runat=“Server” is needed for all ASP.NET server controls. |
| LoginImageUrl | The URL for the Login link image. |
| LoginText | The text for the Login link. |
| LogoutAction | The action to do after the user logs out. This can be specified to Redirect to redirect the user to a page in the LogoutPageUrl attribute, or RedirectToLoginPage to redirect the user to the login page, or Refresh to refresh the current page. |
| LogoutImageUrl | The URL for the Logout link. |
| LogoutPageUrl | The URl for the redirect page after a user logs out if the LogoutAction attribute specifies Redirect. |
| LogoutText | The text for the Logout link. |

**PasswordRecovery Control**

* PasswordRecovery control allows you to easily create a way to retrieve a forgotten password.
* In this control, the user has to enter their security question and the answer to it.
* If the values are valid, the password is reset to a random value and that password is emailed to the email address the user provided when registering to the application.
* The new password can be sent to the user’s email account.
* The PasswordRecovery control in its simplest form will look like the following:

<asp:PasswordRecovery ID="PasswordRecovery1" runat="server">  
</asp:PasswordRecovery>

**Properties of PasswordRecovery Control**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| Id | The ID for the PasswordRecovery control |
| Runat | runat="Server" is needed for all ASP.NET server controls. |
| AnswerLabelText | The text for the Answer label field |
| GeneralFailureText | The text for the Failure text field if the password could not be retrieved. |
| QuestionFailureText | The text for Question failure that is if the user gives a wrong answer for the secret question. |
| QuestionInstructionText | The text for the instructions in the Question’s answer part. |
| QuestionLabelText | The text for the Secret Question label. |
| QuestionTitleText | The text for the title area for the secret question. |
| SubmitButtonImageUrl | The URL for the image of the Submit button. |
| SubmitButtonText | The text for the Submit button. |
| SubmitButtonType | The type of the Submit button can be Button, Link, or Image. |
| SuccessPageUrl | The URL for the Success page if the password has been recovered. |
| SuccessText | The text for the message if the password retrieval is successful, and this is not displayed if the SuccessPageUrl is provided. |
| TextLayout | Here the position of the labels for the username and password textboxes are set. TextOnLeft for the labels to appear on to the left side of the text boxes or TextOnTop for the label to appear above the text boxes. |
| UserNameFailureText | The text for incorrect username. |
| UserNameInstructionText | The text for the instructions for the Username request area. |
| UserNameLabelText | The text for the Username label field. |
| UserNameTitleText | The text for the title in the Username request area. |

**CreateUserWizard**

CreateUserWizard control is a ready to use control that is used to create a new user to the website.

In other words we can say that it a replacement of Registration form of a website.

The control allows user to enter user name, password, confirmation password, email address, security question and security answer.



CreateUserWizard control uses Membership service to create user's details.

|  |  |
| --- | --- |
| Events of CreateUserWizard Control | |
| ContinueButtonClick | Fires when user clicks Continue button in the last wizard step. |
| CreatingUser | Fires before creating a new user. |
| CreatedUser | Fires after new user created. |
| CreateUserError | Fires when creation of user is not successful. |

**ChangePassword**

ChangePassword is an asp.net web server control.

ChangePassword control is a ready to use control to help users to change their password.

ChangePassword control uses Membership service to change user's password.

ChangePassword control enables web site users to change the passwords they use to log in the web site.

ChangePassword server control uses the membership provider defined by the MembershipProvider property to change the password stored in the membership provider data store.

We can configure ChangePassword control to use email services to send the new password to the user.

To send users new password to their email address, we must configure an email server in the Web.config file for an asp.net application.

ChangePassword control has two states or views,

* 1. Change password view and (2) Success view.

(1)The change password view, request the current password and requires inputting new password twice.

(2)Success view display confirmation that the password has been changed.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\***