

Programme: FYBSc. & FYBA

Subject: COMPUTER SCIENCE

Semester : I

Paper Code: CSG110

Paper Title : GE – CLIENT SIDE WEB DEVELOPMENT

Unit : UNIT II

Module Name: Structuring Content-Semantic Tags Part-II.

Module No : 14

HTML - <fieldset> Tag

Description:

The HTML <fieldset> tag is used for grouping related form elements. By using the fieldset tag and the legend tag, you can make your forms much easier to understand for your users.

Syntax: <fieldset> Contents... </fieldset>

Attribute:

1. **disabled:** It specify that the group of related form elements should be disabled.
2. **form:** It specify that one or more forms the fieldset belongs to.
3. **name:** It specify the name for the fieldset.

Example

```
<html>
<body>
<form>
<h2>Fieldset tag</h2>
<fieldset>
Enroll No: <input type = "text"><br />
Name: <input type = "text"><br />
Address: <input type = "text">
</fieldset>
</form>
</body>
</html>
```

OUTPUT:

Fieldset tag

The image shows a screenshot of a web browser displaying a simple HTML form. The title of the browser window is 'Fieldset tag'. Inside the form, there are three text input fields stacked vertically. The first field is labeled 'Enroll No:' followed by an input field. The second field is labeled 'Name:' followed by an input field. The third field is labeled 'Address:' followed by an input field. All labels are in bold black font, and the input fields have a light gray background.

Enroll No:	<input type="text"/>
Name:	<input type="text"/>
Address:	<input type="text"/>

HTML - <Legend> Tag

Description:

The HTML <legend> tag used to define a caption for <fieldset> tag. The legend tag is used to define the title for the child contents. The legend elements are the parent element.

Syntax:

```
<legend> fieldset title... </legend>
```

Attribute:

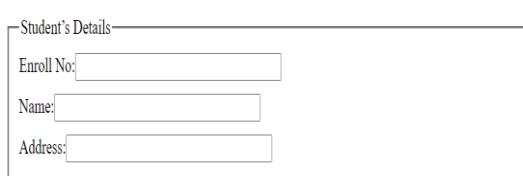
align :Specifies the content alignment.

Example

```
<html>
<body>
<form>
<h2>Legend tag</h2>
<fieldset>
<legend>Student's Details</legend>
Enroll No: <input type="text"> <br>
Name: <input type="text"> <br>
Address: <input type="text">
</fieldset>
</form>
</body>
</html>
```

OUTPUT:

Legend tag



A screenshot of a web browser displaying a form. The form has a legend at the top labeled "Student's Details". Below the legend, there are three input fields: "Enroll No:", "Name:", and "Address:", each preceded by a label and a colon.

HTML WIDGET

Inside the **HTML Widget** you can embed HTML, CSS, Shortcodes, and also include JS scripts. Using widget operations and interactive triggers, you can create a rich, interactive experience that is suitable for many purposes, such as embedding application help, adding social media interactions, or embedding your own limited web browser in the application.

Designing 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 :

```
<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>
```

OUTPUT:

The image shows a rectangular form box with a thin black border. Inside, there are two lines of text: "Username:" followed by an empty text input field, and "Email id:" followed by another empty text input field. Below these fields is a single button labeled "Submit".

Input Element in HTML Forms :

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
- Password Field
- Radio Buttons
- Checkboxes
- File select boxes
- Text area
- Select Boxes
- Reset And Submit Buttons
-

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”.

```
<html>
<h3>Example Of Text Field</h3>
<body>
<form>
<label >Name:</label>
<input type="text" name="Sname" id="Sname">
</form>
</body>
</html>
```

OUTPUT:

Example Of Text Field

Name:

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.

```
<html>
<h3>Example of Password Field</h3>
<body>
<form>
<label >Password: </label>
<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”.

```
<html>
<body>
<form>
SELECT GENDER <br>
<input type="radio" name="gender" id="male">
<label for="male">Male</label>
<input type="radio" name="gender" id="female">
<label for="female">Female</label>
</form>
</body>
</html>
```

OUTPUT:

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”. >

```
<html>
<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="science">Science</label>
</form>
</body>
</html>
```

OUTPUT:

SELECT SUBJECTS			
<input type="checkbox"/>	Maths	<input type="checkbox"/>	Science

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.

```
<html>
<h3>Example of a File Select Box</3>
<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:	<input type="button" value="Choose File"/>	No file chosen

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.

```
<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 :

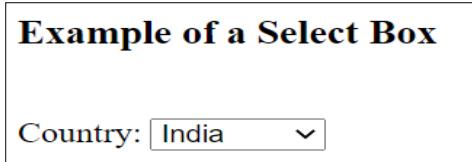
Select boxes are used to allow users to select one or more than one option from a pull-down list of options. List items are defined within the select element.

```
<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: India ▾



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.

```
<html>
<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:

Username:	<input type="text"/>	<input type="button" value="Submit"/>	<input type="button" value="Reset"/>
-----------	----------------------	---------------------------------------	--------------------------------------

Attributes Used in HTML Forms:

- The Action Attribute
- The Target Attribute in HTML Forms
- Name Attribute in Html Forms
- Method Attribute

Attributes Used in HTML Forms**The Action Attribute :**

- The action to be performed after the submission of the form is decided by the action attribute.
- Generally, the form data is sent to a webpage on the web server after the user clicks on the submit button.

Example

:

```
<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>
```

If you click the submit button, the form data would be sent to a page called test.php .

OUTPUT:

Example of a Submit And Reset Button			
Username:	<input type="text"/>	<input type="button" value="Submit"/>	<input type="button" value="Reset"/>

The Target Attribute in HTML Forms :

The Target attribute is used to specify whether the submitted result will open in the current window, a new tab or on a new frame. The default value used is “self” which results in the form submission in the same window. For making the form result open in a new browser

tab, the value should be set to “blank”. After clicking on the submit button, the result will open in a new browser tab.

The Target Attribute in HTML Forms :

```
<html>
<body>
<form action="/test.php" target="_blank">
    Username:
    <input type="text" name="username">
    Password:
    <input type="password" name="password">
    <input type="submit" value="Submit">
</form>
</body>
</html>
```

After clicking on the submit button, the result will open in a new browser tab.

Name Attribute in Html Forms :

The name attribute is required for each input field. If the name attribute is not specified in an input field then the data of that field would not be sent at all.

Name Attribute in Html Forms :

```
<html>
<body>
<form action="/test.php" target="_blank">
    Username:
    <input type="text">
    Password:
    <input type="password" name="password">
    <input type="submit" value="Submit">
</form>
</body>
</html>
```

In the above code, after clicking the submit button, the form data will be sent to a page called /test.php. The data sent would not include the username input field data since the name attribute is omitted.

Method Attribute :

It is used to specify the HTTP method used to send data while submitting the form.

There are two kinds of HTTP Methods:

- GET
- POST.

The GET Method

In the GET method, after the submission of the form, the form values will be visible in the address bar of the new browser tab. This method appends form-data into the URL in name/value pairs. The length of a URL is limited (about 3000 characters) Never use GET to send sensitive data (will be visible in the URL) Useful for form submissions where a user wants to bookmark the result GET is better for non-secure data, like query strings in Google.

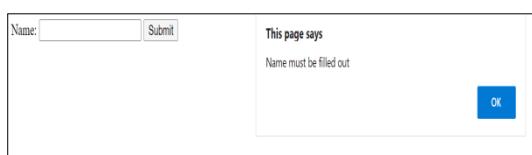
Form Data Validation

The data entered into a form needs to be in the right format and certain fields need to be filled in order to effectively use the submitted form. Username, password, contact information are some details that are mandatory in forms and thus need to be provided by the user.

JavaScript Form Validation

HTML form validation can be done by JavaScript. If a form field (fname) is empty, this function alerts a message, and returns false, to prevent the form from being submitted.

```
<html>
<head>
<script>
function validateForm() {
    var x = document.forms["myForm"]["fname"].value;
    if (x == "") {
        alert("Name must be filled out");
        return false;
    }
}
</script>
</head>
<body>
<form name="myForm" action="/action_page.php" onsubmit="return validateForm()"
method="post">
    Name: <input type="text" name="fname">
    <input type="submit" value="Submit">
</form>
</body>
</html>
➤ OUTPUT:
```



JavaScript Can Validate Numeric Input

```
<script>
function myFunction() {
    var x, text;
    // Get the value of the input field with id="numb"
    x = document.getElementById("numb").value;
    // If x is Not a Number or less than one or greater than 10
    if (isNaN(x) || x < 1 || x > 10) {
        text = "Input not valid";
    } else {
        text = "Input OK";
    }
    document.getElementById("demo").innerHTML = text;
}
</script>
```

OUTPUT:

JavaScript Can Validate Input

Please input a number between 1 and 10:

Input not valid

Automatic HTML Form Validation

HTML form validation can be performed automatically by the browser:If a form field (fname) is empty, the required attribute prevents this form from being submitted:

HTML Form Example

```
<form action="/action_page.php" method="post">
    <input type="text" name="fname" required>
    <input type="submit" value="Submit">
</form>
```