

**Bachelor of Science (Second Year)**  
**CSS104-Web Application Development using Flask**  
**Title of the Unit : HTML**  
**Module Name : Designing form, fieldset and legend widgets**  
**Module Number : 06**

## Designing form

### What is HTML Form :

HTML Form is a document which stores information of a user on a web server using interactive controls. An HTML form contains different kind of information such as username, password, contact number, email id etc. The elements used in an HTML form are check box, input box, radio buttons, submit buttons etc. Using these elements the information of an user is submitted on a web server. The **form** tag is used to create an HTML form.

### Example of an HTML Form :

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

<form>
  Username:<br>
  <input type="text" name="username">
  <br>
  Email id:<br>
  <input type="text" name="email_id">
  <br><br>
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

**Username:**

**Email id:**

Submit

**The Most Common Elements used in an HTML form are <label>, <Input>, <Textarea>, <Select>, etc.**

### The <label> element in HTML Form

The <label> element is the formal way to define a label for an HTML form widget. This is the most important element if you want to build accessible forms — when implemented properly, screenreaders will speak a form element's label along with any related instructions, as well as it being useful for sighted users. Take this example, which we saw in the previous article:

```
<label for="name">Name: </label> <input type="text" id="name" name="user_name">
```

With the <label> associated correctly with the <input> via its for attribute (which contains <input> element's id attribute), a screenreader will read out something like "Name, edit text". There is another way to associate a form control with a label — nest the form control within the <label>, implicitly associating it.

```
<label for="name">  
  Name: <input type="text" id="name" name="user_name">  
</label>
```

### Multiple labels

Strictly speaking, you can put multiple labels on a single widget, but this is not a good idea as some assistive technologies can have trouble handling them. In the case of multiple labels, you should nest a widget and its labels inside a single <label> element.

```
<div>  
  <label for="username">Name: <abbr title="required" aria-label="required">*</abbr></label>  
  <input id="username" type="text" name="username">  
</div>
```

## **Input Element in HTML Form**

Input Elements are the most common elements which are used in HTML Forms. Various user input fields can be created such as textfield, check box, password field, radio button, submit button etc. The most common input elements are listed below:

### **Text Field in HTML Forms :**

The text field is a one line input field allowing the user to input text. Text Field input controls are created using the “input” element with a type attribute having value as “text”.

```
<!DOCTYPE html>  
<html>  
<h3>Example Of Text Field</h3>  
<body>  
  <form>  
    <label for="EMAIL ID">Email Id:</label><br>  
    <input type="text" name="Email id" id="Email id">  
  </form>  
</body>  
</html>
```

**Output :**

### **Example Of Text Field**

Email Id:

### **Password Field in HTML Forms :**

Password fields are a type of text field in which the text entered is masked using asterisk or dots for prevention of user identity from another person who is looking onto the screen. Password Field input controls are created using the “input” element with a type attribute having value as “password”.

```

<!DOCTYPE html>
<html>
<h3>Example of Password Field</h3>
<body>
  <form>
    <label for="user-password">Password:
    </label><br>
    <input type="password" name="user-pwd"
          id="user-password">
  </form>
</body>
</html>

```

**Output :**

### **Example of Password Field**

Password:

### **Radio Buttons in HTML Form :**

Radio Buttons are used to let the user select exactly one option from a list of predefined options. Radio Button input controls are created using the “input” element with a type attribute having value as “radio”.

```

<!DOCTYPE html>
<html>
<h3>Example of Radio Buttons</h3>
<body>
  <form>
    SELECT GENDER
    <br>
    <input type="radio" name="gender" id="male">
    <label for="male">Male</label><br>
    <input type="radio" name="gender" id="female">
    <label for="female">Female</label>
  </form>
</body>
</html>

```

**Output :**

### **Example of Radio Buttons**

SELECT GENDER

- Male
- Female

### **Checkboxes in HTML Form :**

Checkboxes are used to let the user select one or more options from a pre-defined set of options. Checkbox input controls are created using the “input” element with a type attribute having value as “checkbox”.

```

<!DOCTYPE html>
<html>
<h3>Example of HTML Checkboxes</h3>
<body>
  <form>
    <b>SELECT SUBJECTS</b>
    <br>
    <input type="checkbox" name="subject" id="maths">
    <label for="maths">Maths</label>
    <input type="checkbox" name="subject" id="science">
    <label for="sceince">Science</label>
    <input type="checkbox" name="subject" id="english">
    <label for="english">English</label>
  </form>
</body>
</html>

```

**Output :**

### **Example of HTML Checkboxes**

#### **SELECT SUBJECTS**

Maths  Science  English

### **File select boxes in HTML Forms :**

File select boxes are used to allow the user to select a local file and send it as an attachment to the web server. It is similar to a text box with a button which allows the user to browse for a file. Instead of browsing for the file, the path and the name of the file can also be written. File select boxes are created using the “input” element with a type attribute having value as “file”.

```

<!DOCTYPE html>
<html>
<h3>Example of a File Select Box</h3>
<body>
  <form>
    <label for="fileselect">Upload:</label>
    <input type="file" name="upload" id="fileselect">
  </form>
</body>
</html>

```

**Output :**

### **Example of a File Select Box**

Upload:  no file selected

### **Text area in an HTML Form :**

Text Area is a multiple line text input control which allows user to provide a description or text in multiple lines. A Text Area input control is created using the “textarea” element.

```

<!DOCTYPE html>
<html>
<h3>Example of a Text Area Box</h3>
<body>
  <form>
    <label for="Description">Description:</label>
    <textarea rows="5" cols="50" name="Description"
              id="Description"></textarea>
  </form>
</body>
</html>

```

**Output :**

### Example of a Text Area Box

Description:

Select Boxes in HTML Forms :

Select boxes are used to allow users to select one or more than one option from a pull-down list of options. Select boxes are created using two elements which are “select” and “option”. List items are defined within the select element.

```

<!DOCTYPE html>
<html>
<h3>Example of a Select Box</h3>
<body>
  <form>
    <label for="country">Country:</label>
    <select name="country" id="country">
      <option value="India">India</option>
      <option value="Sri Lanka">Sri Lanka</option>
      <option value="Australia">Australia</option>
    </select>
  </form>
</body>
</html>

```

**Output :**

### Example of a Select Box

Country:

### Example of a Select Box

Country 

- India
- Sri Lanka
- Australia

**Reset And Submit Buttons :**

The Submit Button allows the user to send the form data to the web server. The Reset Button is used to reset the form data and use the default values.

```
<!DOCTYPE html>
<html>
<h3>Example of a Submit And Reset Button</h3>
<body>
  <form action="test.php" method="post" id="users">
    <label for="username">Username:</label>
    <input type="text" name="username" id="Username">
    <input type="submit" value="Submit">
    <input type="reset" value="Reset">
  </form>
</body>
</html>
```

**Output :**

### **Example of a Submit And Reset Button**

Username:

### **The <fieldset> and <legend> widgets**

The <fieldset> element is a convenient way to create groups of widgets that share the same purpose, for styling and semantic purposes. You can label a <fieldset> by including a <legend> element just below the opening <fieldset> tag. The text content of the <legend> formally describes the purpose of the <fieldset> it is included inside.

Many assistive technologies will use the <legend> element as if it is a part of the label of each control inside the corresponding <fieldset> element.