WEB PAGES DESIGN

FORMS
(Part I)

Lecturer: M.Sc. Ahmed Sami
Lecture 4.
Class 2.
Department: Informatics Systems Management (ISM)

Introduction to Forms

HTML Forms:

- HTML Forms are required to collect some data from the site visitor. The most common way for a user to communicate information from a Web browser to the server is through a form. forms can be described in HTML and displayed by the browser. HTML provides tags to generate the commonly used objects on a screen form. These objects are called controls.
- There are controls for single-line and multiple-line text collection, checkboxes, radio buttons, and menus, among others. All control tags are inline tags. Most controls are used to gather information from the user in the form of either text or button selections.
- Each control can have a value, usually given through user input. Together, the values of all of the controls (that have values) in a form are called the form data. Every form requires a Submit button. When the user clicks the Submit button, the form data is encoded and sent to the Web server for processing.

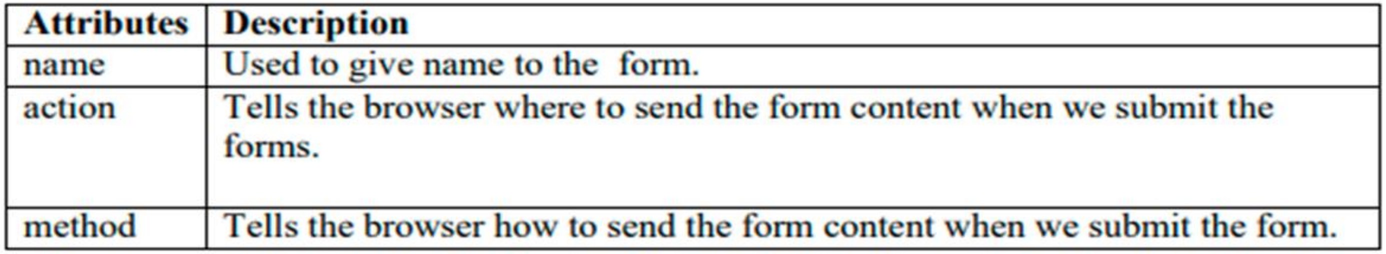
Introduction to Forms

For example, during user registration you would like to collect information such as name, email address, credit card, etc. A form will take input from the site visitor and then will post it to a back-end application such as CGI, ASP Script or PHP script etc. The back-end application will perform required processing on the passed data based on defined business logic inside the application. There are various form elements available like text fields, text area fields, dropdown menus, radio buttons, checkboxes, etc.

Syntax:

<form name=”form1” action="form_action.asp” method="get” > … </form>

<form action="Script URL" method="GET|POST"> form elements like input, text area etc. </form>



Attributes	Description
name	Used to give name to the form.
action	Tells the browser where to send the form content when we submit the forms.
method	Tells the browser how to send the form content when we submit the form.

GET method vs POST method

Form data can be sent to the server in two ways, each corresponding to HTTP methods.

- The GET method, which sends data as part of the URL.
- The POST method, which hides data in something known as the HTTP header.

What the difference is between get and post methods?

- The get method is not appropriate for forms with personal or financial information because it sends data as part of the URL.
- When the method attribute is set to post method, the browser sends a separate server request containing some special headers followed by the data. So that only the server sees the content of this request. Thus it is the best method form sending secure information such as credit card or other personal information.

HTML Forms Controls

There are different types of form controls that you can use to collect data using HTML form:

- Text Input Controls
- Checkboxes Controls
- Radio Box Controls
- Select Box Controls
- File Select boxes
- Hidden Controls
- Click Buttons
- Submit and Reset Button

Text Input Controls

Text Input Controls:- There are three types of text input used on forms:

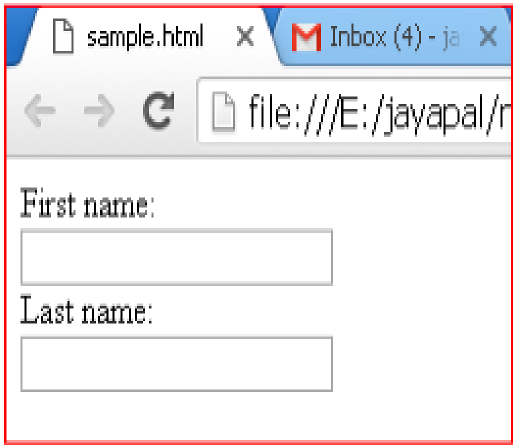
1- Single-line text input controls - This control is used for items that require only one line of user input, such as search boxes or names. They are created using HTML<input> tag.

Syntax:

<input type="text"> defines a one-line input field for text input:

Example:

```
<form>
Firstname:<br>
<input type="text"name="firstname"> <br>
Lastname:<br>
<input type="text" name="lastname">
</form>
```



First name:

Last name:

Text Input Controls

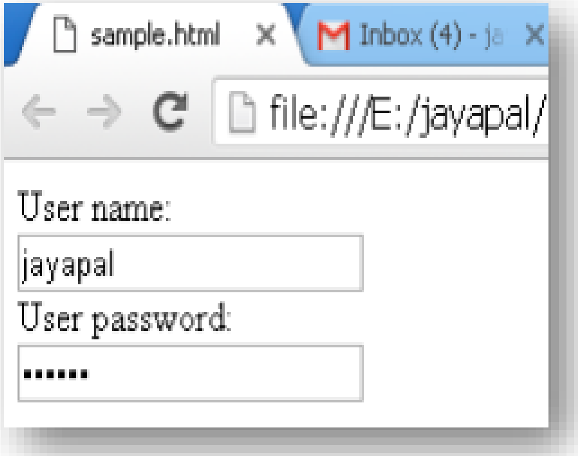
2- Password input controls - This is also a single-line text input but it masks the character as soon as a user enters it. They are also created using HTML <input> tag.

Syntax:

<input type="password"> defines a password field:

Example:

```
<form>
User name:<br>
<input type="text" name="username"> <br>
User password:<br>
<input type="password" name="psw">
</form>
```



User name:

jayapal

User password:

••••••

Text Input Controls

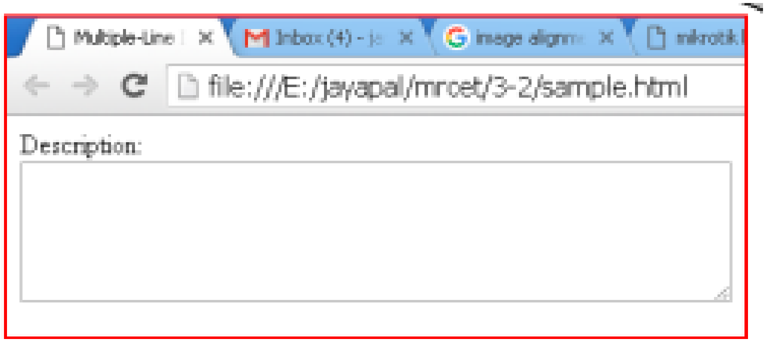
3- Multi-line text input controls - This is used when the user is required to give details that may be longer than a single sentence. Multi-line input controls are created using HTML <textarea> tag.

Syntax:

```
<textarea rows="5" cols="50" name="description"> Enter description
</textarea>
```

Example:

```
<form> Description: <br />
<textarea rows="5" cols="50" name="description"> Enter description
here...
 </textarea>
</form>
```



file:///E:/jayapal/mrcet/3-2/sample.html

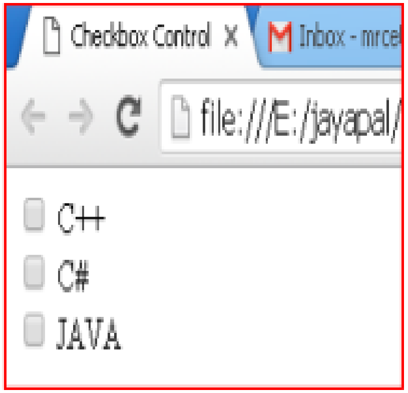
Description:

Checkboxes Controls

Checkboxes Controls:- Checkboxes are used when more than one option is required to be selected. They are also created using HTML <input> tag but type attribute is set to checkbox.

Here is an example HTML code for a form with two checkboxes:

```
<!DOCTYPE html>
<html> <head><title>Checkbox Control</title></head>
<body>
<form>
<input type="checkbox" name="C++" value="on"> C++
<br>
<input type="checkbox" name="C#" value="on"> C#
<br>
<input type="checkbox" name="JAVA" value="on"> JAVA
</form>   </body> </html>
```



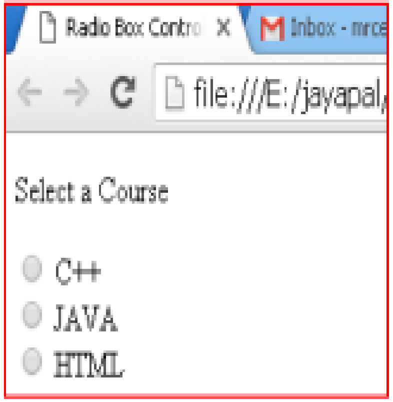
□ C++
□ C#
□ JAVA

Radio Button Control

Radio Button Control:- Radio buttons are used when out of many options, just one option is required to be selected. They are also created using HTML <input> tag but type attribute is set to radio.

Here is an example HTML code for a form with two checkboxes:

```
<html>
<head><title>Radio Box Control</title></head>
<body> <p>Select a Course</p>
<form>
<input type="radio" name="subject" value="C++"> C++
<br>
<input type="radio" name="subject" value="JAVA"> JAVA
<br>
<input type="radio" name="subject" value="HTML"> HTML
</form> </body> </html>
```



Select a Course

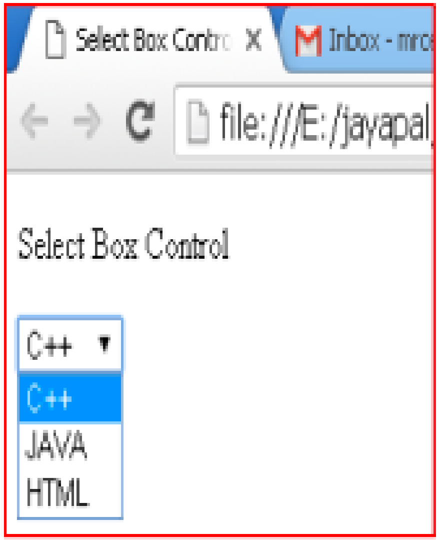
○ C++
○ JAVA
○ HTML

Select Box Controls

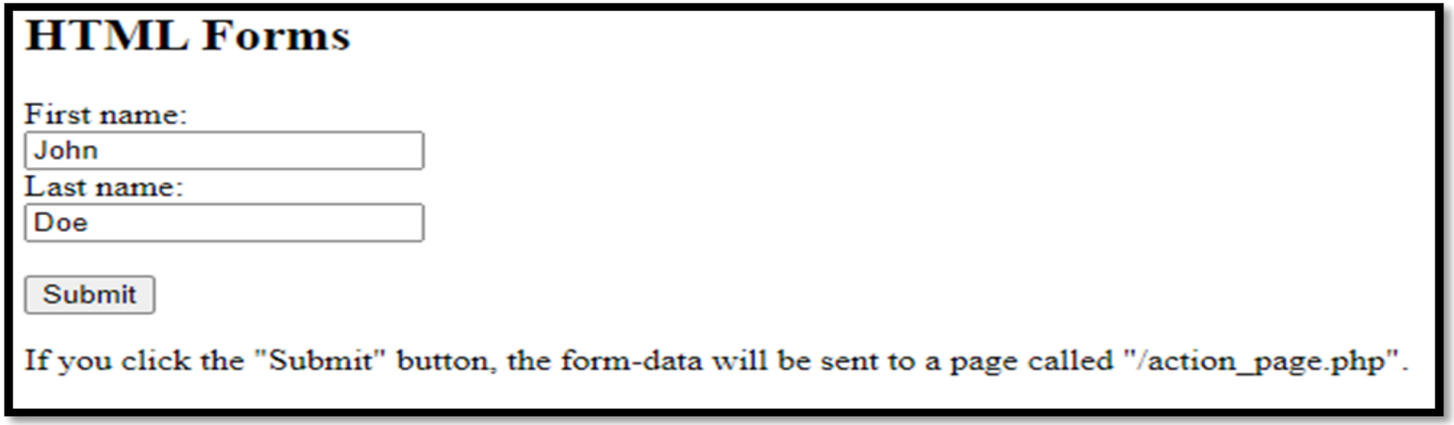
Select Box Controls:- A select box, also called drop down box which provides option to list down various options in the form of drop down list, from where a user can select one or more options.

Here is an example HTML code for a form with two checkboxes:

```
<html>
<head> <title>Select Box Control</title> </head>
<body>
<form>
<select name="dropdown">
<option value="C++" selected> C++ </option>
<option value="JAVA"> JAVA </option>
<option value="HTML"> HTML </option>
</select>
</form>
</body>
</html>
```



HTML Forms



HTML Forms

First name:

John

Last name:

Doe

Submit

If you click the "Submit" button, the form-data will be sent to a page called "/action_page.php".

```
<!DOCTYPE html>
<html>
<body>

<h2>HTML Forms</h2>

<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe"><br><br>
  <input type="submit" value="Submit">
</form>

<p>If you click the "Submit" button, the form-data will be sent to a page
called "/action_page.php".</p>

</body>
</html>
```

The end