

HTML <input> type Attribute

The **HTML <input> type Attribute** is used to specify the type of <input> element to display. The default type of <input> type attribute is text.

Syntax:

```
<input type="value">
```

Attribute Values:

Name	Description
<u>button</u>	Defines clickable button in HTML document, commonly activated with JavaScript scripts.
<u>checkbox</u>	HTML input field for selecting one or more limited options.
<u>color</u>	It is used to define a color picker. The value should be a seven-character hexadecimal notation. Its default value is #000000(black).
<u>date</u>	HTML input element for selecting dates with year, month, and day.
<u>email</u>	HTML field for entering email addresses, automatically validated for format.
<u>file</u>	It is used to specify the file select field and add a button to choose a file for upload to the form.
<u>hidden</u>	Hidden field: Input element in forms, conceals data from users, facilitating backend operations without user interaction, typically for database updates.
<u>image</u>	It is used to define an image as the submit button.
<u>month</u>	It is used to specify the control of month and year field. The value must be in the format of “YYYY-MM”.
<u>number</u>	It is used to specify an input field for entering a number.
<u>password</u>	Input tag attribute for sensitive data; should be transmitted securely (HTTPS) due to its sensitivity.
<u>radio</u>	input element with type=radio, allows single selection from predefined options in forms.
<u>range</u>	Control for user-entered numbers, allows setting restrictions, default range typically 0 to 100.
<u>reset</u>	It is used to defines a reset button. The reset button is used to reset all the form values to its initial values.
<u>search</u>	It is used to define a text field that entered a search string.
<u>submit</u>	Triggers submission of user-input values to form handler, typically activating server-side processing script.
<u>tel</u>	It is used to define a field that entering a user telephone Number.

<u>text</u>	It is used to define a single-line text field . The default width of the text field is 20 characters.
<u>time</u>	It is used to specify the entering time control field.
<u>url</u>	It is used to define a field that entered a URL. This input value is automatically validated before submitted the form.
<u>week</u>	It is used to define a week and a year control field.

HTML <input> type Attribute Examples

Example: In this example we have HTML form for username and password input with submit and reset buttons. Input type attributes: “text” for username, “password” for password.

```
<!DOCTYPE html>
<html>

<body>
  <h3>HTML input type Attribute</h3>

  <form action="#" method="get">
    Username: <input type="text" name="uname">
    <br><br>
    Password: <input type="password" name="pwd">
    <br><br>
    <button type="submit" value="submit">
      Submit
    </button>
    <button type="reset" value="reset">
      Reset
    </button>
  </form>
</body>

</html>
```

Output:

HTML input type Attribute

Username:

Password:

HTML | value Attribute

The **value** attribute in HTML is used to specify the value of the element with which it is used. It has different meaning for different HTML elements.

Usage: It can be used with the following elements: <input>, <button>, <meter>, , <option>, <progress>, and <param>, <output>.

<input>: When the value attribute is present, it specifies the initial value of the input element. It has a different meaning for different input type:

- When present in “button”, “reset” and “submit” it specifies the text on the button.
- When present in “text”, “password” and “hidden” it specifies the initial value of the input field.
- When present in “checkbox”, “radio” and “image” it specifies the value associated with the input.

Syntax:

```
<input value = "value">
```

Example-1:

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML value Attribute</title>
</head>

<body style="text-align:center">
  <h1 style="color:green;">
    ProdipAlways
  </h1>

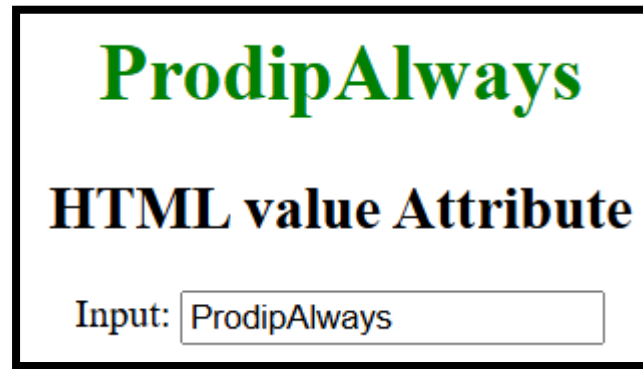
  <h2>
    HTML value Attribute
  </h2>

  Input: <input type="text" value="ProdipAlways">

</body>

</html>
```

Output:



Example-2:

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML value Attribute</title>
</head>

<body style="text-align:center">
  <h1 style="color:green;">
    ProdipAlways
  </h1>

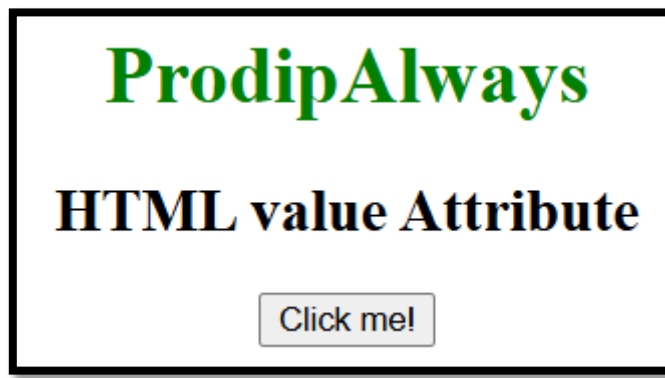
  <h2>
    HTML value Attribute
  </h2>

  <input type="button" value="Click me!">

</body>

</html>
```

Output:



<button>: When the value attribute is present, it specifies the initial value of the button element.

Syntax:

```
<button value = "value">
```

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML value Attribute</title>
</head>

<body style="text-align:center">
  <h1 style="color:green;">
    ProdipAlways
  </h1>

  <h2>
    HTML value Attribute
  </h2>

  <button id="btn" value="ProdipAlways" onclick="prodip()">
    Click me!</button>

  <p id="g"></p>

  <script>
    function prodip() {
      var x = document.getElementById("btn").value;
      document.getElementById("g").innerHTML = "Welcome to " + x;
    }
  </script>
</body>
</html>
```

```
</script>
</body>

</html>
```

Output:

Before clicking the button:



After clicking the button:



<meter>: It specifies the current value of the gauge. The value must be between min and max attribute.

Syntax:

```
<meter value = "value">
```

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML value Attribute</title>
</head>

<body style="text-align:center">
```

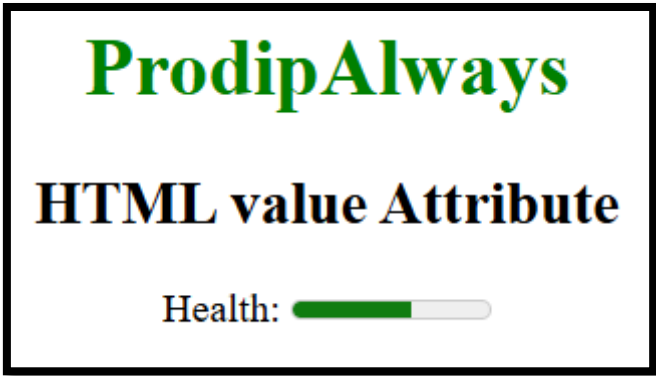
```
<h1 style="color:green;">
  ProdipAlways
</h1>

<h2>
  HTML value Attribute
</h2>

<p>Health: <meter min="0" low="40" high="90" max="100" value="60"></meter></p>
</body>

</html>
```

Output:



: When the value attribute is present, it specifies the initial value of the list item. It is only applicable on the ordered list.

Syntax:

```
<li value = "number">list item </li>
```

- **number:** specifies the value of the list item.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML value Attribute</title>
</head>

<body>
  <h1 style="color: green;">
    ProdipAlways
```



```

</h1>

<h2>
  HTML value Attribute
</h2>

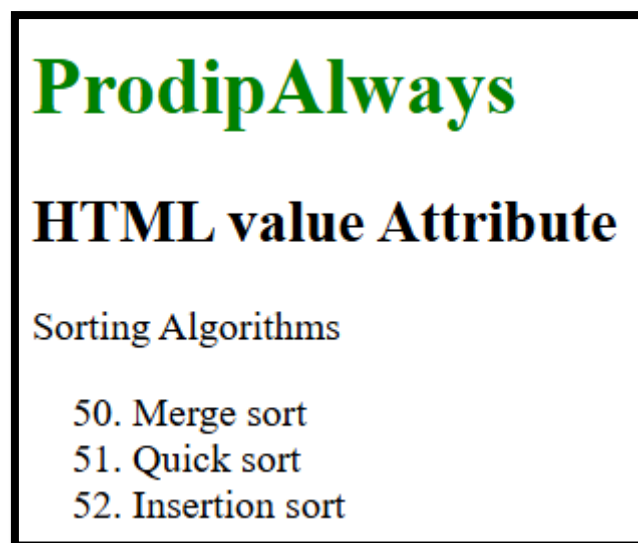
<p>Sorting Algorithms</p>

<ol>
  <li value="50">Merge sort</li>
  <li>Quick sort</li>
  <li>Insertion sort</li>
</ol>
</body>

</html>

```

Output:



<option>: When the value attribute is present, it specifies the value of the option element.

Syntax:

```
<option value = "value"></option>
```

Example:

```

<!DOCTYPE html>
<html>

<head>
  <title>HTML value Attribute</title>

```

```

</head>

<body style="text-align:center">

  <h1 style="color: green;">
    ProdipAlways
  </h1>

  <h2>
    HTML value Attribute
  </h2>

  Sorting Algorithms:
  <select id="opt">
    <option value="quick">Quick sort</option>
    <option value="merge">Merge sort</option>
    <option value="insertion">Insertion sort</option>
  </select>

  <button type="button" onclick="prodip()">Click me!</button>

  <p id="p"></p>

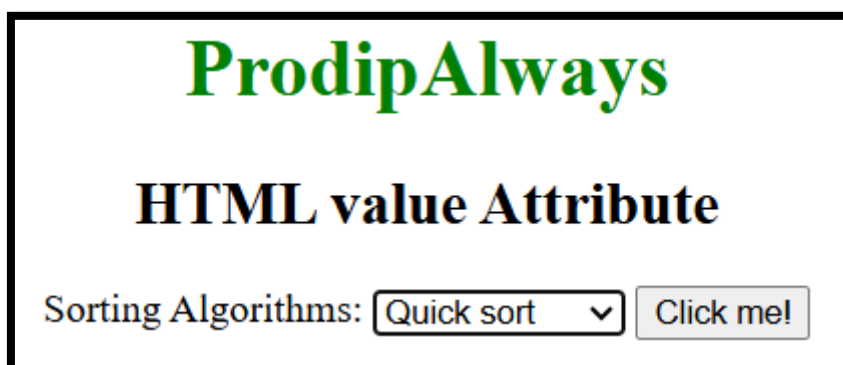
  <script>
    function prodip() {
      var x = document.getElementById("opt").selectedIndex;
      var y = document.getElementsByTagName("option")[x].value;
      document.getElementById("p").innerHTML = "The selected option has value equals " + y + ".";
    }
  </script>
</body>

</html>

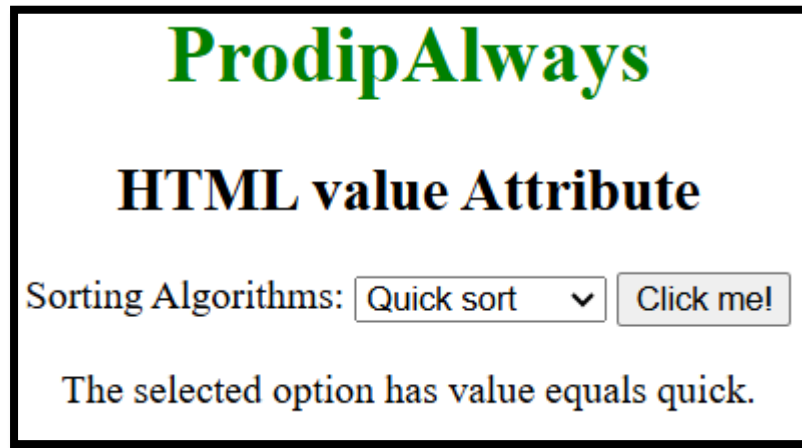
```

Output:

Before clicking the button:



After clicking the button:



<progress>: When the value attribute is present, it specifies the value of the progress element.

Syntax:

```
<progress value = "number"></progress>
```

- **number** specifies the initial value of the progress element.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML value Attribute</title>
</head>

<body style="text-align:center">
  <h1 style="color: green;">
    ProdipAlways
  </h1>

  <h2>
    HTML value Attribute
  </h2>

  Progress:
  <progress value="65" max="100">
  </progress>
</body>

</html>
```

Output:

ProdipAlways

HTML value Attribute

Progress:

HTML placeholder Attribute

HTML placeholder attribute specifies a short hint that describes the expected value of an input field/text area. The short hint is displayed in the field before the user enters a value.

Syntax:

```
<element placeholder="">
```

HTML placeholder Attribute Examples

Example 1: HTML <input> placeholder attribute specifies a short hint or example text displayed in an input field before the user enters a value. It helps users understand the expected input format or content.

```
<!DOCTYPE html>
<html>

<body>
  <h3>
    HTML placeholder Attribute Example
  </h3>
  <!-- HTML placeholder attribute used in input tag -->
  <form action=" ">
    <label for="First Name">Enter your first name:</label>
    <input type="text" name="fname" placeholder="First name">
    <br>
    <br>
    <label for="First Name">Enter your last name:</label>
    <input type="text" name="lname" placeholder="Last name">
    <br>
    <br>
    <input type="submit" value="Submit">
  </form>
</body>

</html>
```

Output: In this example, we demonstrate placeholder attributes in input elements. Two text input fields for first and last names. Placeholder text guides user input, improving form usability.

HTML placeholder Attribute Example

Enter your first name:

Enter your last name:

Example 2: HTML <textarea> placeholder attribute is used to specify the expected value to be displayed before user input in textarea element.

```
<!DOCTYPE html>
<html>

<body>
  <h3>
    HTML placeholder Attribute Example
  </h3>
  <!-- HTML placeholder attribute used in textarea tag -->
  <textarea rows="4" cols="50" placeholder="Describe yourself here...">
  </textarea>
</body>

</html>
```

Output: In this example we demonstrates placeholder attribute in textarea elements. We have a text area with in which user can write their introduction

HTML placeholder Attribute Example

HTML | <input> name Attribute

The **HTML <input> name Attribute** is used to *specify a name for an <input> element*. It is used to reference the form-data after submitting the form or to reference the element in a JavaScript.

Syntax:

```
<input name="name">
```

Attribute Values: It contains a single value name which describes the name of the <input> element.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML Input name Attribute
  </title>
</head>

<body style="text-align:center;">

  <h1>ProdipAlways</h1>

  <h2>HTML Input name Attribute</h2>
  <form id="myProdip">
    <input type="text" id="text_id" name="prodip" pattern="[A-Za-z]{3}" value="Manas Chhabra">
  </form>
  <br>
</body>

</html>
```

Output:

ProdipAlways

HTML Input name Attribute

HTML | <input> alt Attribute

The **HTML <input> alt Attribute** is used to *specify the alternative text for an image when the image attribute is not displayed*. It gives alternative information for the user when the image is not loaded to display.

Syntax:

```
<input alt="text">
```

Attribute Values: It contains single value text which is used to specify the alternative text for input if the image is not displaying.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML Input alt Attribute
  </title>
</head>

<body style="text-align:center;">

  <h1 style="color:green;">
    ProdipAlways
  </h1>

  <h2>HTML Input alt:\Picture\prodip.png Attribute</h2>
  <input id="myImage" type="image" src="prodip.png" alt="Submit" width="48" height="48">
</body>

</html>
```

Output:

ProdipAlways

HTML Input alt:\Picture\prodip.png Attribute



HTML input autofocus Attribute

The **HTML <input> autofocus Attribute** is used to specify that the input field must automatically get focused when the page loads which means the cursor will blink to the specified input field. It is a Boolean attribute.

Syntax:

```
<input autofocus>
```

Example: This Example illustrates the use of autofocus attribute in <input> element.

```
<!DOCTYPE html>
<html>

<head>
  <title> HTML input autofocus Attribute</title>
</head>

<body style="text-align:center">

  <h1 style="color: green;">ProdipAlways</h1>
  <h2> HTML input autofocus Attribute</h2>
  Name:
  <input type="text" autofocus>
  <br><br>
  <!-- Assign id to the Button. -->
  <button id="pdaw">Submit</button>
  <br>
</body>

</html>
```

Output:

ProdipAlways

HTML input autofocus Attribute

Name:

HTML <input>checked Attribute

The **HTML <input checked> attribute** is used with checkboxes or radio buttons to specify that the input should be selected (checked) when the page loads. It's a boolean attribute, meaning its presence implies "true."

Syntax

```
<input type = "checkbox|radio" checked>
```

Example: In this example, we show the use of the checked attribute in checkboxes. The first checkbox is checked by default using checked, while the second checkbox is unchecked by default.

```
<!DOCTYPE html>

<html>

<head>
  <title>
    HTML <input>checked
    Attribute
  </title>
</head>

<body style="text-align: center;">
  <h1 style="color: green;">
    ProdipAlways
  </h1>
  <h2>
    HTML <input>checked
    Attribute
  </h2>
  <form>
    <!-- Below input elements
         have attribute "checked" -->
    <input type="checkbox" name="check" value="1" checked>
    Checked by default
    <br>
    <input type="checkbox" name="check" value="2">
    Not checked by default
    <br>
  </form>
</body>
```

Output:

ProdipAlways

HTMLchecked Attribute

☒ Checked by default

☐ Not checked by default

HTML <input> disabled Attribute

The **disabled** attribute for <input> element in HTML is used to specify that the input field is disabled. A disabled input is un-clickable and unusable. It is a boolean attribute. The disabled <input> elements are not submitted in the form.

Syntax:

```
<input disabled>
```

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML input disabled Attribute</title>
</head>

<body style="text-align:center">
  <h1 style="color: green;">ProdipAlways</h1>
  <h2>HTML input disabled Attribute</h2>

  <label>Input:

    <!--A disabled input-->
    <input type="text" name="value" value="This input field
      is disabled" disabled>

  </label>
</body>

</html>
```

Output:

ProdipAlways

HTML input disabled Attribute

Input:

HTML <input> form Attribute

The **HTML <input> form Attribute** is used to specify interactive input fields for web-based forms. A form can contain multiple input fields to accept inputs from the users. It is the most powerful element in HTML.

Syntax:

```
<input form="form_id">
```

Attribute Value: This attribute contains a single value **form_id** which contains the value i.e **form_id** which specify the one or more than the button element belongs to. The value of this attribute should be the id of the **<form>** element.

Example: This Example illustrates the use of form attribute in **<input>** element.

```
<!DOCTYPE html>
<html>

<body>
  <h1>ProdipAlways</h1>

  <h2>
    HTML &lt;input&gt; form Attribute
  </h2>

  <form id="myProdip"></form>
  <br />
  Name:
  <input type="text" id="btn" name="myProdip" form="myProdip" />
</body>

</html>
```

Output:

ProdipAlways

HTML <input> form Attribute

Name:

HTML | <input>max Attribute

The **HTML|<input> max Attribute** is used to *specify the maximum value for an input field*. It can be used together with the min attribute to create a range of values. It can be used with the many input fields such as number, range, date, datetime, datetime-local, month, time and week. **Syntax:**

```
<input max="number|date">
```

Attribute Values:

- **number:** It contains the value i.e number which specifies the maximum value allowed by the input field.
- **date:** It contains the value i.e date which specifies the maximum date allowed for the <input> date field.

Example:

```
<!DOCTYPE html>
<html>

<body style="text-align:center;">

  <h1 style="color:green;">
    ProdipAlways
  </h1>

  <h2>
    HTML | max Attribute in Input Field
  </h2>

  <form id="myProdip">
    <input type="number" id="myNumber" step="5" name="prodip" placeholder="multiples of 5" max="100">
  </form>

  <br>
  <br>

  <p style="font-size:20px;">
    The maximum value for an input
    field is 100.
  </p>
</body>

</html>
```


Output:

ProdipAlways

HTML | max Attribute in Input Field

The maximum value for an input field is 100.

HTML | <input> required Attribute

The **HTML required Attribute** is a Boolean attribute which is used to specify that the input element must be filled out before submitting the Form.

This attribute works with other types of input like radio, checkbox, number, text, etc.

Syntax:

```
<input required>
```

Example-1: This Example that illustrates the use of required attribute in input Element.

```
<!DOCTYPE html>
<html>

<head>
  <title>
    required Attribute
  </title>
  <style>
    h1,
    h2 {
      color: green;
      font-style: italic;
    }

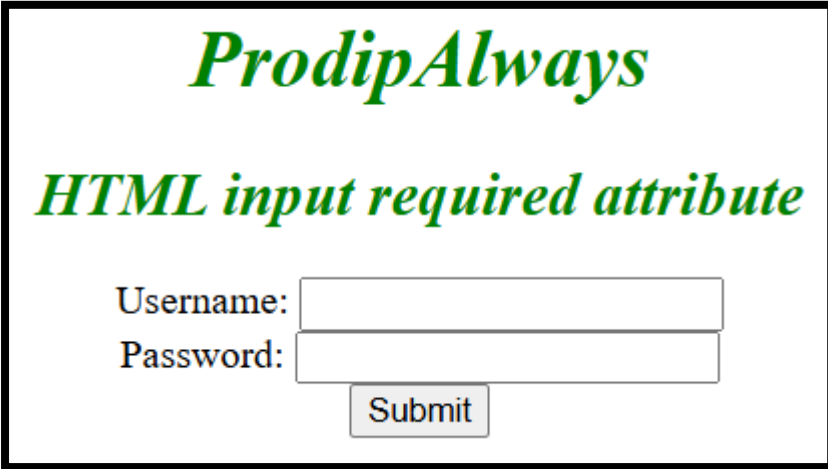
    body {
      text-align: center;
    }
  </style>
</head>

<body>
  <h1>ProdipAlways</h1>
  <h2>
    HTML input required attribute
  </h2>
  <form action="">
    Username:
    <input type="text" name="username" required>
    <br> Password:
    <input type="password" name="password">
    <br>
    <input type="submit">
```

```
</form>
</body>

</html>
```

Output :



ProdipAlways

HTML input required attribute

Username:

Password:

Example-2: This Example that illustrates the use of required attribute in input Element.

```
<!DOCTYPE html>
<html>

<head>
  <title>
    required Attribute
  </title>
  <style>
    h1,
    h2 {
      color: green;
      font-style: italic;
    }

    body {
      text-align: center;
    }
  </style>
</head>

<body>
  <h1>ProdipAlways</h1>
```

```
<h2>
  HTML input required attribute
</h2>
<form action="">
  Required:
  <input type="radio" name="radiocheck" required>
  <br>
  <input type="submit">
</form>
</body>

</html>
```

Output :



HTML | input readonly Attribute

The **readonly** attribute of <input> element in HTML is used to specify that the input field is read-only. If an input is readonly, then it's content cannot be changed but can be copied and highlighted. It is a boolean attribute.

Syntax:

```
<input readonly>
```

Example: This example uses HTML <input> readonly Attribute.

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML Input readonly Attribute</title>
</head>

<body style="text-align:center">
  <h1 style="color: green;">
    ProdipAlways
  </h1>

  <h2>
    HTML Input readonly Attribute
  </h2>

  <label>Input:

  <!--A readonly input-->
  <input type="text" name="value" value="This input field is readonly" readonly>
</label>
</body>

</html>
```

Output:

ProdipAlways

HTML Input readonly Attribute

Input:

HTML accept Attribute

The **HTML <input>** accept attribute is used to control the type of files that can be selected for input, restricting the file selection to specific formats such as image/* for images or .pdf for PDF files.

Syntax:

```
<input accept = "file_extension | audio/* | video/* | image/* | media_type">
```

```
<html>

<body>

  <form>

    <label for="fileUpload">Upload an image file:</label>

    <input type="file" id="fileUpload" name="fileUpload" accept="image/*">

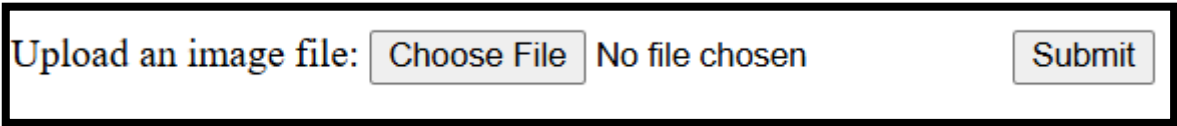
    <input type="submit" value="Submit">

  </form>

</body>

</html>
```

Output:



- The accept=""image/*"" attribute restricts the file input to accept only image files, enhancing user experience by filtering selectable files in the dialog.
- The <input type=""file""> element allows users to browse and select files from their device for upload.

Attribute Values:

Value	Description
file_extension	Specify the file extension(s) like .gif, .jpg, .png, .doc) the user can pick from.
audio/*	The user can pick all sound files.
image/*	A valid media type, with no parameters. Look at IANA Media Types for a complete list of standard media types.
media_type	A valid media type without parameters.

More example of HTML <input> accept Attribute

HTML <input> accept Attribute

```
<html>

<body>

  <form>

    <label for="imageUpload">Upload an image (JPEG or PNG):</label>

    <input type="file" id="imageUpload" name="imageUpload" accept=".jpeg, .jpg, .png">

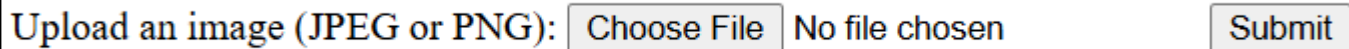
    <input type="submit" value="Submit">

  </form>

</body>

</html>
```

Output:



Upload an image (JPEG or PNG): No file chosen

- The accept attribute is set to .jpeg, .jpg, .png, allowing users to select image files with these specific extensions.
- This ensures that only JPEG and PNG images can be selected for upload.

Styled File Input for PDF Documents

```
<html>

<head>

  <style>

    .file-input {

      display: inline-block;

      padding: 10px 20px;

      font-size: 16px;

      cursor: pointer;

      background-color: #4CAF50;

      color: white;

      border: none;

      border-radius: 4px;

    }

    .file-input:hover {

      background-color: #45a049;
```



```
}

#pdfUpload {
  display: none;
}

</style>
</head>

<body>
  <form>
    <label for="pdfUpload" class="file-input">Choose a PDF file</label>
    <input type="file" id="pdfUpload" name="pdfUpload" accept="application/pdf">
    <input type="submit" value="Submit">
  </form>
</body>

</html>
```

Output:



The screenshot displays a web form with a black border. It contains two buttons: a green button with the text "Choose a PDF file" and a white button with the text "Submit".

HTML <input> align Attribute

The **HTML <input> align attribute** is used with **<input type=" image">** to set the horizontal alignment of the image. It is not supported by HTML 5.

Note: Since **<input type=" image">** is not supported by HTML5 you can use CSS there instead of this attribute. The syntax for CSS **<input type="image" style="float: right">**.

Syntax:

```
<input align="left|right|middle|top|bottom">
```

Attribute Values:

Attribute Values	Description
left	It sets the alignment of the image to the left. it is a default value.
right	It sets the alignment of the image to the right.
middle	It sets the alignment of the image to the middle.
top	It sets the alignment of image to the top.
bottom	It sets the alignment of image to the bottom.

Example1:

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML input align Attribute
  </title>
</head>

<body>
  <h1 style="color:green;">
    ProdipAlways
  </h1>

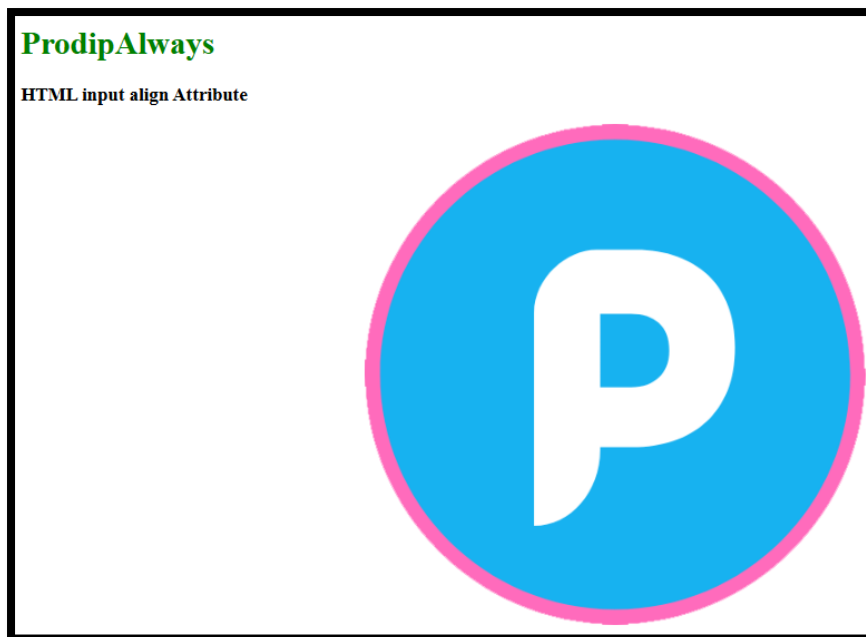
  <h3>
    HTML input align Attribute
  </h3>

  <input type="image" src="prodip.png" alt="ProdipAlways logo" align="right">
```

```
</body>
```

```
</html>
```

Output:



Example 2:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>
```

```
    HTML input align Attribute
```

```
</title>
```

```
</head>
```

```
<body>
```

```
<center>
```

```
<h1 style="color:green;">
```

```
    ProdipAlways
```

```
</h1>
```

```
<h3>
```

```
    HTML input align Attribute
```

```
</h3>
```

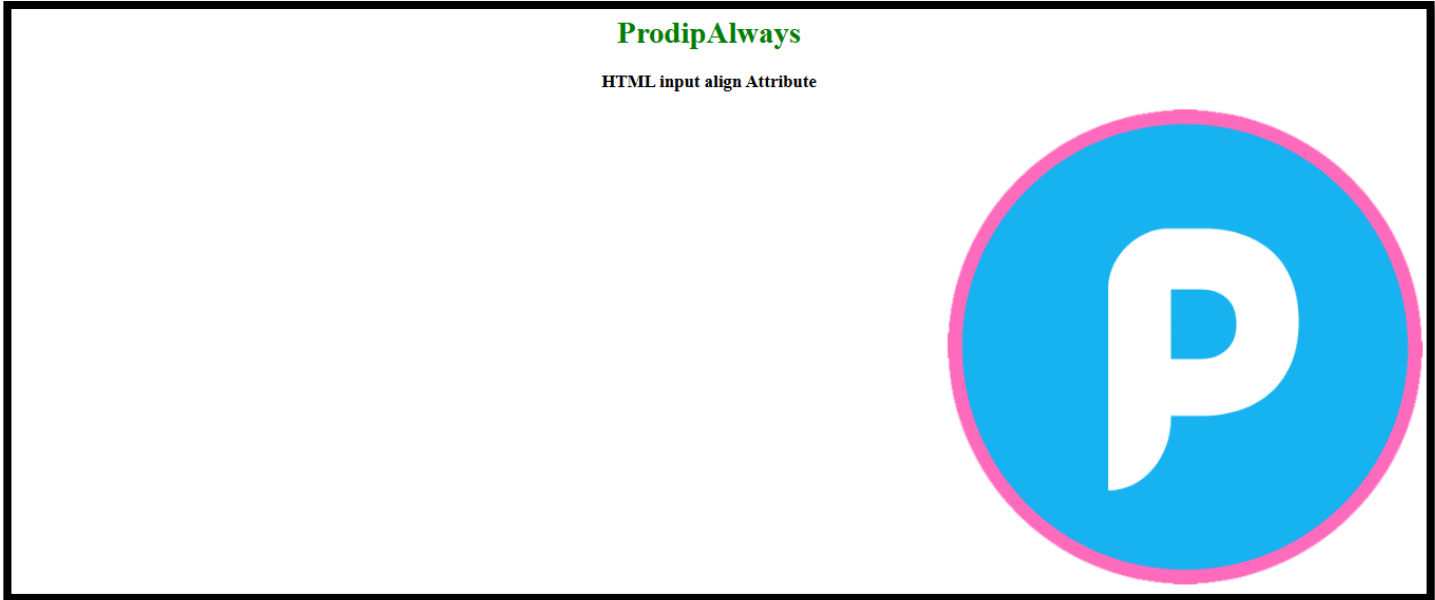
```
</center>
```

```
<div>
```

```
<input type="image" src="prodip.png" alt="ProdipAlways" align="right">
</div>
</body>

</html>
```

Output:



HTML input autocomplete Attribute

The **HTML | <input>autocomplete Attribute** is used to specify whether *the input field has autocompleted and would be on or off*. When the autocomplete attribute is set to on the browser will automatically complete the values based on what the user entered before. It works with many input fields such as text, search, URL, email, password, date pickers, range, and color.

Syntax:

```
<input autocomplete="on|off">
```

Attribute Values:

- **on:** It has a default value. It specifies that autocomplete is enabled.
- **off:** It specifies that the autocomplete is disabled.

Example: This Example illustrates the use of autocomplete attribute in **<input>** element.

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML | input autocomplete Attribute
  </title>
</head>

<body style="text-align:center;">

  <h1>ProdipAlways</h1>
  <h2>
    HTML | input autocomplete Attribute
  </h2>
  <form id="myProdip">
    <input type="text" id="text_id" name="prodip" autocomplete="on">

    <input type="submit">
  </form>
  <br>
</body>

</html>
```

Output:

ProdipAlways

HTML | input autocomplete Attribute

HTML | <input> dirname Attribute

The **HTML input dirname Attribute** is used to enable the text direction of the input and the Textarea Field after submitting the form. The value of the dirname attribute must be the name of the input field and textarea, followed by **“.dir”**. it can be used with both of the **<input>** and **<textarea>** element.

Syntax:

```
<input name="myname" dirname="myname.dir">
```

Attribute Values:

- **name.dir** It specify that the Text direction of the input field will be submitted.

Below Example illustrates the use of dirname attribute in input Element.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>
  </title>
  <style>
    h1 {
      color: green;
    }
  </style>
</head>

<body>

  <form>
    <h1>
      ProdipAlways
    </h1>
    <h2>
      HTML | Input dirname attribute
    </h2> First name:
    <input type="text" name="fname" dirname="fname.dir">

    <input type="submit" value="Submit">
  </form>
```

<p>

After the Submission of the form, the text direction of the input field will also be submitted.

</p>

</body>

</html>

Output:

ProdipAlways

HTML | Input dirname attribute

First name:

After the Submission of the form, the text direction of the input field will also be submitted.

HTML | <input> formaction Attribute

The **HTML <input> formaction Attribute** is used to *specify where to send the data of the form*. After submission of the form the formaction attribute called. The form data is to be sent to the server after submission of the form. It overrides the feature of the action attribute of an <form> element.

Syntax:

```
<input formaction="URL">
```

Attribute Values: It contains single value URL which is used to specify the URL of the document where the data to be sent after submission of the form. The possible value of URL are:

- **absolute URL:** It points to the full address of a page. For example: *www.ProdipAlways.org/data-structure*
- **relative URL:** It is used to point to a file within in a webpage. For Example: *gfg.php*

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML &lt;input&gt; formAction Attribute
  </title>
</head>

<body style="text-align:center;">
  <h1>
    ProdipAlways
  </h1>

  <h2>
    HTML &lt;input&gt; formAction Attribute
  </h2>

  <form action="#" method="get" target="_self">

    <input type="text" id="Prodip" name="myProdip" value="ProdipAlways" formTarget="_blank" formMethod="post"
      formAction="test.php">
    <input type="submit">
  </form>
</body>
```

```
</html>
```

Output:

ProdipAlways

HTML <input> formAction Attribute

ProdipAlways

Submit

HTML | <input> formenctype Attribute

The **HTML <input> formenctype Attribute** is used to specify how to form-data encoded when submitting the form into the server. This attribute works with **method="post"**. This attribute overrides the enctype attribute of the **<form>** element.

Syntax:

```
<input formenctype="value">
```

Attribute Values:

- **application/x-www-form-urlencoded:** It is the default value. It encodes all the characters before sent to the server. It converts all spaces into “+” symbols and special character into its “hex” value.
- **multipart/form-data:** This value does not encode any character.
- **text/plain:** This value convert spaces into “+” symbols but special characters are not converted.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML input formenctype Attribute</title>
</head>

<body style="text-align: center">

  <h1 style="color: green">ProdipAlways</h1>

  <h4>HTML &lt;input&gt; formenctype Attribute</h4>

  <form action="#" method="post">

    First name:
    <input type="text" name="fname"> Last name:
    <input type="text" name="lname">
    <br>
    <br>

    <input type="submit" value="Submit" formenctype="multipart/form-data">
  </form>
```

```
</body>
```

```
</html>
```

Output:

ProdipAlways

HTML <input> formenctype Attribute

First name: Last name:

Submit

HTML | input formmethod Attribute

The **HTML <input> formmethod** Attribute is used to specify the HTTP method for sending form-data to the action URL. This attribute is used to override the method attribute of the <form> element.

Syntax:

```
<input formmethod="get | post">
```

Attribute Values:

- **get:** It has a default value. 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. It has a limited size of about 3000 characters. It is only useful for non-secure data not for sensitive information.
- **post:** In the post method, after the submission of the form, the form values will not be visible in the address bar of the new browser tab as it was visible in the GET method. It appends form data inside the body of the HTTP request. It has no size limitation. This method does not support bookmark the result.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML input formmethod Attribute
  </title>
</head>

<body style="text-align:center;">

  <h1 style="color:green;">
    ProdipAlways
  </h1>

  <h3>HTML &lt;input&gt; formmethod Attribute</h3>

  <form action="#" id="users" action="#" method="GET" target="_blank">

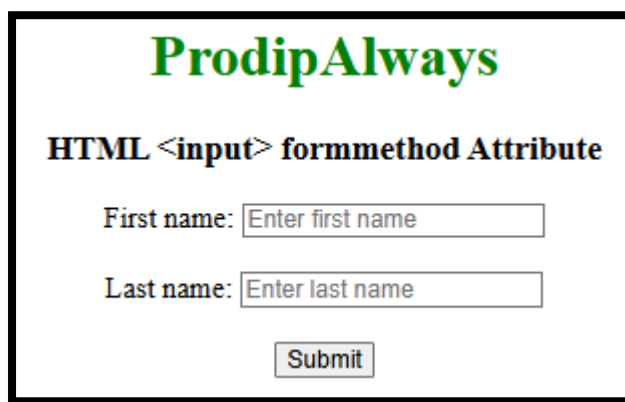
    First name:

    <input type="text" name="fname" placeholder="Enter first name">

    <br>
```

```
<br> Last name:  
<input type="text" name="lname" placeholder="Enter last name">  
<br>  
<br>  
  
<input type="submit" value="Submit" formmethod="post">  
  
</form>  
</body>  
  
</html>
```

Output:



ProdipAlways

HTML `<input>` formmethod Attribute

First name:

Last name:

HTML | <input> formnovalidate Attribute

The **HTML <input> formnovalidate Attribute** is used to specify that the Input Element should not be validated when submitting the form. It is a Boolean attribute, this attribute can be used to override the **novalidate** attribute of the <form> element and it can only be used with the <input type="submit">.

Syntax:

```
<input formnovalidate="formnovalidate">
```

Example: This Example illustrate the use of <input> formnovalidate Attribute.

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML Input formNoValidate Attribute
  </title>
  <style>
    h1 {
      color: green;
    }
  </style>
</head>

<body style="text-align:center;">
  <h1>
    ProdipAlways
  </h1>

  <h4>
    HTML Input formNoValidate Attribute
  </h4>

  <form action="#" method="get" target="_self">
    Username:
    <input type="email" name="userid">
    <input type="submit" id="Prodip" name="myProdip" value="Submit @ ProdipAlways" formTarget="_blank"
formnovalidate>

  </form>
```

</body>

</html>

Output:

ProdipAlways

HTML Input formNoValidate Attribute

Username:

Submit @ ProdipAlways

HTML | <input> formtarget Attribute

The HTML <input> formtarget attribute is used to specify the name or a keyword which indicates where to display the response after submitting the form. This attribute overrides the target attribute of <form> element.

Syntax:

```
<input formtarget="_blank|_self|_parent|_top|frameName">
```

Attribute Values:

- **_blank:** The input response is display in a new window.
- **_self:** The input response is display in the same frame. it is the default value.
- **_parent:** The input response is display in the parent frameset.
- **_top:** The input response is display in the full body of the window.
- **frameName:** The input response is display in the named frame.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML input formtarget Attribute
  </title>
</head>

<body>
  <center>
    <h1 style="color:green;">
      ProdipAlways
    </h1>

    <h2>
      HTML &lt;input&gt; formtarget Attribute
    </h2>

    <form action="#">
      Username:
      <input type="text" name="username">
```

```
<br><br>
```

Password:

```
<input type="text" name="password">
```

```
<br><br>
```

```
<input type="submit" value="Submit" formtarget="_blank">
```

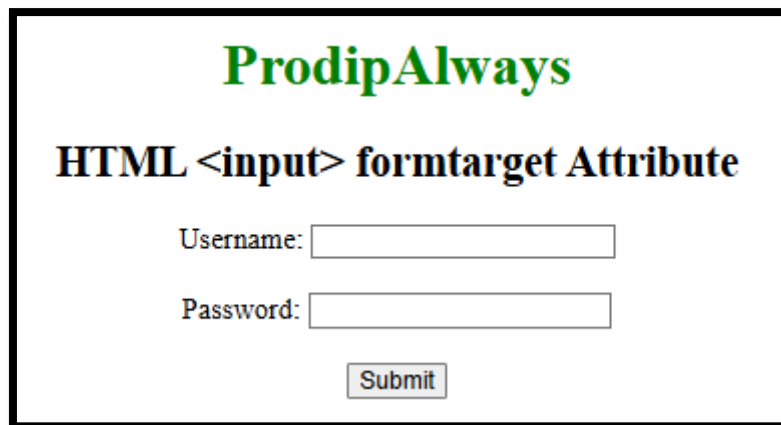
```
</form>
```

```
</center>
```

```
</body>
```

```
</html>
```

Output:



ProdipAlways

HTML <input> formtarget Attribute

Username:

Password:

HTML <input> height Attribute

The **HTML <input> element's height attribute** sets the height of the input field in pixels. It only applies to image inputs (<input type="image">). For other input types, use CSS to control the height instead.

Syntax

```
<input height="pixels">
```

Attribute Values

It contains the value i.e pixels which specify the height of the input Element.

Example: In this example, we display an image input element with a specified height and width using the height attribute. The image is aligned centrally on the page.

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML Input Image height Attribute
  </title>
</head>

<body style="text-align:center;">

  <h1 style="color:green;">
    ProdipAlways
  </h1>

  <h2>HTML Input image Height Attribute</h2>
  <input id="myImage" type="image" src="prodip.png" alt="Submit" width="70" height="96" />

</body>

</html>
```

Output:

ProdipAlways

HTML Input image Height Attribute



HTML | <input> list Attribute

The **HTML <input> list Attribute** is used to *identify a list of pre-defined options for an element to suggest the user.*

Syntax:

```
<input list="datalist_id">
```

Attribute Values:

- **datalist_id:** It is used to specify the Id of the datalist that will be used to make a link up with the input element.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML Input list Attribute
  </title>
</head>

<body>
  <h1>
    ProdipAlways
  </h1>
  <h1 style="color:green">
    HTML Input list Attribute
  </h1>

  <form action="">
    <label>Your Cars Name: </label>
    <input list="cars">
    <datalist id="cars">
      <option value="BMW" />
      <option value="Bentley" />
      <option value="Mercedes" />
      <option value="Audi" />
      <option value="Volkswagen" />
    </datalist>
  </form>
```

```
</body>
```

```
</html>
```

Output:

ProdipAlways

HTML Input list Attribute

Your Cars Name:

- BMW
- Bentley
- Mercedes
- Audi
- Volkswagen

HTML input maxlength Attribute

The maxlength attribute in the HTML <input> element is used to define the maximum number of characters that can be entered into the field. If no value is specified, or if an invalid value is provided, the input field will have no maximum length.

The length is determined in UTF-16 code units, which typically correspond to the number of characters.

Syntax

```
<input maxlength="number">
```

Attribute Value

Attribute Value	Description
number	It contains a single value number which allows the maximum number of characters in <input> element. Its default value is 524288.

Examples of Using the maxlength Attribute

Example 1: Limiting Username and Password Length

In this example, the maxlength attribute is used within <input> elements to restrict the length of text inputs for username and password fields.

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML input maxlength Attribute</title>
</head>

<body>
  <h1>ProdipAlways</h1>
  <h2>HTML <input> maxlength Attribute</h2>
  <form action="#">
    Username:
    <input type="text" name="username" maxlength="12" />
    <br />
    <br />
    Password:
    <input type="text" name="password" maxlength="10" />
    <br />
  </form>
```

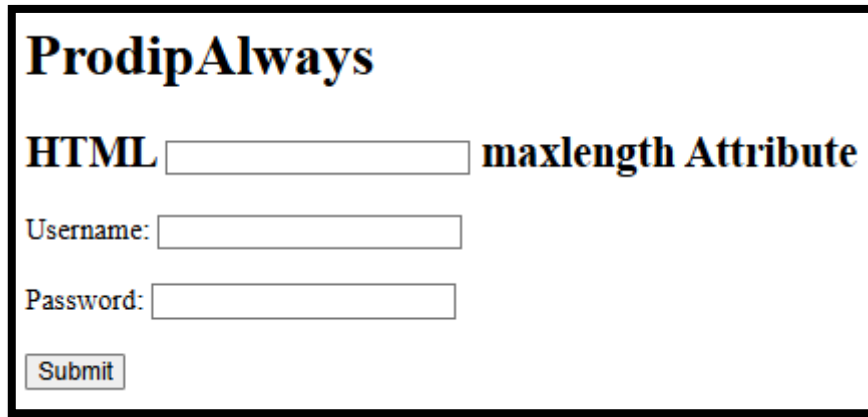
```
<input type="submit" value="Submit" />

</form>

</body>

</html>
```

Output:

A screenshot of a web browser displaying a form. The form has a title "ProdipAlways" in a large, bold, black serif font. Below the title is a heading "HTML <input> maxlength Attribute" in a smaller, bold, black serif font. Underneath the heading are two input fields: "Username:" followed by a text input box, and "Password:" followed by a text input box. At the bottom left of the form is a "Submit" button with a light gray background and a thin black border. The entire form is enclosed in a black rectangular border.

Example 2: Limiting Email Address Length

Here, the maxlength attribute limits the length of an email input field to 30 characters.

```
<!DOCTYPE html>

<html>

<head>

  <title>HTML input maxlength Attribute</title>

</head>

<body>

  <h1>ProdipAlways</h1>

  <h2>HTML <input> maxlength Attribute</h2>

  <form action="#">

    Email:

    <input type="email" name="email" maxlength="30" style="width: 350px" />

    <br /><br />

    <input type="submit" value="Submit" />

  </form>

</body>

</html>
```

Output:

ProdipAlways

HTML maxlength Attribute

Email:

HTML | <input> min Attribute

The **HTML min Attribute** is used to *specify the minimum value for an input field*. It can be used together with the max attribute to create a range of values. It can be used with the many input fields such as number, range, date, datetime, datetime-local, month, time and week. **Syntax:**

```
<input min="number|date">
```

Attribute Values:

- **number:** It contains the value i.e number which specifies the minimum value allowed by the input field.
- **date:** It contains the value i.e date which specifies the minimum date allowed for the <input> date field.

Example: This Example illustrates the use of min attribute in <input> element.

```
<!DOCTYPE html>
<html>

<body style="text-align:center;">

  <h1 style="color:green;">
    ProdipAlways
  </h1>

  <h2>
    HTML | min Attribute in Input Field
  </h2>

  <form id="myProdip">
    <input type="number" id="myNumber" step="5" name="prodip" placeholder="multiples of 5" min="10">
  </form>

  <br>
  <br>

  <p style="font-size:20px;">
    The minimum value for an input field is 10.
  </p>
</body>

</html>
```

Output:

ProdipAlways

HTML | min Attribute in Input Field

multiples of 5

The minimum value for an input field is 10.

HTML <input> multiple Attribute

The **HTML <input> multiple Attribute** is a Boolean Attribute. It specifies that the user is allowed to select more than one value that is present in an element. The multiple attributes work with many input fields such as email, file, etc.

Syntax:

```
<input multiple>
```

HTML <input> multiple Attribute Example

Example: This example uses **multiple attribute** to create a form with multiple file input in single input element.

```
<!DOCTYPE html>
<html>

<body>
  <h2>HTML input multiple Attribute</h2>
  <form action=" ">
    Select images:
    <!-- Using multiple attribute
         for multiple file input-->
    <input type="file" name="img" multiple />
    <input type="submit" />
  </form>
</body>

</html>
```

Output:



HTML input multiple Attribute

Select images: No file chosen

HTML | <input> pattern Attribute

The **HTML <input> pattern attribute** is used to specify the regular expression on which the input elements value is checked against. This attribute works with the following input types: text, password, date, search, email, etc. Use the Global title attribute to describe the pattern for helping the user.

Syntax:

```
<input pattern = "regular_exp">
```

Example: This example illustrates the use of pattern attribute in <input> element.

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML input pattern attribute
  </title>

  <style>
    body {
      text-align: center;
    }

    h1 {
      color: green;
    }
  </style>
</head>

<body>
  <h1>ProdipAlways</h1>

  <h2>HTML &lt;input&gt;pattern attribute</h2>

  <form action="#">
    Password: <input type="text" name="Password" pattern="[A-Za-z]{3}" title="Three letter Password">

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

</html>

Output:

ProdipAlways

HTML <input>pattern attribute

Password:

HTML <input> size Attribute

HTML <input> size Attribute specifies the visible width of an <input> element, typically for text input fields. It determines the width in terms of characters displayed, providing visual control over input size.

Syntax:

```
<input size="number">
```

Attribute values: It contains the numeric value which specifies the width of an input field in terms of the number of characters. Its default value is 20.

HTML <input> size Attribute Examples

Example: Here is the basic implementation of HTML <input> size Attribute.

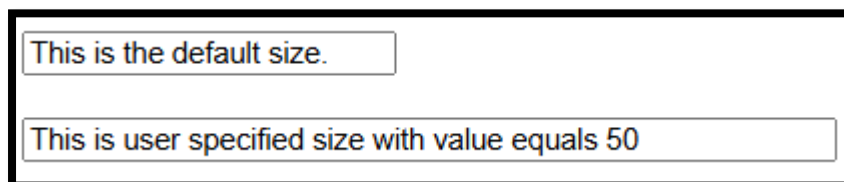
```
<!DOCTYPE html>
<html>

<head>
  <title>HTML input size Attribute</title>
</head>

<body>
  <input type="text" value="This is the default size.">
  <br>
  <br>
  <input type="text" size="50" value="This is user specified size with value equals 50">
</body>

</html>
```

Output:



The screenshot displays two text input fields side-by-side. The first field is narrow, with a width of approximately 20 characters, and contains the text "This is the default size.". The second field is much wider, with a width of 50 characters, and contains the text "This is user specified size with value equals 50". Both fields have a light gray border and a white background.

Explanation:

- In the above example The first <input> element lacks a size attribute, so it displays with the default width for text input fields.
- The second <input> element has size="50", making it wider, accommodating up to 50 characters within the visible area.

- Demonstrates how the size attribute affects the width of input fields, allowing users to input text comfortably based on the specified size.

HTML | <input> src Attribute

The **HTML <input> src Attribute** is used to *specify the URL of the image to be used as a submit button*. This attribute can only be used with **<input type="image">**.

Syntax:

```
<input src="URL">
```

Attribute Values: It contains a single value URL that specifies the link of the source image. There are two types of URL links which are listed below:

- **Absolute URL:** It points to another webpage.
- **Relative URL:** It points to other files of the same web page.

Example:

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML Input src Attribute
  </title>
</head>

<body style="text-align:center;">

  <h1 style="color:green;">
    ProdipAlways
  </h1>

  <h2>HTML Input src Attribute</h2>

  <form>
    <input id="myImage" type="image" src="prodip.png" alt="Submit" width="70" height="48" />
  </form>
</body>

</html>
```

Output:

ProdipAlways

HTML Input src Attribute



HTML <input> step Attribute

The **HTML <input> step Attribute** in HTML is used to *set the discrete step size of the <input> element. The default stepping value for number inputs is 1.*

Usage: It works with the following input types:

- number
- range
- date
- datetime-local
- month
- time
- week

Syntax:

```
<input step = "value">
```

Attribute Values: It contains a value i.e **number** which specifies the legal number interval for the number field. It has a default value which is 1.

Example: This Example illustrates the use of step attribute in <input> element.

```
<!DOCTYPE html>
<html>

<head>
  <title>
    HTML input step Attribute
  </title>
</head>

<body>
  <center>
    <h1 style="color: green;">
      ProdipAlways
    </h1>

    <h2>
      HTML <input>step Attribute
    </h2>
```

```
<input type="number" name="points" step="5" placeholder="multiples of 5">

</center>

</body>

</html>
```

Output:

ProdipAlways

HTML step Attribute

HTML | <input> width Attribute

The **HTML <input> width Attribute** is used to specify the width of the <input> element. This Attribute is only used for input **type="image"**.

Syntax:

```
<input width="pixels">
```

Attribute Values: It contains the value i.e **pixels** which specify the width of the input Element.

Example:

```
<!DOCTYPE html>
<html>

<body style="text-align:center;">

  <h1 style="color:green;">
    ProdipAlways
  </h1>

  <h2>HTML Input width Attribute</h2>
  <input id="myImage" type="image" src="prodip.png" alt="Submit" width="48" height="48" />

</body>

</html>
```

Output:

