

## 1. `<a>` Anchor Element

The `<a>` anchor element is used to create hyperlinks in an HTML document. The hyperlinks can point to other webpages, files on the same server, a location on the same page, or any other URL via the hyperlink reference attribute, `href`. The `href` determines the location the anchor element points to.

```
<!-- Creating text links -->
<a href="http://www.codeillustrated.com">Visit this site</a>

<!-- Creating image links -->
<a href="https://www.codeillustrated.com /academy">
  Click this image
</a>
```

## 2. `<head>` Head Element

The `<head>` element contains general information about an HTML page that isn't displayed on the page itself. This information is called metadata and includes things like the title of the HTML document and links to stylesheets.

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Website</title>
  </head>
</html>
```

## 3. `<html>` HTML Element

The `<html>` element, the root of an HTML document, should be added after the `!DOCTYPE` declaration. All content/structure for an HTML document should be contained between the opening and closing `<html>` tags.

```
<!DOCTYPE html>
<html>

</html>
```

## 4. Whitespace

Whitespace, such as line breaks, added to an HTML document between block-level elements will generally be ignored by the browser and are not added to increase spacing on the rendered HTML page. Rather, whitespace is added for organization and easier reading of the HTML document itself.

```
<p>Test paragraph</p>
```

```
<!-- The whitespace created by this line, and above/below this line is ignored by the browser-->
```

```
<p>Another test paragraph, this will sit right under the first paragraph, no extra space between.</p>
```

## 5. `<title>` Title Element

The `<title>` element contains a text that defines the title of an HTML document. The title is displayed in the browser's title bar or tab in which the HTML page is displayed. The `<title>` element can only be contained inside a document's `<head>` element.

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

## 6. File Path

The following code tells the browser to look for the html file and display the website.

```
<a href="/about.html">Open the about page</a>
```

## 7. HTML

HTML (HyperText Markup Language) is used to give content to a web page and instructs web browsers on how to structure that content

## 8. Element Content

The content of an HTML element is the information between the opening and closing tags of an element.

```
<h1>Code Illustrated is awesome! 😊</h1>
```

## 9. `<li>` List Item Element

The `<li>` list item element create list items inside:

- Ordered lists `<ol>`
- Unordered lists `<ul>`

```
<ol>  
<li>Head east on Prince St</li>  
<li>Turn left on Elizabeth</li>  
</ol>
```

```
<ul>  
<li>Cookies</li>  
<li>Milk</li>  
</ul>
```

## 10. `<video>` Video Element

The `<video>` element embeds a media player for video playback. The `src` attribute will contain the URL to the video. Adding the `controls` attribute will display video controls in the media player.

```
<video src="test-video.mp4" controls>
```

```
</video>
```

## 11. `<em>` Emphasis Element

The `<em>` element emphasizes text and browsers will usually *italicize* the emphasized text by default.

```
<p>This <em>word</em> will be emphasized in italics.</p>
```

## 12. `<ol>` Ordered List Element

The `<ol>` ordered list element creates a list of items in sequential order. Each list item appears numbered by default.

```
<ol>  
  <li>Preheat oven to 325 F 🧺</li>  
  <li>Drop cookie dough 🍪</li>  
  <li>Bake for 15 min 🕒</li>  
</ol>
```

## 13. `<div>` Div Element

The `<div>` element is used as a container that divides an HTML document into sections and is short for “division”. `<div>` elements can contain *flow content* such as headings, paragraphs, links, images, etc.

```
<div>
  <h1>A section of grouped elements</h1>
  <p>Here's some text for the section</p>
</div>
<div>
  <h1>Second section of grouped elements</h1>
  <p>Here's some text</p>
</div>
```

## 14. HTML Structure

HTML is organized into a family tree structure. HTML elements can have parents, grandparents, siblings, children, grandchildren, etc.

```
<body>
  <div>
    <h1>It's div's child and body's grandchild</h1>
    <h2>It's h1's sibling</h2>
  </div>
</body>
```

## Closing Tag

An HTML closing tag is used to denote the end of an HTML element. The syntax for a closing tag is a left angle bracket `<` followed by a forward slash `/` then the element name and a right angle bracket to close `>`

```
<body>
  ...
</body>
```

## 15. Attribute Name and Values

HTML attributes consist of a name and a value using the following syntax: `name="value"` and can be added to the opening tag of an HTML element to configure or change the behavior of the element.

```
<elementName name="value"></elementName>
```

## 16. `<br>` Line Break Element

The `<br>` line break element will create a line break in text (also between elements) and is especially useful where a division of text is required, like in a postal address. The line break element requires only an opening tag and must not have a closing tag.

```
A line break haiku.<br>
Poems are a great use case.<br>
Oh joy! A line break.
```

## 17. `<img>` Image Element

HTML image `<img>` elements embed images in documents. The `src` attribute contains the image URL and is mandatory. `<img>` is an *empty element* meaning it should not have a closing tag.

```

```

## 18. `<h1>`-`<h6>` Heading Elements

HTML can use six different levels of heading elements. The heading elements are ordered from the highest level `<h1>` to the lowest level `<h6>`.

```
<h1>Breaking News</h1>  
<h2>This is the 1st subheading</h2>  
<h3>This is the 2nd subheading</h3>  
...  
<h6>This is the 5th subheading</h6>
```

## 19. `<ul>` Unordered List Element

The `<ul>` unordered list element is used to create a list of items in no particular order. Each individual list item will have a bullet point by default.

```
<ul>  
  <li>Play more music 🎸 </li>  
  <li>Read more books 📖 </li>  
</ul>
```

## 20. `alt` Attribute

An `<img>` element can have alternative text via the `alt` attribute. The alternative text will be displayed if an image fails to render due to an incorrect URL, if the image format is not supported by the browser, if the image is blocked from being displayed, or if the image has not been received from the URL.

The text will be read aloud if screen reading software is used and helps support visually impaired users by providing a text descriptor for the image content on a webpage.

```

```

## 21. `<body>` Body Element

The `<body>` element represents the content of an HTML document. Content inside `<body>` tags are rendered on the web browsers.

**Note:** There can be only one `<body>` element in a document.

## 22. `<span>` Span Element

The `<span>` element is an inline container for text and can be used to group text for styling purposes. However, as `<span>` is a generic container to separate pieces of text from a larger body of text, its use should be avoided if a more semantic element is available

```
<p><span>This text</span> may be styled differently than the surrounding text.</p>
```

## 23. `<strong>` Strong Element

The `<strong>` element highlights important, serious, or urgent text and browsers will normally render this highlighted text in **bold** by default.

```
<p>This is <strong>important</strong> text!</p>
```

## 24. HTML Element

An HTML element is a piece of content in an HTML document and uses the following syntax: opening tag + content + closing tag. In the code provided:

- `<p>` is the opening tag.
- `Hello World!` is the content.
- `</p>` is the closing tag.

```
<p>Hello World!</p>
```

## 25. HTML Tag

The syntax for a single HTML tag is an opening angle bracket `<` followed by the element name and a closing angle bracket `>`. Here is an example of an opening `<div>` tag.

```
<div>
```

## 26. `<tr>` Table Row Element

The table row element, `<tr>`, is used to add rows to a table before adding table data and table headings.

```
<table>  
  <tr>  
    ...  
  </tr>  
</table>
```

## 27. `<td>` Table Data Element

The table data element, `<td>`, can be nested inside a table row element, `<tr>`, to add a cell of data to a table.

```
<table>
  <tr>
    <td>cell one data</td>
    <td>cell two data</td>
  </tr>
</table>
```

## 28. `<th>` Table Heading Element

The table heading element, `<th>`, is used to add titles to rows and columns of a table and must be enclosed in a table row element, `<tr>`.

```
<table>
  <tr>
    <th>column one</th>
    <th>column two</th>
  </tr>
  <tr>
    <td>1</td>
    <td>2</td>
  </tr>
</table>
```

## 29. <table> Table Element

In HTML, the <table> element has content that is used to represent a two-dimensional table made of rows and columns

```
<table>
  <!-- rows and columns will go here -->
</table>
```

## 30. <form> Element

The HTML <form> element is used to collect and send information to an external source.

<form> can contain various input elements. When a user submits the form, information in these input elements is passed to the source which is named in the `action` attribute of the form.

```
<form>
  Enter your name:
  <input type="text" name="fname">
  <br/>
  Enter your age:
  <input type="text" name="age">
  <br/>
  <input type="submit" value="Submit">
</form>
```

## 31. <input> Element

The HTML <input> element is used to render a variety of input fields on a webpage including text fields, checkboxes, buttons, etc. <input> element have a `type` attribute that determines how it gets rendered to a page.

The example code block will create a text input field and a checkbox input field on a webpage.

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

```
<input type="checkbox" name="vehicle" value="Bike"> I own a bike
```

## 32. <label> Element

The HTML <label> element provides identification for a specific <input> based on matching values of the <input>'s id attribute and the <label>'s for attribute. By default, clicking on the <label> will focus the field of the related <input>.

The example code will create a text input field with the label text "Password:" next to it. Clicking on "Password:" on the page will focus the field for the related <input>.

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

## 33. <input>: Checkbox Type

When using an HTML input element, the type="checkbox" attribute will render a single checkbox item. To create a group of checkboxes related to the same topic, they should all use the same name attribute. Since it's a checkbox, multiple checkboxes can be selected for the same topic.

```
<input type="checkbox" name="breakfast" value="bacon">Bacon 🍳<br>  
<input type="checkbox" name="breakfast" value="eggs">Eggs 🍳<br>  
<input type="checkbox" name="breakfast" value="pancakes">Pancakes 🥞<br>
```

## 34. `<input>`: Radio Button Type

HTML `<input>` elements can be given a `type="radio"` attribute that renders a single radio button. Multiple radio buttons of a related topic are given the same `name` attribute value. Only a single option can be chosen from a group of radio buttons.

The value of the selected/checked `<input>`'s `name` and `value` attribute of this element are sent as a key-value pair when the form is submitted.

```
<input name="delivery_option" type="radio" value="pickup" />  
<input name="delivery_option" type="radio" value="delivery" />
```

## 35. `<textarea>` Element

The `textarea` element is used when creating a text-box for multi-line input (e.g. a comment section). The element supports the `rows` and `cols` attributes which determine the height and width, respectively, of the element.

When rendered by the browser, `textarea` fields can be stretched/shrunk in size by the user, but the `rows` and `cols` attributes determine the initial size.

Unlike the `input` element, the `<textarea>` element has both opening and closing tags. The `value` of the element is the content in between these tags (much like a `<p>` element). The code block shows a `<textarea>` of size 10x30 and with a name of "comment".

```
<textarea rows="10" cols="30" name="comment"></textarea>
```