

HTML Element Reference

HTML Tags Ordered Alphabetically

 Search..

 = New in HTML5.

Tag	Description
<u><!--...--></u>	Defines a comment
<u><!DOCTYPE></u>	Defines the document type
<u><a></u>	Defines a hyperlink
<u><abbr></u>	Defines an abbreviation or an acronym
<u><acronym></u>	Not supported in HTML5. Use <u><abbr></u> instead. Defines an acronym
<u><address></u>	Defines contact information for the author/owner of a document
<u><applet></u>	Not supported in HTML5. Use <u><embed></u> or <u><object></u> instead. Defines an embedded applet
<u><area></u>	Defines an area inside an image-map
<u><article></u>	 Defines an article
<u><aside></u>	 Defines content aside from the page content
<u><audio></u>	 Defines sound content
<u></u>	Defines bold text
<u><base></u>	Specifies the base URL/target for all relative URLs in a document
<u><basefont></u>	Not supported in HTML5. Use CSS instead. Specifies a default color, size, and font for all text in a document
<u><bdi></u>	 Isolates a part of text that might be formatted in a different direction from other text outside it
<u><bdo></u>	Overrides the current text direction

Tag	Description
<u><big></u>	Not supported in HTML5. Use CSS instead. Defines big text
<u><blockquote></u>	Defines a section that is quoted from another source
<u><body></u>	Defines the document's body
<u>
</u>	Defines a single line break
<u><button></u>	Defines a clickable button
<u><canvas></u>	5 Used to draw graphics, on the fly, via scripting (usually JavaScript)
<u><caption></u>	Defines a table caption
<u><center></u>	Not supported in HTML5. Use CSS instead. Defines centered text
<u><cite></u>	Defines the title of a work
<u><code></u>	Defines a piece of computer code
<u><col></u>	Specifies column properties for each column within a <colgroup> element
<u><colgroup></u>	Specifies a group of one or more columns in a table for formatting
<u><data></u>	5 Links the given content with a machine-readable translation
<u><datalist></u>	5 Specifies a list of pre-defined options for input controls
<u><dd></u>	Defines a description/value of a term in a description list
<u></u>	Defines text that has been deleted from a document
<u><details></u>	5 Defines additional details that the user can view or hide
<u><dfn></u>	Represents the defining instance of a term
<u><dialog></u>	5 Defines a dialog box or window
<u><dir></u>	Not supported in HTML5. Use instead. Defines a directory list
<u><div></u>	Defines a section in a document
<u><dl></u>	Defines a description list
<u><dt></u>	Defines a term/name in a description list

Tag	Description
<u></u>	Defines emphasized text
<u><embed></u>	 Defines a container for an external (non-HTML) application
<u><fieldset></u>	Groups related elements in a form
<u><figcaption></u>	 Defines a caption for a <figure> element
<u><figure></u>	 Specifies self-contained content
<u></u>	Not supported in HTML5. Use CSS instead. Defines font, color, and size for text
<u><footer></u>	 Defines a footer for a document or section
<u><form></u>	Defines an HTML form for user input
<u><frame></u>	Not supported in HTML5. Defines a window (a frame) in a frameset
<u><frameset></u>	Not supported in HTML5. Defines a set of frames
<u><h1> to <h6></u>	Defines HTML headings
<u><head></u>	Defines information about the document
<u><header></u>	 Defines a header for a document or section
<u><hr></u>	Defines a thematic change in the content
<u><html></u>	Defines the root of an HTML document
<u><i></u>	Defines a part of text in an alternate voice or mood
<u><iframe></u>	Defines an inline frame
<u></u>	Defines an image
<u><input></u>	Defines an input control
<u><ins></u>	Defines a text that has been inserted into a document
<u><kbd></u>	Defines keyboard input
<u><label></u>	Defines a label for an <input> element
<u><legend></u>	Defines a caption for a <fieldset> element
<u></u>	Defines a list item

Tag	Description
<u><link></u>	Defines the relationship between a document and an external resource (most used to link to style sheets)
<u><main></u>	 Specifies the main content of a document
<u><map></u>	Defines a client-side image-map
<u><mark></u>	 Defines marked/highlighted text
<u><meta></u>	Defines metadata about an HTML document
<u><meter></u>	 Defines a scalar measurement within a known range (a gauge)
<u><nav></u>	 Defines navigation links
<u><noframes></u>	Not supported in HTML5. Defines an alternate content for users that do not support frames
<u><noscript></u>	Defines an alternate content for users that do not support client-side scripts
<u><object></u>	Defines an embedded object
<u></u>	Defines an ordered list
<u><optgroup></u>	Defines a group of related options in a drop-down list
<u><option></u>	Defines an option in a drop-down list
<u><output></u>	 Defines the result of a calculation
<u><p></u>	Defines a paragraph
<u><param></u>	Defines a parameter for an object
<u><picture></u>	 Defines a container for multiple image resources
<u><pre></u>	Defines preformatted text
<u><progress></u>	 Represents the progress of a task
<u><q></u>	Defines a short quotation
<u><rp></u>	 Defines what to show in browsers that do not support ruby annotations
<u><rt></u>	 Defines an explanation/pronunciation of characters (for East Asian typography)
<u><ruby></u>	 Defines a ruby annotation (for East Asian typography)

Tag	Description
<u><s></u>	Defines text that is no longer correct
<u><samp></u>	Defines sample output from a computer program
<u><script></u>	Defines a client-side script
<u><section></u>	 Defines a section in a document
<u><select></u>	Defines a drop-down list
<u><small></u>	Defines smaller text
<u><source></u>	 Defines multiple media resources for media elements (<video> and <audio>)
<u></u>	Defines a section in a document
<u><strike></u>	Not supported in HTML5. Use <u></u> or <u><s></u> instead. Defines strikethrough text
<u></u>	Defines important text
<u><style></u>	Defines style information for a document
<u><sub></u>	Defines subscripted text
<u><summary></u>	 Defines a visible heading for a <details> element
<u><sup></u>	Defines superscripted text
<u><svg></u>	Defines a container for SVG graphics
<u><table></u>	Defines a table
<u><tbody></u>	Groups the body content in a table
<u><td></u>	Defines a cell in a table
<u><template></u>	 Defines a template
<u><textarea></u>	Defines a multiline input control (text area)
<u><tfoot></u>	Groups the footer content in a table
<u><th></u>	Defines a header cell in a table
<u><thead></u>	Groups the header content in a table
<u><time></u>	 Defines a date/time

Tag	Description
<u><title></u>	Defines a title for the document
<u><tr></u>	Defines a row in a table
<u><track></u>	 Defines text tracks for media elements (<video> and <audio>)
<u><tt></u>	Not supported in HTML5. Use CSS instead. Defines teletype text
<u><u></u>	Defines text that should be stylistically different from normal text
<u></u>	Defines an unordered list
<u><var></u>	Defines a variable
<u><video></u>	 Defines a video or movie
<u><wbr></u>	 Defines a possible line-break

HTML Element Reference

HTML Tags Ordered by Category

 = New in HTML5.

Basic HTML

Tag	Description
<u><!DOCTYPE></u>	Defines the document type
<u><html></u>	Defines an HTML document
<u><head></u>	Defines information about the document
<u><title></u>	Defines a title for the document
<u><body></u>	Defines the document's body
<u><h1> to <h6></u>	Defines HTML headings
<u><p></u>	Defines a paragraph
<u>
</u>	Inserts a single line break
<u><hr></u>	Defines a thematic change in the content
<u><!--...--></u>	Defines a comment

Formatting

Tag	Description
<u><acronym></u>	Not supported in HTML5. Use <u><abbr></u> instead. Defines an acronym
<u><abbr></u>	Defines an abbreviation or an acronym
<u><address></u>	Defines contact information for the author/owner of a document/article
<u></u>	Defines bold text
<u><bdi></u>	 Isolates a part of text that might be formatted in a different direction from other text outside it
<u><bdo></u>	Overrides the current text direction

Tag	Description
<u><big></u>	Not supported in HTML5. Use CSS instead. Defines big text
<u><blockquote></u>	Defines a section that is quoted from another source
<u><center></u>	Not supported in HTML5. Use CSS instead. Defines centered text
<u><cite></u>	Defines the title of a work
<u><code></u>	Defines a piece of computer code
<u></u>	Defines text that has been deleted from a document
<u><dfn></u>	Represents the defining instance of a term
<u></u>	Defines emphasized text
<u></u>	Not supported in HTML5. Use CSS instead. Defines font, color, and size for text
<u><i></u>	Defines a part of text in an alternate voice or mood
<u><ins></u>	Defines a text that has been inserted into a document
<u><kbd></u>	Defines keyboard input
<u><mark></u>	 Defines marked/highlighted text
<u><meter></u>	 Defines a scalar measurement within a known range (a gauge)
<u><pre></u>	Defines preformatted text
<u><progress></u>	 Represents the progress of a task
<u><q></u>	Defines a short quotation
<u><rp></u>	 Defines what to show in browsers that do not support ruby annotations
<u><rt></u>	 Defines an explanation/pronunciation of characters (for East Asian typography)
<u><ruby></u>	 Defines a ruby annotation (for East Asian typography)
<u><s></u>	Defines text that is no longer correct
<u><samp></u>	Defines sample output from a computer program
<u><small></u>	Defines smaller text
<u><strike></u>	Not supported in HTML5. Use or <s> instead. Defines strikethrough text

Tag	Description
<u></u>	Defines important text
<u><sub></u>	Defines subscripted text
<u><sup></u>	Defines superscripted text
<u><template></u> 	Defines a template
<u><time></u> 	Defines a date/time
<u><tt></u>	Not supported in HTML5. Use CSS instead. Defines teletype text
<u><u></u>	Defines text that should be stylistically different from normal text
<u><var></u>	Defines a variable
<u><wbr></u> 	Defines a possible line-break

Forms and Input

Tag	Description
<u><form></u>	Defines an HTML form for user input
<u><input></u>	Defines an input control
<u><textarea></u>	Defines a multiline input control (text area)
<u><button></u>	Defines a clickable button
<u><select></u>	Defines a drop-down list
<u><optgroup></u>	Defines a group of related options in a drop-down list
<u><option></u>	Defines an option in a drop-down list
<u><label></u>	Defines a label for an <input> element
<u><fieldset></u>	Groups related elements in a form
<u><legend></u>	Defines a caption for a <fieldset> element
<u><datalist></u> 	Specifies a list of pre-defined options for input controls
<u><output></u> 	Defines the result of a calculation

Frames

Tag	Description
<u><frame></u>	Not supported in HTML5. Defines a window (a frame) in a frameset
<u><frameset></u>	Not supported in HTML5. Defines a set of frames
<u><noframes></u>	Not supported in HTML5. Defines an alternate content for users that do not support frames
<u><iframe></u>	Defines an inline frame

Images

Tag	Description
<u></u>	Defines an image
<u><map></u>	Defines a client-side image-map
<u><area></u>	Defines an area inside an image-map
<u><canvas></u>	Used to draw graphics, on the fly, via scripting (usually JavaScript)
<u><figcaption></u>	Defines a caption for a <figure> element
<u><figure></u>	Specifies self-contained content
<u><picture></u>	Defines a container for multiple image resources
<u><svg></u>	Defines a container for SVG graphics

Audio / Video

Tag	Description
<u><audio></u>	Defines sound content
<u><source></u>	Defines multiple media resources for media elements (<video>, <audio> and <picture>)
<u><track></u>	Defines text tracks for media elements (<video> and <audio>)
<u><video></u>	Defines a video or movie

Links

Tag	Description
<u><a></u>	Defines a hyperlink
<u><link></u>	Defines the relationship between a document and an external resource (most used to link to style sheets)
<u><nav></u>	 Defines navigation links

Lists

Tag	Description
<u></u>	Defines an unordered list
<u></u>	Defines an ordered list
<u></u>	Defines a list item
<u><dir></u>	Not supported in HTML5. Use instead. Defines a directory list
<u><dl></u>	Defines a description list
<u><dt></u>	Defines a term/name in a description list
<u><dd></u>	Defines a description of a term/name in a description list

Tables

Tag	Description
<u><table></u>	Defines a table
<u><caption></u>	Defines a table caption
<u><th></u>	Defines a header cell in a table
<u><tr></u>	Defines a row in a table
<u><td></u>	Defines a cell in a table
<u><thead></u>	Groups the header content in a table
<u><tbody></u>	Groups the body content in a table
<u><tfoot></u>	Groups the footer content in a table
<u><col></u>	Specifies column properties for each column within a <colgroup> element
<u><colgroup></u>	Specifies a group of one or more columns in a table for formatting

Styles and Semantics

Tag	Description
<u><style></u>	Defines style information for a document
<u><div></u>	Defines a section in a document
<u></u>	Defines a section in a document
<u><header></u>	Defines a header for a document or section
<u><footer></u>	Defines a footer for a document or section
<u><main></u>	Specifies the main content of a document
<u><section></u>	Defines a section in a document
<u><article></u>	Defines an article
<u><aside></u>	Defines content aside from the page content
<u><details></u>	Defines additional details that the user can view or hide
<u><dialog></u>	Defines a dialog box or window
<u><summary></u>	Defines a visible heading for a <details> element
<u><data></u>	Links the given content with a machine-readable translation

Meta Info

Tag	Description
<u><head></u>	Defines information about the document
<u><meta></u>	Defines metadata about an HTML document
<u><base></u>	Specifies the base URL/target for all relative URLs in a document
<u><basefont></u>	Not supported in HTML5. Use CSS instead. Specifies a default color, size, and font for all text in a document

Programming

Tag	Description
<u><script></u>	Defines a client-side script
<u><noscript></u>	Defines an alternate content for users that do not support client-side scripts
<u><applet></u>	Not supported in HTML5. Use <embed> or <object> instead. Defines an embedded applet
<u><embed></u> 	Defines a container for an external (non-HTML) application
<u><object></u>	Defines an embedded object
<u><param></u>	Defines a parameter for an object

HTML Attribute Reference

All HTML Attributes

Attribute	Belongs to	Description
<u>accept</u>	<u><input></u>	Specifies the types of files that the server accepts (only for type="file")
<u>accept-charset</u>	<u><form></u>	Specifies the character encodings that are to be used for the form submission
<u>accesskey</u>	<u>Global Attributes</u>	Specifies a shortcut key to activate/focus an element
<u>action</u>	<u><form></u>	Specifies where to send the form-data when a form is submitted
<u>align</u>	<u>Not supported in HTML 5.</u>	Specifies the alignment according to surrounding elements. Use CSS instead
<u>alt</u>	<u><area>, , <input></u>	Specifies an alternate text when the original element fails to display
<u>async</u>	<u><script></u>	Specifies that the script is executed asynchronously (only for external scripts)
<u>autocomplete</u>	<u><form>, <input></u>	Specifies whether the <form> or the <input> element should have autocomplete enabled
<u>autofocus</u>	<u><button>, <input>, <select>, <textarea></u>	Specifies that the element should automatically get focus when the page loads
<u>autoplay</u>	<u><audio>, <video></u>	Specifies that the audio/video will start playing as soon as it is ready

Attribute	Belongs to	Description
<u>bgcolor</u>	Not supported in HTML 5.	Specifies the background color of an element. Use CSS instead
<u>border</u>	Not supported in HTML 5.	Specifies the width of the border of an element. Use CSS instead
<u>charset</u>	<u><meta></u> , <u><script></u>	Specifies the character encoding
<u>checked</u>	<u><input></u>	Specifies that an <input> element should be pre-selected when the page loads (for type="checkbox" or type="radio")
<u>cite</u>	<u><blockquote></u> , <u></u> , <u><ins></u> , <u><q></u>	Specifies a URL which explains the quote/deleted/inserted text
<u>class</u>	<u>Global Attributes</u>	Specifies one or more classnames for an element (refers to a class in a style sheet)
<u>color</u>	Not supported in HTML 5.	Specifies the text color of an element. Use CSS instead
<u>cols</u>	<u><textarea></u>	Specifies the visible width of a text area
<u>colspan</u>	<u><td></u> , <u><th></u>	Specifies the number of columns a table cell should span
<u>content</u>	<u><meta></u>	Gives the value associated with the http-equiv or name attribute
<u>contenteditable</u>	<u>Global Attributes</u>	Specifies whether the content of an element is editable or not
<u>controls</u>	<u><audio></u> , <u><video></u>	Specifies that audio/video controls should be displayed (such as a play/pause button etc)
<u>coords</u>	<u><area></u>	Specifies the coordinates of the area
<u>data</u>	<u><object></u>	Specifies the URL of the resource to be used by the object

Attribute	Belongs to	Description
<u>data-*</u>	<u>Global Attributes</u>	Used to store custom data private to the page or application
<u>datetime</u>	<u>, <ins>, <time></u>	Specifies the date and time
<u>default</u>	<u><track></u>	Specifies that the track is to be enabled if the user's preferences do not indicate that another track would be more appropriate
<u>defer</u>	<u><script></u>	Specifies that the script is executed when the page has finished parsing (only for external scripts)
<u>dir</u>	<u>Global Attributes</u>	Specifies the text direction for the content in an element
<u>dirname</u>	<u><input>, <textarea></u>	Specifies that the text direction will be submitted
<u>disabled</u>	<u><button>, <fieldset>, <input>, <optgroup>, <option>, <select>, <textarea></u>	Specifies that the specified element/group of elements should be disabled
<u>download</u>	<u><a>, <area></u>	Specifies that the target will be downloaded when a user clicks on the hyperlink
<u>draggable</u>	<u>Global Attributes</u>	Specifies whether an element is draggable or not
<u>dropzone</u>	<u>Global Attributes</u>	Specifies whether the dragged data is copied, moved, or linked, when dropped
<u>enctype</u>	<u><form></u>	Specifies how the form-data should be encoded when submitting it to the server (only for method="post")
<u>for</u>	<u><label>, <output></u>	Specifies which form element(s) a label/calculation is bound to
<u>form</u>	<u><button>, <fieldset>, <input>, <label>, <meter>, <object>, <output>, <select>, <textarea></u>	Specifies the name of the form the element belongs to

Attribute	Belongs to	Description
<u>formaction</u>	<u><button></u> , <u><input></u>	Specifies where to send the form-data when a form is submitted. Only for type="submit"
<u>headers</u>	<u><td></u> , <u><th></u>	Specifies one or more headers cells a cell is related to
<u>height</u>	<u><canvas></u> , <u><embed></u> , <u><iframe></u> , <u></u> , <u><input></u> , <u><object></u> , <u><video></u>	Specifies the height of the element
<u>hidden</u>	<u>Global Attributes</u>	Specifies that an element is not yet, or is no longer, relevant
<u>high</u>	<u><meter></u>	Specifies the range that is considered to be a high value
<u>href</u>	<u><a></u> , <u><area></u> , <u><base></u> , <u><link></u>	Specifies the URL of the page the link goes to
<u>hreflang</u>	<u><a></u> , <u><area></u> , <u><link></u>	Specifies the language of the linked document
<u>http-equiv</u>	<u><meta></u>	Provides an HTTP header for the information/value of the content attribute
<u>id</u>	<u>Global Attributes</u>	Specifies a unique id for an element
<u>ismap</u>	<u></u>	Specifies an image as a server-side image-map
<u>kind</u>	<u><track></u>	Specifies the kind of text track
<u>label</u>	<u><track></u> , <u><option></u> , <u><optgroup></u>	Specifies the title of the text track
<u>lang</u>	<u>Global Attributes</u>	Specifies the language of the element's content
<u>list</u>	<u><input></u>	Refers to a <datalist> element that contains pre-defined options for an <input> element

Attribute	Belongs to	Description
<u>loop</u>	<u><audio></u> , <u><video></u>	Specifies that the audio/video will start over again, every time it is finished
<u>low</u>	<u><meter></u>	Specifies the range that is considered to be a low value
<u>max</u>	<u><input></u> , <u><meter></u> , <u><progress></u>	Specifies the maximum value
<u>maxlength</u>	<u><input></u> , <u><textarea></u>	Specifies the maximum number of characters allowed in an element
<u>media</u>	<u><a></u> , <u><area></u> , <u><link></u> , <u><source></u> , <u><style></u>	Specifies what media/device the linked document is optimized for
<u>method</u>	<u><form></u>	Specifies the HTTP method to use when sending form-data
<u>min</u>	<u><input></u> , <u><meter></u>	Specifies a minimum value
<u>multiple</u>	<u><input></u> , <u><select></u>	Specifies that a user can enter more than one value
<u>muted</u>	<u><video></u> , <u><audio></u>	Specifies that the audio output of the video should be muted
<u>name</u>	<u><button></u> , <u><fieldset></u> , <u><form></u> , <u><iframe></u> , <u><input></u> , <u><map></u> , <u><meta></u> , <u><object></u> , <u><output></u> , <u><param></u> , <u><select></u> , <u><textarea></u>	Specifies the name of the element
<u>novalidate</u>	<u><form></u>	Specifies that the form should not be validated when submitted
<u>onabort</u>	<u><audio></u> , <u><embed></u> , <u></u> , <u><object></u> , <u><video></u>	Script to be run on abort
<u>onafterprint</u>	<u><body></u>	Script to be run after the document is printed
<u>onbeforeprint</u>	<u><body></u>	Script to be run before the document is printed
<u>onbeforeunload</u>	<u><body></u>	Script to be run when the document is about to be unloaded

Attribute	Belongs to	Description
<u>onblur</u>	All visible elements.	Script to be run when the element loses focus
<u>oncanplay</u>	<u><audio></u> , <u><embed></u> , <u><object></u> , <u><video></u>	Script to be run when a file is ready to start playing (when it has buffered enough to begin)
<u>oncanplaythrough</u>	<u><audio></u> , <u><video></u>	Script to be run when a file can be played all the way to the end without pausing for buffering
<u>onchange</u>	All visible elements.	Script to be run when the value of the element is changed
<u>onclick</u>	All visible elements.	Script to be run when the element is being clicked
<u>oncontextmenu</u>	All visible elements.	Script to be run when a context menu is triggered
<u>oncopy</u>	All visible elements.	Script to be run when the content of the element is being copied
<u>oncuechange</u>	<u><track></u>	Script to be run when the cue changes in a <u><track></u> element
<u>oncut</u>	All visible elements.	Script to be run when the content of the element is being cut
<u>ondblclick</u>	All visible elements.	Script to be run when the element is being double-clicked
<u>ondrag</u>	All visible elements.	Script to be run when the element is being dragged
<u>ondragend</u>	All visible elements.	Script to be run at the end of a drag operation
<u>ondragenter</u>	All visible elements.	Script to be run when an element has been dragged to a valid drop target
<u>ondragleave</u>	All visible elements.	Script to be run when an element leaves a valid drop target

Attribute	Belongs to	Description
<u>ondragover</u>	All visible elements.	Script to be run when an element is being dragged over a valid drop target
<u>ondragstart</u>	All visible elements.	Script to be run at the start of a drag operation
<u>ondrop</u>	All visible elements.	Script to be run when dragged element is being dropped
<u>ondurationchange</u>	<u><audio></u> , <u><video></u>	Script to be run when the length of the media changes
<u>onemptied</u>	<u><audio></u> , <u><video></u>	Script to be run when something bad happens and the file is suddenly unavailable (like unexpectedly disconnects)
<u>onended</u>	<u><audio></u> , <u><video></u>	Script to be run when the media has reached the end (a useful event for messages like "thanks for listening")
<u>onerror</u>	<u><audio></u> , <u><body></u> , <u><embed></u> , <u></u> , <u><object></u> , <u><script></u> , <u><style></u> , <u><video></u>	Script to be run when an error occurs
<u>onfocus</u>	All visible elements.	Script to be run when the element gets focus
<u>onhashchange</u>	<u><body></u>	Script to be run when there has been changes to the anchor part of the a URL
<u>oninput</u>	All visible elements.	Script to be run when the element gets user input
<u>oninvalid</u>	All visible elements.	Script to be run when the element is invalid
<u>onkeydown</u>	All visible elements.	Script to be run when a user is pressing a key
<u>onkeypress</u>	All visible elements.	Script to be run when a user presses a key
<u>onkeyup</u>	All visible elements.	Script to be run when a user releases a key

Attribute	Belongs to	Description
<u>onload</u>	<u><body></u> , <u><iframe></u> , <u></u> , <u><input></u> , <u><link></u> , <u><script></u> , <u><style></u>	Script to be run when the element is finished loading
<u>onloadeddata</u>	<u><audio></u> , <u><video></u>	Script to be run when media data is loaded
<u>onloadedmetadata</u>	<u><audio></u> , <u><video></u>	Script to be run when meta data (like dimensions and duration) are loaded
<u>onloadstart</u>	<u><audio></u> , <u><video></u>	Script to be run just as the file begins to load before anything is actually loaded
<u>onmousedown</u>	All visible elements.	Script to be run when a mouse button is pressed down on an element
<u>onmousemove</u>	All visible elements.	Script to be run as long as the mouse pointer is moving over an element
<u>onmouseout</u>	All visible elements.	Script to be run when a mouse pointer moves out of an element
<u>onmouseover</u>	All visible elements.	Script to be run when a mouse pointer moves over an element
<u>onmouseup</u>	All visible elements.	Script to be run when a mouse button is released over an element
<u>onmousewheel</u>	All visible elements.	Script to be run when a mouse wheel is being scrolled over an element
<u>onoffline</u>	<u><body></u>	Script to be run when the browser starts to work offline
<u>ononline</u>	<u><body></u>	Script to be run when the browser starts to work online
<u>onpagehide</u>	<u><body></u>	Script to be run when a user navigates away from a page
<u>onpageshow</u>	<u><body></u>	Script to be run when a user navigates to a page

Attribute	Belongs to	Description
<u>onpaste</u>	All visible elements.	Script to be run when the user pastes some content in an element
<u>onpause</u>	<u><audio></u> , <u><video></u>	Script to be run when the media is paused either by the user or programmatically
<u>onplay</u>	<u><audio></u> , <u><video></u>	Script to be run when the media has started playing
<u>onplaying</u>	<u><audio></u> , <u><video></u>	Script to be run when the media has started playing
<u>onpopstate</u>	<u><body></u>	Script to be run when the window's history changes.
<u>onprogress</u>	<u><audio></u> , <u><video></u>	Script to be run when the browser is in the process of getting the media data
<u>onratechange</u>	<u><audio></u> , <u><video></u>	Script to be run each time the playback rate changes (like when a user switches to a slow motion or fast forward mode).
<u>onreset</u>	<u><form></u>	Script to be run when a reset button in a form is clicked.
<u>onresize</u>	<u><body></u>	Script to be run when the browser window is being resized.
<u>onscroll</u>	All visible elements.	Script to be run when an element's scrollbar is being scrolled
<u>onsearch</u>	<u><input></u>	Script to be run when the user writes something in a search field (for <u><input="search"></u>)
<u>onseeked</u>	<u><audio></u> , <u><video></u>	Script to be run when the seeking attribute is set to false indicating that seeking has ended

Attribute	Belongs to	Description
<u>onseeking</u>	<u><audio></u> , <u><video></u>	Script to be run when the seeking attribute is set to true indicating that seeking is active
<u>onselect</u>	All visible elements.	Script to be run when the element gets selected
<u>onstalled</u>	<u><audio></u> , <u><video></u>	Script to be run when the browser is unable to fetch the media data for whatever reason
<u>onstorage</u>	<u><body></u>	Script to be run when a Web Storage area is updated
<u>onsubmit</u>	<u><form></u>	Script to be run when a form is submitted
<u>onsuspend</u>	<u><audio></u> , <u><video></u>	Script to be run when fetching the media data is stopped before it is completely loaded for whatever reason
<u>ontimeupdate</u>	<u><audio></u> , <u><video></u>	Script to be run when the playing position has changed (like when the user fast forwards to a different point in the media)
<u>ontoggle</u>	<u><details></u>	Script to be run when the user opens or closes the <u><details></u> element
<u>onunload</u>	<u><body></u>	Script to be run when a page has unloaded (or the browser window has been closed)
<u>onvolumechange</u>	<u><audio></u> , <u><video></u>	Script to be run each time the volume of a video/audio has been changed
<u>onwaiting</u>	<u><audio></u> , <u><video></u>	Script to be run when the media has paused but is expected to resume (like when the media pauses to buffer more data)
<u>onwheel</u>	All visible elements.	Script to be run when the mouse wheel rolls up or down over an element

Attribute	Belongs to	Description
<u>open</u>	<u><details></u>	Specifies that the details should be visible (open) to the user
<u>optimum</u>	<u><meter></u>	Specifies what value is the optimal value for the gauge
<u>pattern</u>	<u><input></u>	Specifies a regular expression that an <input> element's value is checked against
<u>placeholder</u>	<u><input>, <textarea></u>	Specifies a short hint that describes the expected value of the element
<u>poster</u>	<u><video></u>	Specifies an image to be shown while the video is downloading, or until the user hits the play button
<u>preload</u>	<u><audio>, <video></u>	Specifies if and how the author thinks the audio/video should be loaded when the page loads
<u>readonly</u>	<u><input>, <textarea></u>	Specifies that the element is read-only
<u>rel</u>	<u><a>, <area>, <link></u>	Specifies the relationship between the current document and the linked document
<u>required</u>	<u><input>, <select>, <textarea></u>	Specifies that the element must be filled out before submitting the form
<u>reversed</u>	<u></u>	Specifies that the list order should be descending (9,8,7...)
<u>rows</u>	<u><textarea></u>	Specifies the visible number of lines in a text area
<u>rowspan</u>	<u><td>, <th></u>	Specifies the number of rows a table cell should span
<u>sandbox</u>	<u><iframe></u>	Enables an extra set of restrictions for the content in an <iframe>

Attribute	Belongs to	Description
<u>scope</u>	<u><th></u>	Specifies whether a header cell is a header for a column, row, or group of columns or rows
<u>selected</u>	<u><option></u>	Specifies that an option should be pre-selected when the page loads
<u>shape</u>	<u><area></u>	Specifies the shape of the area
<u>size</u>	<u><input>, <select></u>	Specifies the width, in characters (for <input>) or specifies the number of visible options (for <select>)
<u>sizes</u>	<u>, <link>, <source></u>	Specifies the size of the linked resource
<u>span</u>	<u><col>, <colgroup></u>	Specifies the number of columns to span
<u>spellcheck</u>	<u>Global Attributes</u>	Specifies whether the element is to have its spelling and grammar checked or not
<u>src</u>	<u><audio>, <embed>, <iframe>, , <input>, <script>, <source>, <track>, <video></u>	Specifies the URL of the media file
<u>srcdoc</u>	<u><iframe></u>	Specifies the HTML content of the page to show in the <iframe>
<u>srclang</u>	<u><track></u>	Specifies the language of the track text data (required if kind="subtitles")
<u>srcset</u>	<u>, <source></u>	Specifies the URL of the image to use in different situations
<u>start</u>	<u></u>	Specifies the start value of an ordered list
<u>step</u>	<u><input></u>	Specifies the legal number intervals for an input field
<u>style</u>	<u>Global Attributes</u>	Specifies an inline CSS style for an element

Attribute	Belongs to	Description
<u>tabindex</u>	<u>Global Attributes</u>	Specifies the tabbing order of an element
<u>target</u>	<u><a>, <area>, <base>, <form></u>	Specifies the target for where to open the linked document or where to submit the form
<u>title</u>	<u>Global Attributes</u>	Specifies extra information about an element
<u>translate</u>	<u>Global Attributes</u>	Specifies whether the content of an element should be translated or not
<u>type</u>	<u><button>, <embed>, <input>, <link>, <menu>, <object>, <script>, <source>, <style></u>	Specifies the type of element
<u>usemap</u>	<u>, <object></u>	Specifies an image as a client-side image-map
<u>value</u>	<u><button>, <input>, , <option>, <meter>, <progress>, <param></u>	Specifies the value of the element
<u>width</u>	<u><canvas>, <embed>, <iframe>, , <input>, <object>, <video></u>	Specifies the width of the element
<u>wrap</u>	<u><textarea></u>	Specifies how the text in a text area is to be wrapped when submitted in a form

HTML Global Attributes

HTML attributes give elements meaning and context.

The global attributes below can be used on **any** HTML element.

HTML Global Attributes

 = Attribute added in HTML5.

Attribute	Description
<u>accesskey</u>	Specifies a shortcut key to activate/focus an element
<u>class</u>	Specifies one or more classnames for an element (refers to a class in a style sheet)
<u>contenteditable</u>	 Specifies whether the content of an element is editable or not
<u>data-*</u>	 Used to store custom data private to the page or application
<u>dir</u>	Specifies the text direction for the content in an element
<u>draggable</u>	 Specifies whether an element is draggable or not
<u>dropzone</u>	 Specifies whether the dragged data is copied, moved, or linked, when dropped
<u>hidden</u>	 Specifies that an element is not yet, or is no longer, relevant
<u>id</u>	Specifies a unique id for an element
<u>lang</u>	Specifies the language of the element's content
<u>spellcheck</u>	 Specifies whether the element is to have its spelling and grammar checked or not
<u>style</u>	Specifies an inline CSS style for an element
<u>tabindex</u>	Specifies the tabbing order of an element
<u>title</u>	Specifies extra information about an element
<u>translate</u>	 Specifies whether the content of an element should be translated or not

HTML Event Attributes

Global Event Attributes

HTML 4 added the ability to let events trigger actions in a browser, like starting a JavaScript when a user clicks on an element.

To learn more about programming events, please visit our [JavaScript tutorial](#).

Below are the global event attributes that can be added to HTML elements to define event actions.

 = New event attributes in HTML5.

Window Event Attributes

Events triggered for the window object (applies to the <body> tag):

Attribute	Value	Description
<u>onafterprint</u>	 script	Script to be run after the document is printed
<u>onbeforeprint</u>	 script	Script to be run before the document is printed
<u>onbeforeunload</u>	 script	Script to be run when the document is about to be unloaded
<u>onerror</u>	 script	Script to be run when an error occurs
<u>onhashchange</u>	 script	Script to be run when there has been changes to the anchor part of the a URL
<u>onload</u>	script	Fires after the page is finished loading
<u>onmessage</u>	 script	Script to be run when the message is triggered
<u>onoffline</u>	 script	Script to be run when the browser starts to work offline
<u>ononline</u>	 script	Script to be run when the browser starts to work online
<u>onpagehide</u>	 script	Script to be run when a user navigates away from a page
<u>onpageshow</u>	 script	Script to be run when a user navigates to a page
<u>onpopstate</u>	 script	Script to be run when the window's history changes
<u>onresize</u>	 script	Fires when the browser window is resized
<u>onstorage</u>	 script	Script to be run when a Web Storage area is updated
<u>onunload</u>	script	Fires once a page has unloaded (or the browser window has been closed)

Form Events

Events triggered by actions inside a HTML form (applies to almost all HTML elements, but is most used in form elements):

Attribute	Value	Description
<u>onblur</u>	<i>script</i>	Fires the moment that the element loses focus
<u>onchange</u>	<i>script</i>	Fires the moment when the value of the element is changed
<u>oncontextmenu</u> 	<i>script</i>	Script to be run when a context menu is triggered
<u>onfocus</u>	<i>script</i>	Fires the moment when the element gets focus
<u>oninput</u>	 <i>script</i>	Script to be run when an element gets user input
<u>oninvalid</u>	 <i>script</i>	Script to be run when an element is invalid
<u>onreset</u>	 <i>script</i>	Fires when the Reset button in a form is clicked
<u>onsearch</u>	<i>script</i>	Fires when the user writes something in a search field (for <input="search">)
<u>onselect</u>	<i>script</i>	Fires after some text has been selected in an element
<u>onsubmit</u>	<i>script</i>	Fires when a form is submitted

Keyboard Events

Attribute	Value	Description
<u>onkeydown</u>	<i>script</i>	Fires when a user is pressing a key
<u>onkeypress</u>	<i>script</i>	Fires when a user presses a key
<u>onkeyup</u>	<i>script</i>	Fires when a user releases a key

Mouse Events

Attribute	Value	Description
<u>onclick</u>	script	Fires on a mouse click on the element
<u>ondblclick</u>	script	Fires on a mouse double-click on the element
<u>onmousedown</u>	script	Fires when a mouse button is pressed down on an element
<u>onmousemove</u>	script	Fires when the mouse pointer is moving while it is over an element
<u>onmouseout</u>	script	Fires when the mouse pointer moves out of an element
<u>onmouseover</u>	script	Fires when the mouse pointer moves over an element
<u>onmouseup</u>	script	Fires when a mouse button is released over an element
onmousewheel	script	Deprecated. Use the <u>onwheel</u> attribute instead
<u>onwheel</u>	script	Fires when the mouse wheel rolls up or down over an element

Drag Events

Attribute	Value	Description
<u>ondrag</u>	script	Script to be run when an element is dragged
<u>ondragend</u>	script	Script to be run at the end of a drag operation
<u>ondragenter</u>	script	Script to be run when an element has been dragged to a valid drop target
<u>ondragleave</u>	script	Script to be run when an element leaves a valid drop target
<u>ondragover</u>	script	Script to be run when an element is being dragged over a valid drop target
<u>ondragstart</u>	script	Script to be run at the start of a drag operation
<u>ondrop</u>	script	Script to be run when dragged element is being dropped
<u>onscroll</u>	script	Script to be run when an element's scrollbar is being scrolled

Clipboard Events

Attribute	Value	Description
<u>oncopy</u>	<i>script</i>	Fires when the user copies the content of an element
<u>oncut</u>	<i>script</i>	Fires when the user cuts the content of an element
<u>onpaste</u>	<i>script</i>	Fires when the user pastes some content in an element

Media Events

Events triggered by medias like videos, images and audio (applies to all HTML elements, but is most common in media elements, like `<audio>`, `<embed>`, ``, `<object>`, and `<video>`).

Tip: Look at our [HTML Audio and Video DOM Reference](#) for more information.

Attribute	Value	Description
onabort	<i>script</i>	Script to be run on abort
oncanplay	⌚ <i>script</i>	Script to be run when a file is ready to start playing (when it has buffered enough to begin)
oncanplaythrough	⌚ <i>script</i>	Script to be run when a file can be played all the way to the end without pausing for buffering
oncuechange	⌚ <i>script</i>	Script to be run when the cue changes in a <code><track></code> element
ondurationchange	⌚ <i>script</i>	Script to be run when the length of the media changes
onemptied	⌚ <i>script</i>	Script to be run when something bad happens and the file is suddenly unavailable (like unexpectedly disconnects)
onended	⌚ <i>script</i>	Script to be run when the media has reach the end (a useful event for messages like "thanks for listening")
onerror	⌚ <i>script</i>	Script to be run when an error occurs when the file is being loaded
onloadeddata	⌚ <i>script</i>	Script to be run when media data is loaded
onloadedmetadata	⌚ <i>script</i>	Script to be run when meta data (like dimensions and duration) are loaded

Attribute	Value	Description
onloadstart	⌚ script	Script to be run just as the file begins to load before anything is actually loaded
onpause	⌚ script	Script to be run when the media is paused either by the user or programmatically
onplay	⌚ script	Script to be run when the media is ready to start playing
onplaying	⌚ script	Script to be run when the media actually has started playing
onprogress	⌚ script	Script to be run when the browser is in the process of getting the media data
onratechange	⌚ script	Script to be run each time the playback rate changes (like when a user switches to a slow motion or fast forward mode)
onseeked	⌚ script	Script to be run when the seeking attribute is set to false indicating that seeking has ended
onseeking	⌚ script	Script to be run when the seeking attribute is set to true indicating that seeking is active
onstalled	⌚ script	Script to be run when the browser is unable to fetch the media data for whatever reason
onsuspend	⌚ script	Script to be run when fetching the media data is stopped before it is completely loaded for whatever reason
ontimeupdate	⌚ script	Script to be run when the playing position has changed (like when the user fast forwards to a different point in the media)
onvolumechange	⌚ script	Script to be run each time the volume is changed which (includes setting the volume to "mute")
onwaiting	⌚ script	Script to be run when the media has paused but is expected to resume (like when the media pauses to buffer more data)

Misc Events

Attribute	Value	Description
<u>ontoggle</u>	⌚ script	Fires when the user opens or closes the <details> element

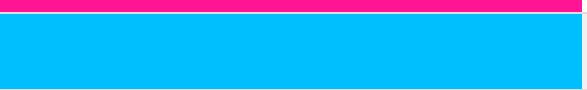
HTML Color Names

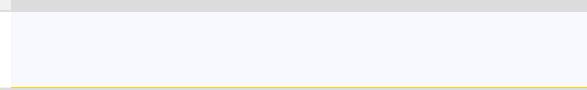
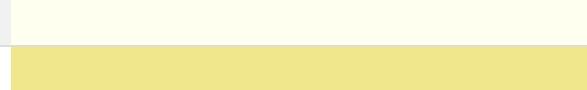
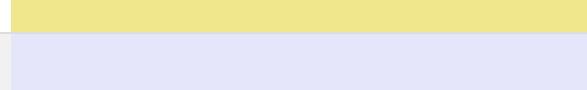
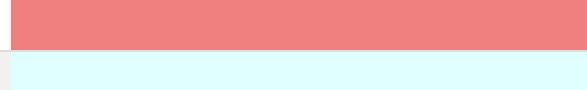
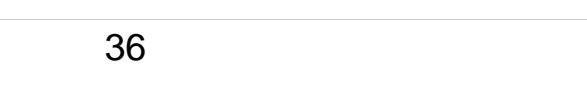
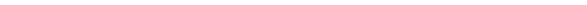
Color Names Supported by All Browsers

All modern browsers support the following 140 color names (click on a color name, or a hex value, to view the color as the background-color along with different text colors):

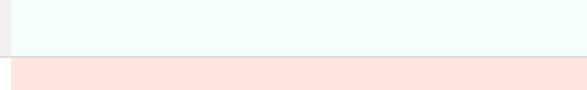
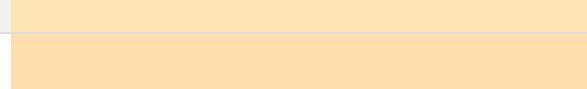
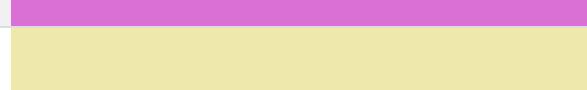
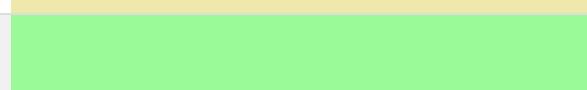
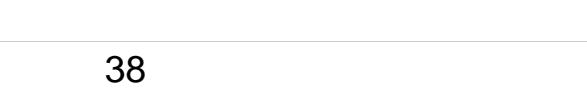
[For a full overview of HTML colors, visit our colors tutorial.](#)

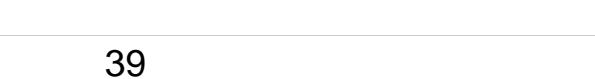
Color Name	HEX	Color	Shades	Mix
AliceBlue	#F0F8FF		Shades	Mix
AntiqueWhite	#FAEBD7		Shades	Mix
Aqua	#00FFFF		Shades	Mix
Aquamarine	#7FFFAD		Shades	Mix
Azure	#F0FFFF		Shades	Mix
Beige	#F5F5DC		Shades	Mix
Bisque	#FFE4C4		Shades	Mix
Black	#000000		Shades	Mix
BlanchedAlmond	#FFEBCD		Shades	Mix
Blue	#0000FF		Shades	Mix
BlueViolet	#8A2BE2		Shades	Mix
Brown	#A52A2A		Shades	Mix
BurlyWood	#DEB887		Shades	Mix
CadetBlue	#5F9EA0		Shades	Mix
Chartreuse	#7FFF00		Shades	Mix
Chocolate	#D2691E		Shades	Mix
Coral	#FF7F50		Shades	Mix
CornflowerBlue	#6495ED		Shades	Mix
Cornsilk	#FFF8DC		Shades	Mix
Crimson	#DC143C		Shades	Mix

Color Name	HEX	Color	Shades	Mix
Cyan	#00FFFF		Shades	Mix
DarkBlue	#00008B		Shades	Mix
DarkCyan	#008B8B		Shades	Mix
DarkGoldenRod	#B8860B		Shades	Mix
DarkGray	#A9A9A9		Shades	Mix
DarkGrey	#A9A9A9		Shades	Mix
DarkGreen	#006400		Shades	Mix
DarkKhaki	#bdb76b		Shades	Mix
DarkMagenta	#8B008B		Shades	Mix
DarkOliveGreen	#556B2F		Shades	Mix
DarkOrange	#FF8C00		Shades	Mix
DarkOrchid	#9932CC		Shades	Mix
DarkRed	#8B0000		Shades	Mix
DarkSalmon	#E9967A		Shades	Mix
DarkSeaGreen	#8FBC8F		Shades	Mix
DarkSlateBlue	#483D8B		Shades	Mix
DarkSlateGray	#2F4F4F		Shades	Mix
DarkSlateGrey	#2F4F4F		Shades	Mix
DarkTurquoise	#00CED1		Shades	Mix
DarkViolet	#9400D3		Shades	Mix
DeepPink	#FF1493		Shades	Mix
DeepSkyBlue	#00BFFF		Shades	Mix
DimGray	#696969		Shades	Mix
DimGrey	#696969		Shades	Mix
DodgerBlue	#1E90FF		Shades	Mix

Color Name	HEX	Color	Shades	Mix
FireBrick	#B22222		Shades	Mix
FloralWhite	#FFFAF0		Shades	Mix
ForestGreen	#228B22		Shades	Mix
Fuchsia	#FF00FF		Shades	Mix
Gainsboro	#DCDCDC		Shades	Mix
GhostWhite	#F8F8FF		Shades	Mix
Gold	#FFD700		Shades	Mix
GoldenRod	#DAA520		Shades	Mix
Gray	#808080		Shades	Mix
Grey	#808080		Shades	Mix
Green	#008000		Shades	Mix
GreenYellow	#ADFF2F		Shades	Mix
HoneyDew	#F0FFF0		Shades	Mix
HotPink	#FF69B4		Shades	Mix
IndianRed	#CD5C5C		Shades	Mix
Indigo	#4B0082		Shades	Mix
Ivory	#FFFFFF		Shades	Mix
Khaki	#F0E68C		Shades	Mix
Lavender	#E6E6FA		Shades	Mix
LavenderBlush	#FFF0F5		Shades	Mix
LawnGreen	#7CFC00		Shades	Mix
LemonChiffon	#FFFACD		Shades	Mix
LightBlue	#ADD8E6		Shades	Mix
LightCoral	#F08080		Shades	Mix
LightCyan	#E0FFFF		Shades	Mix

Color Name	HEX	Color	Shades	Mix
<u>LightGoldenRodYellow</u>	<u>#FAFAD2</u>		<u>Shades</u>	<u>Mix</u>
<u>LightGray</u>	<u>#D3D3D3</u>		<u>Shades</u>	<u>Mix</u>
<u>LightGrey</u>	<u>#D3D3D3</u>		<u>Shades</u>	<u>Mix</u>
<u>LightGreen</u>	<u>#90EE90</u>		<u>Shades</u>	<u>Mix</u>
<u>LightPink</u>	<u>#FFB6C1</u>		<u>Shades</u>	<u>Mix</u>
<u>LightSalmon</u>	<u>#FFA07A</u>		<u>Shades</u>	<u>Mix</u>
<u>LightSeaGreen</u>	<u>#20B2AA</u>		<u>Shades</u>	<u>Mix</u>
<u>LightSkyBlue</u>	<u>#87CEFA</u>		<u>Shades</u>	<u>Mix</u>
<u>LightSlateGray</u>	<u>#778899</u>		<u>Shades</u>	<u>Mix</u>
<u>LightSlateGrey</u>	<u>#778899</u>		<u>Shades</u>	<u>Mix</u>
<u>LightSteelBlue</u>	<u>#B0C4DE</u>		<u>Shades</u>	<u>Mix</u>
<u>LightYellow</u>	<u>#FFFFE0</u>		<u>Shades</u>	<u>Mix</u>
<u>Lime</u>	<u>#00FF00</u>		<u>Shades</u>	<u>Mix</u>
<u>LimeGreen</u>	<u>#32CD32</u>		<u>Shades</u>	<u>Mix</u>
<u>Linen</u>	<u>#FAF0E6</u>		<u>Shades</u>	<u>Mix</u>
<u>Magenta</u>	<u>#FF00FF</u>		<u>Shades</u>	<u>Mix</u>
<u>Maroon</u>	<u>#800000</u>		<u>Shades</u>	<u>Mix</u>
<u>MediumAquaMarine</u>	<u>#66CDAA</u>		<u>Shades</u>	<u>Mix</u>
<u>MediumBlue</u>	<u>#0000CD</u>		<u>Shades</u>	<u>Mix</u>
<u>MediumOrchid</u>	<u>#BA55D3</u>		<u>Shades</u>	<u>Mix</u>
<u>MediumPurple</u>	<u>#9370DB</u>		<u>Shades</u>	<u>Mix</u>
<u>MediumSeaGreen</u>	<u>#3CB371</u>		<u>Shades</u>	<u>Mix</u>
<u>MediumSlateBlue</u>	<u>#7B68EE</u>		<u>Shades</u>	<u>Mix</u>
<u>MediumSpringGreen</u>	<u>#00FA9A</u>		<u>Shades</u>	<u>Mix</u>
<u>MediumTurquoise</u>	<u>#48D1CC</u>		<u>Shades</u>	<u>Mix</u>

Color Name	HEX	Color	Shades	Mix
<u>MediumVioletRed</u>	<u>#C71585</u>		<u>Shades</u>	<u>Mix</u>
<u>MidnightBlue</u>	<u>#191970</u>		<u>Shades</u>	<u>Mix</u>
<u>MintCream</u>	<u>#F5FFFA</u>		<u>Shades</u>	<u>Mix</u>
<u>MistyRose</u>	<u>#FFE4E1</u>		<u>Shades</u>	<u>Mix</u>
<u>Moccasin</u>	<u>#FFE4B5</u>		<u>Shades</u>	<u>Mix</u>
<u>NavajoWhite</u>	<u>#FFDEAD</u>		<u>Shades</u>	<u>Mix</u>
<u>Navy</u>	<u>#000080</u>		<u>Shades</u>	<u>Mix</u>
<u>OldLace</u>	<u>#FDF5E6</u>		<u>Shades</u>	<u>Mix</u>
<u>Olive</u>	<u>#808000</u>		<u>Shades</u>	<u>Mix</u>
<u>OliveDrab</u>	<u>#6B8E23</u>		<u>Shades</u>	<u>Mix</u>
<u>Orange</u>	<u>#FFA500</u>		<u>Shades</u>	<u>Mix</u>
<u>OrangeRed</u>	<u>#FF4500</u>		<u>Shades</u>	<u>Mix</u>
<u>Orchid</u>	<u>#DA70D6</u>		<u>Shades</u>	<u>Mix</u>
<u>PaleGoldenRod</u>	<u>#EEE8AA</u>		<u>Shades</u>	<u>Mix</u>
<u>PaleGreen</u>	<u>#98FB98</u>		<u>Shades</u>	<u>Mix</u>
<u>PaleTurquoise</u>	<u>#AFEEEE</u>		<u>Shades</u>	<u>Mix</u>
<u>PaleVioletRed</u>	<u>#DB7093</u>		<u>Shades</u>	<u>Mix</u>
<u>PapayaWhip</u>	<u>#FFEFD5</u>		<u>Shades</u>	<u>Mix</u>
<u>PeachPuff</u>	<u>#FFDAB9</u>		<u>Shades</u>	<u>Mix</u>
<u>Peru</u>	<u>#CD853F</u>		<u>Shades</u>	<u>Mix</u>
<u>Pink</u>	<u>#FFC0CB</u>		<u>Shades</u>	<u>Mix</u>
<u>Plum</u>	<u>#DDA0DD</u>		<u>Shades</u>	<u>Mix</u>
<u>PowderBlue</u>	<u>#B0E0E6</u>		<u>Shades</u>	<u>Mix</u>
<u>Purple</u>	<u>#800080</u>		<u>Shades</u>	<u>Mix</u>
<u>RebeccaPurple</u>	<u>#663399</u>		<u>Shades</u>	<u>Mix</u>

Color Name	HEX	Color	Shades	Mix
<u>Red</u>	<u>#FF0000</u>		<u>Shades</u>	<u>Mix</u>
<u>RosyBrown</u>	<u>#BC8F8F</u>		<u>Shades</u>	<u>Mix</u>
<u>RoyalBlue</u>	<u>#4169E1</u>		<u>Shades</u>	<u>Mix</u>
<u>SaddleBrown</u>	<u>#8B4513</u>		<u>Shades</u>	<u>Mix</u>
<u>Salmon</u>	<u>#FA8072</u>		<u>Shades</u>	<u>Mix</u>
<u>SandyBrown</u>	<u>#F4A460</u>		<u>Shades</u>	<u>Mix</u>
<u>SeaGreen</u>	<u>#2E8B57</u>		<u>Shades</u>	<u>Mix</u>
<u>SeaShell</u>	<u>#FFF5EE</u>		<u>Shades</u>	<u>Mix</u>
<u>Sienna</u>	<u>#A0522D</u>		<u>Shades</u>	<u>Mix</u>
<u>Silver</u>	<u>#C0C0C0</u>		<u>Shades</u>	<u>Mix</u>
<u>SkyBlue</u>	<u>#87CEEB</u>		<u>Shades</u>	<u>Mix</u>
<u>SlateBlue</u>	<u>#6A5ACD</u>		<u>Shades</u>	<u>Mix</u>
<u>SlateGray</u>	<u>#708090</u>		<u>Shades</u>	<u>Mix</u>
<u>SlateGrey</u>	<u>#708090</u>		<u>Shades</u>	<u>Mix</u>
<u>Snow</u>	<u>#FFFFFA</u>		<u>Shades</u>	<u>Mix</u>
<u>SpringGreen</u>	<u>#00FF7F</u>		<u>Shades</u>	<u>Mix</u>
<u>SteelBlue</u>	<u>#4682B4</u>		<u>Shades</u>	<u>Mix</u>
<u>Tan</u>	<u>#D2B48C</u>		<u>Shades</u>	<u>Mix</u>
<u>Teal</u>	<u>#008080</u>		<u>Shades</u>	<u>Mix</u>
<u>Thistle</u>	<u>#D8bfd8</u>		<u>Shades</u>	<u>Mix</u>
<u>Tomato</u>	<u>#FF6347</u>		<u>Shades</u>	<u>Mix</u>
<u>Turquoise</u>	<u>#40E0D0</u>		<u>Shades</u>	<u>Mix</u>
<u>Violet</u>	<u>#EE82EE</u>		<u>Shades</u>	<u>Mix</u>
<u>Wheat</u>	<u>#F5DEB3</u>		<u>Shades</u>	<u>Mix</u>
<u>White</u>	<u>#FFFFFF</u>		<u>Shades</u>	<u>Mix</u>

Color Name	HEX	Color	Shades	Mix
<u>WhiteSmoke</u>	<u>#F5F5F5</u>		<u>Shades</u>	<u>Mix</u>
<u>Yellow</u>	<u>#FFFF00</u>		<u>Shades</u>	<u>Mix</u>
<u>YellowGreen</u>	<u>#9ACD32</u>		<u>Shades</u>	<u>Mix</u>

HTML Canvas Reference

Description

The HTML5 <canvas> tag is used to draw graphics, on the fly, via scripting (usually JavaScript).

However, the <canvas> element has no drawing abilities of its own (it is only a container for graphics) - you must use a script to actually draw the graphics.

The getContext() method returns an object that provides methods and properties for drawing on the canvas.

This reference will cover the properties and methods of the getContext("2d") object, which can be used to draw text, lines, boxes, circles, and more - on the canvas.

Browser Support

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

Element					
<canvas>	4.0	9.0	2.0	3.1	9.0

Internet Explorer 9, Firefox, Opera, Chrome, and Safari support <canvas> and its properties and methods.

Note: Internet Explorer 8 and earlier versions, do not support the <canvas> element.

Colors, Styles, and Shadows

Property	Description
<u>fillStyle</u>	Sets or returns the color, gradient, or pattern used to fill the drawing
<u>strokeStyle</u>	Sets or returns the color, gradient, or pattern used for strokes
<u>shadowColor</u>	Sets or returns the color to use for shadows
<u>shadowBlur</u>	Sets or returns the blur level for shadows
<u>shadowOffsetX</u>	Sets or returns the horizontal distance of the shadow from the shape
<u>shadowOffsetY</u>	Sets or returns the vertical distance of the shadow from the shape

Method	Description
<u>createLinearGradient()</u>	Creates a linear gradient (to use on canvas content)
<u>createPattern()</u>	Repeats a specified element in the specified direction
<u>createRadialGradient()</u>	Creates a radial/circular gradient (to use on canvas content)
<u>addColorStop()</u>	Specifies the colors and stop positions in a gradient object

Line Styles

Property	Description
<u>lineCap</u>	Sets or returns the style of the end caps for a line
<u>lineJoin</u>	Sets or returns the type of corner created, when two lines meet
<u>lineWidth</u>	Sets or returns the current line width
<u>miterLimit</u>	Sets or returns the maximum miter length

Rectangles

Method	Description
<u>rect()</u>	Creates a rectangle
<u>fillRect()</u>	Draws a "filled" rectangle
<u>strokeRect()</u>	Draws a rectangle (no fill)
<u>clearRect()</u>	Clears the specified pixels within a given rectangle

Paths

Method	Description
<u>fill()</u>	Fills the current drawing (path)
<u>stroke()</u>	Actually draws the path you have defined
<u>beginPath()</u>	Begins a path, or resets the current path
<u>moveTo()</u>	Moves the path to the specified point in the canvas, without creating a line
<u>closePath()</u>	Creates a path from the current point back to the starting point
<u>lineTo()</u>	Adds a new point and creates a line to that point from the last specified point in the canvas
<u>clip()</u>	Clips a region of any shape and size from the original canvas
<u>quadraticCurveTo()</u>	Creates a quadratic Bézier curve
<u>bezierCurveTo()</u>	Creates a cubic Bézier curve
<u>arc()</u>	Creates an arc/curve (used to create circles, or parts of circles)
<u>arcTo()</u>	Creates an arc/curve between two tangents
<u>isPointInPath()</u>	Returns true if the specified point is in the current path, otherwise false

Transformations

Method	Description
<u>scale()</u>	Scales the current drawing bigger or smaller
<u>rotate()</u>	Rotates the current drawing
<u>translate()</u>	Remaps the (0,0) position on the canvas
<u>transform()</u>	Replaces the current transformation matrix for the drawing
<u>setTransform()</u>	Resets the current transform to the identity matrix. Then runs <u>transform()</u> .

Text

Property	Description
<u>font</u>	Sets or returns the current font properties for text content
<u>textAlign</u>	Sets or returns the current alignment for text content
<u>textBaseline</u>	Sets or returns the current text baseline used when drawing text

Method	Description
<u>fillText()</u>	Draws "filled" text on the canvas
<u>strokeText()</u>	Draws text on the canvas (no fill)
<u>measureText()</u>	Returns an object that contains the width of the specified text

Image Drawing

Method	Description
<u>drawImage()</u>	Draws an image, canvas, or video onto the canvas

Pixel Manipulation

Property	Description
<u>width</u>	Returns the width of an ImageData object
<u>height</u>	Returns the height of an ImageData object
<u>data</u>	Returns an object that contains image data of a specified ImageData object

Method	Description
<u>createImageData()</u>	Creates a new, blank ImageData object
<u>getImageData()</u>	Returns an ImageData object that copies the pixel data for the specified rectangle on a canvas
<u>putImageData()</u>	Puts the image data (from a specified ImageData object) back onto the canvas

Compositing

Property	Description
<u>globalAlpha</u>	Sets or returns the current alpha or transparency value of the drawing
<u>globalCompositeOperation</u>	Sets or returns how a new image are drawn onto an existing image

Other

Method	Description
save()	Saves the state of the current context
restore()	Returns previously saved path state and attributes
createEvent()	
getContext()	
toDataURL()	

HTML Audio/Video DOM Reference

HTML Audio and Video DOM Reference

The HTML5 DOM has methods, properties, and events for the `<audio>` and `<video>` elements. These methods, properties, and events allow you to manipulate `<audio>` and `<video>` elements using JavaScript.

HTML Audio/Video Methods

Method	Description
<code>addTextTrack()</code>	Adds a new text track to the audio/video
<code>canPlayType()</code>	Checks if the browser can play the specified audio/video type
<code>load()</code>	Re-loads the audio/video element
<code>play()</code>	Starts playing the audio/video
<code>pause()</code>	Pauses the currently playing audio/video

HTML Audio/Video Properties

Property	Description
<code>audioTracks</code>	Returns an <code>AudioTrackList</code> object representing available audio tracks
<code>autoplay</code>	Sets or returns whether the audio/video should start playing as soon as it is loaded
<code>buffered</code>	Returns a <code>TimeRanges</code> object representing the buffered parts of the audio/video
<code>controller</code>	Returns the <code>MediaController</code> object representing the current media controller of the audio/video
<code>controls</code>	Sets or returns whether the audio/video should display controls (like play/pause etc.)
<code>crossOrigin</code>	Sets or returns the CORS settings of the audio/video
<code>currentSrc</code>	Returns the URL of the current audio/video

Property	Description
<u>currentTime</u>	Sets or returns the current playback position in the audio/video (in seconds)
<u>defaultMuted</u>	Sets or returns whether the audio/video should be muted by default
<u>defaultPlaybackRate</u>	Sets or returns the default speed of the audio/video playback
<u>duration</u>	Returns the length of the current audio/video (in seconds)
<u>ended</u>	Returns whether the playback of the audio/video has ended or not
<u>error</u>	Returns a MediaError object representing the error state of the audio/video
<u>loop</u>	Sets or returns whether the audio/video should start over again when finished
<u>mediaGroup</u>	Sets or returns the group the audio/video belongs to (used to link multiple audio/video elements)
<u>muted</u>	Sets or returns whether the audio/video is muted or not
<u>networkState</u>	Returns the current network state of the audio/video
<u>paused</u>	Returns whether the audio/video is paused or not
<u>playbackRate</u>	Sets or returns the speed of the audio/video playback
<u>played</u>	Returns a TimeRanges object representing the played parts of the audio/video
<u>preload</u>	Sets or returns whether the audio/video should be loaded when the page loads
<u>readyState</u>	Returns the current ready state of the audio/video
<u>seekable</u>	Returns a TimeRanges object representing the seekable parts of the audio/video
<u>seeking</u>	Returns whether the user is currently seeking in the audio/video
<u>src</u>	Sets or returns the current source of the audio/video element
<u>startDate</u>	Returns a Date object representing the current time offset
<u>textTracks</u>	Returns a TextTrackList object representing the available text tracks
<u>videoTracks</u>	Returns a VideoTrackList object representing the available video tracks

Property	Description
<u>volume</u>	Sets or returns the volume of the audio/video

HTML Audio/Video Events

Event	Description
<u>abort</u>	Fires when the loading of an audio/video is aborted
<u>canplay</u>	Fires when the browser can start playing the audio/video
<u>canplaythrough</u>	Fires when the browser can play through the audio/video without stopping for buffering
<u>durationchange</u>	Fires when the duration of the audio/video is changed
<u>emptied</u>	Fires when the current playlist is empty
<u>ended</u>	Fires when the current playlist is ended
<u>error</u>	Fires when an error occurred during the loading of an audio/video
<u>loadeddata</u>	Fires when the browser has loaded the current frame of the audio/video
<u>loadedmetadata</u>	Fires when the browser has loaded meta data for the audio/video
<u>loadstart</u>	Fires when the browser starts looking for the audio/video
<u>pause</u>	Fires when the audio/video has been paused
<u>play</u>	Fires when the audio/video has been started or is no longer paused
<u>playing</u>	Fires when the audio/video is playing after having been paused or stopped for buffering
<u>progress</u>	Fires when the browser is downloading the audio/video
<u>ratechange</u>	Fires when the playing speed of the audio/video is changed
<u>seeked</u>	Fires when the user is finished moving/skipping to a new position in the audio/video
<u>seeking</u>	Fires when the user starts moving/skipping to a new position in the audio/video
<u>stalled</u>	Fires when the browser is trying to get media data, but data is not available
<u>suspend</u>	Fires when the browser is intentionally not getting media data
<u>timeupdate</u>	Fires when the current playback position has changed

Event	Description
<u>volumechange</u>	Fires when the volume has been changed
<u>waiting</u>	Fires when the video stops because it needs to buffer the next frame

HTML Character Sets

Common HTML Character Sets

ANSI was the first official default character set in Windows.

The default character set in HTML 4 was 8859-1.

The default character set in HTML 5 is UTF-8.

For a closer look, visit our [Complete HTML Character Set Reference](#).

Number	ASCII	ANSI	8859-1	UTF-8	Description
32					space
33	!	!	!	!	exclamation mark
34	"	"	"	"	quotation mark
35	#	#	#	#	number sign
36	\$	\$	\$	\$	dollar sign
37	%	%	%	%	percent sign
38	&	&	&	&	ampersand
39	'	'	'	'	apostrophe
40	((((left parenthesis
41))))	right parenthesis
42	*	*	*	*	asterisk
43	+	+	+	+	plus sign
44	,	,	,	,	comma
45	-	-	-	-	hyphen-minus
46	full stop
47	/	/	/	/	solidus
48	0	0	0	0	digit zero
49	1	1	1	1	digit one
50	2	2	2	2	digit two

Number	ASCII	ANSI	8859-1	UTF-8	Description
51	3	3	3	3	digit three
52	4	4	4	4	digit four
53	5	5	5	5	digit five
54	6	6	6	6	digit six
55	7	7	7	7	digit seven
56	8	8	8	8	digit eight
57	9	9	9	9	digit nine
58	:	:	:	:	colon
59	;	;	;	;	semicolon
60	<	<	<	<	less-than sign
61	=	=	=	=	equals sign
62	>	>	>	>	greater-than sign
63	?	?	?	?	question mark
64	@	@	@	@	commercial at
65	A	A	A	A	Latin capital letter A
66	B	B	B	B	Latin capital letter B
67	C	C	C	C	Latin capital letter C
68	D	D	D	D	Latin capital letter D
69	E	E	E	E	Latin capital letter E
70	F	F	F	F	Latin capital letter F
71	G	G	G	G	Latin capital letter G
72	H	H	H	H	Latin capital letter H
73	I	I	I	I	Latin capital letter I
74	J	J	J	J	Latin capital letter J
75	K	K	K	K	Latin capital letter K

Number	ASCII	ANSI	8859-1	UTF-8	Description
76	L	L	L	L	Latin capital letter L
77	M	M	M	M	Latin capital letter M
78	N	N	N	N	Latin capital letter N
79	O	O	O	O	Latin capital letter O
80	P	P	P	P	Latin capital letter P
81	Q	Q	Q	Q	Latin capital letter Q
82	R	R	R	R	Latin capital letter R
83	S	S	S	S	Latin capital letter S
84	T	T	T	T	Latin capital letter T
85	U	U	U	U	Latin capital letter U
86	V	V	V	V	Latin capital letter V
87	W	W	W	W	Latin capital letter W
88	X	X	X	X	Latin capital letter X
89	Y	Y	Y	Y	Latin capital letter Y
90	Z	Z	Z	Z	Latin capital letter Z
91	[[[[left square bracket
92	\	\	\	\	reverse solidus
93]]]]	right square bracket
94	^	^	^	^	circumflex accent
95	-	-	-	-	low line
96	`	`	`	`	grave accent
97	a	a	a	a	Latin small letter a
98	b	b	b	b	Latin small letter b
99	c	c	c	c	Latin small letter c
100	d	d	d	d	Latin small letter d

Number	ASCII	ANSI	8859-1	UTF-8	Description
101	e	e	e	e	Latin small letter e
102	f	f	f	f	Latin small letter f
103	g	g	g	g	Latin small letter g
104	h	h	h	h	Latin small letter h
105	i	i	i	i	Latin small letter i
106	j	j	j	j	Latin small letter j
107	k	k	k	k	Latin small letter k
108	l	l	l	l	Latin small letter l
109	m	m	m	m	Latin small letter m
110	n	n	n	n	Latin small letter n
111	o	o	o	o	Latin small letter o
112	p	p	p	p	Latin small letter p
113	q	q	q	q	Latin small letter q
114	r	r	r	r	Latin small letter r
115	s	s	s	s	Latin small letter s
116	t	t	t	t	Latin small letter t
117	u	u	u	u	Latin small letter u
118	v	v	v	v	Latin small letter v
119	w	w	w	w	Latin small letter w
120	x	x	x	x	Latin small letter x
121	y	y	y	y	Latin small letter y
122	z	z	z	z	Latin small letter z
123	{	{	{	{	left curly bracket
124					vertical line
125	}	}	}	}	right curly bracket

Number	ASCII	ANSI	8859-1	UTF-8	Description
126	~	~	~	~	tilde
127	DEL				
128	€				euro sign
129	□	□	□		NOT USED
130	,				single low-9 quotation mark
131	f				Latin small letter f with hook
132	„				double low-9 quotation mark
133	…				horizontal ellipsis
134	†				dagger
135	‡				double dagger
136	^				modifier letter circumflex accent
137	%o				per mille sign
138	Š				Latin capital letter S with caron
139	<				single left-pointing angle quotation mark
140	Œ				Latin capital ligature OE
141	□	□	□		NOT USED
142	Ž				Latin capital letter Z with caron
143	□	□	□		NOT USED
144	□	□	□		NOT USED
145	‘				left single quotation mark
146	’				right single quotation mark
147	“				left double quotation mark
148	”				right double quotation mark
149	•				bullet
150	-				en dash

Number	ASCII	ANSI	8859-1	UTF-8	Description
151	—				em dash
152	~				small tilde
153	™				trade mark sign
154	š				Latin small letter s with caron
155	>				single right-pointing angle quotation mark
156	œ				Latin small ligature oe
157	□ □ □				NOT USED
158	ž				Latin small letter z with caron
159	Ÿ				Latin capital letter Y with diaeresis
160					no-break space
161	¡ ¡ ¡				inverted exclamation mark
162	¢ ¢ ¢				cent sign
163	£ £ £				pound sign
164	¤ ¤ ¤				currency sign
165	¥ ¥ ¥				yen sign
166	¦ ¦ ¦				broken bar
167	§ § §				section sign
168				diaeresis
169	© © ©				copyright sign
170	ª ª ª				feminine ordinal indicator
171	« « «				left-pointing double angle quotation mark
172	¬ ¬ ¬				not sign
173					soft hyphen
174	® ® ®				registered sign
175	— — —				macron

Number	ASCII	ANSI	8859-1	UTF-8	Description
176	°	°	°	°	degree sign
177	±	±	±	±	plus-minus sign
178	²	²	²	²	superscript two
179	³	³	³	³	superscript three
180	'	'	'	'	acute accent
181	µ	µ	µ	µ	micro sign
182	¶	¶	¶	¶	pilcrow sign
183	·	·	·	·	middle dot
184	¸	¸	¸	¸	cedilla
185	¹	¹	¹	¹	superscript one
186	º	º	º	º	masculine ordinal indicator
187	»	»	»	»	right-pointing double angle quotation mark
188	¼	¼	¼	¼	vulgar fraction one quarter
189	½	½	½	½	vulgar fraction one half
190	¾	¾	¾	¾	vulgar fraction three quarters
191	¿	¿	¿	¿	inverted question mark
192	À	À	À	À	Latin capital letter A with grave
193	Á	Á	Á	Á	Latin capital letter A with acute
194	Â	Â	Â	Â	Latin capital letter A with circumflex
195	Ã	Ã	Ã	Ã	Latin capital letter A with tilde
196	Ä	Ä	Ä	Ä	Latin capital letter A with diaeresis
197	Å	Å	Å	Å	Latin capital letter A with ring above
198	Æ	Æ	Æ	Æ	Latin capital letter AE
199	Ç	Ç	Ç	Ç	Latin capital letter C with cedilla
200	È	È	È	È	Latin capital letter E with grave

Number	ASCII	ANSI	8859-1	UTF-8	Description
201	É	É	É	É	Latin capital letter E with acute
202	Ê	Ê	Ê	Ê	Latin capital letter E with circumflex
203	Ë	Ë	Ë	Ë	Latin capital letter E with diaeresis
204	Ì	Ì	Ì	Ì	Latin capital letter I with grave
205	Í	Í	Í	Í	Latin capital letter I with acute
206	Î	Î	Î	Î	Latin capital letter I with circumflex
207	Ï	Ï	Ï	Ï	Latin capital letter I with diaeresis
208	Đ	Đ	Đ	Đ	Latin capital letter Eth
209	Ñ	Ñ	Ñ	Ñ	Latin capital letter N with tilde
210	Ò	Ò	Ò	Ò	Latin capital letter O with grave
211	Ó	Ó	Ó	Ó	Latin capital letter O with acute
212	Ô	Ô	Ô	Ô	Latin capital letter O with circumflex
213	Õ	Õ	Õ	Õ	Latin capital letter O with tilde
214	Ö	Ö	Ö	Ö	Latin capital letter O with diaeresis
215	×	×	×	×	multiplication sign
216	Ø	Ø	Ø	Ø	Latin capital letter O with stroke
217	Ù	Ù	Ù	Ù	Latin capital letter U with grave
218	Ú	Ú	Ú	Ú	Latin capital letter U with acute
219	Û	Û	Û	Û	Latin capital letter U with circumflex
220	Ü	Ü	Ü	Ü	Latin capital letter U with diaeresis
221	Ý	Ý	Ý	Ý	Latin capital letter Y with acute
222	Þ	Þ	Þ	Þ	Latin capital letter Thorn
223	ß	ß	ß	ß	Latin small letter sharp s
224	à	à	à	à	Latin small letter a with grave
225	á	á	á	á	Latin small letter a with acute

Number	ASCII	ANSI	8859-1	UTF-8	Description
226	â	â	â	â	Latin small letter a with circumflex
227	ã	ã	ã	ã	Latin small letter a with tilde
228	ä	ä	ä	ä	Latin small letter a with diaeresis
229	å	å	å	å	Latin small letter a with ring above
230	æ	æ	æ	æ	Latin small letter ae
231	ç	ç	ç	ç	Latin small letter c with cedilla
232	è	è	è	è	Latin small letter e with grave
233	é	é	é	é	Latin small letter e with acute
234	ê	ê	ê	ê	Latin small letter e with circumflex
235	ë	ë	ë	ë	Latin small letter e with diaeresis
236	ì	ì	ì	ì	Latin small letter i with grave
237	í	í	í	í	Latin small letter i with acute
238	î	î	î	î	Latin small letter i with circumflex
239	ï	ï	ï	ï	Latin small letter i with diaeresis
240	ð	ð	ð	ð	Latin small letter eth
241	ñ	ñ	ñ	ñ	Latin small letter n with tilde
242	ò	ò	ò	ò	Latin small letter o with grave
243	ó	ó	ó	ó	Latin small letter o with acute
244	ô	ô	ô	ô	Latin small letter o with circumflex
245	õ	õ	õ	õ	Latin small letter o with tilde
246	ö	ö	ö	ö	Latin small letter o with diaeresis
247	÷	÷	÷	÷	division sign
248	ø	ø	ø	ø	Latin small letter o with stroke
249	ù	ù	ù	ù	Latin small letter u with grave
250	ú	ú	ú	ú	Latin small letter u with acute

Number	ASCII	ANSI	8859-1	UTF-8	Description
251	û	û	û	û	Latin small letter with circumflex
252	ü	ü	ü	ü	Latin small letter u with diaeresis
253	ý	ý	ý	ý	Latin small letter y with acute
254	þ	þ	þ	þ	Latin small letter thorn
255	ÿ	ÿ	ÿ	ÿ	Latin small letter y with diaeresis

HTML Elements and Valid DOCTYPES

HTML Elements - Valid DOCTYPES

The table below lists all HTML elements, and shows what !DOCTYPE each element appears in.

HTML 4.01 / XHTML 1.0					
Tag	HTML5	Transitional	Strict	Frameset	XHTML 1.1
<u><a></u>	Yes	Yes	Yes	Yes	Yes
<u><abbr></u>	Yes	Yes	Yes	Yes	Yes
<u><acronym></u>	No	Yes	Yes	Yes	Yes
<u><address></u>	Yes	Yes	Yes	Yes	Yes
<u><applet></u>	No	Yes	No	Yes	No
<u><area></u>	Yes	Yes	Yes	Yes	No
<u><article></u>	Yes	No	No	No	No
<u><aside></u>	Yes	No	No	No	No
<u><audio></u>	Yes	No	No	No	No
<u></u>	Yes	Yes	Yes	Yes	Yes
<u><base></u>	Yes	Yes	Yes	Yes	Yes
<u><basefont></u>	No	Yes	No	Yes	No
<u><bdi></u>	Yes	No	No	No	No
<u><bdo></u>	Yes	Yes	Yes	Yes	No
<u><big></u>	No	Yes	Yes	Yes	Yes
<u><blockquote></u>	Yes	Yes	Yes	Yes	Yes
<u><body></u>	Yes	Yes	Yes	Yes	Yes
<u>
</u>	Yes	Yes	Yes	Yes	Yes
<u><button></u>	Yes	Yes	Yes	Yes	Yes
<u><canvas></u>	Yes	No	No	No	No

HTML 4.01 / XHTML 1.0

Tag	HTML5	Transitional	Strict	Frameset	XHTML 1.1
<u><caption></u>	Yes	Yes	Yes	Yes	Yes
<u><center></u>	No	Yes	No	Yes	No
<u><cite></u>	Yes	Yes	Yes	Yes	Yes
<u><code></u>	Yes	Yes	Yes	Yes	Yes
<u><col></u>	Yes	Yes	Yes	Yes	No
<u><colgroup></u>	Yes	Yes	Yes	Yes	No
<u><datalist></u>	Yes	No	No	No	No
<u><dd></u>	Yes	Yes	Yes	Yes	Yes
<u></u>	Yes	Yes	Yes	Yes	No
<u><details></u>	Yes	No	No	No	No
<u><dfn></u>	Yes	Yes	Yes	Yes	Yes
<u><dialog></u>	Yes	No	No	No	No
<u><dir></u>	No	Yes	No	Yes	No
<u><div></u>	Yes	Yes	Yes	Yes	Yes
<u><dl></u>	Yes	Yes	Yes	Yes	Yes
<u><dt></u>	Yes	Yes	Yes	Yes	Yes
<u></u>	Yes	Yes	Yes	Yes	Yes
<u><embed></u>	Yes	No	No	No	No
<u><fieldset></u>	Yes	Yes	Yes	Yes	Yes
<u><figcaption></u>	Yes	No	No	No	No
<u><figure></u>	Yes	No	No	No	No
<u></u>	No	Yes	No	Yes	No
<u><footer></u>	Yes	No	No	No	No
<u><form></u>	Yes	Yes	Yes	Yes	Yes

HTML 4.01 / XHTML 1.0

Tag	HTML5	Transitional	Strict	Frameset	XHTML 1.1
<u><frame></u>	No	No	No	Yes	No
<u><frameset></u>	No	No	No	Yes	No
<u><h1> to <h6></u>	Yes	Yes	Yes	Yes	Yes
<u><head></u>	Yes	Yes	Yes	Yes	Yes
<u><header></u>	Yes	No	No	No	No
<u><hr></u>	Yes	Yes	Yes	Yes	Yes
<u><html></u>	Yes	Yes	Yes	Yes	Yes
<u><i></u>	Yes	Yes	Yes	Yes	Yes
<u><iframe></u>	Yes	Yes	No	Yes	No
<u></u>	Yes	Yes	Yes	Yes	Yes
<u><input></u>	Yes	Yes	Yes	Yes	Yes
<u><ins></u>	Yes	Yes	Yes	Yes	No
<u><kbd></u>	Yes	Yes	Yes	Yes	Yes
<u><label></u>	Yes	Yes	Yes	Yes	Yes
<u><legend></u>	Yes	Yes	Yes	Yes	Yes
<u></u>	Yes	Yes	Yes	Yes	Yes
<u><link></u>	Yes	Yes	Yes	Yes	Yes
<u><main></u>	Yes	No	No	No	No
<u><map></u>	Yes	Yes	Yes	Yes	No
<u><mark></u>	Yes	No	No	No	No
<u><meta></u>	Yes	Yes	Yes	Yes	Yes
<u><meter></u>	Yes	No	No	No	No
<u><nav></u>	Yes	No	No	No	No
<u><noframes></u>	No	Yes	No	Yes	No

HTML 4.01 / XHTML 1.0

Tag	HTML5	Transitional	Strict	Frameset	XHTML 1.1
<u><noscript></u>	Yes	Yes	Yes	Yes	Yes
<u><object></u>	Yes	Yes	Yes	Yes	Yes
<u></u>	Yes	Yes	Yes	Yes	Yes
<u><optgroup></u>	Yes	Yes	Yes	Yes	Yes
<u><option></u>	Yes	Yes	Yes	Yes	Yes
<u><output></u>	Yes	No	No	No	No
<u><p></u>	Yes	Yes	Yes	Yes	Yes
<u><param></u>	Yes	Yes	Yes	Yes	Yes
<u><pre></u>	Yes	Yes	Yes	Yes	Yes
<u><progress></u>	Yes	No	No	No	No
<u><q></u>	Yes	Yes	Yes	Yes	Yes
<u><rp></u>	Yes	No	No	No	No
<u><rt></u>	Yes	No	No	No	No
<u><ruby></u>	Yes	No	No	No	No
<u><s></u>	Yes	Yes	No	Yes	No
<u><samp></u>	Yes	Yes	Yes	Yes	Yes
<u><script></u>	Yes	Yes	Yes	Yes	Yes
<u><section></u>	Yes	No	No	No	No
<u><select></u>	Yes	Yes	Yes	Yes	Yes
<u><small></u>	Yes	Yes	Yes	Yes	Yes
<u><source></u>	Yes	No	No	No	No
<u></u>	Yes	Yes	Yes	Yes	Yes
<u><strike></u>	No	Yes	No	Yes	No
<u></u>	Yes	Yes	Yes	Yes	Yes

HTML 4.01 / XHTML 1.0

Tag	HTML5	Transitional	Strict	Frameset	XHTML 1.1
<u><style></u>	Yes	Yes	Yes	Yes	Yes
<u><sub></u>	Yes	Yes	Yes	Yes	Yes
<u><summary></u>	Yes	No	No	No	No
<u><sup></u>	Yes	Yes	Yes	Yes	Yes
<u><table></u>	Yes	Yes	Yes	Yes	Yes
<u><tbody></u>	Yes	Yes	Yes	Yes	No
<u><td></u>	Yes	Yes	Yes	Yes	Yes
<u><textarea></u>	Yes	Yes	Yes	Yes	Yes
<u><tfoot></u>	Yes	Yes	Yes	Yes	No
<u><th></u>	Yes	Yes	Yes	Yes	Yes
<u><thead></u>	Yes	Yes	Yes	Yes	No
<u><time></u>	Yes	No	No	No	No
<u><title></u>	Yes	Yes	Yes	Yes	Yes
<u><tr></u>	Yes	Yes	Yes	Yes	Yes
<u><track></u>	Yes	No	No	No	No
<u><tt></u>	No	Yes	Yes	Yes	Yes
<u><u></u>	Yes	Yes	No	Yes	No
<u></u>	Yes	Yes	Yes	Yes	Yes
<u><var></u>	Yes	Yes	Yes	Yes	Yes
<u><video></u>	Yes	No	No	No	No
<u><wbr></u>	Yes	No	No	No	No

HTML Language Code Reference

ISO Language Codes

The HTML lang attribute can be used to declare the language of a Web page or a portion of a Web page. This is meant to assist search engines and browsers.

According to the W3C recommendation you should declare the primary language for each Web page with the lang attribute inside the <html> tag, like this:

```
<html lang="en"> ... </html>
```

In XHTML, the language is declared inside the <html> tag as follows:

```
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en"> ...</html>
```

ISO 639-1 Language Codes

ISO 639-1 defines abbreviations for languages.

In HTML and XHTML they can be used in the lang and xml:lang attributes.

See also: next reference for country codes.

Language	ISO Code
Abkhazian	ab
Afar	aa
Afrikaans	af
Akan	ak
Albanian	sq
Amharic	am
Arabic	ar
Aragonese	an
Armenian	hy
Assamese	as
Avaric	av

Language	ISO Code
Avestan	ae
Aymara	ay
Azerbaijani	az
Bambara	bm
Bashkir	ba
Basque	eu
Belarusian	be
Bengali (Bangla)	bn
Bihari	bh
Bislama	bi
Bosnian	bs
Breton	br
Bulgarian	bg
Burmese	my
Catalan	ca
Chamorro	ch
Chechen	ce
Chichewa, Chewa, Nyanja	ny
Chinese	zh
Chinese (Simplified)	zh-Hans
Chinese (Traditional)	zh-Hant
Chuvash	cv
Cornish	kw
Corsican	co
Cree	cr

Language	ISO Code
Croatian	hr
Czech	cs
Danish	da
Divehi, Dhivehi, Maldivian	dv
Dutch	nl
Dzongkha	dz
English	en
Esperanto	eo
Estonian	et
Ewe	ee
Faroese	fo
Fijian	fj
Finnish	fi
French	fr
Fula, Fulah, Pulaar, Pular	ff
Galician	gl
Gaelic (Scottish)	gd
Gaelic (Manx)	gv
Georgian	ka
German	de
Greek	el
Greenlandic	kl
Guarani	gn
Gujarati	gu
Haitian Creole	ht

Language	ISO Code
Hausa	ha
Hebrew	he
Herero	hz
Hindi	hi
Hiri Motu	ho
Hungarian	hu
Icelandic	is
Ido	io
Igbo	ig
Indonesian	id, in
Interlingua	ia
Interlingue	ie
Inuktitut	iu
Inupiaq	ik
Irish	ga
Italian	it
Japanese	ja
Javanese	jv
Kalaallisut, Greenlandic	kl
Kannada	kn
Kanuri	kr
Kashmiri	ks
Kazakh	kk
Khmer	km
Kikuyu	ki

Language	ISO Code
Kinyarwanda (Rwanda)	rw
Kirundi	rn
Kyrgyz	ky
Komi	kv
Kongo	kg
Korean	ko
Kurdish	ku
Kwanyama	kj
Lao	lo
Latin	la
Latvian (Lettish)	lv
Limburgish (Limburger)	li
Lingala	ln
Lithuanian	lt
Luga-Katanga	lu
Luganda, Ganda	lg
Luxembourgish	lb
Manx	gv
Macedonian	mk
Malagasy	mg
Malay	ms
Malayalam	ml
Maltese	mt
Maori	mi
Marathi	mr

Language	ISO Code
Marshallese	mh
Moldavian	mo
Mongolian	mn
Nauru	na
Navajo	nv
Ndonga	ng
Northern Ndebele	nd
Nepali	ne
Norwegian	no
Norwegian bokmål	nb
Norwegian nynorsk	nn
Nuosu	ii
Occitan	oc
Ojibwe	oj
Old Church Slavonic, Old Bulgarian	cu
Oriya	or
Oromo (Afaan Oromo)	om
Ossetian	os
Pāli	pi
Pashto, Pushto	ps
Persian (Farsi)	fa
Polish	pl
Portuguese	pt
Punjabi (Eastern)	pa
Quechua	qu

Language	ISO Code
Romansh	rm
Romanian	ro
Russian	ru
Sami	se
Samoan	sm
Sango	sg
Sanskrit	sa
Serbian	sr
Serbo-Croatian	sh
Sesotho	st
Setswana	tn
Shona	sn
Sichuan Yi	ii
Sindhi	sd
Sinhalese	si
Siswati	ss
Slovak	sk
Slovenian	sl
Somali	so
Southern Ndebele	nr
Spanish	es
Sundanese	su
Swahili (Kiswahili)	sw
Swati	ss
Swedish	sv

Language	ISO Code
Tagalog	tl
Tahitian	ty
Tajik	tg
Tamil	ta
Tatar	tt
Telugu	te
Thai	th
Tibetan	bo
Tigrinya	ti
Tonga	to
Tsonga	ts
Turkish	tr
Turkmen	tk
Twi	tw
Uyghur	ug
Ukrainian	uk
Urdu	ur
Uzbek	uz
Venda	ve
Vietnamese	vi
Volapük	vo
Wallon	wa
Welsh	cy
Wolof	wo
Western Frisian	fy

Language	ISO Code
Xhosa	xh
Yiddish	yi, ji
Yoruba	yo
Zhuang, Chuang	za
Zulu	zu

HTML URL Encoding Reference

URL encoding converts characters into a format that can be transmitted over the Internet.

URL - Uniform Resource Locator

Web browsers request pages from web servers by using a URL.

The URL is the address of a web page, like: **https://www.w3schools.com**.

URL Encoding (Percent Encoding)

URLs can only be sent over the Internet using the ASCII character-set.

Since URLs often contain characters outside the ASCII set, the URL has to be converted into a valid ASCII format.

URL encoding replaces unsafe ASCII characters with a "%" followed by two hexadecimal digits.

URLs cannot contain spaces. URL encoding normally replaces a space with a plus (+) sign or with %20.

Try It Yourself

If you click the "Submit" button below, the browser will URL encode the input before it is sent to the server. A page at the server will display the received input.

Try some other input and click Submit again.

URL Encoding Functions

In JavaScript, PHP, and ASP there are functions that can be used to URL encode a string.

In JavaScript you can use the encodeURI() function.

PHP has the rawurlencode() function, and ASP has the Server.URLEncode() function.

Click the "URL Encode" button to see how the JavaScript function encodes the text.

Note: The JavaScript function encodes space as %20.

ASCII Encoding Reference

Your browser will encode input, according to the character-set used in your page.

The default character-set in HTML5 is UTF-8.

Character	From Windows-1252	From UTF-8
space	%20	%20
!	%21	%21
"	%22	%22
#	%23	%23
\$	%24	%24
%	%25	%25
&	%26	%26
'	%27	%27
(%28	%28
)	%29	%29
*	%2A	%2A
+	%2B	%2B
,	%2C	%2C
-	%2D	%2D
.	%2E	%2E
/	%2F	%2F
0	%30	%30
1	%31	%31
2	%32	%32
3	%33	%33
4	%34	%34
5	%35	%35
6	%36	%36
7	%37	%37
8	%38	%38

Character	From Windows-1252	From UTF-8
9	%39	%39
:	%3A	%3A
;	%3B	%3B
<	%3C	%3C
=	%3D	%3D
>	%3E	%3E
?	%3F	%3F
@	%40	%40
A	%41	%41
B	%42	%42
C	%43	%43
D	%44	%44
E	%45	%45
F	%46	%46
G	%47	%47
H	%48	%48
I	%49	%49
J	%4A	%4A
K	%4B	%4B
L	%4C	%4C
M	%4D	%4D
N	%4E	%4E
O	%4F	%4F
P	%50	%50
Q	%51	%51

Character	From Windows-1252	From UTF-8
R	%52	%52
S	%53	%53
T	%54	%54
U	%55	%55
V	%56	%56
W	%57	%57
X	%58	%58
Y	%59	%59
Z	%5A	%5A
[%5B	%5B
\	%5C	%5C
]	%5D	%5D
^	%5E	%5E
_	%5F	%5F
`	%60	%60
a	%61	%61
b	%62	%62
c	%63	%63
d	%64	%64
e	%65	%65
f	%66	%66
g	%67	%67
h	%68	%68
i	%69	%69
j	%6A	%6A

Character	From Windows-1252	From UTF-8
k	%6B	%6B
l	%6C	%6C
m	%6D	%6D
n	%6E	%6E
o	%6F	%6F
p	%70	%70
q	%71	%71
r	%72	%72
s	%73	%73
t	%74	%74
u	%75	%75
v	%76	%76
w	%77	%77
x	%78	%78
y	%79	%79
z	%7A	%7A
{	%7B	%7B
	%7C	%7C
}	%7D	%7D
~	%7E	%7E
	%7F	%7F
'	%80	%E2%82%AC
□	%81	%81
,	%82	%E2%80%9A
f	%83	%C6%92

Character	From Windows-1252	From UTF-8
"	%84	%E2%80%9E
...	%85	%E2%80%A6
†	%86	%E2%80%A0
‡	%87	%E2%80%A1
^	%88	%CB%86
%oo	%89	%E2%80%B0
Š	%8A	%C5%A0
<	%8B	%E2%80%B9
Œ	%8C	%C5%92
□	%8D	%C5%8D
Ž	%8E	%C5%BD
□	%8F	%8F
□	%90	%C2%90
`	%91	%E2%80%98
'	%92	%E2%80%99
``	%93	%E2%80%9C
''	%94	%E2%80%9D
•	%95	%E2%80%A2
-	%96	%E2%80%93
—	%97	%E2%80%94
~	%98	%CB%9C
™	%99	%E2%84
š	%9A	%C5%A1
>	%9B	%E2%80
œ	%9C	%C5%93

Character	From Windows-1252	From UTF-8
□	%9D	%9D
ž	%9E	%C5%BE
ÿ	%9F	%C5%B8
	%A0	%C2%A0
í	%A1	%C2%A1
¢	%A2	%C2%A2
£	%A3	%C2%A3
¤	%A4	%C2%A4
¥	%A5	%C2%A5
¡	%A6	%C2%A6
§	%A7	%C2%A7
..	%A8	%C2%A8
©	%A9	%C2%A9
¤	%AA	%C2%AA
«	%AB	%C2%AB
¬	%AC	%C2%AC
	%AD	%C2%AD
®	%AE	%C2%AE
—	%AF	%C2%AF
°	%B0	%C2%B0
±	%B1	%C2%B1
²	%B2	%C2%B2
³	%B3	%C2%B3
,	%B4	%C2%B4
µ	%B5	%C2%B5

Character	From Windows-1252	From UTF-8
¶	%B6	%C2%B6
.	%B7	%C2%B7
,	%B8	%C2%B8
¹	%B9	%C2%B9
º	%BA	%C2%BA
»	%BB	%C2%BB
¼	%BC	%C2%BC
½	%BD	%C2%BD
¾	%BE	%C2%BE
¿	%BF	%C2%BF
À	%C0	%C3%80
Á	%C1	%C3%81
Â	%C2	%C3%82
Ã	%C3	%C3%83
Ä	%C4	%C3%84
Å	%C5	%C3%85
Æ	%C6	%C3%86
Ҫ	%C7	%C3%87
È	%C8	%C3%88
É	%C9	%C3%89
Ê	%CA	%C3%8A
Ӯ	%CB	%C3%8B
Ӣ	%CC	%C3%8C
Ӣ	%CD	%C3%8D
Ӣ	%CE	%C3%8E

Character	From Windows-1252	From UTF-8
Ï	%CF	%C3%8F
Ð	%D0	%C3%90
Ñ	%D1	%C3%91
Ò	%D2	%C3%92
Ó	%D3	%C3%93
Ô	%D4	%C3%94
Õ	%D5	%C3%95
Ö	%D6	%C3%96
×	%D7	%C3%97
Ø	%D8	%C3%98
Ù	%D9	%C3%99
Ú	%DA	%C3%9A
Û	%DB	%C3%9B
Ü	%DC	%C3%9C
Ý	%DD	%C3%9D
þ	%DE	%C3%9E
ß	%DF	%C3%9F
à	%E0	%C3%A0
á	%E1	%C3%A1
â	%E2	%C3%A2
ã	%E3	%C3%A3
ä	%E4	%C3%A4
å	%E5	%C3%A5
æ	%E6	%C3%A6
ç	%E7	%C3%A7

Character	From Windows-1252	From UTF-8
è	%E8	%C3%A8
é	%E9	%C3%A9
ê	%EA	%C3%Aâ
ë	%EB	%C3%Aë
ì	%EC	%C3%AC
í	%ED	%C3%AD
î	%EE	%C3%AE
ĩ	%EF	%C3%AF
ð	%F0	%C3%B0
ñ	%F1	%C3%B1
ò	%F2	%C3%B2
ó	%F3	%C3%B3
ô	%F4	%C3%B4
õ	%F5	%C3%B5
ö	%F6	%C3%B6
÷	%F7	%C3%B7
ø	%F8	%C3%B8
ù	%F9	%C3%B9
ú	%FA	%C3%BA
û	%FB	%C3%BB
ü	%FC	%C3%BC
ý	%FD	%C3%BD
þ	%FE	%C3%BE
ÿ	%FF	%C3%BF

URL Encoding Reference

The ASCII control characters **%00-%1F** were originally designed to control hardware devices. Control characters have nothing to do inside a URL.

ASCII Character	Description	URL-encoding
NUL	null character	%00
SOH	start of header	%01
STX	start of text	%02
ETX	end of text	%03
EOT	end of transmission	%04
ENQ	enquiry	%05
ACK	acknowledge	%06
BEL	bell (ring)	%07
BS	backspace	%08
HT	horizontal tab	%09
LF	line feed	%0A
VT	vertical tab	%0B
FF	form feed	%0C
CR	carriage return	%0D
SO	shift out	%0E
SI	shift in	%0F
DLE	data link escape	%10
DC1	device control 1	%11
DC2	device control 2	%12
DC3	device control 3	%13
DC4	device control 4	%14
NAK	negative acknowledge	%15
SYN	synchronize	%16
ETB	end transmission block	%17

ASCII Character	Description	URL-encoding
CAN	cancel	%18
EM	end of medium	%19
SUB	substitute	%1A
ESC	escape	%1B
FS	file separator	%1C
GS	group separator	%1D
RS	record separator	%1E
US	unit separator	%1F

HTML ISO Country Codes Reference

ISO Country Codes

ISO country codes define abbreviations for countries.

In HTML they can be used as an addition to the language value in the lang attribute.

The first two characters of a language code defines the **language**. (see previous reference)

The last two defines the **country**.

```
<html lang="en-US"> ... </html>
```

ISO Country Codes

Country	ISO Code
AFGHANISTAN	AF
ALBANIA	AL
ALGERIA	DZ
AMERICAN SAMOA	AS
ANDORRA	AD
ANGOLA	AO
ANTARCTICA	AQ
ANTIGUA AND BARBUDA	AG
ARGENTINA	AR
ARMENIA	AM
ARUBA	AW
AUSTRALIA	AU
AUSTRIA	AT
AZERBAIJAN	AZ
BAHAMAS	BS
BAHRAIN	BH

Country	ISO Code
BANGLADESH	BD
BARBADOS	BB
BELARUS	BY
BELGIUM	BE
BELIZE	BZ
BENIN	BJ
BERMUDA	BM
BHUTAN	BT
BOLIVIA	BO
BOSNIA AND HERZEGOVINA	BA
BOTSWANA	BW
BOUVET ISLAND	BV
BRAZIL	BR
BRITISH INDIAN OCEAN TERRITORY	IO
BRUNEI DARUSSALAM	BN
BULGARIA	BG
BURKINA FASO	BF
BURUNDI	BI
CAMBODIA	KH
CAMEROON	CM
CANADA	CA
CAPE VERDE	CV
CAYMAN ISLANDS	KY
CENTRAL AFRICAN REPUBLIC	CF
CHAD	TD

Country	ISO Code
CHILE	CL
CHINA	CN
CHRISTMAS ISLAND	CX
COCOS (KEELING) ISLANDS	CC
COLOMBIA	CO
COMOROS	KM
CONGO	CG
CONGO, THE DEMOCRATIC REPUBLIC OF THE	CD
COOK ISLANDS	CK
COSTA RICA	CR
CÔTE D'IVOIRE	CI
CROATIA	HR
CUBA	CU
CYPRUS	CY
CZECH REPUBLIC	CZ
DENMARK	DK
DJIBOUTI	DJ
DOMINICA	DM
DOMINICAN REPUBLIC	DO
ECUADOR	EC
EGYPT	EG
EL SALVADOR	SV
EQUATORIAL GUINEA	GQ
ERITREA	ER
ESTONIA	EE

Country	ISO Code
ETHIOPIA	ET
FALKLAND ISLANDS (MALVINAS)	FK
FAROE ISLANDS	FO
FIJI	FJ
FINLAND	FI
FRANCE	FR
FRENCH GUIANA	GF
FRENCH POLYNESIA	PF
FRENCH SOUTHERN TERRITORIES	TF
GABON	GA
GAMBIA	GM
GEORGIA	GE
GERMANY	DE
GHANA	GH
GIBRALTAR	GI
GREECE	GR
GREENLAND	GL
GRENADA	GD
GUADELOUPE	GP
GUAM	GU
GUATEMALA	GT
GUINEA	GN
GUINEA-BISSAU	GW
GUYANA	GY
HAITI	HT

Country	ISO Code
HEARD ISLAND AND MCDONALD ISLANDS	HM
HONDURAS	HN
HONG KONG	HK
HUNGARY	HU
ICELAND	IS
INDIA	IN
INDONESIA	ID
IRAN, ISLAMIC REPUBLIC OF	IR
IRAQ	IQ
IRELAND	IE
ISRAEL	IL
ITALY	IT
JAMAICA	JM
JAPAN	JP
JORDAN	JO
KAZAKHSTAN	KZ
KENYA	KE
KIRIBATI	KI
KOREA, DEMOCRATIC PEOPLE'S REPUBLIC OF	KP
KOREA, REPUBLIC OF	KR
KUWAIT	KW
KYRGYZSTAN	KG
LAO PEOPLE'S DEMOCRATIC REPUBLIC (LAOS)	LA
LATVIA	LV
LEBANON	LB

Country	ISO Code
LESOTHO	LS
LIBERIA	LR
LIBYAN ARAB JAMAHIRIYA	LY
LIECHTENSTEIN	LI
LITHUANIA	LT
LUXEMBOURG	LU
MACAO	MO
MACEDONIA, THE FORMER YUGOSLAV REPUBLIC OF	MK
MADAGASCAR	MG
MALAWI	MW
MALAYSIA	MY
MALDIVES	MV
MALI	ML
MALTA	MT
MARSHALL ISLANDS	MH
MARTINIQUE	MQ
MAURITANIA	MR
MAURITIUS	MU
MAYOTTE	YT
MEXICO	MX
MICRONESIA, FEDERATED STATES OF	FM
MOLDOVA, REPUBLIC OF	MD
MONACO	MC
MONGOLIA	MN
MONTENEGRO	ME

Country	ISO Code
MONTSERRAT	MS
MOROCCO	MA
MOZAMBIQUE	MZ
MYANMAR	MM
NAMIBIA	NA
NAURU	NR
NEPAL	NP
NETHERLANDS	NL
NETHERLANDS ANTILLES	AN
NEW CALEDONIA	NC
NEW ZEALAND	NZ
NICARAGUA	NI
NIGER	NE
NIGERIA	NG
NIUE	NU
NORFOLK ISLAND	NF
NORTHERN MARIANA ISLANDS	MP
NORWAY	NO
OMAN	OM
PAKISTAN	PK
PALAU	PW
PALESTINIAN TERRITORY, OCCUPIED	PS
PANAMA	PA
PAPUA NEW GUINEA	PG
PARAGUAY	PY

Country	ISO Code
PERU	PE
PHILIPPINES	PH
PITCAIRN	PN
POLAND	PL
PORTUGAL	PT
PUERTO RICO	PR
QATAR	QA
RÉUNION	RE
ROMANIA	RO
RUSSIAN FEDERATION	RU
RWANDA	RW
SAINT HELENA	SH
SAINT KITTS AND NEVIS	KN
SAINT LUCIA	LC
SAINT PIERRE AND MIQUELON	PM
SAINT VINCENT AND THE GRENADINES	VC
SAMOA	WS
SAN MARINO	SM
SAO TOME AND PRINCIPE	ST
SAUDI ARABIA	SA
SENEGAL	SN
SERBIA	RS
SEYCHELLES	SC
SIERRA LEONE	SL
SINGAPORE	SG

Country	ISO Code
SLOVAKIA	SK
SLOVENIA	SI
SOLOMON ISLANDS	SB
SOMALIA	SO
SOUTH AFRICA	ZA
SOUTH GEORGIA AND THE SOUTH SANDWICH ISLANDS	GS
SPAIN	ES
SRI LANKA	LK
SUDAN	SD
SURINAME	SR
SVALBARD AND JAN MAYEN	SJ
SWAZILAND	SZ
SWEDEN	SE
SWITZERLAND	CH
SYRIAN ARAB REPUBLIC	SY
TAIWAN	TW
TAJIKISTAN	TJ
TANZANIA, UNITED REPUBLIC OF	TZ
THAILAND	TH
TIMOR-LESTE	TL
TOGO	TG
TOKELAU	TK
TONGA	TO
TRINIDAD AND TOBAGO	TT
TUNISIA	TN

Country	ISO Code
TURKEY	TR
TURKMENISTAN	TM
TURKS AND CAICOS ISLANDS	TC
TUVALU	TV
UGANDA	UG
UKRAINE	UA
UNITED ARAB EMIRATES	AE
UNITED KINGDOM	GB
UNITED STATES	US
UNITED STATES MINOR OUTLYING ISLANDS	UM
URUGUAY	UY
UZBEKISTAN	UZ
VANUATU	VU
VENEZUELA	VE
VIET NAM	VN
VIRGIN ISLANDS, BRITISH	VG
VIRGIN ISLANDS, U.S.	VI
WALLIS AND FUTUNA	WF
WESTERN SAHARA	EH
YEMEN	YE
ZAMBIA	ZM
ZIMBABWE	ZW

HTTP Status Messages

When a browser requests a service from a web server, an error might occur. This is a list of HTTP status messages that might be returned:

1xx: Information

Message:	Description:
100 Continue	The server has received the request headers, and the client should proceed to send the request body
101 Switching Protocols	The requester has asked the server to switch protocols
103 Checkpoint	Used in the resumable requests proposal to resume aborted PUT or POST requests

2xx: Successful

Message:	Description:
200 OK	The request is OK (this is the standard response for successful HTTP requests)
201 Created	The request has been fulfilled, and a new resource is created
202 Accepted	The request has been accepted for processing, but the processing has not been completed
203 Non-Authoritative Information	The request has been successfully processed, but is returning information that may be from another source
204 No Content	The request has been successfully processed, but is not returning any content
205 Reset Content	The request has been successfully processed, but is not returning any content, and requires that the requester reset the document view
206 Partial Content	The server is delivering only part of the resource due to a range header sent by the client

3xx: Redirection

Message:	Description:
300 Multiple Choices	A link list. The user can select a link and go to that location. Maximum five addresses
301 Moved Permanently	The requested page has moved to a new URL
302 Found	The requested page has moved temporarily to a new URL
303 See Other	The requested page can be found under a different URL
304 Not Modified	Indicates the requested page has not been modified since last requested
306 Switch Proxy	<i>No longer used</i>
307 Temporary Redirect	The requested page has moved temporarily to a new URL
308 Resume Incomplete	Used in the resumable requests proposal to resume aborted PUT or POST requests

4xx: Client Error

Message:	Description:
400 Bad Request	The request cannot be fulfilled due to bad syntax
401 Unauthorized	The request was a legal request, but the server is refusing to respond to it. For use when authentication is possible but has failed or not yet been provided
402 Payment Required	<i>Reserved for future use</i>
403 Forbidden	The request was a legal request, but the server is refusing to respond to it
404 Not Found	The requested page could not be found but may be available again in the future
405 Method Not Allowed	A request was made of a page using a request method not supported by that page

Message:	Description:
406 Not Acceptable	The server can only generate a response that is not accepted by the client
407 Proxy Authentication Required	The client must first authenticate itself with the proxy
408 Request Timeout	The server timed out waiting for the request
409 Conflict	The request could not be completed because of a conflict in the request
410 Gone	The requested page is no longer available
411 Length Required	The "Content-Length" is not defined. The server will not accept the request without it
412 Precondition Failed	The precondition given in the request evaluated to false by the server
413 Request Entity Too Large	The server will not accept the request, because the request entity is too large
414 Request-URI Too Long	The server will not accept the request, because the URL is too long. Occurs when you convert a POST request to a GET request with a long query information
415 Unsupported Media Type	The server will not accept the request, because the media type is not supported
416 Requested Range Not Satisfiable	The client has asked for a portion of the file, but the server cannot supply that portion
417 Expectation Failed	The server cannot meet the requirements of the Expect request-header field

5xx: Server Error

Message:	Description:
500 Internal Server Error	A generic error message, given when no more specific message is suitable
501 Not Implemented	The server either does not recognize the request method, or it lacks the ability to fulfill the request
502 Bad Gateway	The server was acting as a gateway or proxy and received an invalid response from the upstream server
503 Service Unavailable	The server is currently unavailable (overloaded or down)
504 Gateway Timeout	The server was acting as a gateway or proxy and did not receive a timely response from the upstream server
505 HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request
511 Network Authentication Required	The client needs to authenticate to gain network access

HTTP Request Methods

The two most common HTTP methods are: GET and POST.

What is HTTP?

The Hypertext Transfer Protocol (HTTP) is designed to enable communications between clients and servers.

HTTP works as a request-response protocol between a client and server.

A web browser may be the client, and an application on a computer that hosts a web site may be the server.

Example: A client (browser) submits an HTTP request to the server; then the server returns a response to the client. The response contains status information about the request and may also contain the requested content.

HTTP Methods

- **GET**
- **POST**
- **PUT**
- **HEAD**
- **DELETE**
- **PATCH**
- **OPTIONS**

The GET Method

GET is used to request data from a specified resource.

GET is one of the most common HTTP methods.

Note that the query string (name/value pairs) is sent in the URL of a GET request:

```
/test/demo_form.php?name1=value1&name2=value2
```

Some other notes on GET requests:

- GET requests can be cached
- GET requests remain in the browser history
- GET requests can be bookmarked
- GET requests should never be used when dealing with sensitive data
- GET requests have length restrictions
- GET requests is only used to request data (not modify)

The POST Method

POST is used to send data to a server to create/update a resource.

The data sent to the server with POST is stored in the request body of the HTTP request:

```
POST /test/demo_form.php HTTP/1.1 Host: w3schools.com  
name1=value1&name2=value2
```

POST is one of the most common HTTP methods.

Some other notes on POST requests:

- POST requests are never cached
- POST requests do not remain in the browser history
- POST requests cannot be bookmarked
- POST requests have no restrictions on data length

The PUT Method

PUT is used to send data to a server to create/update a resource.

The difference between POST and PUT is that PUT requests are idempotent. That is, calling the same PUT request multiple times will always produce the same result. In contrast, calling a POST request repeatedly have side effects of creating the same resource multiple times.

The HEAD Method

HEAD is almost identical to GET, but without the response body.

In other words, if GET /users returns a list of users, then HEAD /users will make the same request but will not return the list of users.

HEAD requests are useful for checking what a GET request will return before actually making a GET request - like before downloading a large file or response body.

The DELETE Method

The DELETE method deletes the specified resource.

The OPTIONS Method

The OPTIONS method describes the communication options for the target resource.

Compare GET vs. POST

The following table compares the two HTTP methods: GET and POST.

	GET	POST
BACK button/Reload	Harmless	Data will be re-submitted (the browser should alert the user that the data are about to be re-submitted)
Bookmarked	Can be bookmarked	Cannot be bookmarked
Cached	Can be cached	Not cached
Encoding type	application/x-www-form-urlencoded	application/x-www-form-urlencoded or multipart/form-data. Use multipart encoding for binary data
History	Parameters remain in browser history	Parameters are not saved in browser history
Restrictions on data length	Yes, when sending data, the GET method adds the data to the URL; and the length of a URL is limited (maximum URL length is 2048 characters)	No restrictions
Restrictions on data type	Only ASCII characters allowed	No restrictions. Binary data is also allowed
Security	GET is less secure compared to POST because data sent is part of the URL Never use GET when sending passwords or other sensitive information!	POST is a little safer than GET because the parameters are not stored in browser history or in web server logs
Visibility	Data is visible to everyone in the URL	Data is not displayed in the URL

Pixels to Ems Conversion

The tool below allows you to work out the em sizes from pixels (or vice versa).

Pixel to Em Converter

- Set a default pixel size for body (usually 16px)
- Then, convert a pixel value to em, based on the default pixel size
- Or, convert an em value to pixels, based on the default pixel size

Set a default pixel size:

px

Convert PX to EM:

px

Convert EM to PX:

em

Convert

Result:

Body Font Size

In the table below, select a body font size in pixels (px) to display a complete "px to em and percent" conversion table.

Tip: The default font size is usually 16px.

px	em	percent
5px	0.3125em	31.25%
6px	0.3750em	37.50%
7px	0.4375em	43.75%
8px	0.5000em	50.00%
9px	0.5625em	56.25%
10px	0.6250em	62.50%
11px	0.6875em	68.75%
12px	0.7500em	75.00%
13px	0.8125em	81.25%
14px	0.8750em	87.50%
15px	0.9375em	93.75%
16px	1.0000em	100.00%
17px	1.0625em	106.25%
18px	1.1250em	112.50%
19px	1.1875em	118.75%
20px	1.2500em	125.00%
21px	1.3125em	131.25%
22px	1.3750em	137.50%
23px	1.4375em	143.75%
24px	1.5000em	150.00%
25px	1.5625em	156.25%

What is the difference between PX, EM and Percent?

Pixel is a static measurement, while percent and EM are relative measurements. The size of an EM or percent depends on its parent. If the text size of body is 16 pixels, then 150% or 1.5 EM will be 24 pixels ($1.5 * 16$). Look at [CSS Units](#) for more measurement units.

HTML <!--...--> Tag

Example

An HTML comment:

```
<!--This is a comment. Comments are not displayed in the browser-->  
<p>This is a paragraph.</p>
```

Definition and Usage

The comment tag is used to insert comments in the source code. Comments are not displayed in the browsers.

You can use comments to explain your code, which can help you when you edit the source code at a later date. This is especially useful if you have a lot of code.

Browser Support

Element					
<!--...-->	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Tips and Notes

You can also use the comment tag to "hide" scripts from browsers without support for scripts (so they don't show them as plain text):

```
<script type="text/javascript"> <!-- function displayMsg() {      alert("Hello  
World!") } //--> </script>
```

Note: The two forward slashes at the end of comment line (//) is the JavaScript comment symbol. This prevents JavaScript from executing the --> tag.

Standard Attributes

The comment tag does not support any standard attributes.

More information about [Standard Attributes](#).

Event Attributes

The comment tag does not support any event attributes.

More information about [Event Attributes](#).

HTML <!DOCTYPE> Declaration

Example

```
<!DOCTYPE html>
<html>
<head>
<title>Title of the document</title>
</head>

<body>
The content of the document.....
</body>

</html>
```

Definition and Usage

The <!DOCTYPE> declaration must be the very first thing in your HTML document, before the <html> tag.

The <!DOCTYPE> declaration is not an HTML tag; it is an instruction to the web browser about what version of HTML the page is written in.

In HTML 4.01, the <!DOCTYPE> declaration refers to a DTD, because HTML 4.01 was based on SGML. The DTD specifies the rules for the markup language, so that the browsers render the content correctly.

HTML5 is not based on SGML, and therefore does not require a reference to a DTD.

Tip: Always add the <!DOCTYPE> declaration to your HTML documents, so that the browser knows what type of document to expect.

Browser Support

Element					
<!DOCTYPE>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

There are three different <!DOCTYPE> declarations in HTML 4.01. In HTML5 there is only one:

```
<!DOCTYPE html>
```

HTML Elements and Doctypes

Look at our table of all [HTML elements](#), and what Doctype each element appears in.

Tips and Notes

Tip: The <!DOCTYPE> declaration is NOT case sensitive.

Tip: To check if the HTML of your Web documents is valid, go to [W3C's validation service](#).

Common DOCTYPE Declarations

HTML 5

```
<!DOCTYPE html>
```

HTML 4.01 Strict

This DTD contains all HTML elements and attributes, but does NOT INCLUDE presentational or deprecated elements (like font). Framesets are not allowed.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

HTML 4.01 Transitional

This DTD contains all HTML elements and attributes, INCLUDING presentational and deprecated elements (like font). Framesets are not allowed.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

HTML 4.01 Frameset

This DTD is equal to HTML 4.01 Transitional, but allows the use of frameset content.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"  
"http://www.w3.org/TR/html4/frameset.dtd">
```

XHTML 1.0 Strict

This DTD contains all HTML elements and attributes, but does NOT INCLUDE presentational or deprecated elements (like font). Framesets are not allowed. The markup must also be written as well-formed XML.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

XHTML 1.0 Transitional

This DTD contains all HTML elements and attributes, INCLUDING presentational and deprecated elements (like font). Framesets are not allowed. The markup must also be written as well-formed XML.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

XHTML 1.0 Frameset

This DTD is equal to XHTML 1.0 Transitional, but allows the use of frameset content.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```

XHTML 1.1

This DTD is equal to XHTML 1.0 Strict, but allows you to add modules (for example to provide Ruby support for East-Asian languages).

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"  
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

HTML <a> Tag

Example

A link to W3Schools.com:

```
<a href="https://www.w3schools.com">Visit W3Schools.com!</a>
```

More "Try it Yourself" examples below.

Definition and Usage

The <a> tag defines a hyperlink, which is used to link from one page to another.

The most important attribute of the <a> element is the href attribute, which indicates the link's destination.

By default, links will appear as follows in all browsers:

- An unvisited link is underlined and blue
- A visited link is underlined and purple
- An active link is underlined and red

Browser Support

Element					
<a>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: The following attributes: download, hreflang, media, rel, target, and type cannot be present if the href attribute is not present.

Tip: A linked page is normally displayed in the current browser window, unless you specify another target.

Tip: Use CSS to style links: [CSS Links Tutorial](#) and [CSS Buttons Tutorial](#)

Differences Between HTML 4.01 and HTML5

In HTML 4.01, the <a> tag could be either a hyperlink or an anchor. In HTML5, the <a> tag is always a hyperlink, but if it has no href attribute, it is only a placeholder for a hyperlink.

HTML5 has some new attributes, and some HTML 4.01 attributes are no longer supported.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>charset</u>	<i>char_encoding</i>	Not supported in HTML5. Specifies the character-set of a linked document
<u>coords</u>	<i>coordinates</i>	Not supported in HTML5. Specifies the coordinates of a link
<u>download</u>	<i>filename</i>	Specifies that the target will be downloaded when a user clicks on the hyperlink
<u>href</u>	<i>URL</i>	Specifies the URL of the page the link goes to
<u>hreflang</u>	<i>language_code</i>	Specifies the language of the linked document
<u>media</u>	<i>media_query</i>	Specifies what media/device the linked document is optimized for
<u>name</u>	<i>section_name</i>	Not supported in HTML5. Use the global <u>id attribute</u> instead. Specifies the name of an anchor
<u>ping</u>	<i>list_of_URLs</i>	Specifies a space-separated list of URLs to which, when the link is followed, post requests with the body ping will be sent by the browser (in the background). Typically used for tracking.
<u>rel</u>	alternate author bookmarkexternal help license next nofollow noreferrer noopener prev search tag	Specifies the relationship between the current document and the linked document
<u>rev</u>	<i>text</i>	Not supported in HTML5. Specifies the relationship between the linked document and the current document
<u>shape</u>	default rect circle poly	Not supported in HTML5. Specifies the shape of a link
<u>target</u>	_blank _parent _self _top <i>framename</i>	Specifies where to open the linked document
<u>type</u>	<i>media_type</i>	Specifies the media type of the linked document

Global Attributes

The `<a>` tag also supports the [Global Attributes in HTML](#).

Event Attributes

The `<a>` tag also supports the [Event Attributes in HTML](#).

Try it Yourself - Examples

[Create hyperlinks](#) How to create hyperlinks.

[Create a clickable image](#) How to use an image as a link.

[Open a link in a new browser window](#) How to open a link in a new browser window, so that the visitor does not have to leave your Web site.

[Create an email link](#) How to link to a mail message (will only work if you have mail installed).

[Create another email link](#) Another mailto link.

[Create a phone link](#) How to link to a phone number.

[Link to another section on the same page](#) How to link to an element with a specified id within a page (the name attribute is not supported in HTML5).

[Link to a JavaScript](#) How to add a JavaScript inside the href attribute.

HTML <abbr> Tag

Example

An abbreviation is marked up as follows:

The `<abbr title="World Health Organization">WHO</abbr>` was founded in 1948.

Definition and Usage

The <abbr> tag defines an abbreviation or an acronym, like "Mr.", "Dec.", "ASAP", "ATM".

Tip: An abbreviation and an acronym are both shortened versions of something else. Both are often represented as a series of letters.

Marking up abbreviations can give useful information to browsers, translation systems and search-engines.

Browser Support

Element					
<abbr>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: The global title attribute can be used in the <abbr> tag to show the full version of the abbreviation/acronym when you mouse over the <abbr> element.

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The <abbr> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <abbr> tag also supports the [Event Attributes in HTML](#).

HTML <acronym> Tag. Not Supported in HTML5.

Example

An acronym is marked up as follows:

Can I get this <acronym title="as soon as possible">ASAP</acronym>?

Definition and Usage

The <acronym> tag is not supported in HTML5. Use the <abbr> tag instead.

The <acronym> tag defines an acronym.

An acronym must spell out another word. For example: NASA, ASAP, GUI.

Marking up acronyms can give useful information to browsers, translation systems and search-engines.

Browser Support

Element					
<acronym>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: The title attribute can be used to show the full version of the acronym when you mouse over it.

Differences Between HTML 4.01 and HTML5

The <acronym> tag is not supported in HTML5.

HTML <address> Tag

Example

Contact information for Example.com:

```
<address>
Written by <a href="mailto:webmaster@example.com">Jon Doe</a>.<br>
Visit us at:<br>
Example.com<br>
Box 564, Disneyland<br>
USA
</address>
```

Definition and Usage

The <address> tag defines the contact information for the author/owner of a document or an article.

If the <address> element is inside the <body> element, it represents contact information for the document.

If the <address> element is inside an <article> element, it represents contact information for that article.

The text in the <address> element usually renders in *italic*. Most browsers will add a line break before and after the address element.

Browser Support

Element					
<address>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: The <address> tag should NOT be used to describe a postal address, unless it is a part of the contact information.

Tip: The <address> element will typically be included along with other information in a <footer> element.

Differences Between HTML 4.01 and HTML5

HTML 4.01 does not support the <article> tag, so in HTML 4.01 the <address> tag always defines the contact information of the *document's* author/owner.

Global Attributes

The <address> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <address> tag also supports the [Event Attributes in HTML](#).

HTML <applet> Tag. Not Supported in HTML5.

Example

An embedded Java applet:

```
<applet code="Bubbles.class" width="350" height="350">  
Java applet that draws animated bubbles.  
</applet>
```

Definition and Usage

The <applet> tag is not supported in HTML5. Use [<embed>](#) or [<object>](#) instead.

The <applet> tag defines an embedded applet.

Browser Support

Element					
<applet>	Not supported	Not supported	Yes	Yes	Not supported

Note: There is still some support for the <applet> tag in some browsers, but it requires additional plug-ins/installations to work.

Note: The <applet> tag is supported in Internet Explorer 11 and earlier versions, using a plug-in.

Differences Between HTML 4.01 and HTML5

The <applet> tag is not supported in HTML5.

Required Attributes

Attribute	Value	Description
code	URL	Specifies the file name of a Java applet
object	name	Specifies a reference to a serialized representation of an applet

Optional Attributes

Attribute	Value	Description
align	left right top bottom middle baseline	Specifies the alignment of an applet according to surrounding elements
alt	<i>text</i>	Specifies an alternate text for an applet
archive	<i>URL</i>	Specifies the location of an archive file
codebase	<i>URL</i>	Specifies a relative base URL for applets specified in the code attribute
height	<i>pixels</i>	Specifies the height of an applet
hspace	<i>pixels</i>	Defines the horizontal spacing around an applet
name	<i>name</i>	Defines the name for an applet (to use in scripts)
vspace	<i>pixels</i>	Defines the vertical spacing around an applet
width	<i>pixels</i>	Specifies the width of an applet

HTML <area> Tag

Example

An image-map, with clickable areas:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>
```

Definition and Usage

The <area> tag defines an area inside an image-map (an image-map is an image with clickable areas).

The <area> element is always nested inside a <map> tag.

Note: The usemap attribute in the tag is associated with the <map> element's name attribute, and creates a relationship between the image and the map.

Browser Support

Element					
<area>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

HTML5 has some new attributes, and some HTML 4.01 attributes are no longer supported.

Differences Between HTML and XHTML

In HTML the <area> tag has no end tag.

In XHTML the <area> tag must be properly closed.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>alt</u>	<i>text</i>	Specifies an alternate text for the area. Required if the href attribute is present
<u>coords</u>	<i>coordinates</i>	Specifies the coordinates of the area
<u>download</u>	<i>filename</i>	Specifies that the target will be downloaded when a user clicks on the hyperlink
<u>href</u>	<i>URL</i>	Specifies the hyperlink target for the area
<u>hreflang</u>	<i>language_code</i>	Specifies the language of the target URL
<u>media</u>	<i>media query</i>	Specifies what media/device the target URL is optimized for
<u>nohref</u>	<i>value</i>	Not supported in HTML5. Specifies that an area has no associated link
<u>rel</u>	alternate author bookmark help license nextnofollow noreferrer prefetch prev search tag	Specifies the relationship between the current document and the target URL
<u>shape</u>	default rect circle poly	Specifies the shape of the area
<u>target</u>	_blank _parent _self _top <i>framename</i>	Specifies where to open the target URL
<u>type</u>	<i>media_type</i>	Specifies the media type of the target URL

Global Attributes

The <area> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <area> tag also supports the [Event Attributes in HTML](#).

HTML <article> Tag

Example

```
<article>
  <h1>Google Chrome</h1>
  <p>Google Chrome is a free, open-source web browser developed by Google,
  released in 2008.</p>
</article>
```

Definition and Usage

The <article> tag specifies independent, self-contained content.

An article should make sense on its own and it should be possible to distribute it independently from the rest of the site.

Potential sources for the <article> element:

- Forum post
- Blog post
- News story
- Comment

Browser Support

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

Element					
<article>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <article> tag is new in HTML5.

Global Attributes

The <article> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <article> tag also supports the [Event Attributes in HTML](#).

HTML <aside> Tag

Example

```
<p>My family and I visited The Epcot center this summer.</p>

<aside>
  <h4>Epcot Center</h4>
  <p>The Epcot Center is a theme park in Disney World, Florida.</p>
</aside>
```

Definition and Usage

The <aside> tag defines some content aside from the content it is placed in. The aside content should be related to the surrounding content.

Browser Support

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

Element					
<aside>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <aside> tag is new in HTML5.

Tips and Notes

Tip: The <aside> content could be placed as a sidebar in an article.

Global Attributes

The <aside> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <aside> tag also supports the [Event Attributes in HTML](#).

HTML <audio> Tag

Example

Play a sound:

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Your browser does not support the audio tag.
</audio>
```

Definition and Usage

The <audio> tag defines sound, such as music or other audio streams.

Currently, there are 3 supported file formats for the <audio> element: MP3, WAV, and OGG:

Browser	MP3	WAV	OGG
Internet Explorer	YES	NO	NO
Chrome	YES	YES	YES
Firefox	YES	YES	YES
Safari	YES	YES	NO
Opera	YES	YES	YES

MIME Types for Audio Formats

Format	MIME-type
MP3	audio/mpeg
OGG	audio/ogg
WAV	audio/wav

Browser Support

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

Element					
<audio>	4.0	9.0	3.5	4.0	10.5

Differences Between HTML 4.01 and HTML5

The <audio> tag is new in HTML5.

Tips and Notes

Tip: Any text inside the between <audio> and </audio> will be displayed in browsers that do not support the <audio> tag.

Attributes

Attribute	Value	Description
<u>autoplay</u>	autoplay	Specifies that the audio will start playing as soon as it is ready
<u>controls</u>	controls	Specifies that audio controls should be displayed (such as a play/pause button etc)
<u>loop</u>	loop	Specifies that the audio will start over again, every time it is finished
<u>muted</u>	muted	Specifies that the audio output should be muted
<u>preload</u>	auto metadata none	Specifies if and how the author thinks the audio should be loaded when the page loads
<u>src</u>	URL	Specifies the URL of the audio file

Global Attributes

The <audio> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <audio> tag also supports the [Event Attributes in HTML](#).

HTML Tag

Example

```
<p>This is normal text - <b>and this is bold text</b>.</p>
```

Definition and Usage

The tag specifies bold text.

Browser Support

Element					
	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Tips and Notes

Note: According to the HTML 5 specification, the tag should be used as a LAST resort when no other tag is more appropriate. The HTML 5 specification states that headings should be denoted with the <h1> to <h6> tags, emphasized text should be denoted with the tag, important text should be denoted with the tag, and marked/highlighted text should use the <mark> tag.

Tip: You can also use the CSS "font-weight" property to set bold text.

Global Attributes

The tag also supports the [Global Attributes in HTML](#).

Event Attributes

The tag also supports the [Event Attributes in HTML](#).

HTML <base> Tag

Example

Specify a default URL and a default target for all links on a page:

```
<head>
  <base href="https://www.w3schools.com/images/" target="_blank">
</head>

<body>

<a href="https://www.w3schools.com">W3Schools</a>
</body>
```

Definition and Usage

The <base> tag specifies the base URL/target for all relative URLs in a document.

There can be at maximum one <base> element in a document, and it must be inside the <head> element.

Browser Support

Element					
<base>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: If the <base> tag is present, it must have either an href attribute or a target attribute, or both.

Differences Between HTML 4.01 and HTML5

NONE.

Differences Between HTML and XHTML

In HTML the <base> tag has no end tag.

In XHTML the <base> tag must be properly closed.

Attributes

Attribute	Value	Description
<u>href</u>	<i>URL</i>	Specifies the base URL for all relative URLs in the page
<u>target</u>	_blank _parent _self _top <i>framename</i>	Specifies the default target for all hyperlinks and forms in the page

Global Attributes and Events

The <base> tag also supports the [Global Attributes in HTML](#).

Event Attributes

NONE. The <base> tag does not support any event attributes.

HTML <basefont> Tag. Not Supported in HTML5.

Example

Specify a default text-color and font-size for text on page:

```
<head>
<basefont color="red" size="5">
</head>

<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
```

Definition and Usage

The <basefont> tag is not supported in HTML5. Use CSS instead.

The <basefont> tag specifies a default text-color, font-size, or font-family for all the text in a document.

Browser Support

Element					
<basefont>	Not supported	Not supported	Not supported	Not supported	Not supported

Note: The <basefont> tag is supported in Internet Explorer 9, and earlier versions.

Differences Between HTML 4.01 and HTML5

The <basefont> tag is not supported in HTML5.

Compatibility Notes

[CSS Example: Specify a default text-color for a page](#)

[CSS Example: Specify a default font-family for a page](#)

[CSS Example: Specify a default font-size for a page](#)

In our CSS tutorial you can find more information about [CSS text color](#) and [CSS fonts](#).

Optional Attributes

Attribute	Value	Description
<u>color</u>	<i>color</i>	Not supported in HTML5. Specifies the default color for text in a document
<u>face</u>	<i>font_family</i>	Not supported in HTML5. Specifies the default font for text in a document
<u>size</u>	<i>number</i>	Not supported in HTML5. Specifies the default size of text in a document

HTML <bdi> Tag

Example

Isolate the usernames from the surrounding text-direction settings:

```
<ul>
  <li>User <bdi>hrefs</bdi>: 60 points</li>
  <li>User <bdi>jdoe</bdi>: 80 points</li>
  <li>User <bdi>إيهان</bdi>: 90 points</li>
</ul>
```

Definition and Usage

BDI stands for Bi-Directional Isolation.

The <bdi> tag isolates a part of text that might be formatted in a different direction from other text outside it.

This element is useful when embedding user-generated content with an unknown directionality.

Browser Support

Element					
<bdi>	Yes	Not supported	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The <bdi> tag is new in HTML5.

Global Attributes

The <bdi> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <bdi> tag also supports the [Event Attributes in HTML](#).

HTML <bdo> Tag

Example

Specify the text direction:

```
<bdo dir="rtl">  
This text will go right-to-left.  
</bdo>
```

Definition and Usage

bdo stands for Bi-Directional Override.

The <bdo> tag is used to override the current text direction.

Browser Support

Element					
<bdo>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Attributes

Attribute	Value	Description
dir	ltr rtl	Required. Specifies the text direction of the text inside the <bdo> element

Global Attributes

The <bdo> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <bdo> tag also supports the [Event Attributes in HTML](#).

HTML <big> Tag. Not Supported in HTML5.

Example

Make text bigger than normal:

```
<p><big>Bigger text</big></p>
```

Definition and Usage

The <big> tag is not supported in HTML5. Use CSS instead.

The <big> tag defines bigger text.

Browser Support

Element					
<big>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The <big> tag is not supported in HTML5.

Compatibility Notes

[CSS Example: Specify different font-sizes for HTML elements](#)

In our CSS tutorial you can find more information about [CSS fonts](#).

HTML <blockquote> Tag

Example

A section that is quoted from another source:

```
<blockquote cite="http://www.worldwildlife.org/who/index.html">  
For 50 years, WWF has been protecting the future of nature. The world's  
leading conservation organization, WWF works in 100 countries and is  
supported by 1.2 million members in the United States and close to 5  
million globally.  
</blockquote>
```

Definition and Usage

The <blockquote> tag specifies a section that is quoted from another source. Browsers usually indent <blockquote> elements.

Browser Support

Element					
<blockquote>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: Use <q> for inline (short) quotations.

Differences Between HTML 4.01 and HTML5

In HTML 4.01, the <blockquote> tag defines a long quotation.

In HTML5, the <blockquote> tag specifies a section that is quoted from another source.

Differences Between HTML and XHTML

Note: To validate a <blockquote> element as XHTML, it must contain only other block-level elements, like this:

```
<blockquote> <p>Here is a long quotation here is a long quotation.</p> </blockquote>
```

Attributes

Attribute	Value	Description
<u>cite</u>	<i>URL</i>	Specifies the source of the quotation

Global Attributes

The <blockquote> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <blockquote> tag also supports the [Event Attributes in HTML](#).

HTML <body> Tag

Example

A simple HTML document, with the minimum of required tags:

```
<html>
<head>
<title>Title of the document</title>
</head>

<body>
The content of the document.....
</body>

</html>
```

Definition and Usage

The <body> tag defines the document's body.

The <body> element contains all the contents of an HTML document, such as text, hyperlinks, images, tables, lists, etc.

Browser Support

Element					
<body>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

All layout attributes are removed in HTML5.

Attributes

Attribute	Value	Description
<u>alink</u>	<i>color</i>	Not supported in HTML5. Specifies the color of an active link in a document
<u>background</u>	<i>URL</i>	Not supported in HTML5. Specifies a background image for a document
<u>bgcolor</u>	<i>color</i>	Not supported in HTML5. Specifies the background color of a document
<u>link</u>	<i>color</i>	Not supported in HTML5. Specifies the color of unvisited links in a document
<u>text</u>	<i>color</i>	Not supported in HTML5. Specifies the color of the text in a document
<u>vlink</u>	<i>color</i>	Not supported in HTML5. Specifies the color of visited links in a document

Global Attributes

The <body> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <body> tag also supports the [Event Attributes in HTML](#).

HTML
 Tag

Example

A line break is marked up as follows:

This text contains
a line break.

Definition and Usage

The
 tag inserts a single line break.

The
 tag is an empty tag which means that it has no end tag.

Browser Support

Element					
 	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: The
 tag is useful for writing addresses or poems.

Note: Use the
 tag to enter line breaks, not to separate paragraphs.

Differences Between HTML 4.01 and HTML5

NONE.

Differences Between HTML and XHTML

In HTML, the
 tag has no end tag.

In XHTML, the
 tag must be properly closed, like this:
.

Global Attributes

The
 tag also supports the [Global Attributes in HTML](#).

Event Attributes

The
 tag also supports the [Event Attributes in HTML](#).

HTML <button> Tag

Example

A clickable button is marked up as follows:

```
<button type="button">Click Me!</button>
```

Definition and Usage

The <button> tag defines a clickable button.

Inside a <button> element you can put content, like text or images. This is the difference between this element and buttons created with the <input> element.

Tip: Always specify the type attribute for a <button> element. Different browsers use different default types for the <button> element.

Tip: Buttons are often styled with CSS: [Try it »](#)

Browser Support

Element					
<button>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: If you use the <button> element in an HTML form, different browsers may submit different values. Use <input> to create buttons in an HTML form.

Tip: Visit our [CSS Buttons Tutorial](#) to find out how to style buttons with CSS.

Differences Between HTML 4.01 and HTML5

HTML5 has the following new attributes: autofocus, form, formaction, formenctype, formmethod, formnovalidate, and formtarget.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>autofocus</u>	autofocus	Specifies that a button should automatically get focus when the page loads
<u>disabled</u>	disabled	Specifies that a button should be disabled
<u>form</u>	<i>form_id</i>	Specifies one or more forms the button belongs to
<u>formaction</u>	<i>URL</i>	Specifies where to send the form-data when a form is submitted. Only for type="submit"
<u>formenctype</u>	application/x-www-form-urlencoded multipart/form-data text/plain	Specifies how form-data should be encoded before sending it to a server. Only for type="submit"
<u>formmethod</u>	getpost	Specifies how to send the form-data (which HTTP method to use). Only for type="submit"
<u>formnovalidate</u>	formnovalidate	Specifies that the form-data should not be validated on submission. Only for type="submit"
<u>formtarget</u>	_blank _self _parent _top <i>framename</i>	Specifies where to display the response after submitting the form. Only for type="submit"
<u>name</u>	<i>name</i>	Specifies a name for the button
<u>type</u>	button reset submit	Specifies the type of button
<u>value</u>	<i>text</i>	Specifies an initial value for the button

Global Attributes

The <button> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <button> tag also supports the [Event Attributes in HTML](#).

HTML <canvas> Tag

Example

Draw a red square, on the fly, and show it inside the <canvas> element:

```
<canvas id="myCanvas"></canvas>

<script>
var canvas = document.getElementById("myCanvas");
var ctx = canvas.getContext("2d");
ctx.fillStyle = "#FF0000";
ctx.fillRect(0, 0, 80, 80);
</script>
```

Definition and Usage

The <canvas> tag is used to draw graphics, on the fly, via scripting (usually JavaScript).
The <canvas> tag is only a container for graphics, you must use a script to actually draw the graphics.

Browser Support

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

Element					
<canvas>	4.0	9.0	2.0	3.1	9.0

Differences Between HTML 4.01 and HTML5

The <canvas> tag is new in HTML5.

Tips and Notes

Note: Any text inside the <canvas> element will be displayed in browsers that do not support <canvas>.

Tip: Learn more about the <canvas> element in our [HTML Canvas Tutorial](#).

Tip: For a complete reference of all the properties and methods that can be used with the canvas object, go to our [HTML Canvas Reference](#).

Attributes

Attribute	Value	Description
<u>height</u>	<i>pixels</i>	Specifies the height of the canvas
<u>width</u>	<i>pixels</i>	Specifies the width of the canvas

Global Attributes

The <canvas> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <canvas> tag also supports the [Event Attributes in HTML](#).

Default CSS Settings

Most browsers will display the <canvas> element with the following default values:

Example

```
canvas {  
    height: 150px;  
    width: 300px;  
}
```

HTML <caption> Tag

Example

A table with a caption:

```
<table>
  <caption>Monthly savings</caption>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
</table>
```

Definition and Usage

The <caption> tag defines a table caption.

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

Note: You can specify only one caption per table.

Tip: By default, a table caption will be center-aligned above a table. However, the CSS properties text-align and caption-side can be used to align and place the caption.

Browser Support

Element					
<caption>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The align attribute is removed from HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	left right top bottom	Not supported in HTML5. Defines the alignment of the caption

Global Attributes

The <caption> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <caption> tag also supports the [Event Attributes in HTML](#).

HTML <center> Tag. Not Supported in HTML5.

Example

Center-align text in an HTML page:

```
<center>This text will be center-aligned.</center>
```

Definition and Usage

The <center> tag is not supported in HTML5. Use CSS instead.

The <center> tag is used to center-align text.

Browser Support

Element					
<center>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The <center> tag is not supported in HTML5.

Compatibility Notes

[CSS Example: Center-align text](#)

In our CSS tutorial you can find more information about [CSS text alignment](#).

HTML <cite> Tag

Example

Define the title of a work with the <cite> tag:

```
<p><cite>The Scream</cite> by Edward Munch. Painted in 1893.</p>
```

Definition and Usage

The <cite> tag defines the title of a work (e.g. a book, a song, a movie, a TV show, a painting, a sculpture, etc.).

Note: A person's name is not the title of a work.

Browser Support

Element					
<cite>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

In HTML5, the <cite> tag defines the title of a work.

In HTML 4.01, the <cite> tag defines a citation.

Global Attributes

The <cite> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <cite> tag also supports the [Event Attributes in HTML](#).

HTML <code> Tag

Example

Format text in a document:

```
<code>A piece of computer code</code>
```

Definition and Usage

The <code> tag is a phrase tag. It defines a piece of computer code.

Tip: This tag is not deprecated, but it is possible to achieve richer effect with CSS.

All phrase tags:

Tag	Description
<u></u>	Renders as emphasized text
<u></u>	Defines important text
<u><code></u>	Defines a piece of computer code
<u><samp></u>	Defines sample output from a computer program
<u><kbd></u>	Defines keyboard input
<u><var></u>	Defines a variable

Browser Support

Element					
<code>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The <code> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <code> tag also supports the [Event Attributes in HTML](#).

HTML <col> Tag

Example

Set the background color of the three columns with the <colgroup> and <col> tags:

```
<table>
  <colgroup>
    <col span="2" style="background-color:red">
    <col style="background-color:yellow">
  </colgroup>
  <tr>
    <th>ISBN</th>
    <th>Title</th>
    <th>Price</th>
  </tr>
  <tr>
    <td>3476896</td>
    <td>My first HTML</td>
    <td>$53</td>
  </tr>
</table>
```

Definition and Usage

The <col> tag specifies column properties for each column within a <colgroup> element. The <col> tag is useful for applying styles to entire columns, instead of repeating the styles for each cell, for each row.

Browser Support

Element					
<col>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

Most of the attributes in HTML 4.01 are not supported in HTML5.

Differences Between HTML and XHTML

In HTML the `<col>` tag has no end tag.

In XHTML, the `<col>` tag must be properly closed.

Attributes

Attribute	Value	Description
<u>align</u>	left right center justify char	Not supported in HTML5. Specifies the alignment of the content related to a <code><col></code> element
<u>char</u>	<i>character</i>	Not supported in HTML5. Specifies the alignment of the content related to a <code><col></code> element to a character
<u>charoff</u>	<i>number</i>	Not supported in HTML5. Specifies the number of characters the content will be aligned from the character specified by the <code>char</code> attribute
<u>span</u>	<i>number</i>	Specifies the number of columns a <code><col></code> element should span
<u>valign</u>	top middle bottom baseline	Not supported in HTML5. Specifies the vertical alignment of the content related to a <code><col></code> element
<u>width</u>	<i>% pixels</i> <i>relative_length</i>	Not supported in HTML5. Specifies the width of a <code><col></code> element

Global Attributes

The `<col>` tag also supports the [Global Attributes in HTML](#).

Event Attributes

The `<col>` tag also supports the [Event Attributes in HTML](#).

HTML <colgroup> Tag

Example

Set the background color of the three columns with the <colgroup> and <col> tags:

```
<table>
  <colgroup>
    <col span="2" style="background-color:red">
    <col style="background-color:yellow">
  </colgroup>
  <tr>
    <th>ISBN</th>
    <th>Title</th>
    <th>Price</th>
  </tr>
  <tr>
    <td>3476896</td>
    <td>My first HTML</td>
    <td>$53</td>
  </tr>
</table>
```

Definition and Usage

The <colgroup> tag specifies a group of one or more columns in a table for formatting.

The <colgroup> tag is useful for applying styles to entire columns, instead of repeating the styles for each cell, for each row.

Note: The <colgroup> tag must be a child of a <table> element, after any <caption> elements and before any <thead>, <tbody>, <tfoot>, and <tr> elements.

Tip: To define different properties to a column within a <colgroup>, use the <col> tag within the <colgroup> tag.

Browser Support

Element					
<colgroup>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

Most of the attributes in HTML 4.01 are not supported in HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	left right center justify char	Not supported in HTML5. Aligns the content in a column group
<u>char</u>	<i>character</i>	Not supported in HTML5. Aligns the content in a column group to a character
<u>charoff</u>	<i>number</i>	Not supported in HTML5. Sets the number of characters the content will be aligned from the character specified by the char attribute
<u>span</u>	<i>number</i>	Specifies the number of columns a column group should span
<u>valign</u>	top middle bottom baseline	Not supported in HTML5. Vertical aligns the content in a column group
<u>width</u>	<i>pixels % relative_length</i>	Not supported in HTML5. Specifies the width of a column group

Global Attributes

The <colgroup> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <colgroup> tag also supports the [Event Attributes in HTML](#).

HTML <data> Tag

Example

The following example displays product names but also associates each name with a product number:

```
<ul>
  <li><data value="21053">Cherry Tomato</data></li>
  <li><data value="21054">Beef Tomato</data></li>
  <li><data value="21055">Snack Tomato</data></li>
</ul>
```

Definition and Usage

The <data> tag links the given content with a machine-readable translation.

This element provides both a machine-readable value for data processors, and a human-readable value for rendering in a browser.

Tip: If the content is time- or date-related, the <time> element must be used instead.

Browser Support

Element					
<data>	62.0	Yes	22.0	Not supported	49.0

Differences Between HTML 4.01 and HTML5

The <data> tag is new in HTML5.

Attributes

Attribute	Value	Description
value	<i>machine-readable format</i>	Specifies the machine-readable translation of the content of the element

Global Attributes

The <data> tag also supports the [Global Attributes in HTML](#).

HTML <datalist> Tag

Example

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

```
<input list="browsers">

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

Definition and Usage

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

The <datalist> tag is used to provide an "autocomplete" feature on <input> elements. Users will see a drop-down list of pre-defined options as they input data.

Use the <input> element's list attribute to bind it together with a <datalist> element.

Browser Support

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

Element					
<datalist>	20.0	10.0	4.0	Not supported	9.0

Differences Between HTML 4.01 and HTML5

The <datalist> tag is new in HTML5.

Global Attributes

The <datalist> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <datalist> tag also supports the [Event Attributes in HTML](#).

HTML <dd> Tag

Example

A description list, with terms and descriptions:

```
<dl>
  <dt>Coffee</dt>
  <dd>Black hot drink</dd>
  <dt>Milk</dt>
  <dd>White cold drink</dd>
</dl>
```

Definition and Usage

The <dd> tag is used to describe a term/name in a description list.

The <dd> tag is used in conjunction with <dl> (defines a description list) and <dt> (defines terms/names).

Inside a <dd> tag you can put paragraphs, line breaks, images, links, lists, etc.

Browser Support

Element					
<dd>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

In HTML 4.01, the <dd> tag was used to describe an item in a definition list.

In HTML5, the <dd> tag is used to describe a term/name in a description list.

Global Attributes

The <dd> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <dd> tag also supports the [Event Attributes in HTML](#).

HTML Tag

Example

A text with a deleted part, and a new, inserted part:

```
<p>My favorite color is <del>blue</del> <ins>red</ins>!</p>
```

Definition and Usage

The tag defines text that has been deleted from a document.

Browser Support

Element					
	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: Also look at the <ins> tag to markup inserted text.

Tip: Use and <ins> to markup updates and modifications in a document. Browsers will normally strike a line through deleted text and underline inserted text.

Differences Between HTML 4.01 and HTML5

NONE.

Attributes

Attribute	Value	Description
<u>cite</u>	<i>URL</i>	Specifies a URL to a document that explains the reason why the text was deleted
<u>datetime</u>	<i>YYYY-MM-DDThh:mm:ssTZD</i>	Specifies the date and time of when the text was deleted

Global Attributes

The tag also supports the Global Attributes in HTML.

Event Attributes

The `` tag also supports the [Event Attributes in HTML](#).

HTML <details> Tag

Example

Using the <details> element:

```
<details>
  <summary>Copyright 1999-2018.</summary>
  <p> - by Refsnes Data. All Rights Reserved.</p>
  <p>All content and graphics on this web site are the property of the
company Refsnes Data.</p>
</details>
```

Definition and Usage

The <details> tag specifies additional details that the user can view or hide on demand. The <details> tag can be used to create an interactive widget that the user can open and close. Any sort of content can be put inside the <details> tag.

The content of a <details> element should not be visible unless the open attribute is set.

Browser Support

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

Element					
<details>	12.0	Not supported	49.0	6.0	15.0

Differences Between HTML 4.01 and HTML5

The <details> tag is new in HTML5.

Tips and Notes

Tip: The <summary> tag is used to specify a visible heading for the details. The heading can be clicked to view/hide the details.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>open</u>	open	Specifies that the details should be visible (open) to the user

Global Attributes

The <details> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <details> tag also supports the [Event Attributes in HTML](#).

HTML <dfn> Tag

Example

Mark up the defining instance of a term:

```
<p><dfn>HTML</dfn> is the standard markup language for creating web pages.</p>
```

Definition and Usage

The <dfn> tag represents the defining instance of a term in HTML.

The defining instance is often the first use of a term in a document.

The nearest parent of the <dfn> tag must also contain the definition/explanation for the term inside <dfn>.

The term inside the <dfn> tag can be any of the following:

1. The content of the <dfn> element (without a title attribute):

Example

```
<p><dfn>HTML</dfn> is the standard markup language for creating web pages.</p>
```

2. The title attribute of the <dfn> tag:

Example

```
<p><dfn title="HyperText Markup Language">HTML</dfn> is the standard markup language for creating web pages.</p>
```

3. The title attribute of an <abbr> tag inside the <dfn> element:

Example

```
<p><dfn><abbr title="HyperText Markup Language">HTML</abbr></dfn> is the standard markup language for creating web pages.</p>
```

You can also add an id attribute to the `<dfn>` element. Then, whenever a term is used, it can refer back to the definition using an `<a>` tag:

Example

```
<p><dfn id="html-def">HTML</dfn> is the standard markup language for  
creating web pages.</p>  
  
<p>This is some text...</p>  
<p>This is some text...</p>  
<p>Learn <a href="#html-def">HTML</a> now.</p>
```

Browser Support

Element					
<code><dfn></code>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The `<dfn>` tag also supports the [Global Attributes in HTML](#).

Event Attributes

The `<dfn>` tag also supports the [Event Attributes in HTML](#).

HTML <dialog> Tag

Example

Using the <dialog> element:

```
<table>
<tr>
  <th>January <dialog open>This is an open dialog window</dialog></th>
  <th>February</th>
  <th>March</th>
</tr>
<tr>
  <td>31</td>
  <td>28</td>
  <td>31</td>
</tr>
</table>
```

Definition and Usage

The <dialog> tag defines a dialog box or window.

The <dialog> element makes it easy to create popup dialogs and modals on a web page.

Browser Support

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

Element					
<dialog>	37.0	Not supported	59.0	6.0	24.0

Differences Between HTML 4.01 and HTML5

The <dialog> tag is new in HTML5.

Attributes

Attribute	Value	Description
<u>open</u>	open	Specifies that the dialog element is active and that the user can interact with it

Global Attributes

The <dialog> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <dialog> tag also supports the [Event Attributes in HTML](#).

HTML <dir> Tag. Not Supported in HTML5.

Example

A directory list:

```
<dir>
  <li>html</li>
  <li>xhtml</li>
  <li>css</li>
</dir>
```

Definition and Usage

The <dir> tag is not supported in HTML5. Use [](#) or CSS instead.

The <dir> tag is used to list directory titles.

Browser Support

Element					
<dir>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: Use CSS to style lists! In our CSS tutorial you can find more details about [styling lists](#).

Differences Between HTML 4.01 and HTML5

The <dir> tag is not supported in HTML5.

Optional Attributes

Attribute	Value	Description
<u>compact</u>	compact	Not supported in HTML5. Specifies that the list should render smaller than normal

HTML <div> Tag

Example

A section in a document that will have a light blue background color:

```
<div style="background-color:lightblue">
  <h3>This is a heading</h3>
  <p>This is a paragraph.</p>
</div>
```

Definition and Usage

The <div> tag defines a division or a section in an HTML document.

The <div> element is often used as a container for other HTML elements to style them with CSS or to perform certain tasks with JavaScript.

Browser Support

Element					
<div>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: The <div> element is very often used together with CSS, to layout a web page.

Note: By default, browsers always place a line break before and after the <div> element. However, this can be changed with CSS.

Differences Between HTML 4.01 and HTML5

The align attribute not supported in HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	left right center justify	Not supported in HTML5. Specifies the alignment of the content inside a <div> element

Global Attributes

The <div> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <div> tag also supports the [Event Attributes in HTML](#).

HTML <dl> Tag

Example

A description list, with terms and descriptions:

```
<dl>
  <dt>Coffee</dt>
  <dd>Black hot drink</dd>
  <dt>Milk</dt>
  <dd>White cold drink</dd>
</dl>
```

Definition and Usage

The <dl> tag defines a description list.

The <dl> tag is used in conjunction with <dt> (defines terms/names) and <dd> (describes each term/name).

Browser Support

Element					
<dl>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

In HTML 4.01, the <dl> tag defines a definition list.

In HTML 5, the <dl> tag defines a description list.

Global Attributes

The <dl> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <dl> tag also supports the [Event Attributes in HTML](#).

HTML <dt> Tag

Example

A description list, with terms and descriptions:

```
<dl>
  <dt>Coffee</dt>
  <dd>Black hot drink</dd>
  <dt>Milk</dt>
  <dd>White cold drink</dd>
</dl>
```

Definition and Usage

The <dt> tag defines a term/name in a description list.

The <dt> tag is used in conjunction with <dl> (defines a description list) and <dd> (describes each term/name).

Browser Support

Element					
<dt>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

In HTML 4.01, the <dt> tag defines an item in a definition list.

In HTML5, the <dt> tag defines a term/name in a description list.

Global Attributes

The <dt> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <dt> tag also supports the [Event Attributes in HTML](#).

HTML Tag

Example

Format text in a document:

```
<em>Emphasized text</em>
```

Definition and Usage

The tag is a phrase tag. It renders as emphasized text.

Tip: This tag is not deprecated, but it is possible to achieve richer effect with CSS.

All phrase tags:

Tag	Description
<u></u>	Renders as emphasized text
<u></u>	Defines important text
<u><code></u>	Defines a piece of computer code
<u><samp></u>	Defines sample output from a computer program
<u><kbd></u>	Defines keyboard input
<u><var></u>	Defines a variable

Browser Support

Element					
	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The tag also supports the [Global Attributes in HTML](#).

Event Attributes

The tag also supports the [Event Attributes in HTML](#).

HTML <embed> Tag

Example

An embedded flash animation:

```
<embed src="helloworld.swf">
```

Definition and Usage

The <embed> tag defines a container for an external application or interactive content (a plug-in).

Browser Support

Element					
<embed>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The <embed> tag is new in HTML5.

Note: Many web browsers have supported the <embed> tag for a long time. However, the <embed> tag has not been a part of the HTML 4 specification. The <embed> tag is new in HTML5, and will validate in an HTML5 page. However, if you use it in an HTML 4 page, the page will not validate.

Attributes

Attribute	Value	Description
<u>height</u>	<i>pixels</i>	Specifies the height of the embedded content
<u>src</u>	<i>URL</i>	Specifies the address of the external file to embed
<u>type</u>	<i>media_type</i>	Specifies the media type of the embedded content
<u>width</u>	<i>pixels</i>	Specifies the width of the embedded content

Global Attributes

The <embed> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <embed> tag also supports the [Event Attributes in HTML](#).

HTML <fieldset> Tag

Example

Group related elements in a form:

```
<form>
  <fieldset>
    <legend>Personalia:</legend>
    Name: <input type="text"><br>
    Email: <input type="text"><br>
    Date of birth: <input type="text">
  </fieldset>
</form>
```

Definition and Usage

The <fieldset> tag is used to group related elements in a form.

The <fieldset> tag draws a box around the related elements.

Browser Support

Element					
<fieldset>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

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

Differences Between HTML 4.01 and HTML5

HTML5 has added new attributes for <fieldset>.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>disabled</u>	disabled	Specifies that a group of related form elements should be disabled
<u>form</u>	<i>form_id</i>	Specifies one or more forms the fieldset belongs to
<u>name</u>	<i>text</i>	Specifies a name for the fieldset

Global Attributes

The <fieldset> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <fieldset> tag also supports the [Event Attributes in HTML](#).

HTML <figcaption> Tag

Example

Use a <figure> element to mark up a photo in a document.

The <figure> element can also contain a <figcaption>:

```
<figure>
  
  <figcaption>Fig.1 - Trulli, Puglia, Italy.</figcaption>
</figure>
```

Definition and Usage

The <figcaption> tag defines a caption for a <figure> element.

The <figcaption> element can be placed as the first or last child of the <figure> element.

Browser Support

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

Element					
<figcaption>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <figcaption> tag is new in HTML5.

Global Attributes

The <figcaption> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <figcaption> tag also supports the [Event Attributes in HTML](#).

HTML <figure> Tag

Example

Use a <figure> element to mark up a photo in a document:

```
<figure>
  
</figure>
```

Definition and Usage

The <figure> tag specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.

While the content of the <figure> element is related to the main flow, its position is independent of the main flow, and if removed it should not affect the flow of the document.

Browser Support

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

Element					
<figure>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <figure> tag is new in HTML5.

Tips and Notes

Tip: The <figcaption> element is used to add a caption for the <figure> element.

Global Attributes

The <figure> tag also supports the Global Attributes in HTML.

Event Attributes

The <figure> tag also supports the Event Attributes in HTML.

HTML Tag. Not Supported in HTML5.

Example

Specify the font size, font face and color of text:

```
<font size="3" color="red">This is some text!</font>
<font size="2" color="blue">This is some text!</font>
<font face="verdana" color="green">This is some text!</font>
```

Definition and Usage

The tag is not supported in HTML5. Use CSS instead.

The tag specifies the font face, font size, and color of text.

Browser Support

Element					
	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The tag is not supported in HTML5.

Compatibility Notes

[CSS Example: Set the color of text](#)

[CSS Example: Set the font of text](#)

[CSS Example: Set the size of text](#)

In our CSS tutorial you can find more information about [CSS text color](#) and [CSS fonts](#).

Optional Attributes

Attribute	Value	Description
<u>color</u>	<i>rgb(x,x,x)</i> <i>#xxxxxx</i> <i>colorname</i>	Not supported in HTML5. Specifies the color of text
<u>face</u>	<i>font_family</i>	Not supported in HTML5. Specifies the font of text
<u>size</u>	<i>number</i>	Not supported in HTML5. Specifies the size of text

HTML <footer> Tag

Example

A footer section in a document:

```
<footer>
  <p>Posted by: Hege Refsnes</p>
  <p>Contact information: <a href="mailto:someone@example.com">
    someone@example.com</a>.</p>
</footer>
```

Definition and Usage

The <footer> tag defines a footer for a document or section.

A <footer> element should contain information about its containing element.

A <footer> element typically contains:

- authorship information
- copyright information
- contact information
- sitemap
- back to top links
- related documents

You can have several <footer> elements in one document.

Browser Support

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

Element					
<footer>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <footer> tag is new in HTML5.

Tips and Notes

Tip: Contact information inside a <footer> element should go inside an <address> tag.

Global Attributes

The <footer> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <footer> tag also supports the [Event Attributes in HTML](#).

HTML <form> Tag

Example

An HTML form with two input fields and one submit button:

```
<form action="/action_page.php" method="get">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit" value="Submit">
</form>
```

More "Try it Yourself" examples below.

Definition and Usage

The <form> tag is used to create an HTML form for user input.

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

- <input>
- <textarea>
- <button>
- <select>
- <option>
- <optgroup>
- <fieldset>
- <label>

Browser Support

Element					
<form>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

HTML5 has added two new attributes: autocomplete and novalidate, and removed the accept attribute.

Differences Between HTML and XHTML

In XHTML, the name attribute is deprecated. Use the global id attribute instead.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>accept</u>	<i>file_type</i>	Not supported in HTML5. Specifies a comma-separated list of file types that the server accepts (that can be submitted through the file upload)
<u>accept-charset</u>	<i>character_set</i>	Specifies the character encodings that are to be used for the form submission
<u>action</u>	<i>URL</i>	Specifies where to send the form-data when a form is submitted
<u>autocomplete</u>	onoff	Specifies whether a form should have autocomplete on or off
<u>enctype</u>	application/x-www-form-urlencoded multipart/form-data text/plain	Specifies how the form-data should be encoded when submitting it to the server (only for method="post")
<u>method</u>	get post	Specifies the HTTP method to use when sending form-data
<u>name</u>	<i>text</i>	Specifies the name of a form
<u>novalidate</u>	novalidate	Specifies that the form should not be validated when submitted
<u>target</u>	_blank _self _parent _top	Specifies where to display the response that is received after submitting the form

Global Attributes

The <form> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <form> tag also supports the [Event Attributes in HTML](#).

Try it Yourself - Examples

Form with checkboxes A form with two checkboxes, and a submit button.

Form with radiobuttons A form with two radio buttons, and a submit button.

HTML <frame> Tag. Not Supported in HTML5.

Example

A simple three-framed page:

```
<frameset cols="25%,50%,25%">
  <frame src="frame_a.htm">
  <frame src="frame_b.htm">
  <frame src="frame_c.htm">
</frameset>
```

More "Try it Yourself" examples below.

Definition and Usage

The <frame> tag is not supported in HTML5.

The <frame> tag defines one particular window (frame) within a <frameset>.

Each <frame> in a <frameset> can have different attributes, such as border, scrolling, the ability to resize, etc.

Note: If you want to validate a page containing frames, be sure the <!DOCTYPE> is set to either "HTML Frameset DTD" or "XHTML Frameset DTD".

Browser Support

Element					
<frame>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The <frame> tag is not supported in HTML5.

Differences Between HTML and XHTML

In HTML, the <frame> tag has no end tag. In XHTML, the <frame> tag must be properly closed.

Optional Attributes

Attribute	Value	Description
<u>frameborder</u>	0 1	Not supported in HTML5. Specifies whether or not to display a border around a frame
<u>longdesc</u>	URL	Not supported in HTML5. Specifies a page that contains a long description of the content of a frame
<u>marginheight</u>	<i>pixels</i>	Not supported in HTML5. Specifies the top and bottom margins of a frame
<u>marginwidth</u>	<i>pixels</i>	Not supported in HTML5. Specifies the left and right margins of a frame
<u>name</u>	<i>text</i>	Not supported in HTML5. Specifies the name of a frame
<u>noresize</u>	noresize	Not supported in HTML5. Specifies that a frame is not resizable
<u>scrolling</u>	yes no auto	Not supported in HTML5. Specifies whether or not to display scrollbars in a frame
<u>src</u>	URL	Not supported in HTML5. Specifies the URL of the document to show in a frame

Try it Yourself - Examples

[Horizontal frameset](#) How to make a horizontal frameset with three different documents.

[Mixed frameset](#) How to make a frameset with three documents, and how to mix them in rows and columns.

[Frameset with noresize="noresize"](#) How to use the "noresize" attribute. The border between frame A and B is not resizable.

HTML <frameset> Tag. Not Supported in HTML5.

Example

A simple three-framed page:

```
<frameset cols="25%,*,25%">
  <frame src="frame_a.htm">
  <frame src="frame_b.htm">
  <frame src="frame_c.htm">
</frameset>
```

More "Try it Yourself" examples below.

Definition and Usage

The <frameset> tag is not supported in HTML5.

The <frameset> tag defines a frameset.

The <frameset> element holds one or more <frame> elements. Each <frame> element can hold a separate document.

The <frameset> element specifies HOW MANY columns or rows there will be in the frameset, and HOW MUCH percentage/pixels of space will occupy each of them.

Note: If you want to validate a page containing frames, be sure the <!DOCTYPE> is set to either "HTML Frameset DTD" or "XHTML Frameset DTD".

Browser Support

Element					
<frameset>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The <frameset> tag is not supported in HTML5.

Differences Between HTML and XHTML

NONE.

Optional Attributes

Attribute	Value	Description
<u>cols</u>	<i>pixels % *</i>	Not supported in HTML5. Specifies the number and size of columns in a frameset
<u>rows</u>	<i>pixels % *</i>	Not supported in HTML5. Specifies the number and size of rows in a frameset

Try it Yourself - Examples

[Horizontal frameset](#) How to make a horizontal frameset with three different documents.

[Mixed frameset](#) How to make a frameset with three documents, and how to mix them in rows and columns.

[Frameset with noresize="noresize"](#) How to use the "noresize" attribute. The frames are not resizable. Move the mouse over the borders between the frames and notice that you can not move the borders.

HTML <h1> to <h6> Tags

Example

The six different HTML headings:

```
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>
```

Definition and Usage

The <h1> to <h6> tags are used to define HTML headings.

<h1> defines the most important heading. <h6> defines the least important heading.

Browser Support

Element					
<h1> - <h6>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The "align" attribute is not supported in HTML5. Use CSS to align elements.

Attributes

Attribute	Value	Description
<u>align</u>	left center right justify	Not supported in HTML5. Specifies the alignment of a heading

Global Attributes

The <h1> to <h6> tags also supports the [Global Attributes in HTML](#).

Event Attributes

The <h1> to <h6> tags also supports the [Event Attributes in HTML](#).

HTML <head> Tag

Example

An HTML document, with a <title> tag inside the head section:

```
<!DOCTYPE html>
<html>
<head>
  <title>Title of the document</title>
</head>

<body>
The content of the document.....
</body>

</html>
```

More "Try it Yourself" examples below.

Definition and Usage

The <head> element is a container for all the head elements.

The <head> element can include a title for the document, scripts, styles, meta information, and more.

The following elements can go inside the <head> element:

- <title> (this element is required in an HTML document)
- <style>
- <base>
- <link>
- <meta>
- <script>
- <noscript>

Browser Support

Element					
<head>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

In HTML 4.01 the `<head>` element is required.

In HTML5, the `<head>` element can be omitted. The following code will validate as HTML5:

Example

```
<!DOCTYPE html>
<html>
<title>Title of the document</title>

<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>

</html>
```

Attributes

Attribute	Value	Description
profile	URL	Not supported in HTML5. Specifies a URL to a document that contains a set of rules. The rules can be read by browsers to clearly understand the information in the <code><meta></code> tag's content attribute

Global Attributes

The `<head>` tag also supports the [Global Attributes in HTML](#).

Try it Yourself - Examples

[Use of the `<base>` tag in `<head>`](#) How to use the `<base>` tag to specify a default URL and a default target for all links on a page.

[Use of the `<style>` tag in `<head>`](#) How to add style information to the `<head>` section.

[Use of the `<link>` tag in `<head>`](#) How to use the `<link>` tag to link to an external style sheet.

HTML <header> Tag

Example

A header for an <article>:

```
<article>
  <header>
    <h1>Most important heading here</h1>
    <h3>Less important heading here</h3>
    <p>Some additional information here</p>
  </header>
  <p>Lorem Ipsum dolor set amet....</p>
</article>
```

Definition and Usage

The <header> element represents a container for introductory content or a set of navigational links.

A <header> element typically contains:

- one or more heading elements (<h1> - <h6>)
- logo or icon
- authorship information

You can have several <header> elements in one document.

Note: A <header> tag cannot be placed within a <footer>, <address> or another <header> element.

Browser Support

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

Element					
<header>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <header> tag is new in HTML5.

Global Attributes

The <header> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <header> tag also supports the [Event Attributes in HTML](#).

HTML <hr> Tag

Example

Use the <hr> tag to define a thematic change in the content:

```
<h1>HTML</h1>
<p>HTML is a language for describing web pages.....</p>

<hr>

<h1>CSS</h1>
<p>CSS defines how to display HTML elements.....</p>
```

Definition and Usage

The <hr> tag defines a thematic break in an HTML page (e.g. a shift of topic).

The <hr> element is used to separate content (or define a change) in an HTML page.

Browser Support

Element					
<hr>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

In HTML5, the <hr> tag defines a thematic break.

In HTML 4.01, the <hr> tag represents a horizontal rule.

However, the <hr> tag may still be displayed as a horizontal rule in visual browsers, but is now defined in semantic terms, rather than presentational terms.

All the layout attributes are removed in HTML5. Use CSS instead.

Differences Between HTML and XHTML

In HTML, the <hr> tag has no end tag.

In XHTML, the <hr> tag must be properly closed, like this: <hr />.

Attributes

Attribute	Value	Description
<u>align</u>	left center right	Not supported in HTML5. Specifies the alignment of a <hr> element
<u>noshade</u>	noshade	Not supported in HTML5. Specifies that a <hr> element should render in one solid color (noshaded), instead of a shaded color
<u>size</u>	<i>pixels</i>	Not supported in HTML5. Specifies the height of a <hr> element
<u>width</u>	<i>pixels %</i>	Not supported in HTML5. Specifies the width of a <hr> element

Global Attributes

The <hr> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <hr> tag also supports the [Event Attributes in HTML](#).

HTML <html> Tag

Example

A simple HTML5 document:

```
<!DOCTYPE HTML>
<html>
<head>
<title>Title of the document</title>
</head>

<body>
The content of the document.....
</body>

</html>
```

Definition and Usage

The <html> tag tells the browser that this is an HTML document.

The <html> tag represents the root of an HTML document.

The <html> tag is the container for all other HTML elements (except for the <!DOCTYPE> tag).

Browser Support

Element					
<html>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML and XHTML

The `xmlns` attribute is required in XHTML, but is invalid in HTML.

However, the HTML validator at <http://w3.org> does not complain when the `xmlns` attribute is missing in an XHTML document. This is because the namespace "xmlns=http://www.w3.org/1999/xhtml" is default, and will be added to the <html> tag even if you do not include it.

Attributes

Attribute	Value	Description
<u>xmlns</u>	http://www.w3.org/1999/xhtml	Specifies the XML namespace attribute (If you need your content to conform to XHTML)

Global Attributes

The <html> tag also supports the [Global Attributes in HTML](#).

HTML <i> Tag

Example

```
<p>He named his car <i>The lightning</i>, because it was very fast.</p>
```

Definition and Usage

The <i> tag defines a part of text in an alternate voice or mood. The content of the <i> tag is usually displayed in italic.

The <i> tag can be used to indicate a technical term, a phrase from another language, a thought, or a ship name, etc.

Use the <i> element only when there is not a more appropriate semantic element, such as:

- (emphasized text)
- (important text)
- <mark> (marked/highlighted text)
- <cite> (the title of a work)
- <dfn> (a definition term)

Browser Support

Element					
<i>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

In HTML 4.01, the <i> tag was used to render text in italics. However, this is not necessarily the case with HTML5. Style sheets can be used to format the text inside the <i> element.

Global Attributes

The <i> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <i> tag also supports the [Event Attributes in HTML](#).

HTML <iframe> Tag

Example

An inline frame is marked up as follows:

```
<iframe src="https://www.w3schools.com"></iframe>
```

Definition and Usage

The <iframe> tag specifies an inline frame.

An inline frame is used to embed another document within the current HTML document.

Browser Support

Element					
<iframe>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: To deal with browsers that do not support <iframe>, add a text between the opening <iframe> tag and the closing </iframe> tag.

Tip: Use CSS to style the <iframe> (even to include scrollbars).

Differences Between HTML 4.01 and HTML5

HTML5 has added some new attributes, and several HTML 4.01 attributes are removed from HTML5.

Differences Between HTML and XHTML

In XHTML, the name attribute is deprecated, and will be removed. Use the global id attribute instead.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>align</u>	left right top middle bottom	Not supported in HTML5. Specifies the alignment of an <iframe> according to surrounding elements
<u>frameborder</u>	1 0	Not supported in HTML5. Specifies whether or not to display a border around an <iframe>
<u>height</u>	<i>pixels</i>	Specifies the height of an <iframe>
<u>longdesc</u>	<i>URL</i>	Not supported in HTML5. Specifies a page that contains a long description of the content of an <iframe>
<u>marginheight</u>	<i>pixels</i>	Not supported in HTML5. Specifies the top and bottom margins of the content of an <iframe>
<u>marginwidth</u>	<i>pixels</i>	Not supported in HTML5. Specifies the left and right margins of the content of an <iframe>
<u>name</u>	<i>text</i>	Specifies the name of an <iframe>
<u>sandbox</u>	allow-formsallow-pointer-lockallow-popups allow-same-origin allow-scripts allow-top-navigation	Enables an extra set of restrictions for the content in an <iframe>
<u>scrolling</u>	yes no auto	Not supported in HTML5. Specifies whether or not to display scrollbars in an <iframe>
<u>src</u>	<i>URL</i>	Specifies the address of the document to embed in the <iframe>
<u>srcdoc</u>	<i>HTML_code</i>	Specifies the HTML content of the page to show in the <iframe>
<u>width</u>	<i>pixels</i>	Specifies the width of an <iframe>

Global Attributes

The <iframe> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <iframe> tag also supports the [Event Attributes in HTML](#).

HTML Tag

Example

How to insert an image:

```

```

More "Try it Yourself" examples below.

Definition and Usage

The tag defines an image in an HTML page.

The tag has two required attributes: src and alt.

Note: Images are not technically inserted into an HTML page, images are linked to HTML pages. The tag creates a holding space for the referenced image.

Tip: To link an image to another document, simply nest the tag inside <a> tags.

[Try it »](#)

Browser Support

Element					
	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The following attributes: align, border, hspace, and vspace are not supported in HTML5.

Differences Between HTML and XHTML

In HTML the tag has no end tag.

In XHTML the tag must be properly closed.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>align</u>	top bottom middle left right	Not supported in HTML5. Specifies the alignment of an image according to surrounding elements
<u>alt</u>	<i>text</i>	Specifies an alternate text for an image
<u>border</u>	<i>pixels</i>	Not supported in HTML5. Specifies the width of the border around an image
crossorigin	anonymous use-credentials	Allow images from third-party sites that allow cross-origin access to be used with canvas
<u>height</u>	<i>pixels</i>	Specifies the height of an image
<u>hspace</u>	<i>pixels</i>	Not supported in HTML5. Specifies the whitespace on left and right side of an image
<u>ismap</u>	ismap	Specifies an image as a server-side image-map
<u>longdesc</u>	<i>URL</i>	Specifies a URL to a detailed description of an image
sizes		Specifies image sizes for different page layouts
<u>src</u>	<i>URL</i>	Specifies the URL of an image
<u>srcset</u>	<i>URL</i>	Specifies the URL of the image to use in different situations
<u>usemap</u>	# <i>mapname</i>	Specifies an image as a client-side image-map
<u>vspace</u>	<i>pixels</i>	Not supported in HTML5. Specifies the whitespace on top and bottom of an image
<u>width</u>	<i>pixels</i>	Specifies the width of an image

Global Attributes

The tag also supports the [Global Attributes in HTML](#).

Event Attributes

The tag also supports the [Event Attributes in HTML](#).

Try it Yourself - Examples

[Insert images from different locations](#) How to insert images from another folder or from another web site.

[Make a hyperlink of an image](#) How to add a hyperlink to an image.

[Create an image map](#) How to create an image map, with clickable regions. Each region is a hyperlink.

HTML <input> Tag

Example

An HTML form with three input fields; two text fields and one submit button:

```
<form action="/action_page.php">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit" value="Submit">
</form>
```

Definition and Usage

The <input> tag specifies an input field where the user can enter data.

<input> elements are used within a <form> element to declare input controls that allow users to input data.

An input field can vary in many ways, depending on the type attribute.

Browser Support

Element					
<input>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: The <input> element is empty, it contains attributes only.

Tip: Use the <label> element to define labels for <input> elements.

Differences Between HTML 4.01 and HTML5

The "align" attribute is not supported in HTML5.

In HTML5, the <input> tag has several new attributes, and the type attribute has several new values.

Differences Between HTML and XHTML

In HTML, the <input> tag has no end tag.

In XHTML, the <input> tag must be properly closed, like this <input />.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>accept</u>	<i>file_extension audio/* video/* image/* media_type</i>	Specifies the types of files that the server accepts (only for type="file")
<u>align</u>	left right top middle bottom	Not supported in HTML5. Specifies the alignment of an image input (only for type="image")
<u>alt</u>	<i>text</i>	Specifies an alternate text for images (only for type="image")
<u>autocomplete</u>	on off	Specifies whether an <input> element should have autocomplete enabled
<u>autofocus</u>	autofocus	Specifies that an <input> element should automatically get focus when the page loads
<u>checked</u>	checked	Specifies that an <input> element should be pre-selected when the page loads (for type="checkbox" or type="radio")
<u>dirname</u>	<i>inputname.dir</i>	Specifies that the text direction will be submitted
<u>disabled</u>	disabled	Specifies that an <input> element should be disabled
<u>form</u>	<i>form_id</i>	Specifies one or more forms the <input> element belongs to
<u>formaction</u>	<i>URL</i>	Specifies the URL of the file that will process the input control when the form is submitted (for type="submit" and type="image")
<u>formenctype</u>	application/x-www-form-urlencoded multipart/form-data text/plain	Specifies how the form-data should be encoded when submitting it to the server (for type="submit" and type="image")

Attribute	Value	Description
<u>formmethod</u>	getpost	Defines the HTTP method for sending data to the action URL (for type="submit" and type="image")
<u>formnovalidate</u>	formnovalidate	Defines that form elements should not be validated when submitted
<u>formtarget</u>	_blank _self _parent _top <i>framename</i>	Specifies where to display the response that is received after submitting the form (for type="submit" and type="image")
<u>height</u>	<i>pixels</i>	Specifies the height of an <input> element (only for type="image")
<u>list</u>	<i>datalist_id</i>	Refers to a <datalist> element that contains pre-defined options for an <input> element
<u>max</u>	<i>number date</i>	Specifies the maximum value for an <input> element
<u>maxlength</u>	<i>number</i>	Specifies the maximum number of characters allowed in an <input> element
<u>min</u>	<i>number date</i>	Specifies a minimum value for an <input> element
<u>multiple</u>	multiple	Specifies that a user can enter more than one value in an <input> element
<u>name</u>	<i>text</i>	Specifies the name of an <input> element
<u>pattern</u>	<i>regexp</i>	Specifies a regular expression that an <input> element's value is checked against
<u>placeholder</u>	<i>text</i>	Specifies a short hint that describes the expected value of an <input> element
<u>readonly</u>	readonly	Specifies that an input field is read-only

Attribute	Value	Description
<u>required</u>	required	Specifies that an input field must be filled out before submitting the form
<u>size</u>	<i>number</i>	Specifies the width, in characters, of an <input> element
<u>src</u>	<i>URL</i>	Specifies the URL of the image to use as a submit button (only for type="image")
<u>step</u>	<i>number</i>	Specifies the legal number intervals for an input field
<u>type</u>	button checkbox color date datetime-local email file hidden image month number password radio range reset search submit tel text time url week	Specifies the type <input> element to display
<u>value</u>	<i>text</i>	Specifies the value of an <input> element
<u>width</u>	<i>pixels</i>	Specifies the width of an <input> element (only for type="image")

Global Attributes

The <input> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <input> tag also supports the [Event Attributes in HTML](#).

HTML <ins> Tag

Example

A text with a deleted part, and a new, inserted part:

```
<p>My favorite color is <del>blue</del> <ins>red</ins>!</p>
```

Definition and Usage

The <ins> tag defines a text that has been inserted into a document.

Tip: Also look at the tag to markup deleted text.

Browsers will normally strike a line through deleted text and underline inserted text.

Browser Support

Element					
<ins>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: Use <ins> together with to markup updates and modifications in a document.

Differences Between HTML 4.01 and HTML5

NONE.

Attributes

Attribute	Value	Description
<u>cite</u>	<i>URL</i>	Specifies a URL to a document that explains the reason why the text was inserted/changed
<u>datetime</u>	<i>YYYY-MM-DDThh:mm:ssTZD</i>	Specifies the date and time when the text was inserted/changed

Global Attributes

The <ins> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The `<ins>` tag also supports the [Event Attributes in HTML](#).

HTML <kbd> Tag

Example

Format text in a document:

```
<kbd>Keyboard input</kbd>
```

Definition and Usage

The <kbd> tag is a phrase tag. It defines keyboard input.

Tip: This tag is not deprecated, but it is possible to achieve richer effect with CSS.

All phrase tags:

Tag	Description
	Renders as emphasized text
	Defines important text
<code>	Defines a piece of computer code
<samp>	Defines sample output from a computer program
<kbd>	Defines keyboard input
<var>	Defines a variable

Browser Support

Element					
<kbd>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The <kbd> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The `<kbd>` tag also supports the [Event Attributes in HTML](#).

HTML <label> Tag

Example

Three radio buttons with labels:

```
<form action="/action_page.php">
  <label for="male">Male</label>
  <input type="radio" name="gender" id="male" value="male"><br>
  <label for="female">Female</label>
  <input type="radio" name="gender" id="female" value="female"><br>
  <label for="other">Other</label>
  <input type="radio" name="gender" id="other" value="other"><br><br>
  <input type="submit" value="Submit">
</form>
```

Definition and Usage

The <label> tag defines a label for an <input> element.

The <label> element does not render as anything special for the user. However, it provides a usability improvement for mouse users, because if the user clicks on the text within the <label> element, it toggles the control.

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

Browser Support

Element					
<label>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: A label can be bound to an element either by using the "for" attribute, or by placing the element inside the <label> element.

Differences Between HTML 4.01 and HTML5

The "form" attribute is new in HTML5.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>for</u>	<i>element_id</i>	Specifies which form element a label is bound to
<u>form</u>	<i>form_id</i>	Specifies one or more forms the label belongs to

Global Attributes

The <label> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <label> tag also supports the [Event Attributes in HTML](#).

HTML <legend> Tag

Example

Group related elements in a form:

```
<form>
  <fieldset>
    <legend>Personalia:</legend>
    Name: <input type="text" size="30"><br>
    Email: <input type="text" size="30"><br>
    Date of birth: <input type="text" size="10">
  </fieldset>
</form>
```

Definition and Usage

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

Browser Support

Element					
<legend>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The "align" attribute is not supported in HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	top bottom left right	Not supported in HTML5. Specifies the alignment of the caption

Global Attributes

The <legend> tag also supports the Global Attributes in HTML.

Event Attributes

The <legend> tag also supports the [Event Attributes in HTML](#).

HTML Tag

Example

One ordered () and one unordered () HTML list:

```
<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>

<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

More "Try it Yourself" examples below.

Definition and Usage

The tag defines a list item.

The tag is used in ordered lists(), unordered lists (), and in menu lists (<menu>).

Browser Support

Element					
	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The "type" attribute is NOT supported in HTML5.

The "value" attribute was deprecated in HTML 4.01, but IS supported in HTML5.

Tips and Notes

Tip: Use CSS to define the type of list.

Attributes

Attribute	Value	Description
<u>type</u>	1 A a I i disc square circle	Not supported in HTML5. Specifies which kind of bullet point will be used
<u>value</u>	<i>number</i>	Specifies the value of a list item. The following list items will increment from that number (only for lists)

Global Attributes

The tag also supports the [Global Attributes in HTML](#).

Event Attributes

The tag also supports the [Event Attributes in HTML](#).

Try it Yourself - Examples

[A nested list](#) A list inside a list.

[Another nested list](#) A more complicated nested list.

HTML <link> Tag

Example

Link to an external style sheet:

```
<head>
  <link rel="stylesheet" type="text/css" href="theme.css">
</head>
```

Definition and Usage

The <link> tag defines a link between a document and an external resource.

The <link> tag is used to link to external style sheets.

Browser Support

When used for style sheets, the <link> tag is supported in all major browsers. No real support for anything else.

Element					
<link>	Yes	Yes	Yes	Yes	Yes

Note: The <link> element is an empty element, it contains attributes only.

Note: This element goes only in the head section, but it can appear any number of times.

Differences Between HTML 4.01 and HTML5

Some HTML 4.01 attributes are not supported in HTML5.

The "sizes" attribute is new in HTML5.

Differences Between HTML and XHTML

In HTML the <link> tag has no end tag.

In XHTML the <link> tag must be properly closed.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>charset</u>	<i>char_encoding</i>	Not supported in HTML5. Specifies the character encoding of the linked document
crossorigin	anonymoususe-credentials	Specifies how the element handles cross-origin requests
<u>href</u>	<i>URL</i>	Specifies the location of the linked document
<u>hreflang</u>	<i>language_code</i>	Specifies the language of the text in the linked document
<u>media</u>	<i>media_query</i>	Specifies on what device the linked document will be displayed
<u>rel</u>	alternate author dns-prefetch help icon license next pingback preconnect prefetch preload prerender prev search stylesheet	Required. Specifies the relationship between the current document and the linked document
<u>rev</u>	<i>reversed relationship</i>	Not supported in HTML5. Specifies the relationship between the linked document and the current document
<u>sizes</u>	<i>HeightxWidth any</i>	Specifies the size of the linked resource. Only for rel="icon"
<u>target</u>	_blank _self _top _parent <i>frame_name</i>	Not supported in HTML5. Specifies where the linked document is to be loaded
<u>type</u>	<i>media_type</i>	Specifies the media type of the linked document

Global Attributes

The <link> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <link> tag also supports the [Event Attributes in HTML](#).

HTML <main> Tag

Example

```
<main>
  <h1>Web Browsers</h1>
  <p>Google Chrome, Firefox, and Internet Explorer are the most used
browsers today.</p>

  <article>
    <h1>Google Chrome</h1>
    <p>Google Chrome is a free, open-source web browser developed by
Google,
    released in 2008.</p>
  </article>

  <article>
    <h1>Internet Explorer</h1>
    <p>Internet Explorer is a free web browser from Microsoft, released in
1995.</p>
  </article>

  <article>
    <h1>Mozilla Firefox</h1>
    <p>Firefox is a free, open-source web browser from Mozilla, released in
2004.</p>
  </article>
</main>
```

Definition and Usage

The <main> tag specifies the main content of a document.

The content inside the <main> element should be unique to the document. It should not contain any content that is repeated across documents such as sidebars, navigation links, copyright information, site logos, and search forms.

Note: There must not be more than one <main> element in a document. The <main> element must NOT be a descendant of an <article>, <aside>, <footer>, <header>, or <nav> element.

Browser Support

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

Element					
<main>	6.0	12.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <main> tag is new in HTML5.

Global Attributes

The <main> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <main> tag also supports the [Event Attributes in HTML](#).

HTML <map> Tag

Example

An image-map, with clickable areas:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>
```

Definition and Usage

The <map> tag is used to define a client-side image-map. An image-map is an image with clickable areas.

The required name attribute of the <map> element is associated with the 's usemap attribute and creates a relationship between the image and the map.

The <map> element contains a number of <area> elements, that defines the clickable areas in the image map.

Browser Support

Element					
<map>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

Note: In HTML5, if the id attribute of the <map> tag is also specified, it must have the same value as the name attribute.

Differences Between HTML and XHTML

In XHTML, the name attribute is deprecated, and will be removed. Use the global id attribute instead.

Attributes

Attribute	Value	Description
<u>name</u>	<i>mapname</i>	Required. Specifies the name of an image-map

Global Attributes

The `<map>` tag also supports the [Global Attributes in HTML](#).

Event Attributes

The `<map>` tag also supports the [Event Attributes in HTML](#).

HTML <mark> Tag

Example

Highlight parts of a text:

```
<p>Do not forget to buy <mark>milk</mark> today.</p>
```

Definition and Usage

The <mark> tag defines marked text.

Use the <mark> tag if you want to highlight parts of your text.

Browser Support

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

Element					
<mark>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <mark> tag is new in HTML5.

Global Attributes

The <mark> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <mark> tag also supports the [Event Attributes in HTML](#).

HTML <meta> Tag

Example

Describe metadata within an HTML document:

```
<head>
  <meta charset="UTF-8">
  <meta name="description" content="Free Web tutorials">
  <meta name="keywords" content="HTML,CSS,XML,JavaScript">
  <meta name="author" content="John Doe">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
```

Definition and Usage

Metadata is data (information) about data.

The <meta> tag provides metadata about the HTML document. Metadata will not be displayed on the page, but will be machine parsable.

Meta elements are typically used to specify page description, keywords, author of the document, last modified, and other metadata.

The metadata can be used by browsers (how to display content or reload page), search engines (keywords), or other web services.

HTML5 introduced a method to let web designers take control over the viewport (the user's visible area of a web page), through the <meta> tag (See "Setting The Viewport" example below).

Browser Support

Element					
<meta>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: <meta> tags always go inside the <head> element.

Note: Metadata is always passed as name/value pairs.

Note: The content attribute MUST be defined if the name or the http-equiv attribute is defined. If none of these are defined, the content attribute CANNOT be defined.

Setting The Viewport

HTML5 introduced a method to let web designers take control over the viewport, through the <meta> tag.

The viewport is the user's visible area of a web page. It varies with the device, and will be smaller on a mobile phone than on a computer screen.

You should include the following <meta> viewport element in all your web pages:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

A <meta> viewport element gives the browser instructions on how to control the page's dimensions and scaling.

The width=device-width part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).

The initial-scale=1.0 part sets the initial zoom level when the page is first loaded by the browser.

Here is an example of a web page *without* the viewport meta tag, and the same web page *with* the viewport meta tag:

Tip: If you are browsing this page with a phone or a tablet, you can click on the two links below to see the difference.



[Without the](#)



[With the viewport](#)

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet domino

[viewport meta tag](#)

[meta tag](#)

You can read more about the viewport in our [Responsive Web Design - The Viewport Tutorial](#).

Differences Between HTML 4.01 and HTML5

The scheme attribute is not supported in HTML5.

HTML5 has a new attribute, charset, which makes it easier to define charset:

Differences Between HTML and XHTML

In HTML the <meta> tag has no end tag.

In XHTML the <meta> tag must be properly closed.

Examples

Example 1 - Define keywords for search engines:

```
<meta name="keywords" content="HTML, CSS, XML, XHTML, JavaScript">
```

Example 2 - Define a description of your web page:

```
<meta name="description" content="Free Web tutorials on HTML and CSS">
```

Example 3 - Define the author of a page:

```
<meta name="author" content="John Doe">
```

Example 4 - Refresh document every 30 seconds:

```
<meta http-equiv="refresh" content="30">
```

Example 5 - Setting the viewport to make your website look good on all devices:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Attributes

= New in HTML5.

Attribute	Value	Description
<u>charset</u>	<i>character_set</i>	Specifies the character encoding for the HTML document
<u>content</u>	<i>text</i>	Gives the value associated with the http-equiv or name attribute
<u>http-equiv</u>	content-type default-style refresh	Provides an HTTP header for the information/value of the content attribute
<u>name</u>	application-name author description generator keywords viewport	Specifies a name for the metadata
<u>scheme</u>	<i>format/URI</i>	Not supported in HTML5. Specifies a scheme to be used to interpret the value of the content attribute

Global Attributes

The <meta> tag also supports the [Global Attributes in HTML](#).

HTML <meter> Tag

Example

Use the meter element to measure data within a given range (a gauge):

```
<meter value="2" min="0" max="10">2 out of 10</meter><br>
<meter value="0.6">60%</meter>
```

Definition and Usage

The <meter> tag defines a scalar measurement within a known range, or a fractional value. This is also known as a gauge.

Examples: Disk usage, the relevance of a query result, etc.

Note: The <meter> tag should not be used to indicate progress (as in a progress bar). For progress bars, use the <progress> tag.

Browser Support

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

Element					
<meter>	8.0	13.0	6.0	6.0	11.0

Differences Between HTML 4.01 and HTML5

The <meter> tag is new in HTML5.

Attributes

Attribute	Value	Description
<u>form</u>	<i>form_id</i>	Specifies one or more forms the <meter> element belongs to
<u>high</u>	<i>number</i>	Specifies the range that is considered to be a high value
<u>low</u>	<i>number</i>	Specifies the range that is considered to be a low value
<u>max</u>	<i>number</i>	Specifies the maximum value of the range
<u>min</u>	<i>number</i>	Specifies the minimum value of the range
<u>optimum</u>	<i>number</i>	Specifies what value is the optimal value for the gauge
<u>value</u>	<i>number</i>	Required. Specifies the current value of the gauge

Global Attributes

The <meter> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <meter> tag also supports the [Event Attributes in HTML](#).

HTML <nav> Tag

Example

A set of navigation links:

```
<nav>
  <a href="/html/">HTML</a> |
  <a href="/css/">CSS</a> |
  <a href="/js/">JavaScript</a> |
  <a href="/jquery/">jQuery</a>
</nav>
```

Definition and Usage

The <nav> tag defines a set of navigation links.

Notice that NOT all links of a document should be inside a <nav> element. The <nav> element is intended only for major block of **navigation links**.

Browsers, such as screen readers for disabled users, can use this element to determine whether to omit the initial rendering of this content.

Browser Support

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

Element					
<nav>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <nav> tag is new in HTML5.

Global Attributes

The <nav> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <nav> tag also supports the [Event Attributes in HTML](#).

HTML <noframes> Tag. Not Supported in HTML5.

Example

A three-framed page, with a <noframes> tag:

```
<html>

<frameset cols="25%,50%,25%">
  <frame src="frame_a.htm">
  <frame src="frame_b.htm">
  <frame src="frame_c.htm">
  <noframes>Sorry, your browser does not handle frames!</noframes>
</frameset>

</html>
```

Definition and Usage

The <noframes> tag is not supported in HTML5.

The <noframes> tag is a fallback tag for browsers that do not support frames. It can contain all the HTML elements that you can find inside the <body> element of a normal HTML page. The <noframes> element can be used to link to a non-frameset version of the web site or to display a message to users that frames are required.

The <noframes> element goes inside the <frameset> element.

Note: If you want to validate a page containing frames, be sure the <!DOCTYPE> is set to either "HTML Frameset DTD" or "XHTML Frameset DTD".

Browser Support

Element					
<noframes>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The <noframes> tag is not supported in HTML5.

Differences Between HTML and XHTML

Important: In XHTML Frameset DTD, the **text** in the <noframes> element must be enclosed in a <body> element.

HTML <noscript> Tag

Example

Use of the <noscript> tag:

```
<script>
document.write("Hello World!")
</script>
<noscript>Your browser does not support JavaScript!</noscript>
```

Definition and Usage

The <noscript> tag defines an alternate content for users that have disabled scripts in their browser or have a browser that doesn't support script.

The <noscript> element can be used in both <head> and <body>.

When used inside the <head> element: <noscript> must contain only <link>, <style>, and <meta> elements.

The content inside the <noscript> element will be displayed if scripts are not supported, or are disabled in the user's browser.

Browser Support

Element					
<noscript>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: It is also a good practice to use the comment tag to "hide" scripts from browsers without support for client-side scripts (so they don't show them as plain text):

```
<script> <!-- function displayMsg() { alert("Hello World!") } //-->
</script>
```

Differences Between HTML 4.01 and HTML5

In HTML 4.01, the <noscript> tag can only be used inside the <body> element.

In HTML5, the <noscript> tag can be used both inside <head> and <body>.

Differences Between HTML and XHTML

In XHTML, the `<noscript>` tag is not supported.

Global Attributes

The `<noscript>` tag also supports the [Global Attributes in HTML](#).

HTML <object> Tag

Example

How to use the <object> element to embed a Flash file:

```
<object width="400" height="400" data="helloworld.swf"></object>
```

Definition and Usage

The <object> tag defines an embedded object within an HTML document. Use this element to embed multimedia (like audio, video, Java applets, ActiveX, PDF, and Flash) in your web pages.

You can also use the <object> tag to embed another webpage into your HTML document.

You can use the <param> tag to pass parameters to plugins that have been embedded with the <object> tag.

Browser Support

Element					
<object>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: An <object> element must appear inside the <body> element. The text between the <object> and </object> is an alternate text, for browsers that do not support this tag.

Tip: For images use the tag instead of the <object> tag.

Tip: At least one of the "data" or "type" attribute MUST be defined.

Differences Between HTML 4.01 and HTML5

Some HTML 4.01 attributes are not supported in HTML5.

The "form" attribute is new in HTML5.

In HTML5, objects can be used and submitted in forms.

In HTML5, objects can no longer appear inside the <head> element of a document.

Attributes

= New in HTML5.

Attribute	Value	Description
-----------	-------	-------------

<u>align</u>	top bottom middle left right	Not supported in HTML5. Specifies the alignment of the <object> element according to surrounding elements
archive	<i>URL</i>	Not supported in HTML5. A space separated list of URL's to archives. The archives contains resources relevant to the object
<u>border</u>	<i>pixels</i>	Not supported in HTML5. Specifies the width of the border around an <object>
classid	<i>class_ID</i>	Not supported in HTML5. Defines a class ID value as set in the Windows Registry or a URL
codebase	<i>URL</i>	Not supported in HTML5. Defines where to find the code for the object
codetype	<i>media_type</i>	Not supported in HTML5. The media type of the code referred to by the classid attribute
<u>data</u>	<i>URL</i>	Specifies the URL of the resource to be used by the object
declare	declare	Not supported in HTML5. Defines that the object should only be declared, not created or instantiated until needed
<u>form</u>	<i>form_id</i>	Specifies one or more forms the object belongs to
<u>height</u>	<i>pixels</i>	Specifies the height of the object
<u>hspace</u>	<i>pixels</i>	Not supported in HTML5. Specifies the whitespace on left and right side of an object
<u>name</u>	<i>name</i>	Specifies a name for the object
standby	<i>text</i>	Not supported in HTML5. Defines a text to display while the object is loading
<u>type</u>	<i>media_type</i>	Specifies the media type of data specified in the data attribute
<u>usemap</u>	# <i>mapname</i>	Specifies the name of a client-side image map to be used with the object
<u>vspace</u>	<i>pixels</i>	Not supported in HTML5. Specifies the whitespace on top and bottom of an object
<u>width</u>	<i>pixels</i>	Specifies the width of the object

Global Attributes

The <object> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <object> tag also supports the [Event Attributes in HTML](#).

HTML Tag

Example

2 different ordered lists:

```
<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>

<ol start="50">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Definition and Usage

The tag defines an ordered list. An ordered list can be numerical or alphabetical. Use the tag to define list items.

Browser Support

Element					
	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: For unordered list, use the tag.

Tip: Use CSS to style lists.

Differences Between HTML 4.01 and HTML5

The "start" and "type" attributes were deprecated in HTML 4.01, but are supported in HTML5.

The "reversed" attribute is new in HTML5.

The "compact" attribute is not supported in HTML5.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>compact</u>	compact	Not supported in HTML5. Specifies that the list should render smaller than normal
<u>reversed</u>	reversed	Specifies that the list order should be descending (9,8,7...)
<u>start</u>	<i>number</i>	Specifies the start value of an ordered list
<u>type</u>	1 A a I i	Specifies the kind of marker to use in the list

Global Attributes

The tag also supports the [Global Attributes in HTML](#).

Event Attributes

The tag also supports the [Event Attributes in HTML](#).

HTML <optgroup> Tag

Example

Group related options with <optgroup> tags:

```
<select>
  <optgroup label="Swedish Cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
  </optgroup>
  <optgroup label="German Cars">
    <option value="mercedes">Mercedes</option>
    <option value="audi">Audi</option>
  </optgroup>
</select>
```

Definition and Usage

The <optgroup> is used to group related options in a drop-down list.

If you have a long list of options, groups of related options are easier to handle for a user.

Browser Support

Element					
<optgroup>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Attributes

Attribute	Value	Description
<u>disabled</u>	disabled	Specifies that an option-group should be disabled
<u>label</u>	text	Specifies a label for an option-group

Global Attributes

The <optgroup> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <optgroup> tag also supports the [Event Attributes in HTML](#).

HTML <option> Tag

Example

A drop-down list with four options:

```
<select>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi">Audi</option>
</select>
```

Definition and Usage

The <option> tag defines an option in a select list.

<option> elements go inside a <select> or <datalist> element.

Browser Support

Element					
<option>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: The <option> tag can be used without any attributes, but you usually need the **value** attribute, which indicates what is sent to the server.

Tip: If you have a long list of options, you can group related options with the <optgroup> tag.

Attributes

Attribute	Value	Description
<u>disabled</u>	disabled	Specifies that an option should be disabled
<u>label</u>	<i>text</i>	Specifies a shorter label for an option
<u>selected</u>	selected	Specifies that an option should be pre-selected when the page loads
<u>value</u>	<i>text</i>	Specifies the value to be sent to a server

Global Attributes

The <option> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <option> tag also supports the [Event Attributes in HTML](#).

HTML <output> Tag

Example

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

```
<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">
  <input type="range" id="a" value="50">100
  +<input type="number" id="b" value="50">
  =<output name="x" for="a b"></output>
</form>
```

Definition and Usage

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

Browser Support

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

Element					
<output>	10.0	13.0	4.0	5.1	11.0

Note: The form attribute of the <output> element is not supported in IE/Edge.

Differences Between HTML 4.01 and HTML5

The <output> tag is new in HTML5.

Attributes

Attribute	Value	Description
<u>for</u>	<i>element_id</i>	Specifies the relationship between the result of the calculation, and the elements used in the calculation
<u>form</u>	<i>form_id</i>	Specifies one or more forms the output element belongs to
<u>name</u>	<i>name</i>	Specifies a name for the output element

Global Attributes

The <output> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <output> tag also supports the [Event Attributes in HTML](#).

HTML <p> Tag

Example

A paragraph is marked up as follows:

```
<p>This is some text in a paragraph.</p>
```

More "Try it Yourself" examples below.

Definition and Usage

The <p> tag defines a paragraph.

Browsers automatically add some space (margin) before and after each <p> element. The margins can be modified with CSS (with the margin properties).

Browser Support

Element					
<p>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The align attribute is not supported in HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	left right center justify	Not supported in HTML5. Specifies the alignment of the text within a paragraph

Global Attributes

The <p> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <p> tag also supports the [Event Attributes in HTML](#).

Try it Yourself - Examples

More paragraphs Some default behaviors of paragraphs.

Poem problems Examples of how you can NOT write a poem in HTML.

HTML <param> Tag

Example

Set the "autoplay" parameter to "true", so the sound will start playing as soon as the page loads:

```
<object data="horse.wav">
  <param name="autoplay" value="true">
</object>
```

Definition and Usage

The <param> tag is used to define parameters for plugins embedded with an [<object>](#) element.

Tip: HTML 5 also includes two new elements for playing audio or video: The [<audio>](#) and [<video>](#) tags.

Browser Support

The <param> tag is supported in all major browsers. However, the file format defined in <object> may not be supported in all browsers.

Element					
<param>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The "type" and "valuetype" attributes are not supported in HTML5.

Differences Between HTML and XHTML

In HTML the <param> tag has no end tag.

In XHTML the <param> tag must be properly closed, like this <param />.

Attributes

Attribute	Value	Description
<u>name</u>	<i>name</i>	Specifies the name of a parameter
type	<i>media_type</i>	Not supported in HTML5. Specifies the media type of the parameter
<u>value</u>	<i>value</i>	Specifies the value of the parameter
valuetype	data ref object	Not supported in HTML5. Specifies the type of the value

Global Attributes

The <param> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <param> tag also supports the [Event Attributes in HTML](#).

HTML <picture> Tag

Example

How to use the <picture> tag:

```
<picture>
  <source media="(min-width: 650px)" srcset="img_pink_flowers.jpg">
  <source media="(min-width: 465px)" srcset="img_white_flower.jpg">
  
</picture>
```

Definition and Usage

The <picture> tag gives web developers more flexibility in specifying image resources.

The most common use of the <picture> element will be for art direction in responsive designs. Instead of having one image that is scaled up or down based on the viewport width, multiple images can be designed to more nicely fill the browser viewport.

The <picture> element holds two different tags: one or more <source> tags and one tag.

The <source> element has the following attributes:

- srcset (required) - defines the URL of the image to show
- media - accepts any valid media query that would normally be defined in a CSS
- sizes - defines a single width descriptor, a single media query with width descriptor, or a comma-delimited list of media queries with a width descriptor
- type - defines the MIME type

The browser will use the attribute values to load the most appropriate image. The browser will use the first <source> element with a matching hint and ignore any following <source> tags.

The element is required as the last child tag of the <picture> declaration block. The element is used to provide backward compatibility for browsers that do not support the <picture> element, or if none of the <source> tags matched.

The <picture> element works similar to the <video> and <audio> elements. You set up different sources, and the first source that fits the preferences is the one being used.

Browser Support

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

Element					
<picture>	38.0	13.0	38.0	9.1	25.0

Differences Between HTML 4.01 and HTML5

The <picture> tag is new in HTML5.

Global Attributes

The <picture> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <picture> tag also supports the [Event Attributes in HTML](#).

HTML <pre> Tag

Example

Preformatted text:

```
<pre>
Text in a pre element
is displayed in a fixed-width
font, and it preserves
both spaces and
line breaks
</pre>
```

Definition and Usage

The <pre> tag defines preformatted text.

Text in a <pre> element is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks.

Browser Support

Element					
<pre>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: Use the <pre> element when displaying text with unusual formatting, or some sort of computer code.

Differences Between HTML 4.01 and HTML5

The "width" attribute is not supported in HTML5.

Attributes

Attribute	Value	Description
<u>width</u>	<i>number</i>	Not supported in HTML5. Specifies the maximum number of characters per line

Global Attributes

The `<pre>` tag also supports the [Global Attributes in HTML](#).

Event Attributes

The `<pre>` tag also supports the [Event Attributes in HTML](#).

HTML <progress> Tag

Example

Downloading in progress:

```
<progress value="22" max="100"></progress>
```

Definition and Usage

The <progress> tag represents the progress of a task.

Browser Support

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

Element					
<progress>	8.0	10.0	16.0	6.0	11.0

Differences Between HTML 4.01 and HTML5

The <progress> tag is new in HTML5.

Tips and Notes

Tip: Use the <progress> tag in conjunction with JavaScript to display the progress of a task.

Note: The <progress> tag is not suitable for representing a gauge (e.g. disk space usage or relevance of a query result). To represent a gauge, use the [<meter>](#) tag instead.

Attributes

Attribute	Value	Description
<u>max</u>	<i>number</i>	Specifies how much work the task requires in total
<u>value</u>	<i>number</i>	Specifies how much of the task has been completed

Global Attributes

The <progress> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <progress> tag also supports the [Event Attributes in HTML](#).

HTML <q> Tag

Example

Mark up a short quotation:

```
<p>WWF's goal is to:  
<q>Build a future where people live in harmony with nature.</q>  
We hope they succeed.</p>
```

Definition and Usage

The <q> tag defines a short quotation.

Browsers normally insert quotation marks around the quotation.

Browser Support

Element					
<q>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: Use <blockquote> to mark up a section that is quoted from another source.

Differences Between HTML 4.01 and HTML5

NONE.

Attributes

Attribute	Value	Description
<u>cite</u>	<u>URL</u>	Specifies the source URL of the quote

Global Attributes

The <q> tag also supports the Global Attributes in HTML.

Event Attributes

The <q> tag also supports the [Event Attributes in HTML](#).

HTML <rp> Tag

Example

A ruby annotation:

```
<ruby>
漢 <rt><rp>(</rp>) <rp>「</rp></rt>
</ruby>
```

Definition and Usage

The <rp> tag can be used to provide parentheses around a ruby text, to be shown by browsers that do not support ruby annotations.

Use the <rp> tag together with the <ruby> and the <rt> tags: The <ruby> element consists of one or more characters that needs an explanation/pronunciation, and an <rt> element that gives that information, and an optional <rp> element that defines what to show for browsers that not support ruby annotations.

Browser Support

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

Element					
<rp>	5.0	5.5	38.0	5.0	15.0

Differences Between HTML 4.01 and HTML5

The <rp> tag is new in HTML5.

Global Attributes

The <rp> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <rp> tag also supports the [Event Attributes in HTML](#).

Default CSS Settings

None.

HTML <rt> Tag

Example

A ruby annotation:

```
<ruby>  
漢 <rt> ハン </rt>  
</ruby>
```

Definition and Usage

The <rt> tag defines an explanation or pronunciation of characters (for East Asian typography) in a ruby annotation.

Use the <rt> tag together with the <ruby> and the <rp> tags: The <ruby> element consists of one or more characters that needs an explanation/pronunciation, and an <rt> element that gives that information, and an optional <rp> element that defines what to show for browsers that do not support ruby annotations.

Browser Support

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

Element					
<rt>	5.0	5.5	38.0	5.0	15.0

Differences Between HTML 4.01 and HTML5

The <rt> tag is new in HTML5.

Global Attributes

The <rt> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <rt> tag also supports the [Event Attributes in HTML](#).

Default CSS Settings

Most browsers will display the <rt> element with the following default values:

```
rt {  
    line-height: normal;  
}
```

HTML <ruby> Tag

Example

A ruby annotation:

```
<ruby>  
漢 <rt> ㄏㄞˋ </rt>  
</ruby>
```

Definition and Usage

The <ruby> tag specifies a ruby annotation.

A ruby annotation is a small extra text, attached to the main text to indicate the pronunciation or meaning of the corresponding characters. This kind of annotation is often used in Japanese publications.

Use the <ruby> tag together with the <rt> and/or the <rp> tags: The <ruby> element consists of one or more characters that needs an explanation/pronunciation, and an <rt> element that gives that information, and an optional <rp> element that defines what to show for browsers that do not support ruby annotations.

Browser Support

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

Element					
<ruby>	5.0	5.5	38.0	5.0	15.0

Differences Between HTML 4.01 and HTML5

The <ruby> tag is new in HTML5.

Global Attributes

The <ruby> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <ruby> tag also supports the [Event Attributes in HTML](#).

HTML <s> Tag

Example

Mark up text that is no longer correct:

```
<p><s>My car is blue.</s></p>
<p>My new car is silver.</p>
```

Definition and Usage

The <s> tag specifies text that is no longer correct, accurate or relevant.

The <s> tag should not be used to define replaced or deleted text, use the tag to define replaced or deleted text.

Browser Support

Element					
<s>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The <s> element was deprecated in HTML 4.01, and was used to define strikethrough text.

The <s> element is redefined in HTML5, and is now used to define text that is no longer correct.

Global Attributes

The <s> tag also supports the Global Attributes in HTML.

Event Attributes

The <s> tag also supports the Event Attributes in HTML.

HTML <samp> Tag

Example

Format text in a document:

```
<samp>Sample output from a computer program</samp>
```

Definition and Usage

The <samp> tag is a phrase tag. It defines sample output from a computer program.

Tip: This tag is not deprecated, but it is possible to achieve richer effect with CSS.

All phrase tags:

Tag	Description
<u></u>	Renders as emphasized text
<u></u>	Defines important text
<u><code></u>	Defines a piece of computer code
<u><samp></u>	Defines sample output from a computer program
<u><kbd></u>	Defines keyboard input
<u><var></u>	Defines a variable

Browser Support

Element					
<samp>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The <samp> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <code> tag also supports the [Event Attributes in HTML](#).

HTML <script> Tag

Example

Write "Hello JavaScript!" with JavaScript:

```
<script>
document.getElementById("demo").innerHTML = "Hello JavaScript!";
</script>
```

Definition and Usage

The <script> tag is used to define a client-side script (JavaScript).

The <script> element either contains scripting statements, or it points to an external script file through the src attribute.

Common uses for JavaScript are image manipulation, form validation, and dynamic changes of content.

Tip: If you want to learn JavaScript, visit our [JavaScript Tutorial](#).

Browser Support

Element					
<script>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: If the "src" attribute is present, the <script> element must be empty.

Tip: Also look at the <noscript> element for users that have disabled scripts in their browser, or have a browser that doesn't support client-side scripting.

Note: There are several ways an external script can be executed:

- If `async="async"`: The script is executed asynchronously with the rest of the page (the script will be executed while the page continues the parsing)
- If `async` is not present and `defer="defer"`: The script is executed when the page has finished parsing
- If neither `async` or `defer` is present: The script is fetched and executed immediately, before the browser continues parsing the page

Differences Between HTML 4.01 and HTML5

The "type" attribute is required in HTML 4, but optional in HTML5.

The "async" attribute is new in HTML5.

The HTML 4.01 attribute: "xml:space", is not supported in HTML5.

Differences Between HTML and XHTML

In XHTML, the content inside scripts is declared as #PCDATA (instead of CDATA), which means that entities will be parsed.

This means that in XHTML, all special characters should be encoded, or all content should be wrapped inside a CDATA section:

```
<script type="text/javascript"> //<![CDATA[ var i = 10; if (i < 5) { // some code } //]]> </script>
```

Attributes

= New in HTML5.

Attribute	Value	Description
<u>async</u>	async	Specifies that the script is executed asynchronously (only for external scripts)
<u>charset</u>	<i>charset</i>	Specifies the character encoding used in an external script file
<u>defer</u>	defer	Specifies that the script is executed when the page has finished parsing (only for external scripts)
<u>src</u>	<i>URL</i>	Specifies the URL of an external script file
<u>type</u>	<i>media_type</i>	Specifies the media type of the script
xml:space	preserve	Not supported in HTML5.Specifies whether whitespace in code should be preserved

Global Attributes

The `<script>` tag also supports the [Global Attributes in HTML](#).

HTML <section> Tag

Example

A section in a document, explaining what WWF is:

```
<section>
  <h1>WWF</h1>
  <p>The World Wide Fund for Nature (WWF) is....</p>
</section>
```

Definition and Usage

The <section> tag defines sections in a document, such as chapters, headers, footers, or any other sections of the document.

Browser Support

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

Element					
<section>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <section> tag is new in HTML5.

Global Attributes

The <section> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <section> tag also supports the [Event Attributes in HTML](#).

HTML <select> Tag

Example

Create a drop-down list with four options:

```
<select>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="mercedes">Mercedes</option>
  <option value="audi">Audi</option>
</select>
```

Definition and Usage

The <select> element is used to create a drop-down list.

The <option> tags inside the <select> element define the available options in the list.

Browser Support

Element					
<select>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: The <select> element is a form control and can be used in a form to collect user input.

Differences Between HTML 4.01 and HTML5

HTML5 has added some new attributes.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>autofocus</u>	autofocus	Specifies that the drop-down list should automatically get focus when the page loads
<u>disabled</u>	disabled	Specifies that a drop-down list should be disabled
<u>form</u>	<i>form_id</i>	Defines one or more forms the select field belongs to
<u>multiple</u>	multiple	Specifies that multiple options can be selected at once
<u>name</u>	<i>name</i>	Defines a name for the drop-down list
<u>required</u>	required	Specifies that the user is required to select a value before submitting the form
<u>size</u>	<i>number</i>	Defines the number of visible options in a drop-down list

Global Attributes

The <select> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <select> tag also supports the [Event Attributes in HTML](#).

HTML <small> Tag

Example

Define a smaller text:

```
<p>W3Schools.com - the world's largest web development site.</p>
<p><small>Copyright 1999-2050 by Refsnes Data</small></p>
```

Definition and Usage

The <small> tag defines smaller text (and other side comments).

Browser Support

Element					
<small>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The <small> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <small> tag also supports the [Event Attributes in HTML](#).

HTML <source> Tag

Example

An audio player with two source files. The browser should choose which file (if any) it has support for:

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>
```

More "Try it Yourself" examples below.

Definition and Usage

The <source> tag is used to specify multiple media resources for media elements, such as <video>, <audio>, and <picture>.

The <source> tag allows you to specify alternative video/audio/image files which the browser may choose from, based on its media type, codec support or media query.

Browser Support

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

Element					
<source>	4.0	9.0	3.5	4.0	10.5

Differences Between HTML 4.01 and HTML5

The <source> tag is new in HTML5.

Attributes

Attribute	Value	Description
<u>src</u>	<i>URL</i>	Required when <source> is used in <audio> and <video>. Specifies the URL of the media file
<u>srcset</u>	<i>URL</i>	Required when <source> is used in <picture>. Specifies the URL of the image to use in different situations
<u>media</u>	<i>media_query</i>	Accepts any valid media query that would normally be defined in a CSS
sizes		Specifies image sizes for different page layouts
<u>type</u>	<i>MIME-type</i>	Specifies the MIME-type of the resource

Global Attributes

The <source> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <source> tag also supports the [Event Attributes in HTML](#).

More Examples

Example

A <picture> element with two source files, and a fallback image:

```
<picture>
  <source media="(min-width: 650px)" srcset="img_pink_flowers.jpg">
  <source media="(min-width: 465px)" srcset="img_white_flower.jpg">
  
</picture>
```

HTML Tag

Example

A element used to color a part of a text:

```
<p>My mother has <span style="color:blue">blue</span> eyes.</p>
```

Definition and Usage

The tag is used to group inline-elements in a document.

The tag provides no visual change by itself.

The tag provides a way to add a hook to a part of a text or a part of a document.

Browser Support

Element					
	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: When a text is hooked in a element, you can style it with CSS, or manipulate it with JavaScript.

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The tag also supports the [Global Attributes in HTML](#).

Event Attributes

The tag also supports the [Event Attributes in HTML](#).

HTML <strike> Tag. Not Supported in HTML5.

Example

Strikethrough text can be marked up as follows:

```
<p>Version 2.0 is <strike>not yet available!</strike> now available!</p>
```

Definition and Usage

The <strike> tag is not supported in HTML5. Use or <s> instead.

The <strike> tag defines strikethrough text.

Browser Support

Element					
<strike>	Yes	Yes	Yes	Yes	Yes

HTML Tag

Example

Format text in a document:

```
<strong>Strong text</strong>
```

Definition and Usage

The tag is a phrase tag. It defines important text.

Tip: This tag is not deprecated, but it is possible to achieve richer effect with CSS.

All phrase tags:

Tag	Description
	Renders as emphasized text
	Defines important text
<code>	Defines a piece of computer code
<samp>	Defines sample output from a computer program
<kbd>	Defines keyboard input
<var>	Defines a variable

Browser Support

Element					
	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

In HTML 4.01, the tag defines strong emphasized text, but in HTML5 it defines important text.

Global Attributes

The tag also supports the [Global Attributes in HTML](#).

Event Attributes

The tag also supports the [Event Attributes in HTML](#).

HTML <style> Tag

Example

Use of the <style> element in an HTML document:

```
<html>
<head>
<style>
h1 {color:red;}
p {color:blue;}
</style>
</head>
<body>

<h1>A heading</h1>
<p>A paragraph.</p>

</body>
</html>
```

Definition and Usage

The <style> tag is used to define style information for an HTML document.

Inside the <style> element you specify how HTML elements should render in a browser.

Each HTML document can contain multiple <style> tags.

Browser Support

Element					
<style>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: To link to an external style sheet, use the [<link>](#) tag.

Tip: To learn more about style sheets, please read our [CSS Tutorial](#).

Attributes

= New in HTML5.

Attribute	Value	Description
<u>media</u>	<i>media_query</i>	Specifies what media/device the media resource is optimized for
<u>type</u>	text/css	Specifies the media type of the <style> tag

Global Attributes

The <style> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <style> tag also supports the [Event Attributes in HTML](#).

HTML <sub> Tag

Example

Subscript text:

```
<p>This text contains <sub>subscript</sub> text.</p>
```

Definition and Usage

The <sub> tag defines subscript text. Subscript text appears half a character below the normal line, and is sometimes rendered in a smaller font. Subscript text can be used for chemical formulas, like H₂O.

Tip: Use the <sup> tag to define superscripted text.

Browser Support

Element					
<sub>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The <sub> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <sub> tag also supports the [Event Attributes in HTML](#).

HTML <summary> Tag

Example

Using the <summary> element:

```
<details>
  <summary>Copyright 1999-2014.</summary>
  <p> - by Refsnes Data. All Rights Reserved.</p>
  <p>All content and graphics on this web site are the property of the
company Refsnes Data.</p>
</details>
```

Definition and Usage

The <summary> tag defines a visible heading for the <details> element. The heading can be clicked to view/hide the details.

Browser Support

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

Element					
<summary>	12.0	Not supported	48.0	6.0	15.0

Differences Between HTML 4.01 and HTML5

The <summary> tag is new in HTML5.

Tips and Notes

Note: The <summary> element should be the first child element of the <details> element.

Global Attributes

The <summary> tag also supports the Global Attributes in HTML.

Event Attributes

The <summary> tag also supports the Event Attributes in HTML.

HTML <sup> Tag

Example

Superscript text:

```
<p>This text contains <sup>superscript</sup> text.</p>
```

Definition and Usage

The <sup> tag defines superscript text. Superscript text appears half a character above the normal line, and is sometimes rendered in a smaller font. Superscript text can be used for footnotes, like WWW^[1].

Tip: Use the <sub> tag to define subscript text.

Browser Support

Element					
<sup>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The <sup> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <sup> tag also supports the [Event Attributes in HTML](#).

HTML <svg> Tag

Example

Draw a circle:

```
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4"
  fill="yellow" />
</svg>
```

More "Try it Yourself" examples below.

Definition and Usage

The <svg> tag defines a container for SVG graphics.

SVG has several methods for drawing paths, boxes, circles, text, and graphic images.

To learn more about SVG, please read our [SVG Tutorial](#).

Browser Support

Element					
<svg>	4.0	9.0	3.0	3.2	10.1

More Examples

Example

Draw a rectangle:

```
<svg width="400" height="100">
  <rect width="400" height="100" style="fill:rgb(0,0,255);stroke-
width:10;stroke:rgb(0,0,0)" />
</svg>
```

Example

Draw a square with rounded corners:

```
<svg width="400" height="180">
  <rect x="50" y="20" rx="20" ry="20" width="150" height="150"
style="fill:red;stroke:black;stroke-width:5;opacity:0.5" />
</svg>
```

Example

Draw a star:

```
<svg width="300" height="200">
  <polygon points="100,10 40,198 190,78 10,78 160,198"
style="fill:lime;stroke:purple;stroke-width:5;fill-rule:evenodd;" />
</svg>
```

Example

Draw an SVG logo:

```
<svg height="130" width="500">
<defs>
<linearGradient id="grad1" x1="0%" y1="0%" x2="100%" y2="0%">
  <stop offset="0%" style="stop-color:rgb(255,255,0);stop-opacity:1" />
  <stop offset="100%" style="stop-color:rgb(255,0,0);stop-opacity:1" />
</linearGradient>
</defs>

<ellipse cx="100" cy="70" rx="85" ry="55" fill="url(#grad1)" />

<text fill="#ffffff" font-size="45" font-family="Verdana" x="50"
y="86">SVG</text>
</svg>
```

HTML <table> Tag

Example

A simple HTML table, containing two columns and two rows:

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
</table>
```

More "Try it Yourself" examples below.

Definition and Usage

The <table> tag defines an HTML table.

An HTML table consists of the <table> element and one or more <tr>, <th>, and <td> elements.

The <tr> element defines a table row, the <th> element defines a table header, and the <td> element defines a table cell.

A more complex HTML table may also include <caption>, <col>, <colgroup>, <thead>, <tfoot>, and <tbody> elements.

Note: Tables should not be used for page layout! Historically, some Web authors have misused tables in HTML as a way to control their page layout. However, there are a variety of alternatives to using HTML tables for layout, primarily using CSS.

Browser Support

Element					
<table>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

The "align", "bgcolor", "border", "cellpadding", "cellspacing", "frame", "rules", "summary", and "width" attributes are not supported in HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	left center right	Not supported in HTML5. Specifies the alignment of a table according to surrounding text
<u>bgcolor</u>	<i>rgb(x,x,x)</i> <i>#xxxxxx</i> <i>colorname</i>	Not supported in HTML5. Specifies the background color for a table
<u>border</u>	10	Not supported in HTML5. Specifies whether or not the table is being used for layout purposes
<u>cellpadding</u>	<i>pixels</i>	Not supported in HTML5. Specifies the space between the cell wall and the cell content
<u>cellspacing</u>	<i>pixels</i>	Not supported in HTML5. Specifies the space between cells
<u>frame</u>	void above below hsides lhs rhs vsides box border	Not supported in HTML5. Specifies which parts of the outside borders that should be visible
<u>rules</u>	none groups rows cols all	Not supported in HTML5. Specifies which parts of the inside borders that should be visible
<u>summary</u>	<i>text</i>	Not supported in HTML5. Specifies a summary of the content of a table
<u>width</u>	<i>pixels %</i>	Not supported in HTML5. Specifies the width of a table

Global Attributes

The <table> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <table> tag also supports the [Event Attributes in HTML](#).

Try it Yourself - Examples

[Table headers](#) How to create table headers.

Table with a caption An HTML table with a caption.

Tags inside a table How to display elements inside other elements.

Cells that span more than one row/column How to define table cells that span more than one row or one column.

HTML <tbody> Tag

Example

An HTML table with a <thead>, <tbody>, and a <tfoot> element:

```
<table>
  <thead>
    <tr>
      <th>Month</th>
      <th>Savings</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>January</td>
      <td>$100</td>
    </tr>
    <tr>
      <td>February</td>
      <td>$80</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Sum</td>
      <td>$180</td>
    </tr>
  </tfoot>
</table>
```

Definition and Usage

The <tbody> tag is used to group the body content in an HTML table.

The <tbody> element is used in conjunction with the <thead> and <tfoot> elements to specify each part of a table (body, header, footer).

Browsers can use these elements to enable scrolling of the table body independently of the header and footer. Also, when printing a large table that spans multiple pages, these elements can enable the table header and footer to be printed at the top and bottom of each page.

The <tbody> tag must be used in the following context: As a child of a <table> element, after any <caption>, <colgroup>, and <thead> elements.

Browser Support

Element					
<tbody>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: The <tbody> element must have one or more <tr> tags inside.

Tip: The <thead>, <tbody>, and <tfoot> elements will not affect the layout of the table by default. However, you can use CSS to style these elements.

Differences Between HTML 4.01 and HTML5

None of the HTML 4.01 attributes are supported in HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	right left center justify char	Not supported in HTML5. Aligns the content inside the <tbody> element
<u>char</u>	<i>character</i>	Not supported in HTML5. Aligns the content inside the <tbody> element to a character
<u>charoff</u>	<i>number</i>	Not supported in HTML5. Sets the number of characters the content inside the <tbody> element will be aligned from the character specified by the char attribute
<u>valign</u>	top middle bottom baseline	Not supported in HTML5. Vertical aligns the content inside the <tbody> element

Global Attributes

The <tbody> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <tbody> tag also supports the [Event Attributes in HTML](#).

Default CSS Settings

Most browsers will display the <tbody> element with the following default values:

```
tbody {  
    display: table-row-group;  
    vertical-align: middle;  
    border-color: inherit;  
}
```

HTML <td> Tag

Example

A simple HTML table, with two table cells:

```
<table>
  <tr>
    <td>Cell A</td>
    <td>Cell B</td>
  </tr>
</table>
```

More "Try it Yourself" examples below.

Definition and Usage

The <td> tag defines a standard cell in an HTML table.

An HTML table has two kinds of cells:

- Header cells - contains header information (created with the <th> element)
- Standard cells - contains data (created with the <td> element)

The text in <th> elements are bold and centered by default.

The text in <td> elements are regular and left-aligned by default.

Browser Support

Element					
<td>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: Use the colspan and rowspan attribute to let the content span over multiple columns or rows!

Differences Between HTML 4.01 and HTML5

All layout attributes are removed in HTML5.

Attributes

Attribute	Value	Description
<u>abbr</u>	<i>text</i>	Not supported in HTML5. Specifies an abbreviated version of the content in a cell
<u>align</u>	left right center justify char	Not supported in HTML5. Aligns the content in a cell
<u>axis</u>	<i>category_name</i>	Not supported in HTML5. Categorizes cells
<u>bgcolor</u>	<i>rgb(x,x,x)</i> <i>#xxxxxx</i> <i>colorname</i>	Not supported in HTML5. Specifies the background color of a cell
<u>char</u>	<i>character</i>	Not supported in HTML5. Aligns the content in a cell to a character
<u>charoff</u>	<i>number</i>	Not supported in HTML5. Sets the number of characters the content will be aligned from the character specified by the char attribute
<u>colspan</u>	<i>number</i>	Specifies the number of columns a cell should span
<u>headers</u>	<i>header_id</i>	Specifies one or more header cells a cell is related to
<u>height</u>	<i>pixels %</i>	Not supported in HTML5. Sets the height of a cell
<u>nowrap</u>	<i>nowrap</i>	Not supported in HTML5. Specifies that the content inside a cell should not wrap
<u>rowspan</u>	<i>number</i>	Sets the number of rows a cell should span
<u>scope</u>	col colgroup row rowgroup	Not supported in HTML5. Defines a way to associate header cells and data cells in a table
<u>valign</u>	top middle bottom baseline	Not supported in HTML5. Vertical aligns the content in a cell
<u>width</u>	<i>pixels %</i>	Not supported in HTML5. Specifies the width of a cell

Global Attributes

The <td> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The `<td>` tag also supports the [Event Attributes in HTML](#).

Try it Yourself - Examples

[Table headers](#) How to create table headers.

[Table with a caption](#) An HTML table with a caption.

[Tags inside a table](#) How to display elements inside other elements.

[Cells that span more than one row/column](#) How to define table cells that span more than one row or one column.

HTML <template> Tag

Example

The template element holds HTML code without displaying it:

```
<template>
  <h2>Flower</h2>
  
</template>
```

Definition and Usage

The <template> tag holds its content hidden from the client.

Content inside a <template> tag will not be rendered.

The content can be visible and rendered later by using JavaScript.

Use the <template> tag when you have HTML code you want to use over and over again, but not until you ask for it. To do this *without* the <template> tag, you have to create the HTML code with JavaScript to prevent the browser from rendering the code.

Browser Support

Element					
<template>	26.0	13.0	22.0	9	15.0

Differences Between HTML 4.01 and HTML5

The <template> tag is new in HTML5.

Global Attributes

The <template> tag supports the [Global Attributes in HTML](#).

More Examples

Example

Use JavaScript to get the content from a template, and add it to the page:

```
function showContent() {  
    var temp = document.getElementsByTagName("template")[0];  
    var clon = temp.content.cloneNode(true);  
    document.body.appendChild(clon);  
}
```

Example

Use the content of a template for each item in an array:

```
<template>  
    <div class="myClass">I like: </div>  
</template>  
  
<script>  
var myArr = ["Audi", "BMW", "Ford", "Honda", "Jaguar", "Nissan"];  
  
function showContent() {  
    var temp, item, a, i;  
    //get the template element:  
    temp = document.getElementsByTagName("template")[0];  
    //get the DIV element from the template:  
    item = temp.content.querySelector("div");  
    //for each item in the array:  
    for (i = 0; i < myArr.length; i++) {  
        //Create a new node, based on the template:  
        a = document.importNode(item, true);  
        //Add data from the array:  
        a.textContent += myArr[i];  
        //append the new node wherever you like:  
        document.body.appendChild(a);  
    }  
}  
</script>
```

Example

Test browser support for the template element::

```
if (document.createElement("template").content) {  
    /*Code for browsers that supports the TEMPLATE element*/  
} else {  
    /*Alternative code for browsers that do not support the TEMPLATE  
element*/  
}
```

HTML <textarea> Tag

Example

An HTML text area:

```
<textarea rows="4" cols="50">  
At w3schools.com you will learn how to make a website. We offer free  
tutorials in all web development technologies.  
</textarea>
```

Definition and Usage

The <textarea> tag defines a multi-line text input control.

A text area can hold an unlimited number of characters, and the text renders in a fixed-width font (usually Courier).

The size of a text area can be specified by the cols and rows attributes, or even better; through CSS' height and width properties.

Browser Support

Element					
<textarea>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

HTML5 has added several new attributes.

Attributes

= New in HTML5.

Attribute	Value	Description
<u>autofocus</u>	autofocus	Specifies that a text area should automatically get focus when the page loads
<u>cols</u>	<i>number</i>	Specifies the visible width of a text area
<u>dirname</u>	<i>textareaname.dir</i>	Specifies that the text direction of the textarea will be submitted
<u>disabled</u>	disabled	Specifies that a text area should be disabled
<u>form</u>	<i>form_id</i>	Specifies one or more forms the text area belongs to
<u>maxlength</u>	<i>number</i>	Specifies the maximum number of characters allowed in the text area
<u>name</u>	<i>text</i>	Specifies a name for a text area
<u>placeholder</u>	<i>text</i>	Specifies a short hint that describes the expected value of a text area
<u>readonly</u>	readonly	Specifies that a text area should be read-only
<u>required</u>	required	Specifies that a text area is required/must be filled out
<u>rows</u>	<i>number</i>	Specifies the visible number of lines in a text area
<u>wrap</u>	hard soft	Specifies how the text in a text area is to be wrapped when submitted in a form

Global Attributes

The <textarea> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <textarea> tag also supports the [Event Attributes in HTML](#).

HTML <tfoot> Tag

Example

An HTML table with a <thead>, <tbody>, and a <tfoot> element:

```
<table>
  <thead>
    <tr>
      <th>Month</th>
      <th>Savings</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>January</td>
      <td>$100</td>
    </tr>
    <tr>
      <td>February</td>
      <td>$80</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Sum</td>
      <td>$180</td>
    </tr>
  </tfoot>
</table>
```

Definition and Usage

The <tfoot> tag is used to group footer content in an HTML table.

The <tfoot> element is used in conjunction with the <thead> and <tbody> elements to specify each part of a table (footer, header, body).

Browsers can use these elements to enable scrolling of the table body independently of the header and footer. Also, when printing a large table that spans multiple pages, these elements can enable the table header and footer to be printed at the top and bottom of each page.

The <tfoot> tag must be used in the following context: As a child of a <table> element, after any <caption>, <colgroup>, <thead>, and <tbody> elements.

Browser Support

Element					
<tfoot>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: The <tfoot> element must have one or more <tr> tags inside.

Tip: The <thead>, <tbody>, and <tfoot> elements will not affect the layout of the table by default. However, you can use CSS to style these elements.

Differences Between HTML 4.01 and HTML5

None of the HTML 4.01 attributes are supported in HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	right left center justify char	Not supported in HTML5. Aligns the content inside the <tfoot> element
<u>char</u>	<i>character</i>	Not supported in HTML5. Aligns the content inside the <tfoot> element to a character
<u>charoff</u>	<i>number</i>	Not supported in HTML5. Sets the number of characters the content inside the <tfoot> element will be aligned from the character specified by the char attribute
<u>valign</u>	top middle bottom baseline	Not supported in HTML5. Vertical aligns the content inside the <tfoot> element

Global Attributes

The <tfoot> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <tfoot> tag also supports the [Event Attributes in HTML](#).

Default CSS Settings

Most browsers will display the <tfoot> element with the following default values:

```
tfoot {  
    display: table-footer-group;  
    vertical-align: middle;  
    border-color: inherit;  
}
```

HTML <th> Tag

Example

A simple HTML table with two header cells and two data cells:

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
</table>
```

More "Try it Yourself" examples below.

Definition and Usage

The <th> tag defines a header cell in an HTML table.

An HTML table has two kinds of cells:

- Header cells - contains header information (created with the <th> element)
- Standard cells - contains data (created with the <td> element)

The text in <th> elements are bold and centered by default.

The text in <td> elements are regular and left-aligned by default.

Browser Support

Element					
<th>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: Use the colspan and rowspan attribute to let the content span over multiple columns or rows!

Differences Between HTML 4.01 and HTML5

All layout attributes are removed in HTML5.

Attributes

Attribute	Value	Description
<u>abbr</u>	<i>text</i>	Specifies an abbreviated version of the content in a header cell
<u>align</u>	left right center justify char	Not supported in HTML5. Aligns the content in a header cell
<u>axis</u>	<i>category_name</i>	Not supported in HTML5. Categorizes header cells
<u>bgcolor</u>	<i>rgb(x,x,x) #xxxxxx colorname</i>	Not supported in HTML5. Specifies the background color of a header cell
<u>char</u>	<i>character</i>	Not supported in HTML5. Aligns the content in a header cell to a character
<u>charoff</u>	<i>number</i>	Not supported in HTML5. Sets the number of characters the content will be aligned from the character specified by the char attribute
<u>colspan</u>	<i>number</i>	Specifies the number of columns a header cell should span
<u>headers</u>	<i>header_id</i>	Specifies one or more header cells a cell is related to
<u>height</u>	<i>pixels %</i>	Not supported in HTML5. Sets the height of a header cell
<u>nowrap</u>	<i>nowrap</i>	Not supported in HTML5. Specifies that the content inside a header cell should not wrap
<u>rowspan</u>	<i>number</i>	Specifies the number of rows a header cell should span
<u>scope</u>	col colgroup row rowgroup	Specifies whether a header cell is a header for a column, row, or group of columns or rows
<u>sorted</u>	<i>reversed</i> <i>number</i> <i>reversed</i> <i>number</i> <i>number</i> <i>reversed</i>	Defines the sort direction of a column
<u>valign</u>	top middle bottom baseline	Not supported in HTML5. Vertical aligns the content in a header cell
<u>width</u>	<i>pixels %</i>	Not supported in HTML5. Specifies the width of a header cell

Global Attributes

The <th> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <th> tag also supports the [Event Attributes in HTML](#).

Try it Yourself - Examples

[Table headers](#) How to create table headers.

[Table with a caption](#) An HTML table with a caption.

[Tags inside a table](#) How to display elements inside other elements.

[Cells that span more than one row/column](#) How to define table cells that span more than one row or one column.

HTML <thead> Tag

Example

An HTML table with a <thead>, <tbody>, and a <tfoot> element:

```
<table>
  <thead>
    <tr>
      <th>Month</th>
      <th>Savings</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>January</td>
      <td>$100</td>
    </tr>
    <tr>
      <td>February</td>
      <td>$80</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Sum</td>
      <td>$180</td>
    </tr>
  </tfoot>
</table>
```

Definition and Usage

The <thead> tag is used to group header content in an HTML table.

The <thead> element is used in conjunction with the <tbody> and <tfoot> elements to specify each part of a table (header, body, footer).

Browsers can use these elements to enable scrolling of the table body independently of the header and footer. Also, when printing a large table that spans multiple pages, these elements can enable the table header and footer to be printed at the top and bottom of each page.

The `<thead>` tag must be used in the following context: As a child of a `<table>` element, after any `<caption>`, and `<colgroup>` elements, and before any `<tbody>`, `<tfoot>`, and `<tr>` elements.

Browser Support

Element					
<code><thead></code>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: The `<thead>` element must have one or more `<tr>` tags inside.

Tip: The `<thead>`, `<tbody>`, and `<tfoot>` elements will not affect the layout of the table by default. However, you can use CSS to style these elements.

Differences Between HTML 4.01 and HTML5

None of the HTML 4.01 attributes are supported in HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	right left center justify char	Not supported in HTML5. Aligns the content inside the <code><thead></code> element
<u>char</u>	<i>character</i>	Not supported in HTML5. Aligns the content inside the <code><thead></code> element to a character
<u>charoff</u>	<i>number</i>	Not supported in HTML5. Sets the number of characters the content inside the <code><thead></code> element will be aligned from the character specified by the <code>char</code> attribute
<u>valign</u>	top middle bottom baseline	Not supported in HTML5. Vertical aligns the content inside the <code><thead></code> element

Global Attributes

The `<thead>` tag also supports the [Global Attributes in HTML](#).

Event Attributes

The `<thead>` tag also supports the [Event Attributes in HTML](#).

Default CSS Settings

Most browsers will display the <thead> element with the following default values:

```
thead {  
    display: table-header-group;  
    vertical-align: middle;  
    border-color: inherit;  
}
```

HTML <time> Tag

Example

How to define a time and a date:

```
<p>We open at <time>10:00</time> every morning.</p>  
<p>I have a date on <time datetime="2008-02-14 20:00">Valentines  
day</time>.</p>
```

Definition and Usage

The <time> tag defines a human-readable date/time.

This element can also be used to encode dates and times in a machine-readable way so that user agents can offer to add birthday reminders or scheduled events to the user's calendar, and search engines can produce smarter search results.

Browser Support

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

Element					
<time>	6.0	9.0	4.0	5.0	11.1

Differences Between HTML 4.01 and HTML5

The <time> tag is new in HTML5.

Attributes

Attribute	Value	Description
<u>datetime</u>	<i>datetime</i>	Represent a machine-readable date/time of the <time> element

Global Attributes

The <time> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <time> tag also supports the [Event Attributes in HTML](#).

HTML <title> Tag

Example

Define a title for your HTML document:

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML Reference</title>
</head>

<body>
The content of the document.....
</body>

</html>
```

Definition and Usage

The <title> tag is required in all HTML documents and it defines the title of the document.
The <title> element:

- defines a title in the browser toolbar
- provides a title for the page when it is added to favorites
- displays a title for the page in search-engine results

Browser Support

Element					
<title>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Note: You can NOT have more than one <title> element in an HTML document.

Tip: If you omit the <title> tag, the document will not validate as HTML.

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The <title> tag also supports the [Global Attributes in HTML](#).

HTML <tr> Tag

Example

A simple HTML table, containing two columns and two rows:

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
</table>
```

More "Try it Yourself" examples below.

Definition and Usage

The <tr> tag defines a row in an HTML table.

A <tr> element contains one or more <th> or <td> elements.

Browser Support

Element					
<tr>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

All the layout attributes are removed in HTML5.

Attributes

Attribute	Value	Description
<u>align</u>	right left center justify char	Not supported in HTML5. Aligns the content in a table row
<u>bgcolor</u>	<i>rgb(x,x,x)</i> <i>#xxxxxx</i> <i>colorname</i>	Not supported in HTML5. Specifies a background color for a table row
<u>char</u>	<i>character</i>	Not supported in HTML5. Aligns the content in a table row to a character
<u>charoff</u>	<i>number</i>	Not supported in HTML5. Sets the number of characters the content will be aligned from the character specified by the char attribute
<u>valign</u>	top middle bottom baseline	Not supported in HTML5. Vertical aligns the content in a table row

Global Attributes

The <tr> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <tr> tag also supports the [Event Attributes in HTML](#).

Try it Yourself - Examples

[Table headers](#) How to create table headers.

[Table with a caption](#) An HTML table with a caption.

[Tags inside a table](#) How to display elements inside other elements.

[Cells that span more than one row/column](#) How to define table cells that span more than one row or one column.

HTML <track> Tag

Example

A video with two subtitle tracks:

```
<video width="320" height="240" controls>
  <source src="forrest_gump.mp4" type="video/mp4">
  <source src="forrest_gump.ogg" type="video/ogg">
  <track src="subtitles_en.vtt" kind="subtitles" srclang="en"
label="English">
  <track src="subtitles_no.vtt" kind="subtitles" srclang="no"
label="Norwegian">
</video>
```

Definition and Usage

The <track> tag specifies text tracks for media elements (<audio> and <video>). This element is used to specify subtitles, caption files or other files containing text, that should be visible when the media is playing.

Browser Support

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

Element					
<track>	18.0	10.0	31.0	6.0	15.0

Differences Between HTML 4.01 and HTML5

The <track> tag is new in HTML5.

Optional Attributes

Attribute	Value	Description
<u>default</u>	default	Specifies that the track is to be enabled if the user's preferences do not indicate that another track would be more appropriate
<u>kind</u>	captions chapters descriptions metadata subtitles	Specifies the kind of text track
<u>label</u>	<i>text</i>	Specifies the title of the text track
<u>src</u>	<i>URL</i>	Required. Specifies the URL of the track file
<u>srlang</u>	<i>language_code</i>	Specifies the language of the track text data (required if kind="subtitles")

Global Attributes

The <track> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <track> tag also supports the [Event Attributes in HTML](#).

HTML <tt> Tag. Not Supported in HTML5.

Example

Define teletype text:

```
<p><tt>Teletype text</tt></p>
```

Definition and Usage

The <tt> tag is not supported in HTML5.

If <tt> was used for marking up keyboard input, consider the [<kbd>](#) element; for variables, consider the [<var>](#) element; for computer code, consider the [<code>](#) element; and for computer output, consider the [<samp>](#) element, or use CSS instead.

The <tt> tag defines teletype text.

Browser Support

Element					
<tt>	Yes	Yes	Yes	Yes	Yes

Compatibility Notes

CSS syntax: <p style="font-family:'Lucida Console', monospace">

[CSS Example: define teletype/monospace font for a <p> element](#)

In our CSS tutorial you can find more details about specifying the [type of font](#) in a document.

Differences Between HTML 4.01 and HTML5

The <tt> tag is not supported in HTML5.

HTML <u> Tag

Example

Underline a misspelled word with the <u> tag:

```
<p>This is a <u>parragraph</u>.</p>
```

Definition and Usage

The <u> tag represents some text that should be stylistically different from normal text, such as misspelled words or proper nouns in Chinese.

Browser Support

Element					
<u>	Yes	Yes	Yes	Yes	Yes

Tips and Notes

Tip: Avoid using the <u> element where it could be confused for a hyperlink.

Note: The HTML 5 specification reminds developers that other elements are almost always more appropriate than <u>.

Differences Between HTML 4.01 and HTML5

The <u> element was deprecated in HTML 4.01. (the <u> element was used to define underlined text).

The <u> element is redefined in HTML5, to represent text that should be stylistically different from normal text, such as misspelled words or proper nouns in Chinese.

Global Attributes

The <u> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <u> tag also supports the [Event Attributes in HTML](#).

HTML <var> Tag

Example

Format text in a document:

```
<var>Variable</var>
```

Definition and Usage

The <var> tag is a phrase tag. It defines a variable.

Tip: This tag is not deprecated, but it is possible to achieve richer effect with CSS.

All phrase tags:

Tag	Description
<u></u>	Renders as emphasized text
<u></u>	Defines important text
<u><code></u>	Defines a piece of computer code
<u><samp></u>	Defines sample output from a computer program
<u><kbd></u>	Defines keyboard input
<u><var></u>	Defines a variable

Browser Support

Element					
<var>	Yes	Yes	Yes	Yes	Yes

Differences Between HTML 4.01 and HTML5

NONE.

Global Attributes

The <var> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <var> tag also supports the [Event Attributes in HTML](#).

HTML <video> Tag

Example

Play a video:

```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Definition and Usage

The <video> tag specifies video, such as a movie clip or other video streams.

Currently, there are 3 supported video formats for the <video> element: MP4, WebM, and Ogg:

Browser	MP4	WebM	Ogg
Internet Explorer	YES	NO	NO
Chrome	YES	YES	YES
Firefox	YES from Firefox 21 from Firefox 30 for Linux	YES	YES
Safari	YES	NO	NO
Opera	YES From Opera 25	YES	YES

- MP4 = MPEG 4 files with H264 video codec and AAC audio codec
- WebM = WebM files with VP8 video codec and Vorbis audio codec
- Ogg = Ogg files with Theora video codec and Vorbis audio codec

MIME Types for Video Formats

Format	MIME-type
MP4	video/mp4
WebM	video/webm
Ogg	video/ogg

Browser Support

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

Element					
<video>	4.0	9.0	3.5	4.0	10.5

Differences Between HTML 4.01 and HTML5

The <video> tag is new in HTML5.

Tips and Notes

Tip: Any text between the <video> and </video> tags will be displayed in browsers that do not support the <video> element.

Optional Attributes

Attribute	Value	Description
<u>autoplay</u>	autoplay	Specifies that the video will start playing as soon as it is ready
<u>controls</u>	controls	Specifies that video controls should be displayed (such as a play/pause button etc).
<u>height</u>	<i>pixels</i>	Sets the height of the video player
<u>loop</u>	loop	Specifies that the video will start over again, every time it is finished
<u>muted</u>	muted	Specifies that the audio output of the video should be muted
<u>poster</u>	<i>URL</i>	Specifies an image to be shown while the video is downloading, or until the user hits the play button
<u>preload</u>	auto metadata none	Specifies if and how the author thinks the video should be loaded when the page loads
<u>src</u>	<i>URL</i>	Specifies the URL of the video file
<u>width</u>	<i>pixels</i>	Sets the width of the video player

Global Attributes

The <video> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <video> tag also supports the [Event Attributes in HTML](#).

HTML <wbr> Tag

Example

A text with word break opportunities:

```
<p>
To learn AJAX, you must be familiar with the XML<wbr>Http<wbr>Request
Object.
</p>
```

Definition and Usage

The <wbr> (Word Break Opportunity) tag specifies where in a text it would be ok to add a line-break.

Tip: When a word is too long, or you are afraid that the browser will break your lines at the wrong place, you can use the <wbr> element to add word break opportunities.

Browser Support

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

Element					
<wbr>	1.0	12.0	3.0	4.0	11.7

Differences Between HTML 4.01 and HTML5

The <wbr> tag is new in HTML5.

Global Attributes

The <wbr> tag also supports the [Global Attributes in HTML](#).

Event Attributes

The <wbr> tag also supports the [Event Attributes in HTML](#).

HTML accept Attribute

Definition and Usage

The accept attribute specifies the types of files that the server accepts (that can be submitted through a file upload).

Note: The accept attribute can only be used with `<input type="file">`.

Tip: Do not use this attribute as a validation tool. File uploads should be validated on the server.

Applies to

The accept attribute can be used on the following element:

Element	Attribute
<code><input></code>	<u>accept</u>

Example

Input Example

Specify that the server accepts only image files in the file upload:

```
<form action="/action_page.php">
  <input type="file" name="pic" accept="image/*">
  <input type="submit">
</form>
```

Browser Support

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

Attribute					
accept	8.0	10.0	4.0	6.0	15.0

HTML accept-charset Attribute

Definition and Usage

The accept-charset attribute specifies the character encodings that are to be used for the form submission.

The default value is the reserved string "UNKNOWN" (indicates that the encoding equals the encoding of the document containing the <form> element).

Applies to

The accept-charset attribute can be used on the following element:

Element	Attribute
<form>	<u>accept-charset</u>

Form Example

A form with an accept-charset attribute:

```
<form action="/action_page.php" accept-charset="ISO-8859-1">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit" value="Submit">
</form>
```

Browser Support

Attribute					
accept-charset	Yes	Yes	Yes	Yes	Yes

HTML accesskey Attribute

Definition and Usage

The accesskey attribute specifies a shortcut key to activate/focus an element.

Note: The way of accessing the shortcut key is varying in different browsers:

Browser	Windows	Linux	Mac
Internet Explorer	[Alt] + <i>accesskey</i>	N/A	
Chrome	[Alt] + <i>accesskey</i>	[Alt] + <i>accesskey</i>	[Control] [Alt] + <i>accesskey</i>
Firefox	[Alt] [Shift] + <i>accesskey</i>	[Alt] [Shift] + <i>accesskey</i>	[Control] [Alt] + <i>accesskey</i>
Safari	[Alt] + <i>accesskey</i>	N/A	[Control] [Alt] + <i>accesskey</i>
Opera	Opera 15 or newer: [Alt] + <i>accesskey</i> Opera 12.1 or older: [Shift] [Esc] + <i>accesskey</i>		

However, in most browsers the shortcut can be set to another combination of keys.

Tip: The behavior if more than one element has the same access key differs:

- IE, Firefox: The next element with the pressed access key will be activated
- Chrome, Safari: The last element with the pressed access key will be activated
- Opera: The first element with the pressed access key will be activated

Applies to

The accesskey is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	<u>accesskey</u>

Example

Two hyperlinks with specified accesskeys:

```
<a href="https://www.w3schools.com/html5" accesskey="h">HTML5</a><br>
<a href="https://www.w3schools.com/css3" accesskey="c">CSS3</a>
```

Browser Support

Attribute					
accesskey	Yes	Yes	Yes	Yes	Yes

HTML action Attribute

Definition and Usage

The action attribute specifies where to send the form-data when a form is submitted.

Applies to

The action attribute can be used on the following element:

Element	Attribute
<u><form></u>	<u>action</u>

Example

Form Example

On submit, send the form-data to a file named "/action_page.php" (to process the input):

```
<form action="/action_page.php" method="get">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit" value="Submit">
</form>
```

Browser Support

Attribute					
action	Yes	Yes	Yes	Yes	Yes

HTML alt Attribute

Definition and Usage

The alt attribute provides alternative information for an image if a user for some reason cannot view it (because of slow connection, an error in the src attribute, or if the user uses a screen reader).

Note: The alt attribute is required for the `` element.

Note: For `<input>` elements, the alt attribute can only be used with `<input type="image">`.

Tip: To create a tooltip for an image, use the title attribute!

Applies to

The alt attribute can be used on the following elements:

Elements	Attribute
<code><area></code>	<u>alt</u>
<code></code>	<u>alt</u>
<code><input></code>	<u>alt</u>

Examples

Area Example

An image-map, with clickable areas:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>
```

Img Example

An image with an alternate text specified:

```

```

Input Example

An HTML form with an image that represents the submit button:

```
<form action="/action_page.php">
  First name: <input type="text" name="fname"><br>
  <input type="image" src="submit.gif" alt="Submit" width="48" height="48">
</form>
```

Browser Support

The `alt` attribute has the following browser support for each element:

Element					
area	Yes	Yes	Yes	Yes	Yes
img	Yes	Yes	Yes	Yes	Yes
input	Yes	Yes	Yes	Yes	Yes

HTML async Attribute

Definition and Usage

The `async` attribute is a boolean attribute.

When present, it specifies that the script will be executed asynchronously as soon as it is available.

Note: The `async` attribute is only for external scripts (and should only be used if the `src` attribute is present).

Note: There are several ways an external script can be executed:

- If `async` is present: The script is executed asynchronously with the rest of the page (the script will be executed while the page continues the parsing)
- If `async` is not present and `defer` is present: The script is executed when the page has finished parsing
- If neither `async` or `defer` is present: The script is fetched and executed immediately, before the browser continues parsing the page

Applies to

The `async` attribute can be used on the following element:

Elements	Attribute
<code><script></code>	<code>async</code>

Example

Script Example

A script that will be run asynchronously as soon as it is available:

```
<script src="demo_async.js" async></script>
```

Browser Support

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

Attribute					
<code>async</code>	Yes	10.0	3.6	Yes	Yes

HTML autocomplete Attribute

Definition and Usage

The 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.

Tip: It is possible to have autocomplete "on" for the form, and "off" for specific input fields, or vice versa.

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

Applies to

The autocomplete attribute can be used on the following elements:

Elements	Attribute
<form>	<u>autocomplete</u>
<input>	<u>autocomplete</u>

Examples

Form Example

A form with autocomplete on:

```
<form action="/action_page.php" method="get" autocomplete="on">
    First name:<input type="text" name="fname"><br>
    E-mail: <input type="email" name="email"><br>
    <input type="submit">
</form>
```

Input Example

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

```
<form action="/action_page.php" autocomplete="on">
    First name:<input type="text" name="fname"><br>
    Last name: <input type="text" name="lname"><br>
    E-mail: <input type="email" name="email" autocomplete="off"><br>
    <input type="submit">
</form>
```

Browser Support

The `autocomplete` attribute has the following browser support for each element:

Element					
form	Yes	Yes	4.0	5.2	15.0
input	17.0	5.0	4.0	5.2	9.6

HTML autofocus Attribute

Definition and Usage

The autofocus attribute is a boolean attribute.

When present, it specifies that the element should automatically get focus when the page loads.

Applies to

The autofocus attribute can be used on the following elements:

Elements	Attribute
<u><button></u>	<u>autofocus</u>
<u><input></u>	<u>autofocus</u>
<u><select></u>	<u>autofocus</u>
<u><textarea></u>	<u>autofocus</u>

Examples

Button Example

A button with autofocus:

```
<button type="button" autofocus>Click Me!</button>
```

Input Example

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

```
<form action="/action_page.php">
  First name: <input type="text" name="fname" autofocus><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit">
</form>
```

Textarea Example

A text area with autofocus:

```
<textarea autofocus>
At w3schools.com you will learn how to make a website. We offer free
tutorials in all web development technologies.
</textarea>
```

Browser Support

The `autofocus` attribute has the following browser support for each element:

Element					
form	5.0	10.0	4.0	5.0	9.6
input	5.0	10.0	4.0	5.0	9.6
textarea	Yes	10.0	4.0	Yes	Yes

HTML autoplay Attribute

Definition and Usage

The autoplay attribute is a boolean attribute.

When present, the audio/video will automatically start playing as soon as it can do so without stopping.

Applies to

The autoplay attribute can be used on the following elements:

Elements	Attribute
<code><audio></code>	<code>autoplay</code>
<code><video></code>	<code>autoplay</code>

Examples

Example

An audio file that will automatically start playing:

```
<audio controls autoplay>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>
```

Example

A video that will automatically start playing:

```
<video controls autoplay>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Browser Support

The `autoplay` attribute has the following browser support for each element:

Element					
audio	4.0	9.0	3.5	4.0	10.5
video	4.0	9.0	3.5	4.0	10.5

HTML charset Attribute

Definition and Usage

When used by the `<meta>` element, the `charset` attribute specifies the character encoding for the HTML document.

When used by the `<script>` element, the `charset` attribute specifies the character encoding used in an external script file.

Applies to

The `charset` attribute can be used on the following elements:

Elements	Attribute
<code><meta></code>	<code>charset</code>
<code><script></code>	<code>charset</code>

Examples

Meta Example

Specify the character encoding for the HTML document:

```
<head>
<meta charset="UTF-8">
</head>
```

Script Example

An external JavaScript with an UTF-8 character set:

```
<script src="myscripts.js" charset="UTF-8"></script>
```

Browser Support

The `charset` attribute has the following browser support for each element:

Element					
meta	Yes	Yes	Yes	Yes	Yes
script	Yes	Yes	Yes	Yes	Yes

HTML checked Attribute

Definition and Usage

The checked attribute is a boolean attribute.

When present, it specifies that an <input> element should be pre-selected (checked) when the page loads.

The checked attribute can be used with <input type="checkbox"> and <input type="radio">.

The checked attribute can also be set after the page load, with a JavaScript.

Applies to

The checked attribute can be used on the following element:

Element	Attribute
<input>	<u>checked</u>

Examples

Input Example

An HTML form with a pre-selected checkbox:

```
<form action="/action_page.php">
  <input type="checkbox" name="vehicle" value="Bike"> I have a bike<br>
  <input type="checkbox" name="vehicle" value="Car" checked> I have a
car<br>
  <input type="submit" value="Submit">
</form>
```

Browser Support

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

Attribute					
checked	1.0	2.0	1.0	1.0	1.0

HTML cite Attribute

Definition and Usage

The cite attribute specifies a URL to a document that explains the quote, or why the text was inserted/changed.

Applies to

The cite attribute can be used on the following elements:

Elements	Attribute
<u><blockquote></u>	<u>cite</u>
<u></u>	<u>cite</u>
<u><ins></u>	<u>cite</u>
<u><q></u>	<u>cite</u>

Examples

Blockquote Example

A section that is quoted from another source:

```
<blockquote cite="http://www.worldwildlife.org/who/index.html">
For 50 years, WWF has been protecting the future of nature. The world's
leading conservation organization, WWF works in 100 countries and is
supported by 1.2 million members in the United States and close to 5
million globally. WWF's unique way of working combines global reach with a
foundation in science, involves action at every level from local to global,
and ensures the delivery of innovative solutions that meet the needs of
both people and nature.
</blockquote>
```

Del Example

A deleted text, with a URL to a document that explains why the text was deleted:

```
<p><del cite="del_demo_cite.htm">This text has been deleted</del></p>
```

Ins Example

An inserted text, with a URL to a document that explains why the text was inserted:

```
<p>This is a text.  
<ins cite="why_inserted.htm">This is an inserted text.</ins></p>
```

Q Example

Specify the source URL of a quote:

```
<p>WWF's goal is to:  
<q cite="http://www.wwf.org">  
Build a future where people live in harmony with nature.</q>  
We hope they succeed.</p>
```

Browser Support

The `cite` attribute has the following browser support for each element:

Element					
blockquote	Yes	Yes	Yes	Yes	Yes
del	Yes	Yes	Yes	Yes	Yes
ins	Yes	Yes	Yes	Yes	Yes
q	Yes	Yes	Yes	Yes	Yes

HTML class Attribute

Definition and Usage

The class attribute specifies one or more classnames for an element.

The class attribute is mostly used to point to a class in a style sheet. However, it can also be used by a JavaScript (via the HTML DOM) to make changes to HTML elements with a specified class.

Applies to

The class attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	<u>class</u>

Example

Use of the class attribute in an HTML document:

```
<html>
<head>
<style>
h1.intro {
    color: blue;
}

p.important {
    color: green;
}
</style>
</head>
<body>

<h1 class="intro">Header 1</h1>
<p>A paragraph.</p>
<p class="important">Note that this is an important paragraph. :)</p>

</body>
</html>
```

Browser Support

Attribute					
class	Yes	Yes	Yes	Yes	Yes

HTML cols Attribute

Definition and Usage

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

Tip: The size of a textarea can also be set by the CSS height and width properties.

Applies to

The cols attribute can be used on the following element:

Elements	Attribute
<textarea>	<u>cols</u>

Example

Textarea Example

A text area with a specified height and width:

```
<textarea rows="4" cols="50">  
At w3schools.com you will learn how to make a website. We offer free  
tutorials in all web development technologies.  
</textarea>
```

Browser Support

Attribute					
cols	Yes	Yes	Yes	Yes	Yes

HTML colspan Attribute

Definition and Usage

The colspan attribute defines the number of columns a table cell should span.

Applies to

The colspan attribute can be used on the following elements:

Elements	Attribute
<u><td></u>	<u>colspan</u>
<u><th></u>	<u>colspan</u>

Examples

Td Example

An HTML table with a table cell that spans two columns:

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$100</td>
  </tr>
  <tr>
    <td colspan="2">Sum: $180</td>
  </tr>
</table>
```

Th Example

An HTML table with a header cell that spans two columns:

```
<table>
  <tr>
    <th colspan="2">Monthly Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$80</td>
  </tr>
</table>
```

Browser Support

The `colspan` attribute has the following browser support for each element:

Element					
td	Yes	Yes	Yes	Yes	Yes
th	Yes	Yes	Yes	Yes	Yes

HTML content Attribute

Definition and Usage

The content attribute gives the value associated with the http-equiv or name attribute.

Applies to

The content attribute can be used on the following element:

Element	Attribute
<u><meta></u>	<u>content</u>

Example

Meta Example

Describe metadata within an HTML document:

```
<head>
<meta name="description" content="Free Web tutorials">
<meta name="keywords" content="HTML,CSS,XML,JavaScript">
</head>
```

Browser Support

Attribute					
content	Yes	Yes	Yes	Yes	Yes

HTML contenteditable Attribute

Definition and Usage

The contenteditable attribute specifies whether the content of an element is editable or not.

Note: When the contenteditable attribute is not set on an element, the element will inherit it from its parent.

Applies to

The contenteditable attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	contenteditable

Example

Example

An editable paragraph:

```
<p contenteditable="true">This is an editable paragraph.</p>
```

Browser Support

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

Attribute					
contenteditable	4.0	6.0	3.5	3.1	10.1

HTML controls Attribute

Definition and Usage

The controls attribute is a boolean attribute.

When present, it specifies that audio/video controls should be displayed.

Controls should include:

- Play
- Pause
- Seeking
- Volume
- Fullscreen toggle (for video only)
- Captions/Subtitles (for video only, when available)
- Track (for video only, when available)

Applies to

The controls attribute can be used on the following elements:

Elements	Attribute
<code><audio></code>	<u>controls</u>
<code><video></code>	<u>controls</u>

Examples

Audio Example

An `<audio>` element with browser default controls:

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>
```

Video Example

A `<video>` element with browser default controls:

```
<video controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Browser Support

The `controls` attribute has the following browser support for each element:

Element					
audio	4.0	9.0	3.5	4.0	10.5
video	4.0	9.0	3.5	4.0	10.5

HTML coords Attribute

Definition and Usage

The coords attribute specifies the coordinates of an area in an image-map.

The coords attribute is used together with the shape attribute to specify the size, shape, and placement of an area.

Tip: The coordinates of the top-left corner of an area are 0,0.

Applies to

The coords attribute can be used on the following element:

Element	Attribute
<u><area></u>	<u>coords</u>

Example

Area Example

An image-map, with clickable areas:

```
<img src ="planets.gif" width="145" height="126" alt="Planets"
usemap="#planetmap">

<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>
```

Browser Support

Attribute					
coords	Yes	Yes	Yes	Yes	Yes

HTML data Attribute

Definition and Usage

The data attribute specifies the URL of the resource to be used by the object.

Applies to

The data attribute can be used on the following element:

Element	Attribute
<u><object></u>	<u>data</u>

Example

Example

How to use the <object> element to embed a Flash file:

```
<object width="400" height="400" data="helloworld.swf"></object>
```

Browser Support

Attribute					
data	Yes	Yes	Yes	Yes	Yes

HTML data-* Attribute

Definition and Usage

The data-* attributes is used to store custom data private to the page or application.

The data-* attributes gives us the ability to embed custom data attributes on all HTML elements.

The stored (custom) data can then be used in the page's JavaScript to create a more engaging user experience (without any Ajax calls or server-side database queries).

The data-* attributes consist of two parts:

1. The attribute name should not contain any uppercase letters, and must be at least one character long after the prefix "data-"
2. The attribute value can be any string

Note: Custom attributes prefixed with "data-" will be completely ignored by the user agent.

Applies to

The contextmenu attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	<u>data-*</u>

Example

Example

Use the data-* attribute to embed custom data:

```
<ul>
  <li data-animal-type="bird">Owl</li>
  <li data-animal-type="fish">Salmon</li>
  <li data-animal-type="spider">Tarantula</li>
</ul>
```

Browser Support

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

Attribute					
data-*	4.0	5.5	2.0	3.1	9.6

HTML datetime Attribute

Definition and Usage

The `datetime` attribute specifies the date and time when the text was deleted/inserted. When used together with the `<time>` element, it represents a date and/or time of the `<time>` element.

Applies to

The `controls` attribute can be used on the following elements:

Elements	Attribute
<code></code>	<code>datetime</code>
<code><ins></code>	<code>datetime</code>
<code><time></code>	<code>datetime</code>

Examples

Del Example

A deleted text, with a date and time of when the text was deleted:

```
<p>
<del datetime="2015-11-15T22:55:03Z">This text has been deleted</del>
</p>
```

Ins Example

An inserted text, with a date and time of when the text was inserted:

```
<p>This is a text.
<ins datetime="2015-09-15T22:55:03Z">This is an inserted text.</ins></p>
```

Time Example

A time element with a machine-readable datetime attribute:

```
<p>I have a date on <time datetime="2017-02-14">Valentines day</time>. </p>
```

Browser Support

The `datetime` attribute has the following browser support for each element:

Element					
del	Yes	Yes	Yes	Yes	Yes
ins	Yes	Yes	Yes	Yes	Yes
time	Yes	Yes	Yes	Yes	Yes

HTML default Attribute

Definition and Usage

The default attribute is a boolean attribute.

When present, it specifies that the track is to be enabled if the user's preferences do not indicate that another track would be more appropriate.

Note: There must not be more than one track element with a default attribute per media element.

Applies to

The default attribute can be used on the following element:

Element	Attribute
<u><track></u>	<u>default</u>

Example

Example

A video with two subtitle tracks. "English" subtitle is the default:

```
<video width="320" height="240" controls>
  <source src="forrest_gump.mp4" type="video/mp4">
  <source src="forrest_gump.ogg" type="video/ogg">
  <track src="subtitles_en.vtt" kind="subtitles" srclang="en"
        label="English" default>
  <track src="subtitles_no.vtt" kind="subtitles" srclang="no"
        label="Norwegian">
</video>
```

Browser Support

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

Attribute					
default	18.0	10.0	31.0	6.0	15.0

HTML defer Attribute

Definition and Usage

The defer attribute is a boolean attribute.

When present, it specifies that the script is executed when the page has finished parsing.

Note: The defer attribute is only for external scripts (should only be used if the src attribute is present).

Note: There are several ways an external script can be executed:

- If async is present: The script is executed asynchronously with the rest of the page (the script will be executed while the page continues the parsing)
- If async is not present and defer is present: The script is executed when the page has finished parsing
- If neither async or defer is present: The script is fetched and executed immediately, before the browser continues parsing the page

Applies to

The defer attribute can be used on the following element:

Element	Attribute
<script>	<u>defer</u>

Example

Example

A script that will not run until after the page has loaded:

```
<script src="demo_defer.js" defer></script>
```

Browser Support

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

Attribute					
defer	Yes	10.0	3.6	Yes	15.0

HTML dir Attribute

Definition and Usage

The dir attribute specifies the text direction of the element's content.

Applies to

The dir attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	<u>dir</u>

Example

Example

A paragraph with a right-to-left direction:

```
<p dir="rtl">Write this text right-to-left!</p>
```

Browser Support

Attribute					
dir	Yes	Yes	Yes	Yes	Yes

HTML dirname Attribute

Definition and Usage

The dirname attribute enables the submission of the text direction of the input field/textarea
The dirname attribute's value is always the name of the input field/textarea, followed by ".dir".

Applies to

The dirname attribute can be used on the following elements:

Elements	Attribute
<u><input></u>	<u>dirname</u>
<u><textarea></u>	<u>dirname</u>

Examples

Input Example

An HTML form where the field's text direction will be submitted:

```
<form action="/action_page.php">
  First name: <input type="text" name="fname" dirname="fname.dir">
  <input type="submit" value="Submit">
</form>
```

Textarea Example

An HTML form where the field's text direction will be submitted:

```
<form action="/action_page.php">
  Text:
  <textarea name="explanation" dirname="explanation.dir"></textarea>
  <input type="submit" value="Submit">
</form>
```

Browser Support

The `dirname` attribute has the following browser support for each element:

Element					
input	Yes	No	No	Yes	Yes
textarea	Yes	No	No	Yes	Yes

HTML disabled Attribute

Definition and Usage

The disabled attribute is a boolean attribute.

When present, it specifies that the element should be disabled.

A disabled element is unusable.

The disabled attribute can be set to keep a user from using the element until some other condition has been met (like selecting a checkbox, etc.). Then, a JavaScript could remove the disabled value, and make the element usable.

Applies to

The disabled attribute can be used on the following elements:

Elements	Attribute
<code><button></code>	<code>disabled</code>
<code><fieldset></code>	<code>disabled</code>
<code><input></code>	<code>disabled</code>
<code><optgroup></code>	<code>disabled</code>
<code><option></code>	<code>disabled</code>
<code><select></code>	<code>disabled</code>
<code><textarea></code>	<code>disabled</code>

Examples

Button Example

A disabled button:

```
<button type="button" disabled>Click Me!</button>
```

Fieldset Example

Disable a group of related form elements:

```
<fieldset disabled>
  <legend>Personalia:</legend>
  Name: <input type="text"><br>
  Email: <input type="text"><br>
  Date of birth: <input type="text">
</fieldset>
```

Input Example

An HTML form with a disabled input field:

```
<form action="/action_page.php">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname" disabled><br>
  <input type="submit" value="Submit">
</form>
```

Optgroup Example

A disabled option-group:

```
<select>
  <optgroup label="German Cars" disabled>
    <option value="mercedes">Mercedes</option>
    <option value="audi">Audi</option>
  </optgroup>
</select>
```

Option Example

A drop-down list with one disabled option:

```
<select>
  <option value="volvo" disabled>Volvo</option>
  <option value="saab">Saab</option>
  <option value="vw">VW</option>
  <option value="audi">Audi</option>
</select>
```

Select Example

A disabled drop-down list:

```
<select disabled>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="mercedes">Mercedes</option>
  <option value="audi">Audi</option>
</select>
```

Textarea Example

A disabled text area:

```
<textarea disabled>  
At w3schools.com you will learn how to make a website. We offer free  
tutorials in all web development technologies.  
</textarea>
```

Browser Support

The `disabled` attribute has the following browser support for each element:

Element					
button	Yes	Yes	Yes	Yes	Yes
fieldset	Yes	Not supported	Yes	7.0	Yes
input	1.0	6.0	1.0	1.0	1.0
optgroup	1.0	8.0	Yes	Yes	Yes
option	1.0	8.0	1.0	Yes	Yes
select	Yes	Yes	Yes	Yes	Yes
textarea	Yes	Yes	Yes	Yes	Yes

HTML download Attribute

Definition and Usage

The download attribute specifies that the target will be downloaded when a user clicks on the hyperlink.

This attribute is only used if the href attribute is set.

The value of the attribute will be the name of the downloaded file. There are no restrictions on allowed values, and the browser will automatically detect the correct file extension and add it to the file (.img, .pdf, .txt, .html, etc.).

If the value is omitted, the original filename is used.

Applies to

The download attribute can be used on the following elements:

Elements	Attribute
<u><a></u>	<u>download</u>
<u><area></u>	<u>download</u>

Examples

Example

Download file when clicking on the link (instead of navigating to the file):

```
<a href="/images/myw3schoolsimage.jpg" download>
```

Area Example

An image-map with clickable areas that will be downloaded when clicked on:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" alt="Sun"
  href="info_about_the_sun.htm" download="sun">
  <area shape="circle" coords="90,58,3" alt="Mercury" href="merglobe.gif"
  download="mercury">
  <area shape="circle" coords="124,58,8" alt="Venus"
  href="information_about_the_planet_venus.txt" download="venus">
</map>
```

Browser Support

The `download` attribute has the following browser support for each element:

Element					
a	14.0	13.0	20.0	Not supported	15.0
area	14.0	Not supported	20.0	Not supported	15.0

HTML draggable Attribute

Definition and Usage

The draggable attribute specifies whether an element is draggable or not.

Tip: Links and images are draggable by default.

Tip: The draggable attribute is often used in drag and drop operations. Read our [HTML Drag and Drop tutorial](#) to learn more.

Applies to

The draggable attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	draggable

Example

Example

A draggable paragraph:

```
<p draggable="true">This is a draggable paragraph.</p>
```

Browser Support

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

Attribute					
draggable	4.0	9.0	3.5	6.0	12.0

HTML dropzone Attribute

Definition and Usage

The dropzone attribute specifies whether the dragged data is copied, moved, or linked, when it is dropped on an element.

Applies to

The dropzone attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	dropzone

Example

Example

Dropping the data will result in a copy of the dragged data:

```
<div dropzone="copy"></div>
```

Browser Support

Attribute					
dropzone	Not supported	Not supported	Not supported	Not supported	Not supported

HTML enctype Attribute

Definition and Usage

The enctype attribute specifies how the form-data should be encoded when submitting it to the server.

Note: The enctype attribute can be used only if method="post".

Applies to

The enctype attribute can be used on the following element:

Element	Attribute
<u><form></u>	<u>enctype</u>

Example

Example

Send form-data encoded as "multipart/form-data":

```
<form action="/action_page_binary.asp" method="post"
enctype="multipart/form-data">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="submit" value="Submit">
</form>
```

Browser Support

Attribute					
enctype	Yes	Yes	Yes	Yes	Yes

HTML for Attribute

Definition and Usage

When used together with the <label> element, the for attribute specifies which form element a label is bound to.

When used together with the <output> element, the for attribute specifies the relationship between the result of the calculation, and the elements used in the calculation.

Applies to

The for attribute can be used on the following elements:

Elements	Attribute
<u><label></u>	<u>for</u>
<u><output></u>	<u>for</u>

Examples

For Example

Three radio buttons with labels:

```
<form action="/action_page.php">
  <label for="male">Male</label>
  <input type="radio" name="gender" id="male" value="male"><br>
  <label for="female">Female</label>
  <input type="radio" name="gender" id="female" value="female"><br>
  <label for="other">Other</label>
  <input type="radio" name="gender" id="other" value="other"><br><br>
  <input type="submit" value="Submit">
</form>
```

Output Example

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

```
<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">
  <input type="range" id="a" value="50">100
  +<input type="number" id="b" value="50">
  =<output name="x" for="a b"></output>
</form>
```

Browser Support

The `for` attribute has the following browser support for each element:

Element					
label	Yes	Yes	Yes	Yes	Yes
output	10.0	Not supported	4.0	5.1	11.0

HTML form Attribute

Definition and Usage

The form attribute specifies one or more forms the element belongs to.

Applies to

The form attribute can be used on the following elements:

Elements	Attribute
<u><button></u>	<u>form</u>
<u><fieldset></u>	<u>form</u>
<u><input></u>	<u>form</u>
<u><label></u>	<u>form</u>
<u><meter></u>	<u>form</u>
<u><object></u>	<u>form</u>
<u><output></u>	<u>form</u>
<u><select></u>	<u>form</u>
<u><textarea></u>	<u>form</u>

Examples

Button Example

A button located outside a form (but still a part of the form):

```
<form action="/action_page.php" method="get" id="form1">
    First name: <input type="text" name="fname"><br>
    Last name: <input type="text" name="lname"><br>
</form>

<button type="submit" form="form1" value="Submit">Submit</button>
```

Fieldset Example

A `<fieldset>` element located outside a form (but still a part of the form):

```
<form action="/action_page.php" method="get" id="form1">
    What is your favorite color? <input type="text" name="fav_color"><br>
    <input type="submit">
</form>

<fieldset form="form1">
    Name: <input type="text" name="username"><br>
    Email: <input type="text" name="usermail"><br>
</fieldset>
```

Input Example

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

```
<form action="/action_page.php" id="form1">
    First name: <input type="text" name="fname"><br>
    <input type="submit" value="Submit">
</form>

Last name: <input type="text" name="lname" form="form1">
```

Label Example

A `<label>` element located outside a form (but still a part of the form):

```
<form action="/action_page.php" id="form1">
  <input type="radio" name="gender" id="male" value="male"><br>
  <label for="female">Female</label>
  <input type="radio" name="gender" id="female" value="female"><br>
  <label for="other">Other</label>
  <input type="radio" name="gender" id="other" value="other"><br><br>
  <input type="submit" value="Submit">
</form>

<label for="male" form="form1">Male</label>
```

Meter Example

A `<meter>` element located outside a form (but still a part of the form):

```
<form action="/action_page.php" method="get" id="form1">
  First name: <input type="text" name="fname"><br>
  <input type="submit" value="Submit">
</form>

<meter form="form1" name="x1" min="0" low="40" high="90" max="100"
value="95"></meter>
```

Object Example

An `<object>` element located outside a form (but still a part of the form):

```
<form action="/action_page.php" id="form1">
  First name: <input type="text" name="fname"><br>
  <input type="submit" value="Submit">
</form>

<object data="helloworld.swf" height="400" width="400" form="form1"
name="obj1"></object>
```

Output Example

A <output> element located outside a form (but still a part of the form):

```
<form action="/action_page.php" id="numform"
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">
<br><br>
<input type="submit">
</form>

<output form="numform" name="x" for="a b"></output>
```

Select Example

A drop-down list located outside a form (but still a part of the form):

```
<form action="/action_page.php" id="carform">
  Firstname:<input type="text" name="fname">
  <input type="submit">
</form>

<select name="carlist" form="carform">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi">Audi</option>
</select>
```

Textarea Example

A text area located outside a form (but still a part of the form):

```
<form action="/action_page.php" id="usrform">
  Name: <input type="text" name="usrname">
  <input type="submit">
</form>

<textarea name="comment" form="usrform">Enter text here...</textarea>
```

Browser Support

The `form` attribute has the following browser support for each element:

Element					
button	10.0	Not supported	4.0	5.1	9.5
fieldset	Not supported	Not supported	Not supported	Not supported	Not supported
input	9.0	Not supported	4.0	5.1	10.6
label	Yes	Yes	Yes	Yes	Yes
meter	Not supported	Not supported	Not supported	Not supported	Not supported
object	Not supported	Not supported	Not supported	Not supported	Not supported
output	Yes	Not supported	Yes	Yes	Yes
select	Yes	Not supported	Yes	Yes	Yes
textarea	Yes	Not supported	Yes	Yes	Yes

HTML formaction Attribute

Definition and Usage

The formaction attribute specifies where to send the form-data when a form is submitted. This attribute overrides the form's action attribute.

The formaction attribute is only used for inputs/buttons with type="submit".

Applies to

The formaction attribute can be used on the following elements:

Elements	Attribute
<u><button></u>	<u>formaction</u>
<u><input></u>	<u>formaction</u>

Examples

Button Example

A form with two submit buttons. The first submit button submits the form data to "action_page.php", and the second submits to "action_page2.php":

```
<form action="/action_page.php" method="get">
    First name: <input type="text" name="fname"><br>
    Last name: <input type="text" name="lname"><br>
    <button type="submit">Submit</button><br>
    <button type="submit" formaction="/action_page2.php">Submit to another
page</button>
</form>
```

Input Example

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

```
<form action="/action_page.php">
    First name: <input type="text" name="fname"><br>
    Last name: <input type="text" name="lname"><br>
    <input type="submit" value="Submit"><br>
    <input type="submit" formaction="/action_page2.php" value="Submit to
another page">
</form>
```

Browser Support

The `formaction` attribute has the following browser support for each element:

Element					
button	9.0	10.0	4.0	5.1	10.6
input	9.0	10.0	4.0	5.1	10.6

HTML headers Attribute

Definition and Usage

The headers attribute specifies one or more header cells a table cell is related to.

Applies to

The headers attribute can be used on the following elements:

Elements	Attribute
<u><td></u>	<u>headers</u>
<u><th></u>	<u>headers</u>

Examples

Td Example

Specify the <th> element each <td> element is related to:

```
<table>
  <tr>
    <th id="name">Name</th>
    <th id="email">Email</th>
    <th id="phone">Phone</th>
    <th id="addr">Address</th>
  </tr>
  <tr>
    <td headers="name">John Doe</td>
    <td headers="email">someone@example.com</td>
    <td headers="phone">+45342323</td>
    <td headers="addr">Rosevn 56,4300 Sandnes,Norway</td>
  </tr>
</table>
```

Th Example

Specify the `<th>` element each header cell is related to:

```
<table>
  <tr>
    <th id="name" colspan="2">Name</th>
  </tr>
  <tr>
    <th headers="name">Firstname</th>
    <th headers="name">Lastname</th>
  </tr>
</table>
```

Browser Support

The `headers` attribute has the following browser support for each element:

Element					
td	Yes	Yes	Yes	Yes	Yes
th	Yes	Yes	Yes	Yes	Yes

HTML height Attribute

Definition and Usage

The height attribute specifies the height of the element, in pixels.

Applies to

The height attribute can be used on the following elements:

Elements	Attribute
<code><canvas></code>	<code>height</code>
<code><embed></code>	<code>height</code>
<code><iframe></code>	<code>height</code>
<code></code>	<code>height</code>
<code><input></code>	<code>height</code>
<code><object></code>	<code>height</code>
<code><video></code>	<code>height</code>

Examples

Canvas Example

A <canvas> element with a height and width of 200 pixels:

```
<canvas id="myCanvas" width="200" height="200" style="border:1px solid">
```

Embed Example

A flash animation with a height and width of 200 pixels:

```
<embed src="helloworld.swf" width="200" height="200">
```

Iframe Example

An <iframe> with a specified height and width of 200 pixels:

```
<iframe src="/default.asp" width="200" height="200">  
</iframe>
```

Img Example

An image with a height and width of 42 pixels:

```

```

Input Example

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

```
<form action="/action_page.php">  
  First name: <input type="text" name="fname"><br>  
  Last name: <input type="text" name="lname"><br>  
  <input type="image" src="img_submit.gif" alt="Submit" width="48"  
        height="48">  
</form>
```

Object Example

A flash animation with a height and width of 400 pixels::

```
<object data="helloworld.swf" height="400" width="400"></object>
```

Video Example

A video player with a specified height and width:

```
<video width="320" height="240" controls>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogg" type="video/ogg">  
  Your browser does not support the video tag.  
</video>
```

Browser Support

The `height` attribute has the following browser support for each element:

Element					
canvas	4.0	9.0	2.0	3.1	9.0
embed	Yes	Yes	Yes	Yes	Yes
iframe	Yes	Yes	Yes	Yes	Yes
img	Yes	Yes	Yes	Yes	Yes
input	1.0	Yes	16.0	Yes	1.0
object	Yes	Yes	Yes	Yes	Yes
video	Yes	Yes	Yes	Yes	Yes

HTML hidden Attribute

Definition and Usage

The hidden attribute is a boolean attribute.

When present, it specifies that an element is not yet, or is no longer, relevant.

Browsers should not display elements that have the hidden attribute specified.

The hidden attribute can also be used to keep a user from seeing an element until some other condition has been met (like selecting a checkbox, etc.). Then, a JavaScript could remove the hidden attribute, and make the element visible.

Applies to

The hidden attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	<u>hidden</u>

Example

Example

A hidden paragraph:

```
<p hidden>This paragraph should be hidden.</p>
```

Browser Support

Attribute					
hidden	6.0	11.0	4.0	5.1	11.1

HTML high Attribute

Definition and Usage

The high attribute specifies the range where the gauge's value is considered to be a high value.

The high attribute value must be less than the max attribute value, and it also must be greater than the low and min attribute values.

Applies to

The high attribute can be used on the following element:

Element	Attribute
<code><meter></code>	<code>high</code>

Example

Example

A gauge with a current value and min, max, high, and low segments:

```
<meter min="0" low="40" high="90" max="100" value="95"></meter>
```

Browser Support

Attribute	Chrome	Edge	Firefox	Safari	Opera
high	8.0	Not supported	6.0	6.0	11.0

HTML href Attribute

Definition and Usage

For `<a>` and `<area>` elements, the `href` attribute specifies the URL of the page the link goes to.

For `<base>` elements, the `href` attribute specifies the base URL for all relative URLs on a page.

For `<link>` elements, the `href` attribute specifies the location (URL) of the external resource (most often a style sheet file).

Applies to

The `href` attribute can be used on the following elements:

Elements	Attribute
<code><a></code>	<code>href</code>
<code><area></code>	<code>href</code>
<code><base></code>	<code>href</code>
<code><link></code>	<code>href</code>

Examples

A Example

The href attribute specifies the link's destination:

```
<a href="https://www.w3schools.com">Visit W3Schools</a>
```

Area Example

An image-map, with clickable areas:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>
```

Base Example

Specify a base URL for all relative URLs on a page:

```
<head>
<base href="https://www.w3schools.com/images/">
</head>
```

Link Example

Link to an external stylesheet:

```
<link rel="stylesheet" type="text/css" href="theme.css">
```

Browser Support

The `href` attribute has the following browser support for each element:

Element					
a	Yes	Yes	Yes	Yes	Yes
area	Yes	Yes	Yes	Yes	Yes
base	Yes	Yes	Yes	Yes	Yes
link	Yes	Yes	Yes	Yes	Yes

HTML hreflang Attribute

Definition and Usage

The hreflang attribute specifies the language of the linked document.

Note: This attribute is purely advisory.

Applies to

The hreflang attribute can be used on the following elements:

Elements	Attribute
<u><a></u>	<u>hreflang</u>
<u><area></u>	<u>hreflang</u>
<u><link></u>	<u>hreflang</u>

Examples

A Example

The hreflang attribute specifies the language of the document in the link:

```
<a href="https://www.w3schools.com" hreflang="en">W3Schools</a>
```

Area Example

An image-map, with clickable areas:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm"
hreflang="en">
</map>
```

Link Example

Here, the hreflang attribute indicates that the linked document is in English:

```
<link href="tag_link.asp" rel="parent" rev="subsection" hreflang="en">
```

Browser Support

The hreflang attribute has the following browser support for each element:

Element					
a	Yes	Yes	Yes	Yes	Yes
area	Yes	Yes	Yes	Yes	Yes
link	Yes	Yes	Yes	Yes	Yes

HTML http-equiv Attribute

Definition and Usage

The http-equiv attribute provides an HTTP header for the information/value of the content attribute.

The http-equiv attribute can be used to simulate an HTTP response header.

Applies to

The http-equiv attribute can be used on the following element:

Element	Attribute
<u><meta></u>	<u>http-equiv</u>

Example

Meta Example

Refresh document every 30 seconds:

```
<head>
<meta http-equiv="refresh" content="30">
</head>
```

Browser Support

Attribute					
http-equiv	Yes	Yes	Yes	Yes	Yes

HTML id Attribute

Definition and Usage

The id attribute specifies a unique id for an HTML element (the value must be unique within the HTML document).

The id attribute is most used to point to a style in a style sheet, and by JavaScript (via the HTML DOM) to manipulate the element with the specific id.

Applies to

The id attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	<u>id</u>

Example

Example

Use the id attribute to manipulate text with JavaScript:

```
<html>
<body>

<h1 id="myHeader">Hello World!</h1>
<button onclick="displayResult()">Change text</button>

<script>
function displayResult() {
    document.getElementById("myHeader").innerHTML = "Have a nice day!";
}
</script>

</body>
</html>
```

Browser Support

Attribute					
id	Yes	Yes	Yes	Yes	Yes

HTML ismap Attribute

Definition and Usage

The ismap attribute is a boolean attribute.

When present, it specifies that the image is part of a server-side image-map (an image with clickable areas).

When clicking on a server-side image-map, the click coordinates are sent to the server as a URL query string.

Note: The ismap attribute is allowed only if the `` element is a descendant of an `<a>` element with a valid href attribute.

Applies to

The ismap attribute can be used on the following element:

Element	Attribute
<code></code>	<code>ismap</code>

Example

Img Example

A server-side image-map:

```
<a href="/action_page.php">
  
</a>
```

Browser Support

Attribute					
ismap	Yes	Yes	Yes	Yes	Yes

HTML kind Attribute

Definition and Usage

The kind attribute specifies the kind of text track.

Applies to

The kind attribute can be used on the following element:

Element	Attribute
<u><track></u>	<u>kind</u>

Example

Example

A video with two subtitle tracks:

```
<video width="320" height="240" controls>
  <source src="forrest_gump.mp4" type="video/mp4">
  <source src="forrest_gump.ogg" type="video/ogg">
  <track src="subtitles_en.vtt" kind="subtitles" srclang="en"
label="English">
  <track src="subtitles_no.vtt" kind="subtitles" srclang="no"
label="Norwegian">
</video>
```

Browser Support

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

Attribute					
kind	18.0	10.0	31.0	6.0	15.0

HTML label Attribute

Definition and Usage

The label attribute specifies the title of the text track.

The title of the text track is used by the browser when listing available text tracks.

Applies to

The label attribute can be used on the following element:

Element	Attribute
<track>	<u>label</u>

Example

Track Example

A video with two subtitle tracks, both with a label defined:

```
<video width="320" height="240" controls>
  <source src="forrest_gump.mp4" type="video/mp4">
  <source src="forrest_gump.ogg" type="video/ogg">
  <track src="subtitles_en.vtt" kind="subtitles" srclang="en"
        label="English">
  <track src="subtitles_no.vtt" kind="subtitles" srclang="no"
        label="Norwegian">
</video>
```

Browser Support

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

Attribute					
label	18.0	10.0	31.0	6.0	15.0

HTML lang Attribute

Definition and Usage

The lang attribute specifies the language of the element's content.

Common examples are "en" for English, "es" for Spanish, "fr" for France and so on.

Applies to

The lang attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	lang

Example

Example

Some French text in a paragraph:

```
<p lang="fr">Ceci est un paragraphe.</p>
```

Browser Support

Attribute					
lang	Yes	Yes	Yes	Yes	Yes

HTML list Attribute

Definition and Usage

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

Applies to

The list attribute can be used on the following element:

Element	Attribute
<code><input></code>	<code>list</code>

Example

Example

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

```
<input list="browsers">

<datalist id="browsers">
  <option value="Internet Explorer">
  <option value="Firefox">
  <option value="Google Chrome">
  <option value="Opera">
  <option value="Safari">
</datalist>
```

Browser Support

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

Attribute					
list	20.0	10.0	4.0	Not supported	9.6

HTML loop Attribute

Definition and Usage

The loop attribute is a boolean attribute.

When present, it specifies that the audio will start over again, every time it is finished.

Applies to

The loop attribute can be used on the following elements:

Elements	Attribute
<code><audio></code>	<code>loop</code>
<code><video></code>	<code>loop</code>

Examples

Audio Example

A song that will start over again, every time it is finished:

```
<audio controls loop>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>
```

Video Example

A video that will start over again, every time it is finished:

```
<video controls loop>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Browser Support

The `loop` attribute has the following browser support for each element:

Element					
loop	4.0	9.0	3.5	4.0	10.5
loop	4.0	9.0	11.0	4.0	10.5

HTML low Attribute

Definition and Usage

The low attribute specifies the range where the gauge's value is considered to be a low value. The low attribute value must be greater than the min attribute value, and it also must be less than the high and max attribute values.

Applies to

The low attribute can be used on the following element:

Element	Attribute
<code><meter></code>	<code>low</code>

Example

Example

A gauge with a current value and min, max, high, and low segments:

```
<meter min="0" low="40" high="90" max="100" value="95"></meter>
```

Browser Support

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

Attribute					
low	8.0	Not supported	6.0	6.0	11.0

HTML max Attribute

Definition and Usage

The max attribute specifies the maximum value of the element.

When used by the <progress> element, the max attribute specifies how much work the task requires in total.

Applies to

The max attribute can be used on the following elements:

Elements	Attribute
<input>	<u>max</u>
<meter>	<u>max</u>
<progress>	<u>max</u>

Examples

Input Example

Use of the min and max attributes:

```
<form action="/action_page.php">

    Enter a date before 1980-01-01:
    <input type="date" name="bday" max="1979-12-31">

    Enter a date after 2000-01-01:
    <input type="date" name="bday" min="2000-01-02">

    Quantity (between 1 and 5):
    <input type="number" name="quantity" min="1" max="5">

    <input type="submit">

</form>
```

Meter Example

A gauge with a current value and min, max, high, and low segments:

```
<meter min="0" low="40" high="90" max="100" value="95"></meter>
```

Progress Example

Downloading in progress:

```
<progress value="22" max="100"></progress>
```

Browser Support

The `max` attribute has the following browser support for each element:

Element					
input	5.0	10.0	16.0	5.1	10.6
meter	8.0	Not supported	6.0	6.0	11.0
progress	8.0	10.0	16.0	6.0	11.0

HTML maxlength Attribute

Definition and Usage

The maxlength attribute specifies the maximum number of characters allowed in the element.

Applies to

The maxlength attribute can be used on the following elements:

Elements	Attribute
<u><input></u>	<u>maxlength</u>
<u><textarea></u>	<u>maxlength</u>

Examples

Input Example

An <input> element with a maximum length of 10 characters:

```
<form action="/action_page.php">
  Username: <input type="text" name="username" maxlength="10"><br>
  <input type="submit" value="Submit">
</form>
```

Textarea Example

A text area with a maximum length of 50 characters:

```
<textarea maxlength="50">
  Enter text here...
</textarea>
```

Browser Support

The `maxlength` attribute has the following browser support for each element:

Element					
input	1.0	2.0	1.0	1.0	1.0
textarea	Yes	10.0	4.0	Yes	15.0

HTML media Attribute

Definition and Usage

The media attribute specifies what media/device the linked document is optimized for. This attribute is used to specify that the target URL is designed for special devices (like iPhone), speech or print media.

This attribute can accept several values.

Applies to

The media attribute can be used on the following elements:

Elements	Attribute
<u><a></u>	<u>media</u>
<u><area></u>	<u>media</u>
<u><link></u>	<u>media</u>
<u><source></u>	<u>media</u>
<u><style></u>	<u>media</u>

Examples

A Example

A link with a media attribute:

```
<a href="att_a_media.asp?output=print"  
media="print and (resolution:300dpi)">  
Open media attribute page for print.</a>
```

Area Example

An image-map, with a clickable area:

```
  
  
<map name="planetmap">  
  <area shape="rect" coords="0,0,82,126" alt="Sun"  
    href="sun.htm" media="screen and (min-color-index:256)">  
</map>
```

Link Example

Two different style sheets for two different media types (screen and print):

```
<head>  
<link rel="stylesheet" type="text/css" href="theme.css">  
<link rel="stylesheet" type="text/css" href="print.css" media="print">  
</head>
```

Source Example

Use of the media attribute:

```
<source src="movie.ogg" type="video/ogg"  
media="screen and (min-width:320px)">
```

Style Example

Specify the style to use for print:

```
<style media="print">
  h1 {color:#000000;}
  p {color:#000000;}
  body {background-color:#FFFFFF;}
</style>
```

Browser Support

The `media` attribute has the following browser support for each element:

Element					
a	Yes	Yes	Yes	Yes	Yes
area	Yes	Yes	Yes	Yes	Yes
link	Yes	Yes	Yes	Yes	Yes
source	Not supported	Not supported	Not supported	Not supported	Not supported
style	Yes	Yes	Yes	Yes	Yes

HTML method Attribute

Definition and Usage

The method attribute specifies how to send form-data (the form-data is sent to the page specified in the action attribute).

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

Notes on GET:

- Appends form-data into the URL in name/value pairs
- The length of a URL is limited (about 3000 characters)
- Never use GET to send sensitive data! (will be visible in the URL)
- Useful for form submissions where a user want to bookmark the result
- GET is better for non-secure data, like query strings in Google

Notes on POST:

- Appends form-data inside the body of the HTTP request (data is not shown in URL)
- Has no size limitations
- Form submissions with POST cannot be bookmarked

Applies to

The method attribute can be used on the following element:

Element	Attribute
<u><form></u>	<u>method</u>

Example

Example

Submit a form using the "get" method:

```
<form action="/action_page.php" method="get">
    First name: <input type="text" name="fname"><br>
    Last name: <input type="text" name="lname"><br>
    <input type="submit" value="Submit">
</form>
```

Browser Support

Attribute					
method	Yes	Yes	Yes	Yes	Yes

HTML min Attribute

Definition and Usage

The min attribute specifies the minimum value of an element element.

When used together with the <meter> element, the min attribute specifies the lower bound of the gauge.

Applies to

The min attribute can be used on the following elements:

Elements	Attribute
<input>	<u>min</u>
<meter>	<u>min</u>

Examples

Input Example

Use of the min and max attributes:

```
<form action="/action_page.php">

    Enter a date before 1980-01-01:
    <input type="date" name="bday" max="1979-12-31">

    Enter a date after 2000-01-01:
    <input type="date" name="bday" min="2000-01-02">

    Quantity (between 1 and 5):
    <input type="number" name="quantity" min="1" max="5">

    <input type="submit">

</form>
```

Meter Example

A gauge with a current value and min, max, high, and low segments:

```
<meter min="0" low="40" high="90" max="100" value="95"></meter>
```

Browser Support

The `min` attribute has the following browser support for each element:

Element					
input	5.0	10.0	16.0	5.1	10.6
meter	5.0	10.0	16.0	5.1	10.6

HTML multiple Attribute

Definition and Usage

The multiple attribute is a boolean attribute.

When present, it specifies that the user is allowed to enter/select more than one value.

Applies to

The multiple attribute can be used on the following elements:

Elements	Attribute
<code><input></code>	<code>multiple</code>
<code><select></code>	<code>multiple</code>

Examples

Input Example

A file upload field that accepts multiple values:

```
<form action="/action_page.php">
  Select images: <input type="file" name="img" multiple>
  <input type="submit">
</form>
```

Select Example

A drop-down list that allows multiple selections:

```
<select multiple>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi">Audi</option>
</select>
```

Browser Support

The `multiple` attribute has the following browser support for each element:

Element					
input	6.0	10.0	3.6	5.0	11.0
select	Yes	Yes	Yes	Yes	Yes

HTML muted Attribute

Definition and Usage

The muted attribute is a boolean attribute.

When present, it specifies that the audio output of the video should be muted.

Applies to

The muted attribute can be used on the following element:

Element	Attribute
<video>	<u>muted</u>

Example

Video Example

A muted video:

```
<video controls muted>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Browser Support

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

Attribute					
muted	4.0	10.0	11.0	7.1	10.5

HTML name Attribute

Definition and Usage

The name attribute specifies a name for the element.

This name attribute can be used to reference the element in a JavaScript.

For form elements it is also used as a reference when the data is submitted,

For iframe element it can be used to target a form submission.

For the map element, the name attribute is associated with the 's usemap attribute and creates a relationship between the image and the map.

For the meta element, the name attribute specifies a name for the information/value of the content attribute.

For the param element, the name attribute is used together with the value attribute to specify parameters for the plugin specified with the <object> tag.

Applies to

The name attribute can be used on the following elements:

Elements	Attribute
<u><button></u>	<u>name</u>
<u><fieldset></u>	<u>name</u>
<u><form></u>	<u>name</u>
<u><iframe></u>	<u>name</u>
<u><input></u>	<u>name</u>
<u><map></u>	<u>name</u>
<u><meta></u>	<u>name</u>
<u><object></u>	<u>name</u>
<u><output></u>	<u>name</u>
<u><param></u>	<u>name</u>
<u><select></u>	<u>name</u>
<u><textarea></u>	<u>name</u>

Examples

Example

Two buttons with equal names, that submit different values when clicked:

```
<form action="/action_page.php" method="get">
    Choose your favorite subject:
    <button name="subject" type="submit" value="HTML">HTML</button>
    <button name="subject" type="submit" value="CSS">CSS</button>
</form>
```

Fieldset Example

A <fieldset> with a name attribute:

```
<fieldset name="personalia">
    Name: <input type="text"><br>
    Email: <input type="text"><br>
</fieldset>
```

Form Example

An HTML form with a name attribute:

```
<form action="/action_page.php" method="get" name="myForm">
    First name: <input type="text" name="fname"><br>
    Last name: <input type="text" name="lname"><br>
    <input type="button" onclick="formSubmit()" value="Send form data!">
</form>
```

Iframe Example

An <iframe> that act as a target for a link:

```
<iframe src="demo_iframe.htm" name="iframe_a"></iframe>
<a href="https://www.w3schools.com" target="iframe_a">W3Schools.com</a>
```

Input Example

An HTML form with three input fields; two text fields and one submit button:

```
<form action="/action_page.php">
  Name: <input type="text" name="fullname"><br>
  Email: <input type="text" name="email"><br>
  <input type="submit" value="Submit">
</form>
```

Map Example

An image-map, with clickable areas:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>
```

Meta Example

Use the name attribute to define a description, keywords, and the author of an HTML document:

```
<head>
<meta name="description" content="Free Web tutorials">
<meta name="keywords" content="HTML,CSS,JavaScript">
<meta name="author" content="Hege Refsnes">
</head>
```

Object Example

An `<object>` element with a name attribute:

```
<object data="helloworld.swf" width="400" height="400" name="obj1">
</object>
```

Output Example

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

```
<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">
  <input type="range" id="a" value="50">100
  +<input type="number" id="b" value="50">
  =<output name="x" for="a b"></output>
</form>
```

Param Example

Set the "autoplay" parameter to "true", so the sound will start playing as soon as the page loads:

```
<object data="horse.wav">
  <param name="autoplay" value="true">
</object>
```

Select Example

A drop-down list with a name attribute:

```
<select name="cars">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi">Audi</option>
</select>
```

Textarea Example

A text area with a name attribute:

```
<form action="/action_page.php">
  <textarea name="comment">Enter text here...</textarea>
  <input type="submit">
</form>
```

Browser Support

The `multiple` attribute has the following browser support for each element:

Element					
button	Yes	Yes	Yes	Yes	Yes
fieldset	Yes	Not supported	Yes	Yes	Yes
form	Yes	Yes	Yes	Yes	Yes
iframe	Yes	Yes	Yes	Yes	Yes
input	1.0	2.0	1.0	1.0	1.0
map	Yes	Yes	Yes	Yes	Yes
meta	Yes	Yes	Yes	Yes	Yes
object	Yes	Yes	Yes	Yes	Yes
output	10.0	Not supported	4.0	5.1	11.0
param	Yes	Yes	Yes	Yes	Yes
select	Yes	Yes	Yes	Yes	Yes
textarea	Yes	Yes	Yes	Yes	Yes

HTML onabort Attribute

Definition and Usage

The onabort attribute defines a script to be run if the loading of the media file aborts. This event occurs when the media data download has been aborted, and not because of an error.

Applies to

The onabort attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<u><audio></u>	<u>onabort</u>
<u><embed></u>	onabort
<u></u>	onabort
<u><object></u>	onabort
<u><video></u>	<u>onabort</u>

Examples

Video Example

Alert that the loading of a video has been aborted:

```
<video id="myVideo" onabort="alert('Video load aborted')">
```

Browser Support

The `onabort` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
embed	Yes	Not supported	Yes	Yes	Yes
img	Yes	Yes	Yes	Yes	Yes
object	Yes	Not supported	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

Note: The abort event for audio/video, is not supported in Internet Explorer 11 on Windows 7.

HTML novalidate Attribute

Definition and Usage

The novalidate attribute is a boolean attribute.

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

Applies to

The novalidate attribute can be used on the following element:

Element	Attribute
<form>	<u>novalidate</u>

Example

Form Example

Indicate that the form is not to be validated on submit:

```
<form action="/action_page.php" novalidate>
  E-mail: <input type="email" name="user_email">
  <input type="submit">
</form>
```

Browser Support

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

Attribute					
novalidate	10.0	10.0	4.0	Not supported	10.6

HTML onafterprint Attribute

Definition and Usage

The onafterprint attribute fires when a page has started printing, or if the print dialogue box has been closed.

Tip: The onafterprint attribute is often used together with the [onbeforeprint](#) attribute.

Applies to

The onafterprint attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<u><body></u>	<u>onafterprint</u>

Example

Execute a JavaScript when a page has started printing, or if the print dialogue box has been closed:

```
<body onafterprint="myFunction()">
```

Browser Support

Event Attribute					
onafterprint	63	Yes	Yes	Not supported	Not supported

Note: In IE, the onafterprint attribute occurs before the print dialogue box, instead of after.

HTML onbeforeprint Attribute

Definition and Usage

The onbeforeprint attribute fires when a page is about to be printed (before the print dialogue box appears).

Tip: The onbeforeprint attribute is often used together with the [onafterprint](#) attribute.

Applies to

The onbeforeprint attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<u><body></u>	<u>onbeforeprint</u>

Example

Execute a JavaScript when a page is about to be printed:

```
<body onbeforeprint="myFunction()">
```

Browser Support

Event Attribute					
onbeforeprint	63	Yes	Yes	Not supported	Not supported

HTML onbeforeunload Attribute

Definition and Usage

The onbeforeunload event fires when the document is about to be unloaded.

This event allows you to display a message in a confirmation dialog box to inform the user whether he/she wants to stay or leave the current page.

The default message that appears in the confirmation box, is different in different browsers. However, the standard message is something like "Are you sure you want to leave this page?". You cannot remove this message.

However, you can write a custom message together with the default message. See the first example on this page.

Note: In Firefox, only the default message will be displayed (not the custom message (if any)).

Applies to

The onbeforeunload attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<u><body></u>	<u>onbeforeunload</u>

Example

Execute a JavaScript when the page is about to be unloaded:

```
<body onbeforeunload="return myFunction()">
```

Browser Support

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

Event Attribute					
onbeforeunload	Yes	Yes	Yes	Yes	15.0

HTML onblur Attribute

Definition and Usage

The onblur attribute fires the moment that the element loses focus.

Onblur is most often used with form validation code (e.g. when the user leaves a form field).

Tip: The onblur attribute is the opposite of the onfocus attribute.

Applies to

The onblur attribute is part of the Event Attributes, and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onblur</u>

Example

Validate an input field when the user leaves it:

```
<input type="text" name="fname" id="fname" onblur="myFunction()">
```

Browser Support

Event Attribute					
onblur	Yes	Yes	Yes	Yes	Yes

HTML oncanplay Attribute

Definition and Usage

The oncanplay attribute defines a script to run when the browser can start playing the specified media (when it has buffered enough to begin).

Applies to

The oncanplay attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<audio>	canplay
<embed>	canplay
<object>	canplay
<video>	canplay

Examples

Audio Example

Run "myFunction" when the audio is ready to start playing:

```
<audio oncanplay="myFunction()">
```

Video Example

Run "myFunction" when the video is ready to start playing:

```
<video oncanplay="myFunction()">
```

Browser Support

The `oncanplay` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
embed	Not supported	Not supported	Not supported	Not supported	Not supported
object	Not supported	Not supported	Not supported	Not supported	Not supported
video	Yes	9.0	Yes	Yes	Yes

HTML oncanplaythrough Attribute

Definition and Usage

The oncanplaythrough event occurs when the browser estimates it can play through the specified audio/video without having to stop for buffering.

Applies to

The oncanplaythrough attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<audio>	canplaythrough
<video>	canplaythrough

Examples

Audio Example

Run "myFunction" when the audio is ready to start playing:

```
<audio oncanplaythrough="myFunction()">
```

Video Example

Run "myFunction" when the video is ready to start playing:

```
<video oncanplaythrough="myFunction()">
```

Browser Support

The `oncanplaythrough` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onchange Attribute

Definition and Usage

The onchange attribute fires the moment when the value of the element is changed.

Tip: This event is similar to the oninput event. The difference is that the oninput event occurs immediately after the value of an element has changed, while onchange occurs when the element loses focus. The other difference is that the onchange event also works on <select> elements.

Applies to

The onchange attribute is part of the Event Attributes, and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onchange</u>

Examples

Select Example

Execute a JavaScript when a user changes the selected option of a <select> element:

```
<select onchange="myFunction()">
```

Input Example

Execute a JavaScript when the user changes the content of an input field:

```
<input type="text" name="txt" value="Hello"
onchange="myFunction(this.value)">
```

Browser Support

Event Attribute					
onchange	Yes	Yes	Yes	Yes	Yes

HTML onclick Attribute

Definition and Usage

The onclick attribute fires on a mouse click on the element.

Applies to

The onclick attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onclick</u>

Examples

Button Example

Execute a JavaScript when a button is clicked:

```
<button onclick="myFunction()">Click me</button>
```

P Example

Click on a <p> element to change its text color to red:

```
<p id="demo" onclick="myFunction()">Click me to change my text color.</p>

<script>
function myFunction() {
    document.getElementById("demo").style.color = "red";
}
</script>
```

Browser Support

Event Attribute					
onclick	Yes	Yes	Yes	Yes	Yes

HTML oncontextmenu Attribute

Definition and Usage

The oncontextmenu attribute fires when the user right-clicks on an element to open the context menu.

Note: Although the oncontextmenu event is supported in all browsers, the contextmenu attribute is currently only supported in Firefox.

Applies to

The oncontextmenu attribute is part of the Event Attributes, and can be used on any HTML elements.

Elements	Event
All HTML elements	oncontextmenu

Example

Execute a JavaScript when a context menu is triggered:

```
<div oncontextmenu="myFunction()" contextmenu="mymenu">
```

Browser Support

Event Attribute					
oncontextmenu	Yes	Yes	Yes	Yes	Yes

HTML oncopy Attribute

Definition and Usage

The oncopy attribute fires when the user copies the content of an element.

Tip: The oncopy attribute also fires when the user copies an element, for example, an image, created with the element.

Tip: The oncopy attribute is mostly used on <input> elements with type="text".

Tip: There are three ways to copy an element/the content of an element:

- Press CTRL + C
- Select "Copy" from the Edit menu in your browser
- Right click to display the context menu and select the "Copy" command

Applies to

The oncopy attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	oncopy

Examples

Input Example

Execute a JavaScript when copying some text of an <input> element:

```
<input type="text" oncopy="myFunction()" value="Try to copy this text">
```

P Example

Execute a JavaScript when copying some text of a <p> element:

```
<p oncopy="myFunction()">Try to copy this text</p>
```

Img Example

Execute a JavaScript when copying an image:

```

```

Browser Support

Event Attribute					
oncopy	Yes	Yes	Yes	Yes	Yes

Note: The oncopy attribute may not work as expected in some browsers when trying to copy an image (See example above).

HTML oncuechange Attribute

Definition and Usage

The oncuechange attribute defines a script to run when the cue changes in a <track> element.

Applies to

The oncuechange attribute is part of the [Event Attributes](#), and can be used on the following element:

Element	Event
<track>	oncuechange

Browser Support

The `oncuechange` attribute has the following browser support:

Element					
track	Yes	Yes	Not supported	Yes	Yes

HTML oncut Attribute

Definition and Usage

The oncut attribute fires when the user cuts the content of an element.

Note: Although the oncut attribute is supported by all HTML elements, it is not actually possible to cut the content of, for example, a `<p>` element, UNLESS the element has set contenteditable to "true" (See "More Examples" below).

Tip: There are three ways to cut the content of an element:

- Press CTRL + X
- Select "Cut" from the Edit menu in your browser
- Right click to display the context menu and select the "Cut" command

Applies to

The oncut attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>oncut</u>

Examples

Input Example

Execute a JavaScript when cutting some text in an `<input>` element:

```
<input type="text" oncut="myFunction()" value="Try to cut this text">
```

P Example

Execute a JavaScript when cutting some text of a `<p>` element (Note that contenteditable is set to "true"):

```
<p contenteditable="true" oncut="myFunction()">Try to cut this text</p>
```

Browser Support

Event Attribute					
oncut	Yes	Yes	Yes	Yes	Yes

HTML ondblclick Attribute

Definition and Usage

The ondblclick attribute fires on a mouse double-click on the element.

Applies to

The ondblclick attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	ondblclick

Examples

Button Example

Execute a JavaScript when a button is double-clicked:

```
<button ondblclick="myFunction()">Double-click me</button>
```

P Example

Double-click on a `<p>` element to change its text color to red:

```
<p id="demo" ondblclick="myFunction()">Double-click me to change my text  
color.</p>  
  
<script>  
function myFunction() {  
    document.getElementById("demo").style.color = "red";  
}  
</script>
```

Browser Support

Event Attribute					
ondblclick	Yes	Yes	Yes	Yes	Yes

HTML ondrag Attribute

Definition and Usage

The ondrag attribute fires when an element or text selection is being dragged.

To learn about Drag and Drop, read our HTML Tutorial on [HTML5 Drag and Drop](#).

Tip: Links and images are draggable by default, and do not need the draggable attribute.

There are many event attributes that are used, and can occur, in the different stages of a drag and drop operation:

- **Events fired on the draggable target** (the source element):
 - ondragstart - fires when the user starts to drag an element
 - ondrag - fires when an element is being dragged
 - ondragend - fires when the user has finished dragging the element
- **Events fired on the drop target:**
 - ondragenter - fires when the dragged element enters the drop target
 - ondragover - fires when the dragged element is over the drop target
 - ondragleave - fires when the dragged element leaves the drop target
 - ondrop - fires when the dragged element is dropped on the drop target

Note: While dragging an element, the ondrag event fires every 350 milliseconds.

Applies to

The ondrag attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>ondrag</u>

Example

P Example

Execute a JavaScript when a `<p>` element is being dragged:

```
<p draggable="true" ondrag="myFunction(event)">Drag me!</p>
```

Browser Support

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

Event Attribute					
ondrag	4.0	9.0	3.5	6.0	12.0

HTML ondragend Attribute

Definition and Usage

The ondragend attribute fires when the user has finished dragging an element or text selection.

To learn about Drag and Drop, read our HTML Tutorial on [HTML5 Drag and Drop](#).

Tip: Links and images are draggable by default, and do not need the draggable attribute.

There are many event attributes that are used, and can occur, in the different stages of a drag and drop operation:

- **Events fired on the draggable target** (the source element):
 - [ondragstart](#) - fires when the user starts to drag an element
 - [ondrag](#) - fires when an element is being dragged
 - [ondragend](#) - fires when the user has finished dragging the element
- **Events fired on the drop target:**
 - [ondragenter](#) - fires when the dragged element enters the drop target
 - [ondragover](#) - fires when the dragged element is over the drop target
 - [ondragleave](#) - fires when the dragged element leaves the drop target
 - [ondrop](#) - fires when the dragged element is dropped on the drop target

Applies to

The ondragend attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	ondragend

Example

P Example

Execute a JavaScript when the user has finished dragging a `<p>` element:

```
<p draggable="true" ondragend="myFunction(event)">Drag me!</p>
```

Browser Support

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

Event Attribute					
ondragend	4.0	9.0	3.5	6.0	12.0

HTML ondragenter Attribute

Definition and Usage

The ondragenter attribute fires when a draggable element or text selection enters a valid drop target.

The ondragenter and ondragleave events can help the user to understand that a draggable element is about to enter or leave a drop target. This can be done by, for example, setting a background color when the draggable element enters the drop target, and removing the color when the element is moved out of the target.

To learn about Drag and Drop, read our HTML Tutorial on [HTML5 Drag and Drop](#).

Tip: Links and images are draggable by default, and do not need the draggable attribute.

There are many event attributes that are used, and can occur, in the different stages of a drag and drop operation:

- **Events fired on the draggable target** (the source element):
 - ondragstart - fires when the user starts to drag an element
 - ondrag - fires when an element is being dragged
 - ondragend - fires when the user has finished dragging the element
- **Events fired on the drop target:**
 - ondragenter - fires when the dragged element enters the drop target
 - ondragover - fires when the dragged element is over the drop target
 - ondragleave - fires when the dragged element leaves the drop target
 - ondrop - fires when the dragged element is dropped on the drop target

Applies to

The ondragenter attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>ondragenter</u>

Example

Div Example

Execute a JavaScript when a draggable element enters a drop target:

```
<div ondragenter="myFunction(event)"></div>
```

Browser Support

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

Event Attribute					
ondragenter	4.0	9.0	3.5	6.0	12.0

HTML ondragleave Attribute

Definition and Usage

The ondragleave attribute fires when a draggable element or text selection leaves a valid drop target.

The ondragenter and ondragleave events can help the user to understand that a draggable element is about to enter or leave a drop target. This can be done by, for example, setting a background color when the draggable element enters the drop target, and removing the color when the element is moved out of the target.

To learn about Drag and Drop, read our HTML Tutorial on [HTML5 Drag and Drop](#).

Tip: Links and images are draggable by default, and do not need the draggable attribute.

There are many event attributes that are used, and can occur, in the different stages of a drag and drop operation:

- **Events fired on the draggable target** (the source element):
 - [ondragstart](#) - fires when the user starts to drag an element
 - [ondrag](#) - fires when an element is being dragged
 - [ondragend](#) - fires when the user has finished dragging the element
- **Events fired on the drop target:**
 - [ondragenter](#) - fires when the dragged element enters the drop target
 - [ondragover](#) - fires when the dragged element is over the drop target
 - [ondragleave](#) - fires when the dragged element leaves the drop target
 - [ondrop](#) - fires when the dragged element is dropped on the drop target

Applies to

The ondragleave attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	ondragleave

Example

Div Example

Execute a JavaScript when a draggable element is moved out of a drop target:

```
<div ondragleave="myFunction(event)"></div>
```

Browser Support

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

Event Attribute					
ondragleave	4.0	9.0	3.5	6.0	12.0

HTML ondragover Attribute

Definition and Usage

The ondragover attribute fires when a draggable element or text selection is being dragged over a valid drop target.

By default, data/elements cannot be dropped in other elements. To allow a drop, we must prevent the default handling of the element. This is done by calling the event.preventDefault() method for the ondragover attribute.

To learn about Drag and Drop, read our HTML Tutorial on [HTML5 Drag and Drop](#).

Tip: Links and images are draggable by default, and do not need the draggable attribute.

There are many event attributes that are used, and can occur, in the different stages of a drag and drop operation:

- **Events fired on the draggable target** (the source element):
 - ondragstart - fires when the user starts to drag an element
 - ondrag - fires when an element is being dragged
 - ondragend - fires when the user has finished dragging the element
- **Events fired on the drop target:**
 - ondragenter - fires when the dragged element enters the drop target
 - ondragover - fires when the dragged element is over the drop target
 - ondragleave - fires when the dragged element leaves the drop target
 - ondrop - fires when the dragged element is dropped on the drop target

Note: While dragging an element, the ondragover event fires every 350 milliseconds.

Applies to

The ondragover attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>ondragover</u>

Example

Div Example

Execute a JavaScript when an element is being dragged over a drop target:

```
<div ondragover="myFunction(event)"></div>
```

Browser Support

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

Event Attribute					
ondragover	4.0	9.0	3.5	6.0	12.0

HTML ondragstart Attribute

Definition and Usage

The ondragstart attribute fires when the user starts to drag an element or text selection.

To learn about Drag and Drop, read our HTML Tutorial on [HTML5 Drag and Drop](#).

Tip: Links and images are draggable by default, and do not need the draggable attribute.

There are many event attributes that are used, and can occur, in the different stages of a drag and drop operation:

- **Events fired on the draggable target** (the source element):
 - ondragstart - fires when the user starts to drag an element
 - ondrag - fires when an element is being dragged
 - ondragend - fires when the user has finished dragging the element
- **Events fired on the drop target:**
 - ondragenter - fires when the dragged element enters the drop target
 - ondragover - fires when the dragged element is over the drop target
 - ondragleave - fires when the dragged element leaves the drop target
 - ondrop - fires when the dragged element is dropped on the drop target

Applies to

The ondragstart attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>ondragstart</u>

Example

P Example

Execute a JavaScript when the user starts to drag a <p> element:

```
<p draggable="true" ondragstart="myFunction(event)">Drag me!</p>
```

Browser Support

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

Event Attribute					
ondragstart	4.0	9.0	3.5	6.0	12.0

HTML ondrop Attribute

Definition and Usage

The ondrop attribute fires when a draggable element or text selection is dropped on a valid drop target.

To learn about Drag and Drop, read our HTML Tutorial on [HTML5 Drag and Drop](#).

Tip: Links and images are draggable by default, and do not need the draggable attribute.

There are many event attributes that are used, and can occur, in the different stages of a drag and drop operation:

- **Events fired on the draggable target** (the source element):
 - ondragstart - fires when the user starts to drag an element
 - ondrag - fires when an element is being dragged
 - ondragend - fires when the user has finished dragging the element
- **Events fired on the drop target:**
 - ondragenter - fires when the dragged element enters the drop target
 - ondragover - fires when the dragged element is over the drop target
 - ondragleave - fires when the dragged element leaves the drop target
 - ondrop - fires when the dragged element is dropped on the drop target

Applies to

The ondrop attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>ondrop</u>

Example

Div Example

Execute a JavaScript when a draggable element is dropped in a <div> element:

```
<div ondrop="myFunction(event)"></div>
```

Browser Support

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

Event Attribute					
ondrop	4.0	9.0	3.5	6.0	12.0

HTML ondurationchange Attribute

Definition and Usage

The durationchange event occurs when the duration data of the specified audio/video is changed.

Note: When an audio/video is loaded, the duration will change from "NaN" to the actual duration of the audio/video.

During the loading process of an audio/video, the following events occur, in this order:

1. loadstart
2. durationchange
3. loadedmetadata
4. loadeddata
5. progress
6. canplay
7. canplaythrough

Applies to

The ondurationchange attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<u><audio></u>	<u>durationchange</u>
<u><video></u>	<u>durationchange</u>

Examples

Audio Example

Run "myFunction" when the audio is ready to start playing:

```
<audio ondurationchange="myFunction()">
```

Video Example

Run "myFunction" when the video is ready to start playing:

```
<video ondurationchange="myFunction()">
```

Browser Support

The `ondurationchange` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onemptied Attribute

Definition and Usage

Fires when the current playlist is empty

Applies to

The onemptied attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<code><audio></code>	onemptied
<code><video></code>	onemptied

HTML onended Attribute

Definition and Usage

The onended event occurs when the audio/video has reached the end.

This event is useful for messages like "thanks for listening", "thanks for watching", etc.

Applies to

The onended attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<code><audio></code>	<code>ended</code>
<code><video></code>	<code>ended</code>

Examples

Audio Example

Run "myFunction" when the audio has finished playing:

```
<audio onended="myFunction()">
```

Video Example

Run "myFunction" when the video is finished:

```
<video onended="myFunction()">
```

Browser Support

The `onended` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onerror Attribute

Definition and Usage

The error event occurs when an error occurred during the loading of a media file.

Applies to

The onerror attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<u><audio></u>	<u>onerror</u>
<u><body></u>	onerror
<u><embed></u>	onerror
<u></u>	onerror
<u><link></u>	onerror
<u><object></u>	onerror
<u><script></u>	onerror
<u><video></u>	<u>onerror</u>

Examples

Audio Example

Run "myFunction" if an error occurs during loading:

```
<audio onerror="myFunction()">
```

Video Example

Run "myFunction" if an error occurs during loading:

```
<video onerror="myFunction()">
```

Browser Support

The `onerror` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
body	Yes	Yes	Yes	Yes	Yes
embed	Yes	Yes	Yes	Yes	Yes
img	Yes	Yes	Yes	Yes	Yes
object	Yes	Yes	Yes	Yes	Yes
link	Yes	Yes	Yes	Yes	Yes
script	Yes	Yes	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onfocus Attribute

Definition and Usage

The onfocus attribute fires the moment that the element gets focus.

Onfocus is most often used with <input>, <select>, and <a>.

Tip: The onfocus attribute is the opposite of the onblur attribute.

Applies to

The onfocus attribute is part of the Event Attributes, and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onfocus</u>

Example

Input Example

Execute a JavaScript when an input field gets focus:

```
<input type="text" id="fname" onfocus="myFunction(this.id)">
```

Browser Support

Event Attribute					
onfocus	Yes	Yes	Yes	Yes	Yes

HTML onhashchange Attribute

Definition and Usage

The onhashchange attribute fires when there has been changes to the anchor part (begins with a '#' symbol) of the current URL.

An example of what an anchor part actually is: Assume that the current URL is <http://www.example.com/test.htm#part2> - The anchor part of this URL would be #part2. To invoke this event, you can:

- Change the anchor part by setting the [location.hash](#) or [location.href](#) property of the [Location Object](#)
- Navigate to the current page with a different bookmark (Use the "back" or "forward" buttons)
- Click on a link to a bookmark anchor

Applies to

The onhashchange attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<code><body></code>	<code>onhashchange</code>

Example

Execute a JavaScript when the anchor part has been changed:

```
<body onhashchange="myFunction()">
```

Browser Support

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

Event Attribute					
onhashchange	5.0	8.0	3.6	5.0	10.6

HTML oninput Attribute

Definition and Usage

The oninput attribute fires when an element gets user input.

The oninput attribute fires when the value of an <input> or <textarea> element is changed.

Tip: This event is similar to the onchange event. The difference is that the oninput event occurs immediately after the value of an element has changed, while onchange occurs when the element loses focus. The other difference is that the onchange event also works on <select> elements.

Applies to

The oninput attribute is part of the Event Attributes, and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>oninput</u>

Example

Input Example

Execute a JavaScript when a user writes something in an <input> field:

```
<input type="text" oninput="myFunction()">
```

Browser Support

Event Attribute					
oninput	Yes	Yes	Yes	Yes	Yes

HTML oninvalid Attribute

Definition and Usage

The oninvalid event occurs when a submittable <input> element is invalid.

For example, the input field is invalid if the required attribute is set and the field is empty (the required attribute specifies that the input field must be filled out before submitting the form).

Applies to

The oninvalid attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	oninvalid

Example

Input Example

Execute a JavaScript when an input field is invalid:

```
<input type="text" oninvalid="alert('You must fill out the form!');" required>
```

Browser Support

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

Event Attribute					
oninvalid	Yes	10.0	Yes	Not supported	Yes

HTML onkeydown Attribute

Definition and Usage

The onkeydown attribute fires when the user is pressing a key (on the keyboard).

Tip: The order of events related to the onkeydown event:

1. onkeydown
2. onkeypress
3. onkeyup

Applies to

The onkeydown attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onkeydown</u>

Example

Input Example

Execute a JavaScript when a user is pressing a key:

```
<input type="text" onkeydown="myFunction()">
```

Browser Support

Event Attribute					
onkeydown	Yes	Yes	Yes	Yes	Yes

HTML onkeypress Attribute

Definition and Usage

The onkeypress attribute fires when the user presses a key (on the keyboard).

Tip: The order of events related to the onkeypress event:

1. onkeydown
2. onkeypress
3. onkeyup

Note: The onkeypress event is not fired for all keys (e.g. ALT, CTRL, SHIFT, ESC) in all browsers. To detect only whether the user has pressed a key, use onkeydown instead, because it works for all keys.

Applies to

The onkeypress attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onkeypress</u>

Example

Input Example

Execute a JavaScript when a user presses a key:

```
<input type="text" onkeypress="displayResult()">
```

Browser Support

Event Attribute					
onkeypress	Yes	Yes	Yes	Yes	Yes

HTML onkeyup Attribute

Definition and Usage

The onkeyup attribute fires when the user releases a key (on the keyboard).

Tip: The order of events related to the onkeyup event:

1. onkeydown
2. onkeypress
3. onkeyup

Applies to

The onkeyup attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onkeyup</u>

Example

Input Example

Execute a JavaScript when a user releases a key:

```
<input type="text" onkeyup="myFunction()">
```

Browser Support

Event Attribute					
onkeyup	Yes	Yes	Yes	Yes	Yes

HTML onload Attribute

Definition and Usage

The `onload` attribute fires when an object has been loaded.

`onload` is most often used within the `<body>` element to execute a script once a web page has completely loaded all content (including images, script files, CSS files, etc.). However, it can be used on other elements as well (see "Supported HTML tags" below).

For input elements, the `onload` attribute is only supported when `<input type="image">`

The `onload` attribute can be used to check the visitor's browser type and browser version, and load the proper version of the web page based on the information.

Applies to

The `onload` attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements

[<body>](#)

[<iframe>](#)

[](#)

[<input>](#)

[<link>](#)

[<script>](#)

[<style>](#)

Examples

Body Example

Execute a JavaScript immediately after a page has been loaded:

```
<body onload="myFunction()">
```

Img Example

Using onload on an `` element. Alert "Image is loaded" immediately after an image has been loaded:

```


<script>
function loadImage() {
    alert("Image is loaded");
}
</script>
```

Browser Support

The `onload` attribute has the following browser support for each element:

Element					
body	Yes	Yes	Yes	Yes	Yes
iframe	Yes	Yes	Yes	Yes	Yes
img	Yes	Yes	Yes	Yes	Yes
link	Yes	Yes	Yes	Yes	Yes
script	Yes	Yes	Yes	Yes	Yes
style	Yes	Yes	Yes	Yes	Yes

HTML loadeddata Attribute

Definition and Usage

The loadeddata event occurs when data for the current frame is loaded, but not enough data to play next frame of the specified audio/video.

During the loading process of an audio/video, the following events occur, in this order:

1. onloadstart
2. ondurationchange
3. onloadedmetadata
4. loadeddata
5. onprogress
6. oncanplay
7. oncanplaythrough

Applies to

The loadeddata attribute is part of the Event Attributes, and can be used on the following elements:

Elements	Event
<u><audio></u>	<u>loadeddata</u>
<u><video></u>	<u>loadeddata</u>

Examples

Audio Example

Run "myFunction" when the loadeddata event occurs:

```
<audio onloadeddata="myFunction()">
```

Video Example

Run "myFunction" when the loadeddata event occurs:

```
<video onloadeddata="myFunction()">
```

Browser Support

The `onloadeddata` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML `onloadedmetadata` Attribute

Definition and Usage

The `loadedmetadata` event occurs when meta data for the specified audio/video has been loaded.

Meta data for audio/video consists of: duration, dimensions (video only) and text tracks. During the loading process of an audio/video, the following events occur, in this order:

1. [loadstart](#)
2. [durationchange](#)
3. `loadedmetadata`
4. [loadeddata](#)
5. [progress](#)
6. [canplay](#)
7. [canplaythrough](#)

Applies to

The `onloadedmetadata` attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<code><audio></code>	<u>loadedmetadata</u>
<code><video></code>	<u>loadedmetadata</u>

Examples

Audio Example

Run "myFunction" when the loadedmetadata event occurs:

```
<audio onloadedmetadata="myFunction()">
```

Video Example

Run "myFunction" when the loadedmetadata event occurs:

```
<video onloadedmetadata="myFunction()">
```

Browser Support

The `onloadedmetadata` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onloadstart Attribute

Definition and Usage

The `onloadstart` event occurs when the browser starts looking for the specified audio/video. This is when the loading process starts.

During the loading process of an audio/video, the following events occur, in this order:

1. `loadstart`
2. `durationchange`
3. `loadedmetadata`
4. `loadeddata`
5. `progress`
6. `canplay`
7. `canplaythrough`

Applies to

The `onloadstart` attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<code><audio></code>	<code>loadstart</code>
<code><video></code>	<code>loadstart</code>

Examples

Audio Example

Run "myFunction" when the data starts to load:

```
<audio onloadstart="myFunction()">
```

Video Example

Run "myFunction" when the data starts to load:

```
<video onloadstart="myFunction()">
```

Browser Support

The `onloadstart` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onmousedown Attribute

Definition and Usage

The onmousedown attribute fires when a mouse button is pressed down on the element.

Tip: The order of events related to the onmousedown event (for the left/middle mouse button):

1. onmousedown
2. onmouseup
3. onclick

The order of events related to the onmousedown event (for the right mouse button):

1. onmousedown
2. onmouseup
3. oncontextmenu

Applies to

The onmousedown attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onmousedown</u>

Example

P Example

Execute a JavaScript when pressing a mouse button over a paragraph:

```
<p onmousedown="mouseDown()">Click the text!</p>
```

Browser Support

Event Attribute					
onmousedown	Yes	Yes	Yes	Yes	Yes

HTML onmousemove Attribute

Definition and Usage

The onmousemove attribute fires when the pointer is moving while it is over an element.

Applies to

The onmousemove attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	onmousemove

Example

Img Example

Execute a JavaScript when moving the mouse pointer over an image:

```

```

Browser Support

Event Attribute					
onmousemove	Yes	Yes	Yes	Yes	Yes

HTML onmouseout Attribute

Definition and Usage

The onmouseout attribute fires when the mouse pointer moves out of an element.

Tip: The onmouseout attribute is often used together with the [onmouseover](#) attribute.

Applies to

The onmouseout attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onmouseout</u>

Example

Img Example

Execute a JavaScript when moving the mouse pointer out of an image:

```

```

Browser Support

Event Attribute					
onmouseout	Yes	Yes	Yes	Yes	Yes

HTML onmouseover Attribute

Definition and Usage

The onmouseover attribute fires when the mouse pointer moves over an element.

Tip: The onmouseover attribute is often used together with the onmouseout attribute.

Applies to

The onmouseover attribute is part of the Event Attributes, and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onmouseover</u>

Example

Img Example

Execute a JavaScript when moving the mouse pointer over an image:

```

```

Browser Support

Event Attribute					
onmouseover	Yes	Yes	Yes	Yes	Yes

HTML onmouseup Attribute

Definition and Usage

The onmouseup attribute fires when a mouse button is released over the element.

Tip: The order of events related to the onmouseup event (for the left/middle mouse button):

1. onmousedown
2. onmouseup
3. onclick

The order of events related to the onmouseup event (for the right mouse button):

1. onmousedown
2. onmouseup
3. oncontextmenu

Applies to

The onmouseup attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onmouseup</u>

Example

P Example

Execute a JavaScript when releasing a mouse button over a paragraph:

```
<p onmouseup="mouseUp()">Click the text!</p>
```

Browser Support

Event Attribute					
onmouseup	Yes	Yes	Yes	Yes	Yes

HTML onmousewheel Attribute

Definition and Usage

The onmousewheel attribute fires when the mouse wheel is rolled up or down over an element.

Deprecated. The onmousewheel attribute is deprecated, you should use the [onwheel](#) attribute instead.

Applies to

The onmousewheel attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	onmousewheel

Example

Img Example

Execute a JavaScript when the user rolls the mouse wheel over a <div> element:

```
<div onmousewheel="myFunction()">Roll the mouse wheel over me</div>
```

Browser Support

Event Attribute					
onmousewheel	Yes	Yes	Not supported	Yes	Yes

HTML onoffline Attribute

Definition and Usage

The onoffline attribute fires when the browser starts to work offline.

Tip: The onoffline attribute is the opposite of the [ononline](#) attribute.

Applies to

The onoffline attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<u><body></u>	<u>onoffline</u>

Body Example

Execute a JavaScript when the browser starts to work offline:

```
<body onoffline="myFunction()">
```

Browser Support

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

Event Attribute					
onoffline	Not supported	Deprecated as of IE18.0	3.0	Not supported	Not supported

HTML ononline Attribute

Definition and Usage

The ononline attribute fires when the browser starts to work online.

Tip: The ononline attribute is the opposite of the [onoffline](#) attribute.

Applies to

The ononline attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<u><body></u>	<u>ononline</u>

Body Example

Execute a JavaScript when the browser starts to work online:

```
<body ononline="myFunction()">
```

Browser Support

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

Event Attribute					
ononline	Not supported	Deprecated as of IE18.0	3.0	Not supported	Not supported

HTML onpageshow Attribute

Definition and Usage

The onpageshow event occurs when a user navigates to a webpage.

The onpageshow event is similar to the [onload](#) event, except that it occurs after the onload event when the page first loads. Also, the onpageshow event occurs every time the page is loaded, whereas the onload event does not occur when the page is loaded from the cache.

Applies to

The onpageshow attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<code><body></code>	<code>onpageshow</code>

Example

Execute a JavaScript when a user navigates to a webpage:

```
<body onpageshow="myFunction()">
```

Browser Support

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

Event Attribute					
onpageshow	Yes	11.0	Yes	5.0	Yes

HTML onpaste Attribute

Definition and Usage

The onpaste attribute fires when the user pastes some content in an element.

Note: Although the onpaste attribute is supported by all HTML elements, it is not actually possible to paste some content in, for example, a `<p>` element, UNLESS the element has set contenteditable to "true" (See "More Examples" below).

Tip: The onpaste attribute is mostly used on `<input>` elements with type="text".

Tip: There are three ways to paste some content in an element:

- Press CTRL + V
- Select "Paste" from the Edit menu in your browser
- Right click to display the context menu and select the "Paste" command

Applies to

The onpaste attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	onpaste

Example

Input Example

Execute a JavaScript when pasting some text in an `<input>` element:

```
<input type="text" onpaste="myFunction()" value="Paste something in here">
```

P Example

Execute a JavaScript when pasting some text in a `<p>` element (Note that contenteditable is set to "true"):

```
<p contenteditable="true" onpaste="myFunction()">Try to paste something  
inside this paragraph.</p>
```

Browser Support

Event Attribute					
onpaste	Yes	Yes	Yes	Yes	Yes

HTML onpause Attribute

Definition and Usage

The onpause attribute defines a script to be run when the audio/video is paused either by the user or programmatically.

Tip: The onplay attribute is used to define a script to run when the audio/video has been started or is no longer paused.

Applies to

The onpause attribute is part of the Event Attributes, and can be used on the following elements:

Elements	Event
<u><audio></u>	<u>pause</u>
<u><video></u>	<u>pause</u>

Examples

Audio Example

Run "myFunction" if the audio is paused:

```
<audio onpause="myFunction()">
```

Video Example

Run "myFunction" if the video is paused:

```
<video onpause="myFunction()">
```

Browser Support

The `onpause` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onplay Attribute

Definition and Usage

The onplay attribute defines a script to be run when the audio/video has been started or is no longer paused.

Tip: The onpause attribute is used to define a script when the audio/video has been paused.

Applies to

The onplay attribute is part of the Event Attributes, and can be used on the following elements:

Elements	Event
<audio>	play
<video>	play

Examples

Audio Example

Run "myFunction" when the audio file is being played:

```
<audio onplay="myFunction()">
```

Video Example

Run "myFunction" when the video is being played:

```
<video onplay="myFunction()">
```

Browser Support

The `onplay` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onplaying Attribute

Definition and Usage

The onplaying attributes defines a script to run when the audio/video has started playing.

Applies to

The onplaying attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<audio>	playing
<video>	playing

Examples

Audio Example

Run "myFunction" when the audio file is being played:

```
<audio onplaying="myFunction()">
```

Video Example

Run "myFunction" when the video is being played:

```
<video onplaying="myFunction()">
```

Browser Support

The `onplaying` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onprogress Attribute

Definition and Usage

The onprogress attribute defines a script to be run when the browser is downloading the specified audio/video.

During the loading process of an audio/video, the following events occur, in this order:

1. onloadstart
2. ondurationchange
3. onloadedmetadata
4. onloadeddata
5. onprogress
6. oncanplay
7. oncanplaythrough

Applies to

The onprogress attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<u><audio></u>	<u>progress</u>
<u><video></u>	<u>progress</u>

Examples

Audio Example

Run "myFunction" when the audio file is being downloaded:

```
<audio onprogress="myFunction()">
```

Video Example

Run "myFunction" when the video is being downloaded:

```
<video onprogress="myFunction()">
```

Browser Support

The `onprogress` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onratechange Attribute

Definition and Usage

The onratechange attribute defines a script to be run when the playing speed of the audio/video is changed (like when a user switches to a slow motion or fast forward mode).

Applies to

The onratechange attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<code><audio></code>	<code>ratechange</code>
<code><video></code>	<code>ratechange</code>

Examples

Audio Example

Run "myFunction" when the audio changes the rate:

```
<audio onratechange="myFunction()">
```

Video Example

Run "myFunction" when the video changes the rate:

```
<video onratechange="myFunction()">
```

Browser Support

The `onratechange` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onreset Attribute

Definition and Usage

The onreset attribute fires when a form is reset.

Applies to

The onreset attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<form>	<u>onreset</u>

Example

Execute a JavaScript when the Reset button in a form is clicked:

```
<form onreset="myFunction()>
  Enter name: <input type="text">
  <input type="reset">
</form>
```

Browser Support

Event Attribute					
onreset	Yes	Yes	Yes	Yes	Yes

HTML onresize Attribute

Definition and Usage

The onresize attribute fires when the browser window is resized.

Applies to

The onresize attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<body>	onresize

Example

Execute a JavaScript when the browser window is resized:

```
<body onresize="myFunction()">
```

Browser Support

Event Attribute					
onresize	Yes	Yes	Yes	Yes	Yes

HTML onscroll Attribute

Definition and Usage

The onscroll attribute fires when an element's scrollbar is being scrolled.

Tip: use the CSS overflow style property to create a scrollbar for an element.

Applies to

The onscroll attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onscroll</u>

Example

Div Example

Execute a JavaScript when a <div> element is being scrolled:

```
<div onscroll="myFunction()">
```

Browser Support

Event Attribute					
onscroll	Yes	Yes	Yes	Yes	Yes

HTML onsearch Attribute

Definition and Usage

The onsearch attribute fires when a user presses the "ENTER" key or clicks the "x" button in an <input> element with type="search".

Applies to

The onsearch attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<input type="search">	onsearch

Example

Execute a JavaScript when submitting a search:

```
<input type="search" onsearch="myFunction()">
```

Browser Support

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

Event Attribute					
onsearch	Yes	Not supported	Not supported	Yes	15.0

HTML onseeked Attribute

Definition and Usage

The onseeked attribute defines a script to run when the user is finished moving/skipping to a new position in the audio/video.

Tip: Use the currentTime property of the Audio/Video Object to get the current playback position.

Applies to

The onseeked attribute is part of the Event Attributes, and can be used on the following elements:

Elements	Event
<u><audio></u>	<u>seeked</u>
<u><video></u>	<u>seeked</u>

Examples

Audio Example

Run "myFunction" when the audio changes the position:

```
<audio onseeked="myFunction()">
```

Video Example

Run "myFunction" when the video changes the position:

```
<video onseeked="myFunction()">
```

Browser Support

The `onseeked` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onseeking Attribute

Definition and Usage

The onseeking attribute defines a script to run when the user *starts* moving/skipping to a new position in the audio/video.

Tip: Use the currentTime property of the Audio/Video Object to get the current playback position.

Applies to

The onseeking attribute is part of the Event Attributes, and can be used on the following elements:

Elements	Event
<u><audio></u>	<u>seeking</u>
<u><video></u>	<u>seeking</u>

Examples

Audio Example

Run "myFunction" when the user starts changing the position of the audio:

```
<audio onseeking="myFunction()">
```

Video Example

Run "myFunction" when the user starts changing the position of the video:

```
<video onseeking="myFunction()">
```

Browser Support

The `onseeking` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onselect Attribute

Definition and Usage

The onselect attribute fires after some text has been selected in an element.

Applies to

The onselect attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	<u>onselect</u>

Example

Input Example

Execute a JavaScript after some text has been selected in an <input> element:

```
<input type="text" onselect="myFunction()" value="Hello world!">
```

Browser Support

Event Attribute					
onselect	Yes	Yes	Yes	Yes	Yes

HTML `onstalled` Attribute

Definition and Usage

The `onstalled` attribute defines a script to run when the browser is trying to get media data, but data is not available.

Tip: Related events that occurs when there is some kind of disturbance to the media loading process, are:

- [onabort](#)
- [onemptied](#)
- [onerror](#)
- [onsuspend](#)

Applies to

The `onstalled` attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<code><audio></code>	<u>stalled</u>
<code><video></code>	<u>stalled</u>

Browser Support

The `onstalled` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onsubmit Attribute

Definition and Usage

The onsubmit attribute fires when a form is submitted.

Applies to

The onsubmit attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<form>	<u>onsubmit</u>

Example

Execute a JavaScript when a form is submitted:

```
<form onsubmit="myFunction()>
  Enter name: <input type="text">
  <input type="submit">
</form>
```

Browser Support

Event Attribute					
onsubmit	Yes	Yes	Yes	Yes	Yes

HTML onsuspend Attribute

Definition and Usage

The `onsuspend` attribute defines a script to run when the browser is intentionally not getting media data.

This event occurs when the loading of the media is suspended (prevented from continuing). This can happen when the download has completed, or because it has been paused for some reason.

Tip: Related events that occurs when there is some kind of disturbance to the media loading process, are:

- [onabort](#)
- [onemptied](#)
- [onerror](#)
- [onstalled](#)

Applies to

The `onsuspend` attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<code><audio></code>	<u>suspend</u>
<code><video></code>	<u>suspend</u>

Browser Support

The `onsuspend` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML ontimeupdate Attribute

Definition and Usage

The ontimeupdate attribute defines a script to run when the playing position of an audio/video has changed.

This event is invoked by:

- Playing the audio/video
- Moving the playback position (like when the user fast forwards to a different point in the audio/video)

Tip: This timeupdate event is often used together with the `currentTime` property of the Audio/Video Object, which returns the current position of the audio/video playback, in seconds.

Applies to

The onseeking attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<code><audio></code>	<code>seeking</code>
<code><video></code>	<code>seeking</code>

Examples

Audio Example

Run "myFunction" when the position of the audio changes:

```
<audio ontimetimeupdate="myFunction()">
```

Video Example

Run "myFunction" when the position of the video changes:

```
<video ontimetimeupdate="myFunction()">
```

Browser Support

The `ontimeupdate` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML ontoggle Attribute

Definition and Usage

The ontoggle attribute fires when the user opens or closes the <details> element.

The <details> element specifies additional details that the user can view or hide on demand.

Applies to

The ontoggle attribute is part of the [Event Attributes](#), and can be used on the following element:

Elements	Event
<u><details></u>	<u>ontoggle</u>

Example

Execute a JavaScript when a <details> element is opened or closed:

```
<details ontoggle="myFunction()">
```

Browser Support

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

Event Attribute					
ontoggle	12.0	Not supported	Not supported	6.0	15.0

HTML onunload Attribute

Definition and Usage

The onunload attribute fires once a page has unloaded (or the browser window has been closed).

onunload occurs when the user navigates away from the page (by clicking on a link, submitting a form, closing the browser window, etc.)

Note: If you reload a page, you will also trigger the onunload event (and the onload event).

Applies to

The onunload attribute is part of the Event Attributes, and can be used on the following element:

Elements	Event
<u><body></u>	<u>onunload</u>

Example

Execute a JavaScript when a user unloads the document:

```
<body onunload="myFunction()">
```

Browser Support

Event					
onunload	Yes	Yes	Yes	Yes	Yes

Note: Due to different browser settings, this event may not always work as expected.

HTML onvolumechange Attribute

Definition and Usage

The onvolumechange attribute defines a script to run each time the volume of a video/audio has been changed.

This event is invoked by:

- Increasing or decreasing the volume
- Muting or unmuting the media player

Tip: Use the volume property of the Audio/Video Object to set or return the audio volume of an audio/video.

Applies to

The onvolumechange attribute is part of the Event Attributes, and can be used on the following elements:

Elements	Event
<u><audio></u>	<u>volumechange</u>
<u><video></u>	<u>volumechange</u>

Examples

Audio Example

Run "myFunction" when the volume of the audio changes:

```
<audio onvolumechange="myFunction()">
```

Video Example

Run "myFunction" when the volume of the video changes:

```
<video onvolumechange="myFunction()">
```

Browser Support

The `onvolumechange` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onwaiting Attribute

Definition and Usage

The onwaiting attribute defines a script to run when the video stops because it needs to buffer the next frame.

This event can also be used on <audio> elements, but it is mostly used for videos.

Applies to

The onwaiting attribute is part of the [Event Attributes](#), and can be used on the following elements:

Elements	Event
<audio>	waiting
<video>	waiting

Examples

Audio Example

Run "myFunction" when the audio needs to buffer:

```
<audio onwaiting="myFunction()">
```

Video Example

Run "myFunction" when the video needs to buffer:

```
<video onwaiting="myFunction()">
```

Browser Support

The `onwaiting` attribute has the following browser support for each element:

Element					
audio	Yes	9.0	Yes	Yes	Yes
video	Yes	9.0	Yes	Yes	Yes

HTML onwheel Attribute

Definition and Usage

The onwheel attribute fires when the wheel of a pointing device is rolled up or down over an element.

The onwheel attribute also fires when the user scrolls or zooms on an element by using a touchpad (like the "mouse" of a laptop).

Applies to

The onwheel attribute is part of the [Event Attributes](#), and can be used on any HTML elements.

Elements	Event
All HTML elements	onwheel

Example

Div Example

Execute a JavaScript when the user rolls the mouse wheel over a <div> element:

```
<div onwheel="myFunction()">Roll the mouse wheel over me</div>
```

Browser Support

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

Event Attribute					
onwheel	31.0	Not supported	17.0	Not supported	18.0

HTML open Attribute

Definition and Usage

The open attribute is a boolean attribute.

When present, it specifies that the details should be visible (open) by default.

Applies to

The open attribute can be used on the following element:

Element	Attribute
<u><details></u>	<u>open</u>

Example

An open/visible <details> element:

```
<details open>
  <summary>Copyright 1999-2014.</summary>
  <p> - by Refsnes Data. All Rights Reserved.</p>
  <p>All content and graphics on this web site are the property of the
  company Refsnes Data.</p>
</details>
```

Browser Support

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

Attribute					
open	12.0	Not supported	Not supported	6.0	15.0

HTML optimum Attribute

Definition and Usage

The optimum attribute specifies the range where the gauge's value is considered to be an optimal value.

Applies to

The optimum attribute can be used on the following element:

Element	Attribute
<code><meter></code>	<u>optimum</u>

Example

A gauge with an optimal value of 0.5:

```
<meter value="0.3" high="0.9" low="0.1" optimum="0.5"></meter>
```

Browser Support

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

Attribute					
optimum	8.0	Not supported	6.0	6.0	11.0

HTML pattern Attribute

Definition and Usage

The pattern attribute specifies a regular expression that the <input> element's value is checked against.

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

Tip: Use the global title attribute to describe the pattern to help the user.

Tip: Learn more about regular expressions in our JavaScript tutorial.

Applies to

The pattern attribute can be used on the following element:

Element	Attribute
<input>	<u>pattern</u>

Examples

Input Example

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

```
<form action="/action_page.php">
Country code: <input type="text" name="country_code"
pattern="[A-Za-z]{3}" title="Three letter country code">
<input type="submit">
</form>
```

Password Example

An <input> element with type="password" that must contain 6 or more characters:

```
<form action="/action_page.php">  
  Password: <input type="password" name="pw" pattern=".{6,}" title="Six or  
  more characters">  
  <input type="submit">  
</form>
```

Password Example

An <input> element with type="password" that must contain 8 or more characters that are of at least one number, and one uppercase and lowercase letter:

```
<form action="/action_page.php">  
  Password: <input type="password" name="pw" pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}" title="Must contain at least one number and one uppercase  
  and lowercase letter, and at least 8 or more characters">  
  <input type="submit">  
</form>
```

Browser Support

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

Attribute					
pattern	5.0	10.0	4.0	Not supported	9.6

HTML placeholder Attribute

Definition and Usage

The placeholder attribute specifies a short hint that describes the expected value of a input field / textarea.

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

Applies to

The placeholder attribute can be used on the following elements:

Elements	Attribute
<code><input></code>	<code>placeholder</code>
<code><textarea></code>	<code>placeholder</code>

Examples

Input Example

Two input fields with a placeholder text:

```
<form action="/action_page.php">
  <input type="text" name="fname" placeholder="First name"><br>
  <input type="text" name="lname" placeholder="Last name"><br>
  <input type="submit" value="Submit">
</form>
```

Textarea Example

A text area with a placeholder text:

```
<textarea placeholder="Describe yourself here..."></textarea>
```

Browser Support

The `placeholder` attribute has the following browser support for each element:

Element					
input	10.0	10.0	4.0	5.0	11.0
textarea	Yes	10.0	4.0	5.0	11.5

HTML poster Attribute

Definition and Usage

The poster attribute specifies an image to be shown while the video is downloading, or until the user hits the play button. If this is not included, the first frame of the video will be used instead.

Applies to

The poster attribute can be used on the following element:

Elements	Attribute
<u><video></u>	<u>poster</u>

Example

Video Example

A video player with a poster image:

```
<video controls poster="/images/w3html5.gif">
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Browser Support

The **poster** attribute has the following browser support:

Element					
video	4.0	9.0	3.5	4.0	10.5

HTML preload Attribute

Definition and Usage

The preload attribute specifies if and how the author thinks that the media file should be loaded when the page loads.

The preload attribute allows the author to provide a hint to the browser about what he/she thinks will lead to the best user experience. This attribute may be ignored in some instances.

Note: The preload attribute is ignored if autoplay is present.

Applies to

The preload attribute can be used on the following elements:

Elements	Attribute
<u><audio></u>	<u>preload</u>
<u><video></u>	<u>preload</u>

Examples

Audio Example

Author thinks that the sound should NOT be loaded when the page loads:

```
<audio controls preload="none">
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>
```

Video Example

Author thinks that the video should NOT be loaded when the page loads:

```
<video controls preload="none">
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Browser Support

The `preload` attribute has the following browser support for each element:

Element					
audio	4.0	9.0	4.0	4.0	10.5
video	4.0	Not supported	4.0	4.0	10.5

HTML readonly Attribute

Definition and Usage

The readonly attribute is a boolean attribute.

When present, it specifies that an input field or textarea is read-only.

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

The readonly attribute can be set to keep a user from changing the value until some other conditions have been met (like selecting a checkbox, etc.). Then, a JavaScript can remove the readonly value, and make the input field editable.

Applies to

The readonly attribute can be used on the following elements:

Elements	Attribute
<u><input></u>	<u>readonly</u>
<u><textarea></u>	<u>readonly</u>

Examples

Input Example

An HTML form with a read-only input field:

```
<form action="/action_page.php">
    Country: <input type="text" name="country" value="Norway" readonly><br>
    <input type="submit" value="Submit">
</form>
```

Textarea Example

A read-only text area:

```
<textarea readonly>
At w3schools.com you will learn how to make a website. We offer free
tutorials in all web development technologies.
</textarea>
```

Browser Support

The `readonly` attribute has the following browser support for each element:

Element					
input	1.0	6.0	1.0	1.0	1.0
textarea	Yes	Yes	Yes	Yes	Yes

HTML rel Attribute

Definition and Usage

The rel attribute specifies the relationship between the current document and the linked document/resource.

Applies to

The rel attribute can be used on the following elements:

Elements	Attribute
<u><a></u>	<u>rel</u>
<u><area></u>	<u>rel</u>
<u><link></u>	<u>rel</u>

Examples

A Example

A link with a rel attribute:

```
<a rel="nofollow" href="http://www.functravel.com/">Cheap Flights</a>
```

Area Example

An image-map, with a clickable area:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" alt="Sun"
    href="sun.htm" rel="alternate">
</map>
```

Link Example

An external stylesheet:

```
<link rel="stylesheet" type="text/css" href="theme.css">
```

Browser Support

The `rel` attribute has the following browser support for each element:

Element					
a	Yes	Yes	Yes	Yes	Yes
area	Yes	Yes	Yes	Yes	Yes
link	Yes	Yes	Yes	Yes	Yes

HTML required Attribute

Definition and Usage

The required attribute is a boolean attribute.

When present, it specifies that the element must be filled out before submitting the form.

Applies to

The required attribute can be used on the following elements:

Elements	Attribute
<u><input></u>	<u>required</u>
<u><select></u>	<u>required</u>
<u><textarea></u>	<u>required</u>

Examples

Input Example

An HTML form with a required input field:

```
<form action="/action_page.php">
  Username: <input type="text" name="username" required>
  <input type="submit">
</form>
```

Select Example

An HTML form with a required drop-down list:

```
<select required>
  <option value="">None</option>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="mercedes">Mercedes</option>
  <option value="audi">Audi</option>
</select>
```

Textarea Example

A form with a required text area:

```
<form action="/action_page.php">
  <textarea name="comment" required></textarea>
  <input type="submit">
</form>
```

Browser Support

The `required` attribute has the following browser support for each element:

Element					
input	5.0	10.0	4.0	Not supported	9.6
select	Yes	10.0	Yes	Not supported	Yes
textarea	5.0	10.0	4.0	Not supported	Yes

HTML reversed Attribute

Definition and Usage

The reversed attribute is a boolean attribute.

When present, it specifies that the list order should be descending (9,8,7...), instead of ascending (1, 2, 3...).

Applies to

The reversed attribute can be used on the following element:

Element	Attribute
<u></u>	<u>reversed</u>

Example

Descending list order:

```
<ol reversed>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Browser Support

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

Attribute					
reversed	18.0	Not supported	18.0	5.2	Yes

HTML rows Attribute

Definition and Usage

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

Note: The size of a textarea can also be specified by the CSS height and width properties.

Applies to

The rows attribute can be used on the following element:

Element	Attribute
<u><textarea></u>	<u>rows</u>

Example

A text area with a specified height and width:

```
<textarea rows="4" cols="50">  
At w3schools.com you will learn how to make a website. We offer free  
tutorials in all web development technologies.  
</textarea>
```

Browser Support

Attribute					
rows	Yes	Yes	Yes	Yes	Yes

HTML rowspan Attribute

Definition and Usage

The rowspan attribute specifies the number of rows a cell should span.

Applies to

The rowspan attribute can be used on the following elements:

Elements	Attribute
<u><td></u>	<u>rowspan</u>
<u><th></u>	<u>rowspan</u>

Examples

Td Example

An HTML table with a table cell that spans two rows:

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
    <th>Savings for holiday!</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
    <td rowspan="2">$50</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$80</td>
  </tr>
</table>
```

Th Example

An HTML table with a header cell that spans three rows:

```
<table>
  <tr>
    <th>Month</th>
    <th>Savings</th>
    <th rowspan="3">Savings for holiday!</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$80</td>
  </tr>
</table>
```

Browser Support

The `rowspan` attribute has the following browser support for each element:

Element					
td	Yes	Yes	Yes	Yes	Yes
th	Yes	Yes	Yes	Yes	Yes

HTML sandbox Attribute

Definition and Usage

The sandbox attribute enables an extra set of restrictions for the content in the iframe.

When the sandbox attribute is present, and it will:

- treat the content as being from a unique origin
- block form submission
- block script execution
- disable APIs
- prevent links from targeting other browsing contexts
- prevent content from using plugins (through `<embed>`, `<object>`, `<applet>`, or other)
- prevent the content to navigate its top-level browsing context
- block automatically triggered features (such as automatically playing a video or automatically focusing a form control)

The value of the sandbox attribute can either be just `sandbox` (then all restrictions are applied), or a space-separated list of pre-defined values that will REMOVE the particular restrictions.

Applies to

The sandbox attribute can be used on the following element:

Element	Attribute
<code><iframe></code>	<code>sandbox</code>

Example

An `<iframe>` with extra restrictions:

```
<iframe src="demo_iframe_sandbox.htm" sandbox></iframe>
```

Browser Support

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

Attribute					
<code>sandbox</code>	4.0	10.0	17.0	5.0	15.0

HTML scope Attribute

Definition and Usage

The scope attribute specifies whether a header cell is a header for a column, row, or group of columns or rows.

The scope attribute has no visual effect in ordinary web browsers, but can be used by screen readers.

Applies to

The scope attribute can be used on the following element:

Element	Attribute
<code><th></code>	<code>scope</code>

Example

Specify that the two header cells are headers for columns:

```
<table>
  <tr>
    <th></th>
    <th scope="col">Month</th>
    <th scope="col">Savings</th>
  </tr>
  <tr>
    <td>1</td>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>2</td>
    <td>February</td>
    <td>$80</td>
  </tr>
</table>
```

Browser Support

Attribute					
scope	Yes	Yes	Yes	Yes	Yes

HTML selected Attribute

Definition and Usage

The selected attribute is a boolean attribute.

When present, it specifies that an option should be pre-selected when the page loads.

The pre-selected option will be displayed first in the drop-down list.

Tip: The selected attribute can also be set after the page loads, with a JavaScript.

Applies to

The selected attribute can be used on the following element:

Element	Attribute
<code><option></code>	<u>selected</u>

Example

A drop-down list with a pre-selected option:

```
<select>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="vw">VW</option>
  <option value="audi" selected>Audi</option>
</select>
```

Browser Support

Attribute					
selected	Yes	Yes	Yes	Yes	Yes

HTML shape Attribute

Definition and Usage

The shape attribute specifies the shape of an area.

The shape attribute is used together with the coords attribute to specify the size, shape, and placement of an area.

Applies to

The shape attribute can be used on the following element:

Element	Attribute
<u><area></u>	<u>shape</u>

Example

An image-map, with clickable areas:

```
<img src ="planets.gif" width="145" height="126" alt="Planets"
usemap="#planetmap">

<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>
```

Browser Support

Attribute					
shape	Yes	Yes	Yes	Yes	Yes

HTML size Attribute

Definition and Usage

For input elements, the size attribute specifies the visible width, in characters, of an <input> element.

For select elements, the size attribute specifies the number of visible options in a drop-down list.

Applies to

The size attribute can be used on the following elements:

Elements	Attribute
<input>	<u>size</u>
<select>	<u>size</u>

Examples

Input Example

An HTML form with two input fields with a width of 35 and 4 characters:

```
<form action="/action_page.php">
  Email: <input type="text" name="email" size="35"><br>
  PIN: <input type="text" name="pin" maxlength="4" size="4"><br>
  <input type="submit" value="Submit">
</form>
```

Select Example

A drop-down list with three visible options:

```
<select size="3">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi">Audi</option>
</select>
```

Browser Support

The `size` attribute has the following browser support for each element:

Element					
input	1.0	2.0	1.0	1.0	1.0
select	Yes	Yes	Yes	Yes	Yes

HTML sizes Attribute

Definition and Usage

The sizes attribute specifies the sizes of icons for visual media. This attribute is only used if rel="icon".

Applies to

The sizes attribute can be used on the following element:

Element	Attribute
<link>	<u>sizes</u>

Example

Icon with specified size:

```
<link rel="icon" href="demo_icon.gif" type="image/gif" sizes="16x16">
```

Browser Support

Attribute					
sizes	Not supported	Not supported	Not supported	Not supported	Not supported

HTML span Attribute

Definition and Usage

The span attribute defines the number of columns a <col>/<colgroup> element should span.

Applies to

The span attribute can be used on the following elements:

Elements	Attribute
<col>	<u>span</u>
<colgroup>	<u>span</u>

Examples

Col Example

Here, the first two columns should have a background color of red:

```
<table>
  <colgroup>
    <col span="2" style="background-color:red">
    <col style="background-color:yellow">
  </colgroup>
  <tr>
    <th>ISBN</th>
    <th>Title</th>
    <th>Price</th>
  </tr>
  <tr>
    <td>3476896</td>
    <td>My first HTML</td>
    <td>$53</td>
  </tr>
</table>
```

Colgroup Example

Set the background color of the first two columns using the `<colgroup>` `span` attribute:

```
<table>
  <colgroup span="2" style="background:red"></colgroup>
  <tr>
    <th>ISBN</th>
    <th>Title</th>
    <th>Price</th>
  </tr>
  <tr>
    <td>3476896</td>
    <td>My first HTML</td>
    <td>$53</td>
  </tr>
  <tr>
    <td>5869207</td>
    <td>My first CSS</td>
    <td>$49</td>
  </tr>
</table>
```

Browser Support

The `span` attribute has the following browser support for each element:

Element					
col	Yes	Yes	Yes	Yes	Yes
colgroup	Yes	Yes	Yes	Yes	Yes

HTML spellcheck Attribute

Definition and Usage

The spellcheck attribute specifies whether the element is to have its spelling and grammar checked or not.

The following can be spellchecked:

- Text values in input elements (not password)
- Text in <textarea> elements
- Text in editable elements

Applies to

The spellcheck attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	<u>spellcheck</u>

Example

P Example

An editable paragraph with spellcheck:

```
<p contenteditable="true" spellcheck="true">This is a paragraph.</p>
```

Browser Support

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

Attribute					
spellcheck	9.0	10.0	2.0	5.1	10.5

HTML src Attribute

Definition and Usage

The src attribute specifies the location (URL) of the external resource.

Applies to

The src attribute can be used on the following elements:

Elements	Attribute
<u><audio></u>	<u>src</u>
<u><embed></u>	<u>src</u>
<u><iframe></u>	<u>src</u>
<u></u>	<u>src</u>
<u><input></u>	<u>src</u>
<u><script></u>	<u>src</u>
<u><source></u>	<u>src</u>
<u><track></u>	<u>src</u>
<u><video></u>	<u>src</u>

Examples

Audio Example

An audio player:

```
<audio src="horse.ogg" controls>  
Your browser does not support the audio element.  
</audio>
```

Embed Example

An embedded flash animation:

```
<embed src="helloworld.swf">
```

Iframe Example

An <iframe> in its simplest use:

```
<iframe src="/default.asp"></iframe>
```

Img Example

An image is marked up as follows:

```

```

Input Example

An HTML form with an image that represents the submit button:

```
<form action="/action_page.php">  
First name: <input type="text" name="fname"><br>  
<input type="image" src="submit.gif" alt="Submit">  
</form>
```

Script Example

Point to an external JavaScript file:

```
<script src="myscripts.js"></script>
```

Source Example

An audio player with two source files. The browser should choose which file (if any) it has support for:

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
Your browser does not support the audio element.
</audio>
```

Track Example

A video with two subtitle tracks:

```
<video width="320" height="240" controls>
  <source src="forrest_gump.mp4" type="video/mp4">
  <source src="forrest_gump.ogv" type="video/ogg">
  <track src="subtitles_en.vtt" kind="subtitles" srclang="en"
        label="English">
  <track src="subtitles_no.vtt" kind="subtitles" srclang="no"
        label="Norwegian">
</video>
```

Video Example

Play a video:

```
<video src="movie.ogg" controls>
Your browser does not support the video tag.
</video>
```

Browser Support

The `src` attribute has the following browser support for each element:

Element					
audio	4.0	9.0	3.5	4.0	10.5
embed	Yes	Yes	Yes	Yes	Yes
iframe	Yes	Yes	Yes	Yes	Yes
img	Yes	Yes	Yes	Yes	Yes
input	1.0	2.0	1.0	1.0	1.0
script	Yes	Yes	Yes	Yes	Yes
source	4.0	9.0	3.5	4.0	10.5
track	18.0	10.0	31.0	6.0	15.0
video	4.0	9.0	3.5	4.0	10.5

HTML srcdoc Attribute

Definition and Usage

The srcdoc attribute specifies the HTML content of the page to show in the inline frame.

Tip: This attribute is expected to be used together with the sandbox and seamless attributes.

If a browser supports the srcdoc attribute, it will override the content specified in the src attribute (if present).

If a browser does NOT support the srcdoc attribute, it will show the file specified in the src attribute instead (if present).

Applies to

The srcdoc attribute can be used on the following element:

Element	Attribute
<u><iframe></u>	<u>srcdoc</u>

Example

An <iframe> with a srcdoc attribute:

```
<iframe srcdoc="<p>Hello world!</p>" src="demo_iframe_srcdoc.htm"></iframe>
```

Browser Support

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

Attribute					
srcdoc	20.0	Not supported	25.0	6.0	15.0

HTML srclang Attribute

Definition and Usage

The srclang attribute specifies the language of the track text data.

This attribute is required if kind="subtitles".

Tip: To view all available language codes, go to our [Language code reference](#).

Applies to

The srclang attribute can be used on the following element:

Element	Attribute
<code><track></code>	<code>srclang</code>

Example

A video with two subtitle tracks:

```
<video width="320" height="240" controls>
  <source src="forrest_gump.mp4" type="video/mp4">
  <source src="forrest_gump.ogg" type="video/ogg">
  <track src="subtitles_en.vtt" kind="subtitles" srclang="en"
        label="English">
  <track src="subtitles_no.vtt" kind="subtitles" srclang="no"
        label="Norwegian">
</video>
```

Browser Support

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

Attribute					
srclang	18.0	10.0	31.0	6.0	15.0

HTML <source> srcset Attribute

< HTML <source> tag

Example

A <picture> element with two source files, and a fallback image:

```
<picture>
  <source media="(min-width: 650px)" srcset="img_pink_flowers.jpg">
  <source media="(min-width: 465px)" srcset="img_white_flower.jpg">
  
</picture>
```

Definition and Usage

The srcset attribute specifies the URL of the image to use in different situations.

This attribute is required when <source> is used in <picture>.

Browser Support

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

Attribute					
srcset	38.0	Yes	38.0	9.1	25.0

Differences Between HTML 4.01 and HTML5

The <source> tag is new in HTML5.

Syntax

```
<source srcset="URL">
```

Attribute Values

Value	Description
<i>URL</i>	<p>Specifies the URL of the image.</p> <p>Possible values:</p> <ul style="list-style-type: none">• An absolute URL - points to another web site (like href="http://www.example.com/flower.jpg")• A relative URL - points to a file within a web site (like href="flower.jpg")

< HTML <source> tag

HTML start Attribute

Definition and Usage

The start attribute specifies the start value of the first list item in an ordered list.

Applies to

The start attribute can be used on the following element:

Element	Attribute
<u></u>	<u>start</u>

Example

An ordered list starting at "50":

```
<ol start="50">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

Browser Support

Attribute					
start	Yes	Yes	Yes	Yes	Yes

HTML step Attribute

Definition and Usage

The step attribute specifies the legal number intervals for an <input> element.

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

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

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

Applies to

The step attribute can be used on the following element:

Element	Attribute
<input>	<u>step</u>

Example

An HTML form with an input field with a specified legal number intervals:

```
<form action="/action_page.php">
  <input type="number" name="points" step="3">
  <input type="submit">
</form>
```

Browser Support

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

Attribute					
step	6.0	10.0	16.0	5.0	10.6

HTML style Attribute

Definition and Usage

The style attribute specifies an inline style for an element.

The style attribute will override any style set globally, e.g. styles specified in the <style> tag or in an external style sheet.

Applies to

The style attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	<u>style</u>

Example

Use of the style attribute in an HTML document:

```
<h1 style="color:blue;text-align:center">This is a header</h1>
<p style="color:green">This is a paragraph.</p>
```

Browser Support

Attribute					
style	Yes	Yes	Yes	Yes	Yes

HTML tabindex Attribute

Definition and Usage

The tabindex attribute specifies the tab order of an element (when the "tab" button is used for navigating).

Applies to

The tabindex attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	tabindex

Example

Links with a specified tab order:

```
<a href="https://www.w3schools.com/" tabindex="2">W3Schools</a>
<a href="http://www.google.com/" tabindex="1">Google</a>
<a href="http://www.microsoft.com/" tabindex="3">Microsoft</a>
```

Browser Support

Attribute					
tabindex	Yes	Yes	Yes	Yes	Yes

HTML target Attribute

Definition and Usage

For a elements, and area elements, the target attribute specifies where to open the linked document.

For base elements the target attribute specifies the default target for all hyperlinks and forms in the page.

For form elements, the target attribute specifies a name or a keyword that indicates where to display the response that is received after submitting the form.

Applies to

The target attribute can be used on the following elements:

Elements	Attribute
<u><a></u>	<u>src</u>
<u><area></u>	<u>src</u>
<u><base></u>	<u>src</u>
<u><form></u>	<u>src</u>

Examples

A Example

The target attribute specifies where to open the linked document:

```
<a href="https://www.w3schools.com" target="_blank">Visit W3Schools</a>
```

Area Example

An image-map, with clickable areas, and a target attribute:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun"
target="_blank">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>
```

Base Example

Specify a default target for all hyperlinks and forms on a page:

```
<head>
<base target="_blank">
</head>
```

Form Example

Display the response received in a new window or tab:

```
<form action="/action_page.php" method="get" target="_blank">
    First name: <input type="text" name="fname"><br>
    Last name: <input type="text" name="lname"><br>
    <input type="submit" value="Submit">
</form>
```

Browser Support

The `target` attribute has the following browser support for each element:

Element					
a	Yes	Yes	Yes	Yes	Yes
area	Yes	Yes	Yes	Yes	Yes
base	Yes	Yes	Yes	Yes	Yes
form	Yes	Yes	Yes	Yes	Yes

HTML title Attribute

Definition and Usage

The title attribute specifies extra information about an element.

The information is most often shown as a tooltip text when the mouse moves over the element.

Applies to

The title attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	<u>title</u>

Example

Use of the title attribute in an HTML document:

```
<p><abbr title="World Health Organization">WHO</abbr> was founded in 1948.  
</p>  
<p title="Free Web tutorials">W3Schools.com</p>
```

Browser Support

Attribute					
title	Yes	Yes	Yes	Yes	Yes

HTML translate Attribute

Definition and Usage

The translate attribute specifies whether the content of an element should be translated or not.

Test: Use the Google translate box (at the top of the page) to change to another language, and look what happens to the word "ice cream" below:

Here we use translate="no": ice cream.

Here we use class="nottranslate": ice cream.

Tip: Use class="nottranslate" instead.

Applies to

The translate attribute is part of the [Global Attributes](#), and can be used on any HTML element.

Element	Attribute
All HTML elements	translate

Example

Specify that some elements should not be translated:

```
<p translate="no">Don't translate this!</p>
<p>This can be translated to any language.</p>
```

Browser Support

Attribute					
translate	Not supported	Not supported	Not supported	Not supported	Not supported

HTML type Attribute

Definition and Usage

For button elements, the type attribute specifies the type of button.

For input elements, the type attribute specifies the type of <input> element to display.

For embed, link, object, script, source, and style elements, the type attribute specifies the Internet media type (formerly known as MIME type).

Applies to

The type attribute can be used on the following elements:

Elements	Attribute
<u><button></u>	<u>type</u>
<u><embed></u>	<u>type</u>
<u><input></u>	<u>type</u>
<u><link></u>	<u>type</u>
<u><menu></u>	<u>type</u>
<u><object></u>	<u>type</u>
<u><script></u>	<u>type</u>
<u><source></u>	<u>type</u>
<u><style></u>	<u>type</u>

Examples

Button Example

Two button elements that act as one submit button and one reset button (in a form):

```
<form action="/action_page.php" method="get">
    First name: <input type="text" name="fname"><br>
    Last name: <input type="text" name="lname"><br>
    <button type="submit" value="Submit">Submit</button>
    <button type="reset" value="Reset">Reset</button>
</form>
```

Embed Example

An embedded flash animation with a specified media type:

```
<embed src="helloworld.swf" type="application/vnd.adobe.flash-movie">
```

Input Example

An HTML form with two different input types; text and submit:

```
<form action="/action_page.php">
    Username: <input type="text" name="usrname"><br>
    <input type="submit" value="Submit">
</form>
```

Link Example

In the following example, the type attribute indicates that the linked document is an external style sheet:

```
<head>
<link rel="stylesheet" type="text/css" href="theme.css">
</head>
```

Object Example

An <object> element with a specified media type:

```
<object width="400" height="400" data="helloworld.swf"  
type="application/vnd.adobe.flash-movie"></object>
```

Script Example

A script with the type attribute specified:

```
<script type="text/javascript">  
document.getElementById("demo").innerHTML = "Hello JavaScript!";  
</script>
```

Source Example

Use of the type attribute:

```
<audio controls>  
  <source src="horse.ogg" type="audio/ogg">  
  <source src="horse.mp3" type="audio/mpeg">  
Your browser does not support the audio element.  
</audio>
```

Style Example

Use the type attribute to specify the media type of the <style> tag :

```
<style type="text/css">  
h1 {color:red;}  
p {color:blue;}  
</style>
```

Browser Support

The `type` attribute has the following browser support for each element:

Element					
button	Yes	Yes	Yes	Yes	Yes
embed	Yes	Yes	Yes	Yes	Yes
input	Yes	Yes	Yes	Yes	Yes
link	Yes	Yes	Yes	Yes	Yes
object	Yes	Yes	Yes	Yes	Yes
script	Yes	Yes	Yes	Yes	Yes
source	4.0	9.0	3.5	4.0	10.5
style	Yes	Yes	Yes	Yes	Yes

HTML usemap Attribute

Definition and Usage

The usemap attribute specifies an image (or an object) as an image-map (an image-map is an image with clickable areas).

The usemap attribute is associated with a <map> element's name or id attribute, and creates a relationship between the and the <map>.

Note: The usemap attribute cannot be used if the element is a descendant of an <a> or <button> element.

Applies to

The usemap attribute can be used on the following elements:

Elements	Attribute
<u></u>	<u>usemap</u>
<u><object></u>	<u>usemap</u>

Examples

Example

An image-map, with clickable areas:

```


<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" href="sun.htm" alt="Sun">
  <area shape="circle" coords="90,58,3" href="mercur.htm" alt="Mercury">
  <area shape="circle" coords="124,58,8" href="venus.htm" alt="Venus">
</map>
```

Example

An `<object>` element using an image map:

```
<object data="planets.gif" width="145" height="126" usemap="#planetmap">
</object>
```

Browser Support

The `usemap` attribute has the following browser support for each element:

Element					
img	Yes	Yes	Yes	Yes	Yes
object	Not supported	Yes	Yes	Not supported	Not supported

HTML value Attribute

Definition and Usage

For button, input and option elements, the value attribute specifies the initial value of the element.

For li elements, the value attribute sets the value of an ordered list item. The following list items will increment from that number.

For progress elements, the value attribute specifies how much of the task has been completed.

For param elements, the value attribute specifies the value of a <param> element.

Applies to

The type attribute can be used on the following elements:

Elements	Attribute
<button>	<u>value</u>
<input>	<u>value</u>
	<u>value</u>
<option>	<u>value</u>
<progress>	<u>value</u>
<param>	<u>value</u>

Examples

Button Example

Two buttons with equal names, that submit different values when clicked:

```
<form action="/action_page.php" method="get">
    Choose your favorite subject:
    <button name="subject" type="submit" value="fav_HTML">HTML</button>
    <button name="subject" type="submit" value="fav_CSS">CSS</button>
</form>
```

Input Example

An HTML form with initial (default) values:

```
<form action="/action_page.php">
    First name: <input type="text" name="fname" value="John"><br>
    Last name: <input type="text" name="lname" value="Doe"><br>
    <input type="submit" value="Submit form">
</form>
```

Li Example

Use of the value attribute in an ordered list:

```
<ol>
    <li value="100">Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
    <li>Water</li>
    <li>Juice</li>
    <li>Beer</li>
</ol>
```

Option Example

A drop-down list inside an HTML form:

```
<form action="/action_page.php">
  <select name="cars">
    <option value="volvo">Volvo XC90</option>
    <option value="saab">Saab 95</option>
    <option value="mercedes">Mercedes SLK</option>
    <option value="audi">Audi TT</option>
  </select>
  <input type="submit" value="Submit">
</form>
```

Progress Example

Downloading in progress:

```
<progress value="22" max="100"></progress>
```

Param Example

Set the "autoplay" parameter to "true", so the sound will start playing as soon as the page loads:

```
<object data="horse.wav">
  <param name="autoplay" value="true">
</object>
```

Browser Support

The `value` attribute has the following browser support for each element:

Element					
button	Yes	Yes	Yes	Yes	Yes
input	1.0	2.0	1.0	1.0	1.0
li	Yes	Yes	Yes	Yes	Yes
option	Yes	Yes	Yes	Yes	Yes
progress	8.0	10.0	16.0	6.0	11.0
param	Yes	Yes	Yes	Yes	Yes

HTML width Attribute

Definition and Usage

The width attribute specifies the width of the element, in pixels.

Note: For input elements, the width attribute is used only with `<input type="image">`.

Applies to

The width attribute can be used on the following elements:

Elements	Attribute
<code><canvas></code>	<code>width</code>
<code><embed></code>	<code>width</code>
<code><iframe></code>	<code>width</code>
<code></code>	<code>width</code>
<code><input></code>	<code>width</code>
<code><object></code>	<code>width</code>
<code><video></code>	<code>width</code>

Examples

Canvas Example

A <canvas> element with a height and width of 200 pixels:

```
<canvas id="myCanvas" width="200" height="200" style="border:1px solid">
```

Embed Example

A flash animation with a height and width of 200 pixels:

```
<embed src="helloworld.swf" width="200" height="200">
```

Iframe Example

An <iframe> with a specified height and width of 200 pixels:

```
<iframe src="/default.asp" width="200" height="200"></iframe>
```

Img Example

An image with a height and a width of 42 pixels:

```

```

Input Example

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

```
<form action="/action_page.php">
  First name: <input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  <input type="image" src="img_submit.gif" alt="Submit" width="48"
        height="48">
</form>
```

Object Example

A flash animation with a height and width of 400 pixels::

```
<object data="helloworld.swf" width="400" height="400"></object>
```

Video Example

A video player with a specified width and height:

```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Browser Support

The `width` attribute has the following browser support for each element:

Element					
canvas	4.0	9.0	2.0	3.1	9.0
embed	Yes	Yes	Yes	Yes	Yes
iframe	Yes	Yes	Yes	Yes	Yes
img	Yes	Yes	Yes	Yes	Yes
input	1.0	Yes	16.0	Yes	1.0
object	Yes	Yes	Yes	Yes	Yes
width	Yes	Yes	Yes	Yes	Yes

HTML wrap Attribute

Definition and Usage

The wrap attribute specifies how the text in a text area is to be wrapped when submitted in a form.

Applies to

The wrap attribute can be used on the following element:

Element	Attribute
<u><textarea></u>	<u>wrap</u>

Example

The text in a text area with wrap="hard" will contain newlines (if any) when submitted in a form:

```
<textarea rows="2" cols="20" wrap="hard">  
At W3Schools you will find free Web-building tutorials.  
</textarea>
```

Browser Support

Attribute					
wrap	Yes	Yes	Yes	Yes	Yes