HTML:

The full meaning of HTML is Hyper Text Markup Language.

Hypertext:

The term is invented by Ted Nelson in 1965. It is a machine readable language. Hypertext is text displayed on a [computer display](https://en.wikipedia.org/wiki/Computer_display) or other [electronic devices](https://en.wikipedia.org/wiki/Electronic_devices) with references or [hyperlinks](https://en.wikipedia.org/wiki/Hyperlinks) to other text that the reader can immediately access. Hypertext documents are interconnected by hyperlinks, which are typically activated by a [mouse](https://en.wikipedia.org/wiki/Mouse_(computing)) click, key press set or by touching the screen.

Markup Language:

Markup languages are designed for the processing, definition and presentation of [text](https://www.webopedia.com/TERM/T/text.html). The [language](https://www.webopedia.com/TERM/L/language.html) specifies [code](https://www.webopedia.com/TERM/C/code.html) for formatting, both the layout and style, within a [text file](https://www.webopedia.com/TERM/T/text_file.html). The code used to specify the formatting is called [tags](https://www.webopedia.com/TERM/T/tag.html). [HTML](https://www.webopedia.com/TERM/H/HTML.html) is an example of a widely known and used markup language.

Semantic Element:

A semantic element clearly describes its meaning to both the browser and the developer. Semantic elements are given below:

* <a>
* <article>
* <aside>
* <address>
* <audio>
* <b>
* <blockquote>
* <cite>
* <details>
* <form>
* <figcaption>
* <figure>
* <footer>
* <hr>
* <header>
* <h1> to <h6>
* <i>
* <img>
* <li>
* <main>
* <mark>
* <nav>
* <ol>
* <p>
* <strong>
* <section>
* <summary>
* <time>
* <table>
* <ul>
* <video>

Block Level Element:

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.

Examples:

* <div>
* <nav>
* <section>
* <p>
* <table>
* <form>

Inline Level Element:

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

Examples:

* <a>
* <button>
* <img>
* <i>
* <input>
* <strong>

CSS Selector:

Basic selectors

|  |  |  |
| --- | --- | --- |
| Selector | Example | Example description |
| class | Intro | Selects the element with class="intro" |
| id | First name | Selects the element with id="firstname" |
| **\*** | **\*** | Selects the elements |
| element | p | Selects all <p> elements |
| element>p | div>p | Selects all <p> elements where the parent is a div element |
| element+p | div+p | Selects all <p> elements that are placed immediately after <div> elements |

Attribute Selector

|  |  |  |
| --- | --- | --- |
| Selector | Description | Example |
| [attribute] | Matches elements **containing a given attribute.** | a[href] {color: red;} |
| [attribute\*="x"] | Matches elements containing a given attribute with a value that **contains** something. | a[href\*="htmldog"] { color: red; } |
| [attribute~="x"] | Matches elements containing a given attribute with a value that contains a sub-value within a **space-separated list.** | abbr[title~="Style"] {color: red;} |
| [attribute="x"] | Matches elements **containing a given attribute with a given value.** | a[href="/sitemap/"] {color: red;} |
| [attribute|="x"] | Matches elements containing a given attribute with a value that contains a sub-value within a **hyphen-separated list.** | html[lang|="en"] {color: red;} |
| [attribute^="x"] | Matches elements containing a given attribute with a value that **starts** with something. | a[href^="http://"] {color: red;} |
| [attribute$="x"] | Matches elements containing a given attribute with a value that **ends** with something. | a[href$=".com"] {color: red;} |

Pseudo-classes

|  |  |  |
| --- | --- | --- |
| Selector | Description | Example |
| :link | Matches a **link that has not been visited.** | a:link { color: blue;} |
| :hover | Matches an element whose box is being **hovered over** by a cursor. | a:hover { text-decoration: none; } |
| :focus | Matches an element that has **focus**, such as one that has been tabbed to. | a:focus { border: 1px solid yellow; } |
| :first-child | Matches the **first child** of an element. | p:first-child { color: red; } |
| :last-child | Matches the **last child** of an element. | div p:last-child { color: blue; } |
| :nth-child | Matches an element that is the **ordinal number child** of its parent. | p:nth-child(3) { color: red; } |

Pseudo-elements

|  |  |  |
| --- | --- | --- |
| Selector | Description | Example |
| ::before | Used with the [content](http://htmldog.com/references/css/properties/content/) property to generate content **before** the initial content of an element. | h1::before { content: "\*"; } |
| ::after | Used with the [content](http://htmldog.com/references/css/properties/content/) property to generate content **after** the initial content of an element. | h1::after { content: "+"; } |
| ::first-line | Matches the **first textual line** in an element. | p::first-line { font-weight: bold; } |
| ::first-letter | Matches the **first letter** in an element. | p::first-letter { font-size: 2em; } |

Combinatiors

|  |  |  |
| --- | --- | --- |
| Selector | Description | Example |
| selector selector | **Descendant** combinator. Matches elements that are descendants of another element. | aside p { color: red; } |
| selector > selector | **Child** combinator. Matches elements that are children of another element | .warning > p { color: red; } |
| selector + selector | **Adjacent sibling** combinator. Matches elements that immediately follow another element. | h1 + \* { color: red; } |
| selector ~ selector | **General sibling** combinator. Matches elements that follow another element. | h2 ~ p { color: red; } |