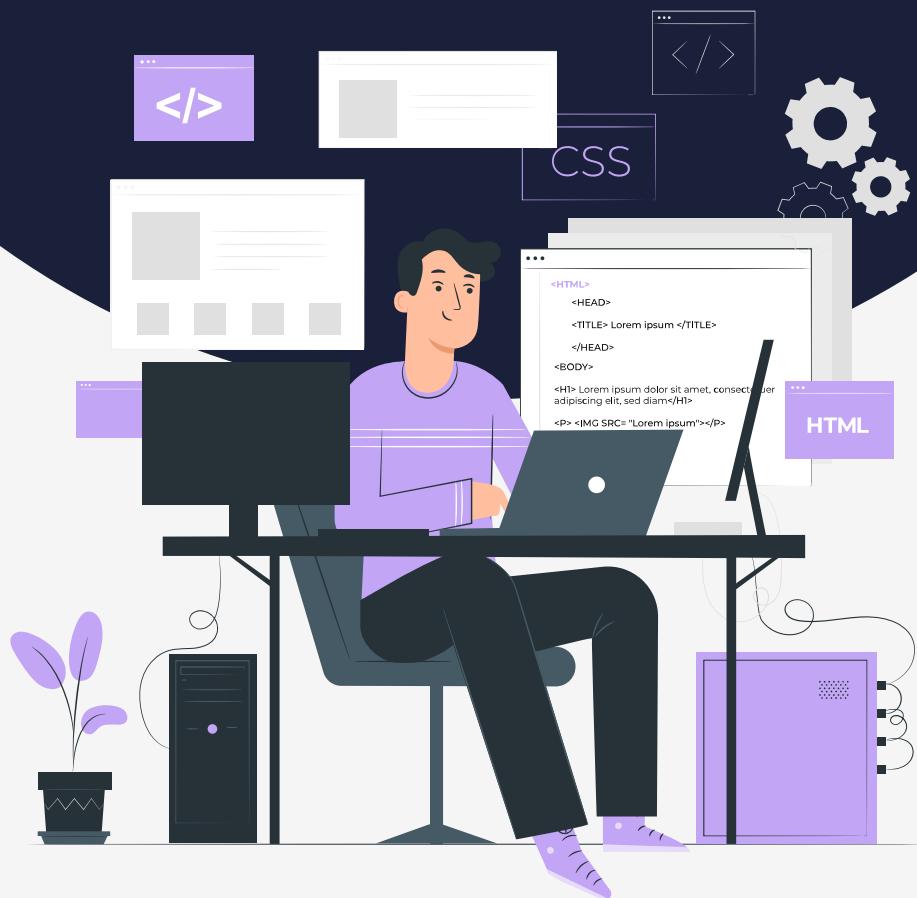


Lesson:

Input tags and their attributes



Topics Covered

- Input Tag
- Input Attributes
- Common Input Attributes
- Legend

Input Tag

The **<input>** HTML element is used to create interactive controls for web-based forms in order to accept data from the user;

A wide variety of types of input data and control widgets are available.

A simple example,

```
Unset
<label for="email">Enter Email: </label>
<input type="email" name="email" id="email" required>
```

Above code defines an **email input** for entering an email address. It also checks for a valid email address.

Here **<label>** tags help users to know the purpose of the form control element (**input**), and **for** attribute specifies the target control element, which allows users to click on the label to focus on the associated form control.

Input Attributes

- **type:** This attribute specifies the type of input control to be created. It is a required attribute and can take various values such as **text, password, email, number, checkbox, radio, submit**, and many more.
- **name:** This attribute specifies the name of the input control, which is used as a reference when the form is submitted. The value of this attribute is sent to the server as part of the form data.
- **required:** This attribute indicates that the input field is required to be filled out by the user before the form can be submitted. If the field is empty, it will show a warning message.
- **value:** This attribute specifies the default value of the input control. It is used to pre-fill the input field with a value that the user can modify.
- **placeholder:** This attribute specifies a short hint or example text that is displayed in the input field to provide guidance to the user on what should be entered.

Example of an input attribute

Unset

```
<!DOCTYPE html>
<html>
<head>
  <title>Simple HTML Input</title>
</head>
<body>
  <form action="/subscribe" method="get">
    <label for="email">Enter Email: </label>
    <input
      type="number"
      name="salary"
      placeholder="Enter your Salary"
      value="100000"
      required
    >
  </form>
</body>
</html>
```

Browser Output:

Initially Input will look like below	When the input is empty, we can see placeholder value.	When a user tries to submit, without filing a value.
Enter Salary: <input type="text" value="100000"/> <div style="display: flex; justify-content: space-around; width: 100%;"> label default value </div>	Enter Salary: <input placeholder="Enter your Salary" type="text"/> <div style="display: flex; justify-content: center; width: 100%;"> placeholder </div>	Enter Salary: <input placeholder="Enter your Salary" type="text"/> <div style="display: flex; align-items: center;"> ! Please fill out this field. </div> <div style="text-align: right;">Required Field</div>

Common Input Types

1. Text

It accepts a single line string. Let's define a text input,

Unset

```
<label for="mailid">Mail Id</label>
<input type="text" placeholder="Text Input" id="mailid"
name="mailid" required/>
```

Text Input

2. Email

An input element with type="email" is used to create an email input field in HTML.

Unset

```
<label for="mailid">Mail Id</label>
<input type="email" id="mailid" name="mailid"/>
```

Mail Id

! Please include an '@' in the email address.
 'anurag' is missing an '@'.

When the form is submitted, the value of the email input field will be included in the form data with the key specified by the **name** attribute. If the **email address entered in the input field is not a valid** email address, the **browser will throw a warning** message to the user before allowing the form to be submitted.

3. Password

It allows users to enter passwords secretly. Let's define password input,

Unset

```
<label for="pass">Password</label>
<input type="password" id="pass" name="pass"/>
```

Password 

4. Number

Only numbers allowed. Let's define number input,

Unset

```
<label for="salary">Salary</label>
<input type="number" id="salary" name="salary"/>
```

Salary

5. Checkbox

Checkboxes are a type of input field that **allows the user to select one or more options** from a list of predefined options. The **type** attribute of the `<input>` tag should be set to "**checkbox**" to create a checkbox. The **name** attribute is used to group related checkboxes together, and the **value** attribute specifies the value of the checkbox when it is selected. When the user selects a checkbox, the value associated with that checkbox is submitted with the form data. If you want to pre-select a checkbox by default, you can add the checked attribute to the `<input>` tag.

Unset

```

<label for="option1">option1</label>
<input type="checkbox" id="option1" name="check"
value="option1"/>

<label for="option2">option2</label>
<input type="checkbox" id="option2" name="check"
value="option2"/>

<label for="option3"> option3</label>
<input type="checkbox" id="option3" name="check"
value="option3"/>

```

option1 option2 option3

6. Radio

Radio buttons are a type of input field that allows the user to **select one option** from a list of predefined options. the "**name**" attribute in the radio button (`<input type="radio">`) tag **is used to group related radio buttons together** so that **only one radio button can be selected** at a time **within a given group**. By giving all the radio buttons in a group the same name attribute, you ensure that only one value is submitted for that group of radio buttons. This is important when you have a list of options where only one option can be selected, such as in a multiple-choice question or a preference selection.

Unset

```

<label for="option1">option1</label>
<input type="radio" id="option1" name="selected"
value="option1" />

<label for="option2">option2</label>
<input type="radio" id="option2" name="selected"
value="option2"/>

<label for="option3">option3</label>
<input type="radio" id="option3" name="selected"
value="option3"/>

```

option1 option2 option3

Note: Don't forget to give a **value** in the case of **checkboxes** and **radio** buttons.

7. Select and Options

Allows users to select one option from a dropdown of multiple options. Let's define Dropdown,

Unset

```
<select name="selected">
  <option value="Asus">Asus</option>
  <option value="Lenovo">Lenovo</option>
  <option value="Apple">Apple</option>
  <option value="Dell">Dell</option>
</select>
```



8. Textarea

allows users to input multiple lines of input.

9. Details

The **<details>** HTML element generates a disclosure widget that displays information only when the widget is switched to the **open** state.

Unset

```
<details>
  <summary>Topic Name</summary>
  Topic Details Here
</details>
```

Close State	Open State
► Topic Name	▼ Topic Name Topic Details are Here

10. Submit

allows you to submit information to the server.

Let's define Submit input,

Unset

```
<input type="submit" />
```

Submit

11. Datalist

The `<datalist>` tag is used to provide an **autocomplete** feature for `<input>` elements. Users will see a drop-down list of predefined options as they input data.

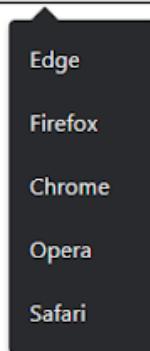
The `<datalist>` element's `id` attribute must be equal to the `<input>` element's `list` attribute (this binds them together).

Unset

```
<!DOCTYPE html>
<html>
<body>
<h1>The datalist element</h1>
<form action="/action_page.php" method="get">
<label for="browser">Choose your browser from the
list:</label>
<input list="browsers" name="browser" id="browser">
<datalist id="browsers">
<option value="Edge">
<option value="Firefox">
<option value="Chrome">
<option value="Opera">
<option value="Safari">
</datalist>
<input type="submit">
</form>
</body>
</html>
```

The datalist element

Choose your browser from the list:



Legend

The **<legend>** tag is used within a **<fieldset>** to provide a **caption or title** for the fieldset. It is often used to group related form controls together and provide context or instructions for the user. The text within the **<legend> tag** is typically styled to stand out from the rest of the form controls, making it easier for the user to identify the purpose of the group of controls.

Unset

```
<fieldset>
  <legend>Choose your favorite monster</legend>

  <input type="radio" id="kraken" name="monster" value="K">
  <label for="kraken">Kraken</label><br>

  <input type="radio" id="sasquatch" name="monster"
value="S">
  <label for="sasquatch">Sasquatch</label><br>

  <input type="radio" id="mothman" name="monster" value="M">
  <label for="mothman">Mothman</label>
</fieldset>
```

—Choose your favorite monster—

- Kraken
- Sasquatch
- Mothman

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>input</title>
</head>
<body>
  <fieldset>
    <legend>user details</legend>
    <form action="#" method="GET">
      ←!— name (input text) —→
      <div>
        <label for="name"> Name </label>
        <input
          type="text"
          name="Name"
          id="name"
          required
          placeholder="Name"
        />
      </div>
      ←!— age (input number)—→
      <div>
        <label for="age">Age</label>
        <input type="number" name="Age" id="age" required placeholder="Age" />
      </div>
      ←!— email (input email)—→
      <div>
        <label for="email"> Email</label>
        <input
          type="email"
          name="Email"
          id="email"
          required
          placeholder="Email"
        />
      </div>
      ←!— gender (input radio) —→
      <div>
        Gender:
        <label for="male">male</label>
        <input type="radio" id="male" name="Gender" value="male" />
      </div>
    </form>
  </fieldset>
</body>

```

```

<label for="female">female</label>
<input type="radio" id="female" name="Gender" value="female" />

<label for="other">other</label>
<input type="radio" id="other" name="Gender" value="other" />
</div>
<!-- skills (input checkbox) -->
<div>
  Skills :
  <label for="HTML">HTML</label>
  <input type="checkbox" id="HTML" name="skill" value="HTML" checked />

  <label for="CSS">CSS</label>
  <input type="checkbox" id="CSS" name="skill" value="CSS" />

  <label for="javascript">javascript</label>
  <input
    type="checkbox"
    id="javascript"
    name="skill"
    value="javascript"
  />
</div>
<!-- submit -->
<input type="submit">
</form>
</fieldset>
</body>
</html>

```

Browser output

user details

Name	<input type="text" value="jhon"/>
Age	<input type="text" value="22"/>
Email	<input type="text" value="jhon@gmail.com"/>
Gender: male <input checked="" type="radio"/> female <input type="radio"/> other <input type="radio"/>	
Skills : HTML <input checked="" type="checkbox"/> CSS <input type="checkbox"/> javascript <input type="checkbox"/>	
<input type="button" value="submit"/>	

Here we create the simple html form that takes the user's details, like name, age, email, gender, and skills. After we submit the form, the form data will be sent to the server for processing.

Since we have used the "GET" method, you will see the form data in the browser tab in key value pairs (key=value), where key is the **name** that we provided for the inputs, and the **value** will be the same as entered by the user.

browser tab:

Unset

```
http://127.0.0.1:5500/index.html?Name=jon&Age=22&Email=jhon%40  
gmail.com &Gender=male&skill=HTML&skill=CSS#
```