HTML Tables

HTML tables allow web developers to arrange data into rows and columns

Define an HTML Table

The <table> tag defines an HTML table.

Each table row is defined with a <tr> tag. Each table header is defined with a <th> tag. Each table data/cell is defined with a <td> tag.

By default, the text in <th> elements are bold and centered.

By default, the text in <td> elements are regular and left-aligned.

Example

A simple HTML table:

<table style="width:100%">  
  <tr>  
    <th>Firstname</th>  
    <th>Lastname</th>  
    <th>Age</th>  
  </tr>  
  <tr>  
    <td>Jill</td>  
    <td>Smith</td>  
    <td>50</td>  
  </tr>  
  <tr>  
    <td>Eve</td>  
    <td>Jackson</td>  
    <td>94</td>  
  </tr>  
</table>

**Note:** The <td> elements are the data containers of the table.  
They can contain all sorts of HTML elements; text, images, lists, other tables, etc.

HTML Table - Add a Border

To add a border to a table, use the CSS border property:

Example

table, th, td {  
  border: 1px solid black;  
}

Remember to define borders for both the table and the table cells.

HTML Table - Collapsed Borders

To let the borders collapse into one border, add the CSS border-collapse property:

Example

table, th, td {  
  border: 1px solid black;  
  border-collapse: collapse;  
}

HTML Table - Add Cell Padding

Cell padding specifies the space between the cell content and its borders.

If you do not specify a padding, the table cells will be displayed without padding.

To set the padding, use the CSS padding property:

Example

th, td {  
  padding: 15px;  
}

HTML Table - Left-align Headings

By default, table headings are bold and centered.

To left-align the table headings, use the CSS text-align property:

Example

th {  
  text-align: left;  
}

HTML Table - Add Border Spacing

Border spacing specifies the space between the cells.

To set the border spacing for a table, use the CSS border-spacing property:

Example

table {  
  border-spacing: 5px;  
}

**Note:** If the table has collapsed borders, border-spacing has no effect.

HTML Table - Cell that Span Many Columns

To make a cell span more than one column, use the colspan attribute:

Example

<table style="width:100%">  
  <tr>  
    <th>Name</th>  
    <th colspan="2">Telephone</th>  
  </tr>  
  <tr>  
    <td>Bill Gates</td>  
    <td>55577854</td>  
    <td>55577855</td>  
  </tr>  
</table>

HTML Table - Cell that Span Many Rows

To make a cell span more than one row, use the rowspan attribute:

Example

<table style="width:100%">  
  <tr>  
    <th>Name:</th>  
    <td>Bill Gates</td>  
  </tr>  
  <tr>  
    <th rowspan="2">Telephone:</th>  
    <td>55577854</td>  
  </tr>  
  <tr>  
    <td>55577855</td>  
  </tr>  
</table>

HTML Table - Add a Caption

To add a caption to a table, use the <caption> tag:

Example

<table style="width:100%">  
  <caption>Monthly savings</caption>  
  <tr>  
    <th>Month</th>  
    <th>Savings</th>  
  </tr>  
  <tr>  
    <td>January</td>  
    <td>$100</td>  
  </tr>  
  <tr>  
    <td>February</td>  
    <td>$50</td>  
  </tr>  
</table>

**Note:** The <caption> tag must be inserted immediately after the <table> tag.

A Special Style for One Table

To define a special style for one particular table, add an id attribute to the table:

Example

<table id="t01">  
  <tr>  
    <th>Firstname</th>  
    <th>Lastname</th>  
    <th>Age</th>  
  </tr>  
  <tr>  
    <td>Eve</td>  
    <td>Jackson</td>  
    <td>94</td>  
  </tr>  
</table>

Now you can define a special style for this table:

#t01 {  
  width: 100%;  
  background-color: #f1f1c1;  
}

And add more styles:

#t01 tr:nth-child(even) {  
  background-color: #eee;  
}  
#t01 tr:nth-child(odd) {  
  background-color: #fff;  
}  
#t01 th {  
  color: white;  
  background-color: black;  
}

Chapter Summary

* Use the HTML <table> element to define a table
* Use the HTML <tr> element to define a table row
* Use the HTML <td> element to define a table data
* Use the HTML <th> element to define a table heading
* Use the HTML <caption> element to define a table caption
* Use the CSS border property to define a border
* Use the CSS border-collapse property to collapse cell borders
* Use the CSS padding property to add padding to cells
* Use the CSS text-align property to align cell text
* Use the CSS border-spacing property to set the spacing between cells
* Use the colspan attribute to make a cell span many columns
* Use the rowspan attribute to make a cell span many rows
* Use the id attribute to uniquely define one table

# HTML Forms

An HTML form is used to collect user input. The user input is most often sent to a server for processing.

## The <form> Element

The HTML <form> element is used to create an HTML form for user input:

<form>  
.  
*form elements*  
.  
</form>

The <form> element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc.

All the different form elements are covered in this chapter: [HTML Form Elements](https://www.w3schools.com/html/html_form_elements.asp).

## The <input> Element

The HTML <input> element is the most used form element.

An <input> element can be displayed in many ways, depending on the type attribute.

Here are some examples:

|  |  |
| --- | --- |
| **Type** | **Description** |
| <input type="text"> | Displays a single-line text input field |
| <input type="radio"> | Displays a radio button (for selecting one of many choices) |
| <input type="checkbox"> | Displays a checkbox (for selecting zero or more of many choices) |
| <input type="submit"> | Displays a submit button (for submitting the form) |
| <input type="button"> | Displays a clickable button |

All the different input types are covered in this chapter: [HTML Input Types](https://www.w3schools.com/html/html_form_input_types.asp).

## Text Fields

The <input type="text"> defines a single-line input field for text input.

### Example

A form with input fields for text:

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_text)

This is how the HTML code above will be displayed in a browser:

Top of Form

First name:  
  
Last name:  


Bottom of Form

**Note:** The form itself is not visible. Also note that the default width of an input field is 20 characters.

## The <label> Element

Notice the use of the <label> element in the example above.

The <label> tag defines a label for many form elements.

The <label> element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.

The <label> element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the <label> element, it toggles the radio button/checkbox.

The for attribute of the <label> tag should be equal to the id attribute of the <input> element to bind them together.

## Radio Buttons

The <input type="radio"> defines a radio button.

Radio buttons let a user select ONE of a limited number of choices.

### Example

A form with radio buttons:

<form>  
  <input type="radio" id="male" name="gender" value="male">  
  <label for="male">Male</label><br>  
  <input type="radio" id="female" name="gender" value="female">  
  <label for="female">Female</label><br>  
  <input type="radio" id="other" name="gender" value="other">  
  <label for="other">Other</label>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_radio)

This is how the HTML code above will be displayed in a browser:

 Male  
 Female  
 Other

## Checkboxes

The <input type="checkbox"> defines a **checkbox**.

Checkboxes let a user select ZERO or MORE options of a limited number of choices.

### Example

A form with checkboxes:

<form>  
  <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">  
  <label for="vehicle1"> I have a bike</label><br>  
  <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">  
  <label for="vehicle2"> I have a car</label><br>  
  <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">  
  <label for="vehicle3"> I have a boat</label>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_checkbox)

This is how the HTML code above will be displayed in a browser:

 I have a bike  
 I have a car  
 I have a boat

## The Submit Button

The <input type="submit"> defines a button for submitting the form data to a form-handler.

The form-handler is typically a file on the server with a script for processing input data.

The form-handler is specified in the form's action attribute.

### Example

A form with a submit button:

<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>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_submit)

This is how the HTML code above will be displayed in a browser:

Top of Form

First name:  
  
Last name:  
  
  


Bottom of Form

## The Name Attribute for <input>

Notice that each input field must have a name attribute to be submitted.

If the name attribute is omitted, the value of the input field will not be sent at all.

### Example

This example will not submit the value of the "First name" input field:

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

HTML Form Attribute

This chapter describes the different attributes for the HTML <form> element.

The Action Attribute

The action attribute defines the action to be performed when the form is submitted.

Usually, the form data is sent to a file on the server when the user clicks on the submit button.

In the example below, the form data is sent to a file called "action\_page.php". This file contains a server-side script that handles the form data:

Example

On submit, send form data to "action\_page.php":

<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>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_submit)

**Tip:** If the action attribute is omitted, the action is set to the current page.

The Target Attribute

The target attribute specifies where to display the response that is received after submitting the form.

The target attribute can have one of the following values:

|  |  |
| --- | --- |
| **Value** | **Description** |
| \_blank | The response is displayed in a new window or tab |
| \_self | The response is displayed in the current window |
| \_parent | The response is displayed in the parent frame |
| \_top | The response is displayed in the full body of the window |
| *framename* | The response is displayed in a named iframe |

The default value is \_self which means that the response will open in the current window.

Example

Here, the submitted result will open in a new browser tab:

<form action="/action\_page.php" target="\_blank">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_target)

The Method Attribute

The method attribute specifies the HTTP method to use used when submitting the form data.

The form-data can be sent as URL variables (with method="get") or as HTTP post transaction (with method="post").

The default HTTP method when submitting form data is GET.

Example

This example uses the GET method when submitting the form data:

<form action="/action\_page.php" method="get">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_get)

Example

This example uses the POST method when submitting the form data:

<form action="/action\_page.php" method="post">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_post)

**Notes on GET:**

* Appends the form data to the URL, in name/value pairs
* NEVER use GET to send sensitive data! (the submitted form data is visible in the URL!)
* The length of a URL is limited (2048 characters)
* Useful for form submissions where a user wants to bookmark the result
* GET is good for non-secure data, like query strings in Google

**Notes on POST:**

* Appends the form data inside the body of the HTTP request (the submitted form data is not shown in the URL)
* POST has no size limitations, and can be used to send large amounts of data.
* Form submissions with POST cannot be bookmarked

**Tip:** Always use POST if the form data contains sensitive or personal information!

The Autocomplete Attribute

The autocomplete attribute specifies whether a form should have autocomplete on or off.

When autocomplete is on, the browser automatically complete values based on values that the user has entered before.

Example

A form with autocomplete on:

<form action="/action\_page.php" autocomplete="on">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_autocomplete)

The Novalidate Attribute

The novalidate attribute is a boolean attribute.

When present, it specifies that the form-data (input) should not be validated when submitted.

Example

A form with a novalidate attribute:

<form action="/action\_page.php" novalidate>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_novalidate)

HTML Exercises

Top of Form

Test Yourself With Exercises

Exercise:

Add a submit button, and specify that the form should go to "/action\_page.php".

<form ="/action\_page.php">  
Name: <input type="text" name="name">  
<>  
</form>

Submit Answer »

[Start the Exercise](https://www.w3schools.com/html/exercise.asp?filename=exercise_html_forms_attributes1)

Bottom of Form

List of All <form> Attributes

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| [accept-charset](https://www.w3schools.com/tags/att_form_accept_charset.asp) | Specifies the character encodings used for form submission |
| [action](https://www.w3schools.com/tags/att_form_action.asp) | Specifies where to send the form-data when a form is submitted |
| [autocomplete](https://www.w3schools.com/tags/att_form_autocomplete.asp) | Specifies whether a form should have autocomplete on or off |
| [enctype](https://www.w3schools.com/tags/att_form_enctype.asp) | Specifies how the form-data should be encoded when submitting it to the server (only for method="post") |
| [method](https://www.w3schools.com/tags/att_form_method.asp) | Specifies the HTTP method to use when sending form-data |
| [name](https://www.w3schools.com/tags/att_form_name.asp) | Specifies the name of the form |
| [novalidate](https://www.w3schools.com/tags/att_form_novalidate.asp) | Specifies that the form should not be validated when submitted |
| [rel](https://www.w3schools.com/tags/att_form_rel.asp) | Specifies the relationship between a linked resource and the current document |
| [target](https://www.w3schools.com/tags/att_form_target.asp) | Specifies where to display the response that is received after submitting the form |

HTML Form Elements

This chapter describes all the different HTML form elements.

The HTML <form> Elements

The HTML <form> element can contain one or more of the following form elements:

* <input>
* <label>
* <select>
* <textarea>
* <button>
* <fieldset>
* <legend>
* <datalist>
* <output>
* <option>
* <optgroup>

The <input> Element

One of the most used form element is the <input> element.

The <input> element can be displayed in several ways, depending on the type attribute.

Example

<label for="fname">First name:</label>  
<input type="text" id="fname" name="fname">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_input)

All the different values of the type attribute are covered in the next chapter: [HTML Input Types](https://www.w3schools.com/html/html_form_input_types.asp).

The <label> Element

The <label> element defines a label for several form elements.

The <label> element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.

The <label> element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the <label> element, it toggles the radio button/checkbox.

The for attribute of the <label> tag should be equal to the id attribute of the <input> element to bind them together.

The <select> Element

The <select> element defines a drop-down list:

Example

<label for="cars">Choose a car:</label>  
<select id="cars" name="cars">  
  <option value="volvo">Volvo</option>  
  <option value="saab">Saab</option>  
  <option value="fiat">Fiat</option>  
  <option value="audi">Audi</option>  
</select>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_select)

The <option> elements defines an option that can be selected.

By default, the first item in the drop-down list is selected.

To define a pre-selected option, add the selected attribute to the option:

Example

<option value="fiat" selected>Fiat</option>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_select_pre)

Visible Values:

Use the size attribute to specify the number of visible values:

Example

<label for="cars">Choose a car:</label>  
<select id="cars" name="cars" size="3">  
  <option value="volvo">Volvo</option>  
  <option value="saab">Saab</option>  
  <option value="fiat">Fiat</option>  
  <option value="audi">Audi</option>  
</select>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_select_size)

Allow Multiple Selections:

Use the multiple attribute to allow the user to select more than one value:

Example

<label for="cars">Choose a car:</label>  
<select id="cars" name="cars" size="4"multiple>  
  <option value="volvo">Volvo</option>  
  <option value="saab">Saab</option>  
  <option value="fiat">Fiat</option>  
  <option value="audi">Audi</option>  
</select>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_select_multiple)

The <textarea> Element

The <textarea> element defines a multi-line input field (a text area):

Example

<textarea name="message" rows="10" cols="30">  
The cat was playing in the garden.  
</textarea>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_textarea)

The rows attribute specifies the visible number of lines in a text area.

The cols attribute specifies the visible width of a text area.

This is how the HTML code above will be displayed in a browser:



You can also define the size of the text area by using CSS:

Example

<textarea name="message" style="width:200px; height:600px;">  
The cat was playing in the garden.  
</textarea>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_textarea_style)

The <button> Element

The <button> element defines a clickable button:

Example

<button type="button" onclick="alert('Hello World!')">Click Me!</button>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_button)

This is how the HTML code above will be displayed in a browser:

Click Me!

**Note:** Always specify the type attribute for the button element. Different browsers may use different default types for the button element.

The <fieldset> and <legend> Elements

The <fieldset> element is used to group related data in a form.

The <legend> element defines a caption for the <fieldset> element.

Example

<form action="/action\_page.php">  
  <fieldset>  
    <legend>Personalia:</legend>  
    <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">  
  </fieldset>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_legend)

This is how the HTML code above will be displayed in a browser:

Top of Form

Personalia:First name:  
  
Last name:  
  
  


Bottom of Form

The <datalist> Element

The <datalist> element specifies a list of pre-defined options for an <input> element.

Users will see a drop-down list of the pre-defined options as they input data.

The list attribute of the <input> element, must refer to the id attribute of the <datalist> element.

Example

<form action="/action\_page.php">  
  <input list="browsers">  
  <datalist id="browsers">  
    <option value="Internet Explorer">  
    <option value="Firefox">  
    <option value="Chrome">  
    <option value="Opera">  
    <option value="Safari">  
  </datalist>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_datalist)

The <output> Element

The <output> element represents the result of a calculation (like one performed by a script).

Example

Perform a calculation and show the result in an <output> element:

<form action="/action\_page.php"  
  oninput="x.value=parseInt(a.value)+parseInt(b.value)">  
  0  
  <input type="range"  id="a" name="a" value="50">  
  100 +  
  <input type="number" id="b" name="b" value="50">  
  =  
  <output name="x" for="a b"></output>  
  <br><br>  
  <input type="submit">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_output)

HTML Exercises

Top of Form

Test Yourself With Exercises

Exercise:

In the form below, add an empty drop down list with the name "cars".

<form action="/action\_page.php">  
<>  
</>  
</form>

Submit Answer »

[Start the Exercise](https://www.w3schools.com/html/exercise.asp?filename=exercise_html_form_elements1)

Bottom of Form

HTML Form Elements

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<form>](https://www.w3schools.com/tags/tag_form.asp) | Defines an HTML form for user input |
| [<input>](https://www.w3schools.com/tags/tag_input.asp) | Defines an input control |
| [<textarea>](https://www.w3schools.com/tags/tag_textarea.asp) | Defines a multiline input control (text area) |
| [<label>](https://www.w3schools.com/tags/tag_label.asp) | Defines a label for an <input> element |
| [<fieldset>](https://www.w3schools.com/tags/tag_fieldset.asp) | Groups related elements in a form |
| [<legend>](https://www.w3schools.com/tags/tag_legend.asp) | Defines a caption for a <fieldset> element |
| [<select>](https://www.w3schools.com/tags/tag_select.asp) | Defines a drop-down list |
| [<optgroup>](https://www.w3schools.com/tags/tag_optgroup.asp) | Defines a group of related options in a drop-down list |
| [<option>](https://www.w3schools.com/tags/tag_option.asp) | Defines an option in a drop-down list |
| [<button>](https://www.w3schools.com/tags/tag_button.asp) | Defines a clickable button |
| [<datalist>](https://www.w3schools.com/tags/tag_datalist.asp) | Specifies a list of pre-defined options for input controls |
| [<output>](https://www.w3schools.com/tags/tag_output.asp) | Defines the result of a calculation |

HTML Input Types

This chapter describes the different types for the HTML <input> element.

HTML Input Types

Here are the different input types you can use in HTML:

* <input type="button">
* <input type="checkbox">
* <input type="color">
* <input type="date">
* <input type="datetime-local">
* <input type="email">
* <input type="file">
* <input type="hidden">
* <input type="image">
* <input type="month">
* <input type="number">
* <input type="password">
* <input type="radio">
* <input type="range">
* <input type="reset">
* <input type="search">
* <input type="submit">
* <input type="tel">
* <input type="text">
* <input type="time">
* <input type="url">
* <input type="week">

**Tip:** The default value of the type attribute is "text".

Input Type Text

<input type="text"> defines a **single-line text input field**:

Example

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_text)

This is how the HTML code above will be displayed in a browser:

First name:  
  
Last name:  


Input Type Password

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

Example

<form>  
  <label for="username">Username:</label><br>  
  <input type="text" id="username" name="username"><br>  
  <label for="pwd">Password:</label><br>  
  <input type="password" id="pwd" name="pwd">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_password)

This is how the HTML code above will be displayed in a browser:

Username:  
  
Password:  


The characters in a password field are masked (shown as asterisks or circles).

Input Type Submit

<input type="submit"> defines a button for **submitting** form data to a **form-handler**.

The form-handler is typically a server page with a script for processing input data.

The form-handler is specified in the form's action attribute:

Example

<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>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_submit)

This is how the HTML code above will be displayed in a browser:

Top of Form

First name:  
  
Last name:  
  
  


Bottom of Form

If you omit the submit button's value attribute, the button will get a default text:

Example

<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">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_submit_nn)

Input Type Reset

<input type="reset"> defines a **reset button** that will reset all form values to their default values:

Example

<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">  
  <input type="reset">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_reset)

This is how the HTML code above will be displayed in a browser:

Top of Form

First name:  
  
Last name:  
  
  
 

Bottom of Form

If you change the input values and then click the "Reset" button, the form-data will be reset to the default values.

Input Type Radio

<input type="radio"> defines a **radio button**.

Radio buttons let a user select ONLY ONE of a limited number of choices:

Example

<form>  
  <input type="radio" id="male" name="gender" value="male">  
  <label for="male">Male</label><br>  
  <input type="radio" id="female" name="gender" value="female">  
  <label for="female">Female</label><br>  
  <input type="radio" id="other" name="gender" value="other">  
  <label for="other">Other</label>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_radio)

This is how the HTML code above will be displayed in a browser:

 Male  
 Female  
 Other

Input Type Checkbox

<input type="checkbox"> defines a **checkbox**.

Checkboxes let a user select ZERO or MORE options of a limited number of choices.

Example

<form>  
  <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">  
  <label for="vehicle1"> I have a bike</label><br>  
  <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">  
  <label for="vehicle2"> I have a car</label><br>  
  <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">  
  <label for="vehicle3"> I have a boat</label>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_checkbox)

This is how the HTML code above will be displayed in a browser:

 I have a bike  
 I have a car  
 I have a boat

Input Type Button

<input type="button"> defines a **button**:

Example

<input type="button" onclick="alert('Hello World!')" value="Click Me!">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_button)

This is how the HTML code above will be displayed in a browser:

Input Type Color

The <input type="color"> is used for input fields that should contain a color.

Depending on browser support, a color picker can show up in the input field.

Example

<form>  
  <label for="favcolor">Select your favorite color:</label>  
  <input type="color" id="favcolor" name="favcolor">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_color)

Input Type Date

The <input type="date"> is used for input fields that should contain a date.

Depending on browser support, a date picker can show up in the input field.

Example

<form>  
  <label for="birthday">Birthday:</label>  
  <input type="date" id="birthday" name="birthday">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_date)

You can also use the min and max attributes to add restrictions to dates:

Example

<form>  
  <label for="datemax">Enter a date before 1980-01-01:</label>  
  <input type="date" id="datemax" name="datemax" max="1979-12-31"><br><br>  
  <label for="datemin">Enter a date after 2000-01-01:</label>  
  <input type="date" id="datemin" name="datemin" min="2000-01-02">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_date_max_min)

Input Type Datetime-local

The <input type="datetime-local"> specifies a date and time input field, with no time zone.

Depending on browser support, a date picker can show up in the input field.

Example

<form>  
  <label for="birthdaytime">Birthday (date and time):</label>  
  <input type="datetime-local" id="birthdaytime" name="birthdaytime">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_datetime-local)

Input Type Email

The <input type="email"> is used for input fields that should contain an e-mail address.

Depending on browser support, the e-mail address can be automatically validated when submitted.

Some smartphones recognize the email type, and add ".com" to the keyboard to match email input.

Example

<form>  
  <label for="email">Enter your email:</label>  
  <input type="email" id="email" name="email">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_email)

Input Type File

The <input type="file"> defines a file-select field and a "Browse" button for file uploads.

Example

<form>  
  <label for="myfile">Select a file:</label>  
  <input type="file" id="myfile" name="myfile">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_file)

Input Type Month

The <input type="month"> allows the user to select a month and year.

Depending on browser support, a date picker can show up in the input field.

Example

<form>  
  <label for="bdaymonth">Birthday (month and year):</label>  
  <input type="month" id="bdaymonth" name="bdaymonth">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_month)

Input Type Number

The <input type="number"> defines a **numeric** input field.

You can also set restrictions on what numbers are accepted.

The following example displays a numeric input field, where you can enter a value from 1 to 5:

Example

<form>  
  <label for="quantity">Quantity (between 1 and 5):</label>  
  <input type="number" id="quantity" name="quantity" min="1" max="5">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_number)

Input Restrictions

Here is a list of some common input restrictions:

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| checked | Specifies that an input field should be pre-selected when the page loads (for type="checkbox" or type="radio") |
| disabled | Specifies that an input field should be disabled |
| max | Specifies the maximum value for an input field |
| maxlength | Specifies the maximum number of character for an input field |
| min | Specifies the minimum value for an input field |
| pattern | Specifies a regular expression to check the input value against |
| readonly | Specifies that an input field is read only (cannot be changed) |
| required | Specifies that an input field is required (must be filled out) |
| size | Specifies the width (in characters) of an input field |
| step | Specifies the legal number intervals for an input field |
| value | Specifies the default value for an input field |

You will learn more about input restrictions in the next chapter.

The following example displays a numeric input field, where you can enter a value from 0 to 100, in steps of 10. The default value is 30:

Example

<form>  
  <label for="quantity">Quantity:</label>  
  <input type="number" id="quantity" name="quantity" min="0" max="100" step="10" value="30">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_number_step)

Input Type Range

The <input type="range"> defines a control for entering a number whose exact value is not important (like a slider control). Default range is 0 to 100. However, you can set restrictions on what numbers are accepted with the min, max, and step attributes:

Example

<form>  
  <label for="vol">Volume (between 0 and 50):</label>  
  <input type="range" id="vol" name="vol" min="0" max="50">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_range)

Input Type Search

The <input type="search"> is used for search fields (a search field behaves like a regular text field).

Example

<form>  
  <label for="gsearch">Search Google:</label>  
  <input type="search" id="gsearch" name="gsearch">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_search)

Input Type Tel

The <input type="tel"> is used for input fields that should contain a telephone number.

Example

<form>  
  <label for="phone">Enter your phone number:</label>  
  <input type="tel" id="phone" name="phone" pattern="[0-9]{3}-[0-9]{2}-[0-9]{3}">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_tel)

Input Type Time

The <input type="time"> allows the user to select a time (no time zone).

Depending on browser support, a time picker can show up in the input field.

Example

<form>  
  <label for="appt">Select a time:</label>  
  <input type="time" id="appt" name="appt">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_time)

Input Type Url

The <input type="url"> is used for input fields that should contain a URL address.

Depending on browser support, the url field can be automatically validated when submitted.

Some smartphones recognize the url type, and adds ".com" to the keyboard to match url input.

Example

<form>  
  <label for="homepage">Add your homepage:</label>  
  <input type="url" id="homepage" name="homepage">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_url)

Input Type Week

The <input type="week"> allows the user to select a week and year.

Depending on browser support, a date picker can show up in the input field.

Example

<form>  
  <label for="week">Select a week:</label>  
  <input type="week" id="week" name="week">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_week)

HTML Exercises

Top of Form

Test Yourself With Exercises

Exercise:

In the form below, add an input field for text, with the name "username" .

<form action="/action\_page.php">  
<>  
</form>

Submit Answer »

[Start the Exercise](https://www.w3schools.com/html/exercise.asp?filename=exercise_html_form_input_types1)

Bottom of Form

HTML Input Type Attribute

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<input type="">](https://www.w3schools.com/tags/att_input_type.asp) | Specifies the input type to display |

# HTML Input Attributes

[❮ Previous](https://www.w3schools.com/html/html_form_input_types.asp)[Next ❯](https://www.w3schools.com/html/html_form_attributes_form.asp)

This chapter describes the different attributes for the HTML <input> element.

## The value Attribute

The input value attribute specifies an initial value for an input field:

### Example

Input fields with initial (default) values:

<form>  
  <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">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_attributes_value)

## The readonly Attribute

The input readonly attribute specifies that an input field is read-only.

A read-only input field cannot be modified (however, a user can tab to it, highlight it, and copy the text from it).

The value of a read-only input field will be sent when submitting the form!

### Example

A read-only input field:

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John" readonly><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_attributes_readonly)

## The disabled Attribute

The input disabled attribute specifies that an input field should be disabled.

A disabled input field is unusable and un-clickable.

The value of a disabled input field will not be sent when submitting the form!

### Example

A disabled input field:

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John" disabled><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_attributes_disabled)

## The size Attribute

The input size attribute specifies the visible width, in characters, of an input field.

The default value for size is 20.

**Note:** The size attribute works with the following input types: text, search, tel, url, email, and password.

### Example

Set a width for an input field:

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" size="50"><br>  
  <label for="pin">PIN:</label><br>  
  <input type="text" id="pin" name="pin" size="4">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_attributes_size)

## The maxlength Attribute

The input maxlength attribute specifies the maximum number of characters allowed in an input field.

**Note:** When a maxlength is set, the input field will not accept more than the specified number of characters. However, this attribute does not provide any feedback. So, if you want to alert the user, you must write JavaScript code.

### Example

Set a maximum length for an input field:

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" size="50"><br>  
  <label for="pin">PIN:</label><br>  
  <input type="text" id="pin" name="pin" maxlength="4" size="4">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_attributes_maxlength)

## The min and max Attributes

The input min and max attributes specify the minimum and maximum values for an input field.

The min and max attributes work with the following input types: number, range, date, datetime-local, month, time and week.

**Tip:** Use the max and min attributes together to create a range of legal values.

### Example

Set a max date, a min date, and a range of legal values:

<form>  
  <label for="datemax">Enter a date before 1980-01-01:</label>  
  <input type="date" id="datemax" name="datemax" max="1979-12-31"><br><br>  
  
  <label for="datemin">Enter a date after 2000-01-01:</label>  
  <input type="date" id="datemin" name="datemin" min="2000-01-02"><br><br>  
  
  <label for="quantity">Quantity (between 1 and 5):</label>  
  <input type="number" id="quantity" name="quantity" min="1" max="5">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_max_min)

## The multiple Attribute

The input multiple attribute specifies that the user is allowed to enter more than one value in an input field.

The multiple attribute works with the following input types: email, and file.

### Example

A file upload field that accepts multiple values:

<form>  
  <label for="files">Select files:</label>  
  <input type="file" id="files" name="files" multiple>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_multiple)

## The pattern Attribute

The input pattern attribute specifies a regular expression that the input field's value is checked against, when the form is submitted.

The pattern attribute works with the following input types: text, date, search, url, tel, email, and password.

**Tip:** Use the global [title](https://www.w3schools.com/tags/att_global_title.asp) attribute to describe the pattern to help the user.

**Tip:** Learn more about [regular expressions](https://www.w3schools.com/js/js_regexp.asp) in our JavaScript tutorial.

### Example

An input field that can contain only three letters (no numbers or special characters):

<form>  
  <label for="country\_code">Country code:</label>  
  <input type="text" id="country\_code" name="country\_code"  
  pattern="[A-Za-z]{3}" title="Three letter country code">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_pattern)

## The placeholder Attribute

The input placeholder attribute specifies short a hint that describes the expected value of an input field (a sample value or a short description of the expected format).

The short hint is displayed in the input field before the user enters a value.

The placeholder attribute works with the following input types: text, search, url, tel, email, and password.

### Example

An input field with a placeholder text:

<form>  
  <label for="phone">Enter a phone number:</label>  
  <input type="tel" id="phone" name="phone"  
  placeholder="123-45-678"  
  pattern="[0-9]{3}-[0-9]{2}-[0-9]{3}">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_placeholder)

## The required Attribute

The input required attribute specifies that an input field must be filled out before submitting the form.

The required attribute works with the following input types: text, search, url, tel, email, password, date pickers, number, checkbox, radio, and file.

### Example

A required input field:

<form>  
  <label for="username">Username:</label>  
  <input type="text" id="username" name="username" required>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_required)

## The step Attribute

The input step attribute specifies the legal number intervals for an input field.

Example: if step="3", legal numbers could be -3, 0, 3, 6, etc.

**Tip:** This attribute can be used together with the max and min attributes to create a range of legal values.

The step attribute works with the following input types: number, range, date, datetime-local, month, time and week.

### Example

An input field with a specified legal number intervals:

<form>  
  <label for="points">Points:</label>  
  <input type="number" id="points" name="points" step="3">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_step)

**Note:** Input restrictions are not foolproof, and JavaScript provides many ways to add illegal input. To safely restrict input, it must also be checked by the receiver (the server)!

## The autofocus Attribute

The input autofocus attribute specifies that an input field should automatically get focus when the page loads.

### Example

Let the "First name" input field automatically get focus when the page loads:

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" autofocus><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_autofocus)

## The height and width Attributes

The input height and width attributes specify the height and width of an <input type="image"> element.

**Tip:** Always specify both the height and width attributes for images. If height and width are set, the space required for the image is reserved when the page is loaded. Without these attributes, the browser does not know the size of the image, and cannot reserve the appropriate space to it. The effect will be that the page layout will change during loading (while the images load).

### Example

Define an image as the submit button, with height and width attributes:

<form>  
  <label for="fname">First name:</label>  
  <input type="text" id="fname" name="fname"><br><br>  
  <label for="lname">Last name:</label>  
  <input type="text" id="lname" name="lname"><br><br>  
  <input type="image" src="img\_submit.gif" alt="Submit" width="48" height="48">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_height_width)

## The list Attribute

The input list attribute refers to a <datalist> element that contains pre-defined options for an <input> element.

### Example

An <input> element with pre-defined values in a <datalist>:

<form>  
  <input list="browsers">  
  <datalist id="browsers">  
    <option value="Internet Explorer">  
    <option value="Firefox">  
    <option value="Chrome">  
    <option value="Opera">  
    <option value="Safari">  
  </datalist>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_datalist)

## The autocomplete Attribute

The input autocomplete attribute specifies whether a form or an input field should have autocomplete on or off.

Autocomplete allows the browser to predict the value. When a user starts to type in a field, the browser should display options to fill in the field, based on earlier typed values.

The autocomplete attribute works with <form> and the following <input> types: text, search, url, tel, email, password, datepickers, range, and color.

### Example

An HTML form with autocomplete on, and off for one input field:

<form action="/action\_page.php" autocomplete="on">  
  <label for="fname">First name:</label>  
  <input type="text" id="fname" name="fname"><br><br>  
  <label for="lname">Last name:</label>  
  <input type="text" id="lname" name="lname"><br><br>  
  <label for="email">Email:</label>  
  <input type="email" id="email" name="email" autocomplete="off"><br><br>  
  <input type="submit" value="Submit">  
</form>

HTML Input form\* Attributes

[❮ Previous](https://www.w3schools.com/html/html_form_attributes.asp)[Next ❯](https://www.w3schools.com/html/html5_canvas.asp)

This chapter describes the different form\* attributes for the HTML <input> element.

The form Attribute

The input form attribute specifies the form the <input> element belongs to.

The value of this attribute must be equal to the id attribute of the <form> element it belongs to.

Example

An input field located outside of the HTML form (but still a part of the form):

<form action="/action\_page.php" id="form1">  
  <label for="fname">First name:</label>  
  <input type="text" id="fname" name="fname"><br><br>  
  <input type="submit" value="Submit">  
</form>  
  
<label for="lname">Last name:</label>  
<input type="text" id="lname" name="lname" form="form1">

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_form)

The formaction Attribute

The input formaction attribute specifies the URL of the file that will process the input when the form is submitted.

**Note:** This attribute overrides the action attribute of the <form> element.

The formaction attribute works with the following input types: submit and image.

Example

An HTML form with two submit buttons, with different actions:

<form action="/action\_page.php">  
  <label for="fname">First name:</label>  
  <input type="text" id="fname" name="fname"><br><br>  
  <label for="lname">Last name:</label>  
  <input type="text" id="lname" name="lname"><br><br>  
  <input type="submit" value="Submit">  
  <input type="submit" formaction="/action\_page2.php" value="Submit as Admin">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_formaction)

The formenctype Attribute

The input formenctype attribute specifies how the form-data should be encoded when submitted (only for forms with method="post").

**Note:** This attribute overrides the enctype attribute of the <form> element.

The formenctype attribute works with the following input types: submit and image.

Example

A form with two submit buttons. The first sends the form-data with default encoding, the second sends the form-data encoded as "multipart/form-data":

<form action="/action\_page\_binary.asp" method="post">  
  <label for="fname">First name:</label>  
  <input type="text" id="fname" name="fname"><br><br>  
  <input type="submit" value="Submit">  
  <input type="submit" formenctype="multipart/form-data"  
  value="Submit as Multipart/form-data">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_formenctype)

The formmethod Attribute

The input formmethod attribute defines the HTTP method for sending form-data to the action URL.

**Note:** This attribute overrides the method attribute of the <form> element.

The formmethod attribute works with the following input types: submit and image.

The form-data can be sent as URL variables (method="get") or as an HTTP post transaction (method="post").

**Notes on the "get" method:**

* This method appends the form-data to the URL in name/value pairs
* This method is useful for form submissions where a user want to bookmark the result
* There is a limit to how much data you can place in a URL (varies between browsers), therefore, you cannot be sure that all of the form-data will be correctly transferred
* Never use the "get" method to pass sensitive information! (password or other sensitive information will be visible in the browser's address bar)

**Notes on the "post" method:**

* This method sends the form-data as an HTTP post transaction
* Form submissions with the "post" method cannot be bookmarked
* The "post" method is more robust and secure than "get", and "post" does not have size limitations

Example

A form with two submit buttons. The first sends the form-data with method="get". The second sends the form-data with method="post":

<form action="/action\_page.php" method="get">  
  <label for="fname">First name:</label>  
  <input type="text" id="fname" name="fname"><br><br>  
  <label for="lname">Last name:</label>  
  <input type="text" id="lname" name="lname"><br><br>  
  <input type="submit" value="Submit using GET">  
  <input type="submit" formmethod="post" value="Submit using POST">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_formmethod)

The formtarget Attribute

The input formtarget a attribute specifies a name or a keyword that indicates where to display the response that is received after submitting the form.

**Note:** This attribute overrides the target attribute of the <form> element.

The formtarget attribute works with the following input types: submit and image.

Example

A form with two submit buttons, with different target windows:

<form action="/action\_page.php">  
  <label for="fname">First name:</label>  
  <input type="text" id="fname" name="fname"><br><br>  
  <label for="lname">Last name:</label>  
  <input type="text" id="lname" name="lname"><br><br>  
  <input type="submit" value="Submit">  
  <input type="submit" formtarget="\_blank" value="Submit to a new window/tab">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_formtarget)

The formnovalidate Attribute

The input formnovalidate attribute specifies that an <input> element should not be validated when submitted.

**Note:** This attribute overrides the novalidate attribute of the <form> element.

The formnovalidate attribute works with the following input types: submit.

Example

A form with two submit buttons (with and without validation):

<form action="/action\_page.php">  
  <label for="email">Enter your email:</label>  
  <input type="email" id="email" name="email"><br><br>  
  <input type="submit" value="Submit">  
  <input type="submit" formnovalidate="formnovalidate"  
  value="Submit without validation">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml5_input_formnovalidate)

The novalidate Attribute

The novalidate attribute is a <form> attribute.

When present, novalidate specifies that all of the form-data should not be validated when submitted.

Example

Specify that no form-data should be validated on submit:

<form action="/action\_page.php" novalidate>  
  <label for="email">Enter your email:</label>  
  <input type="email" id="email" name="email"><br><br>  
  <input type="submit" value="Submit">  
</form>

CSS Margins

[❮ Previous](https://www.w3schools.com/css/css_border_rounded.asp)[Next ❯](https://www.w3schools.com/css/css_margin_collapse.asp)

This element has a margin of 70px.

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_margin_intro)

CSS Margins

The CSS margin properties are used to create space around elements, outside of any defined borders.

With CSS, you have full control over the margins. There are properties for setting the margin for each side of an element (top, right, bottom, and left).

Margin - Individual Sides

CSS has properties for specifying the margin for each side of an element:

* margin-top
* margin-right
* margin-bottom
* margin-left

All the margin properties can have the following values:

* auto - the browser calculates the margin
* *length* - specifies a margin in px, pt, cm, etc.
* *%* - specifies a margin in % of the width of the containing element
* inherit - specifies that the margin should be inherited from the parent element

**Tip:** Negative values are allowed.

Example

Set different margins for all four sides of a <p> element:

p {  
  margin-top: 100px;  
  margin-bottom: 100px;  
  margin-right: 150px;  
  margin-left: 80px;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_margin_sides)

Margin - Shorthand Property

To shorten the code, it is possible to specify all the margin properties in one property.

The margin property is a shorthand property for the following individual margin properties:

* margin-top
* margin-right
* margin-bottom
* margin-left

So, here is how it works:

If the margin property has four values:

* **margin: 25px 50px 75px 100px;**
  + top margin is 25px
  + right margin is 50px
  + bottom margin is 75px
  + left margin is 100px

Example

Use the margin shorthand property with four values:

p {  
  margin: 25px 50px 75px 100px;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_margin_shorthand_4val)

If the margin property has three values:

* **margin: 25px 50px 75px;**
  + top margin is 25px
  + right and left margins are 50px
  + bottom margin is 75px

Example

Use the margin shorthand property with three values:

p {  
  margin: 25px 50px 75px;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_margin_shorthand_3val)

If the margin property has two values:

* **margin: 25px 50px;**
  + top and bottom margins are 25px
  + right and left margins are 50px

Example

Use the margin shorthand property with two values:

p {  
  margin: 25px 50px;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_margin_shorthand_2val)

If the margin property has one value:

* **margin: 25px;**
  + all four margins are 25px

Example

Use the margin shorthand property with one value:

p {  
  margin: 25px;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_margin_shorthand_1val)

The auto Value

You can set the margin property to auto to horizontally center the element within its container.

The element will then take up the specified width, and the remaining space will be split equally between the left and right margins.

Example

Use margin: auto:

div {  
  width: 300px;  
  margin: auto;  
  border: 1px solid red;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_margin_auto)

The inherit Value

This example lets the left margin of the <p class="ex1"> element be inherited from the parent element (<div>):

Example

Use of the inherit value:

div {  
  border: 1px solid red;  
  margin-left: 100px;  
}  
  
p.ex1 {  
  margin-left: inherit;  
}

CSS Layout - float and clear

[❮ Previous](https://www.w3schools.com/css/css_overflow.asp)[Next ❯](https://www.w3schools.com/css/css_float_clear.asp)

The CSS float property specifies how an element should float.

The CSS clear property specifies what elements can float beside the cleared element and on which side.

The float Property

The float property is used for positioning and formatting content e.g. let an image float left to the text in a container.

The float property can have one of the following values:

* left - The element floats to the left of its container
* right - The element floats to the right of its container
* none - The element does not float (will be displayed just where it occurs in the text). This is default
* inherit - The element inherits the float value of its parent

In its simplest use, the float property can be used to wrap text around images.

Example - float: right;

The following example specifies that an image should float to the **right** in a text:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac...

Example

img {  
  float: right;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_layout_float)

Example - float: left;

The following example specifies that an image should float to the **left** in a text:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac...

Example

img {  
  float: left;  
}

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_layout_float2)

Example - No float

In the following example the image will be displayed just where it occurs in the text (float: none;):

 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac...

Example

img {  
  float: none;  
}

# CSS display Property

[❮ Previous](https://www.w3schools.com/cssref/pr_text_direction.asp)[Complete CSS Reference](https://www.w3schools.com/cssref/default.asp)[Next ❯](https://www.w3schools.com/cssref/pr_tab_empty-cells.asp)

### Example

Use of some different display values:

p.ex1 {display: none;}  
p.ex2 {display: inline;}  
p.ex3 {display: block;}  
p.ex4 {display: inline-block;}

[Try it Yourself »](https://www.w3schools.com/cssref/tryit.asp?filename=trycss_display)

More "Try it Yourself" examples below.

## Definition and Usage

The display property specifies the display behavior (the type of rendering box) of an element.

In HTML, the default display property value is taken from the HTML specifications or from the browser/user default style sheet. The default value in XML is inline, including SVG elements.

|  |  |
| --- | --- |
| **Default value:** | ? |
| **Inherited:** | no |
| **Animatable:** | no. [Read about animatable](https://www.w3schools.com/cssref/css_animatable.asp) |
| **Version:** | CSS1 |
| **JavaScript syntax:** | *object*.style.display="none"[Try it](https://www.w3schools.com/cssref/tryit.asp?filename=trycss_js_display) |

## Browser Support

The numbers in the table specify the first browser version that fully supports the property.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Property |  |  |  |  |  |
| display | 4.0 | 8.0 | 3.0 | 3.1 | 7.0 |

**Note:** The values "flex" and "inline-flex" requires the -webkit- prefix to work in Safari.

**Note:** "display: contents" does not work in Edge/Internet Explorer.

## CSS Syntax

display: *value*;

## Property Values

|  |  |  |
| --- | --- | --- |
| **Value** | **Description** | **Play it** |
| inline | Displays an element as an inline element (like <span>). Any height and width properties will have no effect | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=inline) |
| block | Displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=block) |
| contents | Makes the container disappear, making the child elements children of the element the next level up in the DOM | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=contents) |
| flex | Displays an element as a block-level flex container | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=flex) |
| grid | Displays an element as a block-level grid container | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=grid) |
| inline-block | Displays an element as an inline-level block container. The element itself is formatted as an inline element, but you can apply height and width values | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=inline-block) |
| inline-flex | Displays an element as an inline-level flex container | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=inline-flex) |
| inline-grid | Displays an element as an inline-level grid container | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=inline-grid) |
| inline-table | The element is displayed as an inline-level table | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=inline-table) |
| list-item | Let the element behave like a <li> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=list-item) |
| run-in | Displays an element as either block or inline, depending on context | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=run-in) |
| table | Let the element behave like a <table> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=table) |
| table-caption | Let the element behave like a <caption> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=table-caption) |
| table-column-group | Let the element behave like a <colgroup> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=table-column-group) |
| table-header-group | Let the element behave like a <thead> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=table-header-group) |
| table-footer-group | Let the element behave like a <tfoot> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=table-footer-group) |
| table-row-group | Let the element behave like a <tbody> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=table-row-group) |
| table-cell | Let the element behave like a <td> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=table-cell) |
| table-column | Let the element behave like a <col> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=table-column) |
| table-row | Let the element behave like a <tr> element | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=table-row) |
| none | The element is completely removed | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=none) |
| initial | Sets this property to its default value. [Read about initial](https://www.w3schools.com/cssref/css_initial.asp) | [Play it »](https://www.w3schools.com/cssref/playit.asp?filename=playcss_display&preval=initial) |
| inherit | Inherits this property from its parent element. [Read about inherit](https://www.w3schools.com/cssref/css_inherit.asp) |  |

## More Examples

### Example

A demonstration of how to use the contents property value. In the following example the .a container will disappear, and making the child elements (.b) children of the element the next level up in the DOM:

.a {  
  display: contents;  
  border: 2px solid red;  
  background-color: #ccc;  
  padding: 10px;  
  width: 200px;  
}  
  
.b {  
  border: 2px solid blue;  
  background-color: lightblue;  
  padding: 10px;  
}

[Try it Yourself »](https://www.w3schools.com/cssref/tryit.asp?filename=trycss_display_contents)

### Example

A demonstration of how to use the inherit property value:

body {  
  display: inline;  
}  
  
p {  
  display: inherit;  
}

[Try it Yourself »](https://www.w3schools.com/cssref/tryit.asp?filename=trycss_display_inline2)

### Example

Set the direction of some flexible items inside a <div> element in reverse order:

div {  
  display: flex;  
  flex-direction: row-reverse;  
}

# How TO - Toggle Hide and Show

[❮ Previous](https://www.w3schools.com/howto/howto_js_toggle_like.asp)[Next ❯](https://www.w3schools.com/howto/howto_js_toggle_dark_mode.asp)

Toggle between hiding and showing an element with JavaScript.

Toggle Hide and Show

## Toggle (Hide/Show) an Element

##### **Step 1) Add HTML:**

### Example

<button onclick="myFunction()">Click Me</button>  
  
<div id="myDIV">  
  This is my DIV element.  
</div>

##### **Step 2) Add JavaScript:**

### Example

function myFunction() {  
  var x = document.getElementById("myDIV");  
  if (x.style.display === "none") {  
    x.style.display = "block";  
  } else {  
    x.style.display = "none";  
  }  
}

JavaScript Forms

[❮ Previous](https://www.w3schools.com/js/js_json.asp)[Next ❯](https://www.w3schools.com/js/js_validation_api.asp)

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:

JavaScript Example

function validateForm() {  
  var x = document.forms["myForm"]["fname"].value;  
  if (x == "") {  
    alert("Name must be filled out");  
    return false;  
  }  
}

The function can be called when the form is submitted:

HTML Form Example

<form name="myForm" action="/action\_page.php" **onsubmit="return validateForm()"** method="post">  
Name: <input type="text" name="fname">  
<input type="submit" value="Submit">  
</form>

[Try it Yourself »](https://www.w3schools.com/js/tryit.asp?filename=tryjs_validation_js)

JavaScript Can Validate Numeric Input

JavaScript is often used to validate numeric input:

Please input a number between 1 and 10

 Submit

[Try it Yourself »](https://www.w3schools.com/js/tryit.asp?filename=tryjs_validation_number)

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>

[Try it Yourself »](https://www.w3schools.com/js/tryit.asp?filename=tryjs_validation_html)

Automatic HTML form validation does not work in Internet Explorer 9 or earlier.

Data Validation

Data validation is the process of ensuring that user input is clean, correct, and useful.

Typical validation tasks are:

* has the user filled in all required fields?
* has the user entered a valid date?
* has the user entered text in a numeric field?

Most often, the purpose of data validation is to ensure correct user input.

Validation can be defined by many different methods, and deployed in many different ways.

**Server side validation** is performed by a web server, after input has been sent to the server.

**Client side validation** is performed by a web browser, before input is sent to a web server.

HTML Constraint Validation

HTML5 introduced a new HTML validation concept called **constraint validation**.

HTML constraint validation is based on:

* Constraint validation **HTML** **Input Attributes**
* Constraint validation **CSS Pseudo Selectors**
* Constraint validation **DOM Properties and Methods**

Constraint Validation HTML Input Attributes

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| disabled | Specifies that the input element should be disabled |
| max | Specifies the maximum value of an input element |
| min | Specifies the minimum value of an input element |
| pattern | Specifies the value pattern of an input element |
| required | Specifies that the input field requires an element |
| type | Specifies the type of an input element |

For a full list, go to [HTML Input Attributes](https://www.w3schools.com/html/html_form_attributes.asp).

Constraint Validation CSS Pseudo Selectors

|  |  |
| --- | --- |
| **Selector** | **Description** |
| :disabled | Selects input elements with the "disabled" attribute specified |
| :invalid | Selects input elements with invalid values |
| :optional | Selects input elements with no "required" attribute specified |
| :required | Selects input elements with the "required" attribute specified |
| :valid | Selects input elements with valid values |