**List of attributes of every tags in htmls**

1. **accept**: Specifies the types of files that the server accepts (input)
2. **accept-charset**: Specifies the character encodings that are to be used for form submission (form)
3. **accesskey**: Defines a keyboard shortcut for an element
4. **action**: Specifies the URL to which the form's data will be submitted (form)
5. **align**: Specifies the alignment of an element (deprecated)
6. **alt**: Defines an alternative text for an image (img, area)
7. **async**: Specifies that the script should be executed asynchronously (script)
8. **autocomplete**: Specifies whether a form should have autocomplete enabled (form, input)
9. **autofocus**: Specifies that the element should automatically get focus when the page loads (button, input, select, textarea)
10. **autoplay**: Specifies that the audio/video will start playing automatically (audio, video)
11. **background**: Specifies the background image for an element (body, table)
12. **bgcolor**: Specifies the background color of an element (deprecated)
13. **border**: Specifies the width of the border of an element (table)
14. **charset**: Specifies the character encoding of the linked resource (a, link, script)
15. **checked**: Specifies that an input element should be pre-selected when the page loads (input)
16. **cite**: Specifies a URL which explains the quote or citation (blockquote, del, ins, q)
17. **class**: Specifies one or more class names for an element (global attribute)
18. **color**: Specifies the text color of an element (deprecated)
19. **cols**: Specifies the number of columns in a textarea (textarea)
20. **colspan**: Specifies the number of columns a table cell should span (td, th)
21. **content**: Specifies the value associated with the http-equiv or name attribute (meta)
22. **contenteditable**: Specifies whether the content of an element is editable or not
23. **controls**: Specifies that audio/video controls should be displayed (audio, video)
24. **coords**: Specifies the coordinates of the area in an image map (area)
25. **crossorigin**: Specifies how the element handles crossorigin requests (img, link, script)
26. **data-\***: Used to store custom data private to the page or application (global attribute)
27. **datetime**: Specifies the date and time of the element's content (del, ins, time)
28. **defer**: Specifies that the script should be executed after the page has finished parsing (script)
29. **dir**: Specifies the text direction of the element's content (global attribute)
30. **disabled**: Specifies that an input element should be disabled (button, fieldset, input, optgroup, option, select, textarea)
31. **download**: Specifies that the target will be downloaded when a user clicks on the hyperlink (a, area)
32. **draggable**: Specifies whether an element is draggable or not
33. **dropzone**: Specifies whether the dragged data is copied, moved, or linked when dropped (global attribute)
34. **enctype**: Specifies how form data should be encoded when submitted to the server (form)
35. **for**: Specifies which form element a label is bound to (label)
36. **form**: Specifies the form that the element belongs to (button, fieldset, input, label, meter, object, output, select, textarea)
37. **formaction**: Specifies the URL to which the form's data will be submitted (button, input)
38. **headers**: Specifies the cells that the header cell is related to (th)
39. **height**: Specifies the height of an element (canvas, embed, iframe, img, input, object, video)
40. **hidden**: Specifies that an element is not yet, or is no longer, relevant (global attribute)
41. **high**: Specifies the high value of a meter element (meter)
42. **href**: Specifies the URL of the linked resource (a, area, base, link)
43. **hreflang**: Specifies the language of the linked resource (a, area, link)
44. **http-equiv**: Provides an HTTP header for the information/value of the content attribute (meta)
45. **id**: Specifies a unique id for an element (global attribute)
46. **ismap**: Specifies an image as a server-side image map (img)
47. **kind**: Specifies the kind of text track (subtitles, captions, descriptions, chapters, metadata) (track)
48. **label**: Specifies the title of the track (track)
49. **lang**: Specifies the language of the element's content (global attribute)
50. **list**: Specifies a list of pre-defined options for input controls (input)
51. **loop**: Specifies that the audio/video will start over again, every time it is finished (audio, video)
52. **low**: Specifies the low value of a meter element (meter)
53. **manifest**: Specifies the URL of the app's cache manifest (html)
54. **max**: Specifies the maximum value for an input element (input, meter, progress)
55. **maxlength**: Specifies the maximum number of characters allowed in an element (input, textarea)
56. **media**: Specifies what media/device the linked document is optimized for (a, area, link, source, style)
57. **method**: Specifies the HTTP method to use when submitting the form (form)
58. **min**: Specifies the minimum value for an input element (input, meter)
59. **multiple**: Specifies that a user can enter more than one value in an input element (input, select)
60. **muted**: Specifies that the audio output of the video should be muted (video)
61. **name**: Specifies the name of an element (button, form, iframe, input, map, meta, object, output, param, select, textarea)
62. **novalidate**: Specifies that the form should not be validated when submitted (form)
63. **onabort**: Specifies a function to be called when the loading of an audio/video is aborted (audio, video)
64. **onafterprint**: Specifies a function to be called after the document is printed (body)
65. **onbeforeprint**: Specifies a function to be called before the document is printed (body)
66. **onbeforeunload**: Specifies a function to be called before the document is unloaded (body)
67. **onblur**: Specifies a function to be called when an element loses focus (button, input, select, textarea)
68. **oncanplay**: Specifies a function to be called when the browser can start playing the audio/video (audio, video)
69. **oncanplaythrough**: Specifies a function to be called when the browser can play through the audio/video without stopping for buffering (audio, video)
70. **onchange**: Specifies a function to be called when the value of an element changes (input, select, textarea)
71. **onclick**: Specifies a function to be called when an element is clicked (a, area, button, input, object, select, textarea)
72. **oncontextmenu**: Specifies a function to be called when the right-click context menu is triggered (body)
73. **oncopy**: Specifies a function to be called when the content of an element is being copied (body)
74. **oncut**: Specifies a function to be called when the content of an element is being cut (body)
75. **ondblclick**: Specifies a function to be called when an element is double-clicked (body)
76. **ondrag**: Specifies a function to be called when an element is being dragged (body)
77. **ondragend**: Specifies a function to be called when the user has finished dragging an element (body)
78. **ondragenter**: Specifies a function to be called when an element enters a drop target (body)
79. **ondragleave**: Specifies a function to be called when an element leaves a drop target (body)
80. **ondragover**: Specifies a function to be called when an element is being dragged over a drop target (body)
81. **ondragstart**: Specifies a function to be called when the user starts to drag an element (body)
82. **ondrop**: Specifies a function to be called when the dragged element is being dropped (body)
83. **ondurationchange**: Specifies a function to be called when the duration of the audio/video is changed (audio, video)
84. **onemptied**: Specifies a function to be called when the media element is reset to its initial state (audio, video)
85. **onended**: Specifies a function to be called when the audio/video has ended (audio, video)
86. **onerror**: Specifies a function to be called when an error occurs during the loading of an audio/video (audio, video)
87. **onfocus**: Specifies a function to be called when an element gets focus (button, input, select, textarea)
88. **onhashchange**: Specifies a function to be called when there has been changes to the anchor part of a URL (body)
89. **oninput**: Specifies a function to be called when an element gets user input (input, select, textarea)
90. **oninvalid**: Specifies a function to be called when an element is invalid (input, select, textarea)
91. **onkeydown**: Specifies a function to be called when the user presses a key (body)
92. **onkeypress**: Specifies a function to be called when the user presses a key (body)
93. **onkeyup**: Specifies a function to be called when the user releases a key (body)
94. **onload**: Specifies a function to be called when the page is fully loaded (body)
95. **onloadeddata**: Specifies a function to be called when the browser has loaded the current frame of the audio/video (audio, video)
96. **onloadedmetadata**: Specifies a function to be called when the browser has loaded meta data for the audio/video (audio, video)
97. **onloadstart**: Specifies a function to be called when the browser starts to load the audio/video (audio, video)
98. **onmousedown**: Specifies a function to be called when a mouse button is pressed down on an element (body)
99. **onmousemove**: Specifies a function to be called when the mouse pointer is moving while it is over an element (body)
100. **onmouseout**: Specifies a function to be called when the mouse pointer moves out of an element (body)
101. **onmouseover**: Specifies a function to be called when the mouse pointer moves over an element (body)
102. **onmouseup**: Specifies a function to be called when a mouse button is released over an element (body)
103. **onmousewheel**: Specifies a function to be called when the mouse wheel rolls up or down over an element (body)
104. **onoffline**: Specifies a function to be called when the browser starts to work offline (body)
105. **ononline**: Specifies a function to be called when the browser starts to work online (body)
106. **onpagehide**: Specifies a function to be called when the user navigates away from a page (body)
107. **onpageshow**: Specifies a function to be called when the user navigates to a page (body)
108. **onpaste**: Specifies a function to be called when the content of an element is being pasted (body)
109. **onpause**: Specifies a function to be called when the audio/video is paused either by the user or programmatically (audio, video)
110. **onplay**: Specifies a function to be called when the audio/video has been started or is no longer paused (audio, video)
111. **onplaying**: Specifies a function to be called when the audio/video is playing after having been paused or stopped for buffering (audio, video)
112. **onpopstate**: Specifies a function to be called when the window's history changes (body)
113. **onprogress**: Specifies a function to be called when the browser is in the process of getting the media data (audio, video)
114. **onratechange**: Specifies a function to be called when the playing speed of the audio/video is changed (audio, video)
115. **onreadystatechange**: Specifies a function to be called when the ready-state changes (body)
116. **onredo**: Specifies a function to be called when the document performs a redo (body)
117. **onresize**: Specifies a function to be called when the window is resized (body)
118. **onscroll**: Specifies a function to be called when the element is being scrolled (body)
119. **onseeked**: Specifies a function to be called when the user is finished moving/skipping to a new position in the audio/video (audio, video)
120. **onseeking**: Specifies a function to be called when the user starts moving/skipping to a new position in the audio/video (audio, video)
121. **onselect**: Specifies a function to be called when the text in the element is being selected by the user (body)
122. **onshow**: Specifies a function to be called when the object is about to be made visible (body)
123. **onstalled**: Specifies a function to be called when the browser is unable to fetch the media data for whatever reason (audio, video)
124. **onstorage**: Specifies a function to be called when a web storage area is updated (body)
125. **onsubmit**: Specifies a function to be called when a form is submitted (form)
126. **onsuspend**: Specifies a function to be called when the browser is intentionally not getting media data (audio, video)
127. **ontimeupdate**: Specifies a function to be called when the playing position of the audio/video is changed (audio, video)
128. **onundo**: Specifies a function to be called when the document performs an undo (body)
129. **onunload**: Specifies a function to be called when the document is unloaded (body)
130. **onvolumechange**: Specifies a function to be called when the volume of the audio/video is changed (audio, video)
131. **onwaiting**: Specifies a function to be called when the video stops because it needs to buffer the next frame (audio, video)
132. **open**: Specifies that the details should be visible (details)
133. **optimum**: Specifies the optimal value of a meter element (meter)
134. **pattern**: Specifies a regular expression that the input element's value is checked against (input)
135. **ping**: Specifies a space-separated list of URLs to which, when the a element is clicked, a ping request should be sent (a)
136. **placeholder**: Specifies a short hint that describes the expected value of an input element (input, textarea)
137. **poster**: Specifies an image to be shown while the video is downloading, or until the user hits the play button (video)
138. **preload**: Specifies if and how the author thinks the audio/video should be loaded when the page loads (audio, video)
139. **readonly**: Specifies that an input field is read-only (input, textarea)
140. **rel**: Specifies the relationship between the current document and the linked document (a, area, link)
141. **required**: Specifies that an input field must be filled out before submitting the form (input, select, textarea)
142. **reversed**: Specifies that the list order should be descending (ol)
143. **rows**: Specifies the number of rows in a textarea (textarea)
144. **rowspan**: Specifies the number of rows a table cell should span (td, th)
145. **sandbox**: Enables an extra set of restrictions for the content in an iframe (iframe)
146. **scope**: Specifies whether a header cell is a header for a column, row, or group of columns or rows (th)
147. **scoped**: Specifies that the styles only apply to this element's parent and its child elements (style)
148. **selected**: Specifies that an option should be pre-selected when the page loads (option)
149. **shape**: Specifies the shape of the area in an image map (area)
150. **size**: Specifies the width of an input element (in characters) (input)
151. **sizes**: Specifies the sizes of the icons for visual media (link, source)
152. **span**: Specifies the number of columns to span an element (col, colgroup)
153. **spellcheck**: Specifies whether the element is to have its spelling and grammar checked or not (global attribute)
154. **src**: Specifies the URL of the media file (audio, embed, iframe, img, input, script, source, track, video)
155. **srcdoc**: Specifies the HTML content of the page to show in the iframe (iframe)
156. **srclang**: Specifies the language of the track text data (track)
157. **srcset**: Specifies the URL of the image to use in different situations (img, source)
158. **start**: Specifies the start value of a list (ol)
159. **step**: Specifies the legal number intervals for an input field (input)
160. **style**: Specifies an inline CSS style for an element (global attribute)
161. **tabindex**: Specifies the tabbing order of an element (global attribute)
162. **target**: Specifies where to open the linked document (a, area, base, form)
163. **title**: Specifies extra information about an element (global attribute)
164. **translate**: Specifies whether the content of an element should be translated or not (global attribute)
165. **type**: Specifies the type of the element (button, input, link, script, source, style)
166. **usemap**: Specifies an image as a client-side image map (img)
167. **value**: Specifies the value of an element (button, input, li, option, meter, progress, param)
168. **width**: Specifies the width of an element (canvas, embed, iframe, img, input, object, video)
169. **wrap**: Specifies how the text in a text area is to be wrapped when submitted in a form (textarea)