Attributes provide additional information about HTML elements.

- **alt** Specifies an alternative text for an image, when the image cannot be displayed
- **disabled** Specifies that an input element should be disabled
- **href** Specifies the URL (web address) for a link
- **id** Specifies a unique id for an element
- **src** Specifies the URL (web address) for an image
- **style** Specifies an inline CSS style for an element
- **title** Specifies extra information about an element (displayed as a tool tip)
- **Headings are defined with the <h1> to <h6> tags.**

<tagname style="property:value;">

<body style="background-color:powderblue;">

<b>	Defines bold text
	Defines emphasized text
<i>	Defines italic text
<small>	Defines smaller text
	Defines important text
<sub>	Defines subscripted text
<sup>	Defines superscripted text
<ins>	Defines inserted text
<del>	Defines deleted text
<mark>	Defines marked/highlighted text

- **A link does not have to be text. It can be an image or any other HTML element.**

```
<img src="pic_trulli.jpg" alt="Italian Trulli">
```

```
<img src="/images/html5.gif" alt="HTML5 Icon"
style="width:128px;height:128px;">
```

- ## Image as a Link

To use an image as a link, put the <img> tag inside the <a> tag

- **Float**

```
<p><img src="smiley.gif" alt="Smiley face"
style="float:right;width:42px;height:42px;">
The image will float to the right of the text.</p>
```

- **Background Image**

```
<body style="background-image:url('clouds.jpg')">
```

- # HTML Table - Cells that Span Many Rows

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

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

Unordered HTML List

An unordered list starts with the <ul> tag. Each list item starts with the <li> tag.

```
<ul style="list-style-type:disc">
<ol type="1">
```

disc	Sets the list item marker to a bullet (default)
circle	Sets the list item marker to a circle

square	Sets the list item marker to a square
none	The list items will not be marked

Ordered HTML List - The Type Attribute

The `type` attribute of the `<ol>` tag, defines the type of the list item marker:

type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters
type="a"	The list items will be numbered with lowercase letters
type="I"	The list items will be numbered with uppercase roman numbers
type="i"	The list items will be numbered with lowercase roman numbers

```
<ol type="1">
```

HTML Block and Inline Elements

Block-level Elements

A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can).

The element is a block-level element.

<address><article><aside><blockquote>

<canvas><dd> <dl><dt><fieldset>

<figcaption><figure><footer><form><h1>-<h6>

<header><hr>
main><nav><noscript>

<output><p><pre><section><table><tfoot>

<video>

Inline Elements

An inline element does not start on a new line and only takes up as much width as necessary.

This is an inline element inside a paragraph.

<a><abbr><acronym><b><bdo><big>

<button><cite><code>

<dfn><i><label><map>
<img><input><kbd><object>
<q><samp><script><select><small>
<sub><sup><textarea><time><tt><var>

	Defines a section in a document (block-level)
****	**Defines a section in a document (inline)**

CLASS:

HTML elements can have more than one class name, each class name must be separated by a space.

ID:-

The id attribute specifies a unique id for an HTML element (the value must be unique within the HTML document).
write a hash (#) character
An HTML element can only have one unique id that belongs to that single element, while a class name can be used by multiple elements

HTML Iframes

An iframe is used to display a web page within a web page.

```
<iframe src="URL"></iframe>
```

Iframe - Target for a Link

An iframe can be used as the target frame for a link

<iframe>	Defines an inline frame

JavaScript can change HTML content

```
document.getElementById("demo").innerHTML = "Hello JavaScript!";
```

avaScript can change HTML styles

```
document.getElementById("demo").style.fontSize = "25px";
document.getElementById("demo").style.color = "red";
document.getElementById("demo").style.backgroundColor = "yellow";
```

JavaScript can change HTML attributes

```
document.getElementById("image").src = "picture.gif";
```

The <noscript> tag is used to provide an alternate content for users that have disabled scripts in their browser or have a browser that doesn't support client-side scripts:

```
<noscript>Sorry, your browser does not support JavaScript!</noscript>
```

<img src="/images/picture.jpg">	picture.jpg is located in the images folder at the root of the current web
<img src="../picture.jpg">	picture.jpg is located in the folder one level up from the current folder

The <head> element is a container for metadata (data about data) and is placed between the <html> tag and the <body> tag.

EX META

```
<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">
```

VIEWPORT:

```
<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).

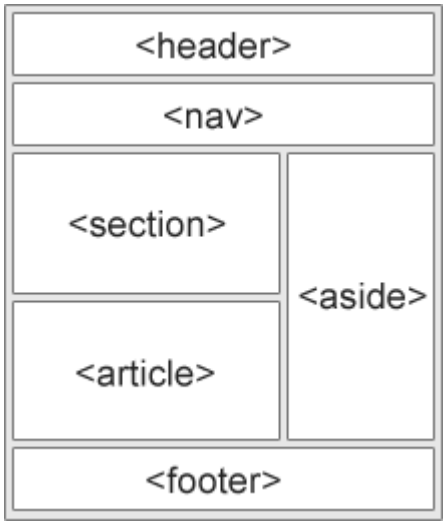
The initial-scale=1.0 part sets the initial zoom level when the page is first loaded by the browser.

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

HTML Layout Elements

Websites often display content in multiple columns (like a magazine or newspaper).

HTML5 offers new semantic elements that define the different parts of a web page:



- <header> - Defines a header for a document or a section
- <nav> - Defines a container for navigation links
- <section> - Defines a section in a document
- <article> - Defines an independent self-contained article
- <aside> - Defines content aside from the content (like a sidebar)
- <footer> - Defines a footer for a document or a section
- <details> - Defines additional details
- <summary> - Defines a heading for the <details> element

What is Responsive Web Design?

Responsive Web Design is about using HTML and CSS to automatically resize, hide, shrink, or enlarge, a website, to make it look good on all devices (desktops, tablets, and phones):

Media Queries

In addition to resize text and images, it is also common to use media queries in responsive web pages.

With media queries you can define completely different styles for different browser sizes.

Example: resize the browser window to see that the three div elements below will display horizontally on large screens and stacked vertically on small screens

```
/* Use a media query to add a breakpoint at 800px: */
@media screen and (max-width: 800px) {
  .left, .main, .right {
    width: 100%; /* The width is 100%, when the viewport is 800px or
smaller */
  }
```

Responsive Web Design - Frameworks

There are many existing CSS Frameworks that offer Responsive Design.

W3.css
Using bootstrap

Tag	Description
<code>	Defines programming code
<kbd>	Defines keyboard input
<samp>	Defines computer output
<var>	Defines a variable
<pre>	Defines preformatted text

Some Greek Letters Supported by HTML

HTML Encoding (Character Sets)

ASCII was the first **character encoding standard** (also called character set). ASCII defined 128 different alphanumeric characters that could be used on the internet: numbers (0-9), English letters (A-Z), and some special characters like ! $ + - ( ) @ < > .

UTF-8 (Unicode) covers almost all of the characters and symbols in the world.

The UTF-8 Character Set

UTF-8 is identical to ASCII for the values from 0 to 127.

UTF-8 does not use the values from 128 to 159.

UTF-8 is identical to both ANSI and 8859-1 for the values from 160 to 255.

UTF-8 continues from the value 256 with more than 10 000 different characters.

For a closer look, study our Complete HTML Character Set Reference.

The @charset CSS Rule

You can use the CSS `@charset` rule to specify the character encoding used in a style sheet:

```
@charset "UTF-8";
```

Form

<input type="text">	Defines a one-line text input field
<input type="radio">	Defines a radio button (for selecting one of many choices)
<input type="submit">	Defines a submit button (for submitting the form)

When to Use GET?

The default method when submitting form data is GET.

However, when GET is used, the submitted form data will be **visible in the page address field**:

```
/action_page.php?firstname=Mickey&lastname=Mouse
```

Always use POST if the form data contains sensitive or personal information. The POST method does not display the submitted form data in the page address field.

```
<form action="/action_page.php">
  <fieldset>
    <legend>Personal information:</legend>
    First name:<br>
    <input type="text" name="firstname" value="Mickey"><br>
    Last name:<br>
    <input type="text" name="lastname" value="Mouse"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```

Allow Multiple Selections:

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

Example

```
<select name="cars" size="4" multiple>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="fiat">Fiat</option>
  <option value="audi">Audi</option>
</select>
```

HTML5 Input Types

HTML5 added several new input types:

- color
- date
- datetime-local
- email
- month
- number
- range
- search
- tel
- time
- url
- week

Input Type Search

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

Example

```
<form>
  Search Google:
  <input type="search" name="googlesearch">
</form>
```

```
HTML5....
```

The most interesting new HTML5 elements are:

New **semantic elements** like `<header>`, `<footer>`, `<article>`, and `<section>`.

New **attributes of form elements** like number, date, time, calendar, and range.

New **graphic elements**: `<svg>` and `<canvas>`.

New **multimedia elements**: `<audio>` and `<video>`.

New HTML5 API's (Application Programming Interfaces)

The most interesting new API's in HTML5 are:

- HTML Geolocation
- HTML Drag and Drop
- HTML Local Storage
- HTML Application Cache
- HTML Web Workers
- HTML SSE

A semantic element clearly describes its meaning to both the browser and the developer.

Examples of **non-semantic** elements: `<div>` and `<span>` - Tells nothing about its content.

Examples of **semantic** elements: `<form>`, `<table>`, and `<article>` - Clearly defines its content.

- <article>
- <aside>
- <details>
- <figcaption>
- <figure>
- <footer>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <summary>
- <time>

HTML5 <figure> and <figcaption> Elements

The purpose of a figure caption is to add a visual explanation to an image.

MIGRATION HTML4 TO 5

Change to HTML5 Doctype

Change to HTML5 Encoding

HTML5 GRAPHICS

The HTML <canvas> element is used to draw graphics on a web page.
You must use JavaScript to actually draw the graphics.

What is SVG?

- SVG stands for Scalable Vector Graphics
- SVG is used to define graphics for the Web

SVG is a language for describing 2D graphics in XML.

Canvas draws 2D graphics, on the fly (with a JavaScript).

Multimedia on the web is sound, music, videos, movies, and animations.
<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>

HTML Helpers (Plug-ins)

Helper applications (plug-ins) are computer programs that extend the standard functionality of a web browser.

Ex

HTML5 Geolocation

The HTML Geolocation API is used to locate a user's position.