Forms are used to collect data inputted by a user. They can be used as an interface for a web application, for example, or to send data across the web. form defines the form and within this tag, if you are using a form for a user to submit information (which we are assuming at this level), an action attribute is needed to tell the form where its contents will be sent to.

The input tag is the daddy of the form world. It can take a multitude of guises, the most common of which are outlined below (see the input reference page for the whole crazy family):

<input type="text"> or simply <input> is a standard textbox. This can also have a value attribute, which sets the initial text in the textbox.

<input type="password"> is similar to the textbox, but the characters typed in by the user will be hidden.

<input type="checkbox"> is a checkbox, which can be toggled on and off by the user. This can also have a checked attribute (<input type="checkbox" checked> - the attribute doesn’t require a value), and makes the initial state of the check box to be switched on, as it were.

<input type="radio"> is similar to a checkbox, but the user can only select one radio button in a group. This can also have a checked attribute.

<input type="submit"> is a button that when selected will submit the form. You can control the text that appears on the submit button with the value attribute, for example <input type="submit" value="Ooo. Look. Text on a button. Wow">.

We can, of course, use extension objects to pass data through to third-party controls, for example, Flash or Silverlight. But we can also fully control the way we display the data using HTML and JavaScript, especially if we employ one of the many visualization libraries available. HTML Tables used to be employed to create entire website layouts. Now, they should only be used in selective cases when appropriate. Each row of your table needs to be marked up using <tr> </tr> tag pairs. For now, it is best if you space out your code similar to above, so that you can see each table row clearly. There aren’t really any attributes of the <tr> though we’ll look at some neat tricks you can do in CSS3 for Tables at a later point. If your table has ‘header’ information – like the top row label in an Excel Spreadsheet – then it can be helpful to mark these as <th> tags. Often browsers will automatically centre-align the text in the cell and make it bold. You can adjust this behaviour in CSS but for the purposes of this tutorial, we’ll leave it in place. Each header cell must have a corresponding </th> tag so that you are clearly telling your page what content is to be contained inside that particular cell. All other cells which are not headers are called table-data cells and are contained in the <td> </td> tag pair. These have no default behaviour, unlike your table headers.