

# Assignment

## HTML:

HTML stands for Hyper Text Markup Language.

## Hypertext:

Hypertext is a text which contains links that can be linked to another document or text. Hypertext covers both textual hyperlinks and graphical ones. The term was coined by Ted Nelson around 1965.

## Markup language:

A markup language is a computer language that uses tags for the processing, definition and presentation of text.

HTML is a machine- readable data.

## Semantic Element:

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

## Semantic elements name:

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

- <address>
- <audio>
- <video>
- <img>
- <h1> to <h6>
- <strong>
- <mark>
- <cite>
- <blockquote>
- <a>
- <p>
- <hr>

## 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).

Example:

```
<div>Hello</div>
```

```
<h1>Hello</h1>
```

`<p>Hello</p>`

## Inline level element:

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

Example:

`<span>Hello</span>`

``

`<a href="#">FB</a>`

## CSS Selector:

### Basic Selector

selector	Description	Example
element	<b>Type</b> selector. Matches an element.	<code>p{color: red;}</code> <code>a{text-decoration: none;}</code>

.class	<b>Class</b> selector. Matches the value of a class attribute	.header{background-color: red;}
#id	<b>ID</b> selector. Matches the value of an id attribute.	#id{color:#fff;}
*	<b>Universal</b> selector. Matches everything.	*{margin:0; padding:0}

## Attribute Selector

Selector	Description	Example
[attribute]	Matches elements <b>containing a given attribute.</b>	a[href] {color: red;}
[attribute="x"]	Matches elements <b>containing a given attribute with a given value.</b>	a[href="/sitemap/"] {color: red;}
[attribute~="x"]	Matches elements containing a given attribute with a value that contains a sub-value within a <b>space-separated list.</b>	abbr[title~="Style"] {color: red;}
[attribute ="x"]	Matches elements containing a given attribute with a value that contains a sub-value within a <b>hyphen-separated list.</b>	html[lang ="en"] {color: red;}
[attribute^="x"]	Matches elements containing a given attribute	a[href^="http://"] {color: red;}

	with a value that <b>starts</b> with something.	
[attribute\$="x"]	Matches elements containing a given attribute with a value that <b>ends</b> with something.	a[href\$=".com"] {color: red;}
[attribute*="x"]	Matches elements containing a given attribute with a <b>value</b> that contains something.	a[href*="htmlldog"] { color: red; }

## Pseudo-classes

Selector	Description	Example
:link	Matches a <b>link that has not been visited</b> .	a:link { color: blue;}
:visited	Matches a <b>link that has been visited</b> .	div p:last-child { color: blue; }
:hover	Matches an element whose box is being <b>hovered over</b> by a cursor.	p:nth-child(3) { color: red; }
:focus	Matches an element that has <b>focus</b> , such as one that has been tabbed to.	a:focus { border: 1px solid yellow; }
:first-child	Matches the <b>first child</b> of an element.	p:first-child { color: red; }
:last-child	Matches the <b>last child</b> of an element.	div p:last-child { color: blue; }

:nth-child()	Matches an element that is the <b>ordinal number child</b> of its parent.	p:nth-child(3) { color: red; }
--------------	---	--------------------------------

## Pseudo-elements

Selector	Description	Example
::first-line	Matches the <b>first textual line</b> in an element.	p::first-line { font-weight: bold; }
::first-letter	Matches the <b>first letter</b> in an element.	p::first-letter { font-size: 2em; }
::before	Used with the content property to generate content <b>before</b> the initial content of an element.	h1::before { content: "*"; }
::after	Used with the content property to generate content <b>after</b> the initial content of an element.	h1::after { content: "+"; }

## Combinators

Selector	Description	Example
selector selector	<b>Descendant</b> combinator. Matches elements that are descendants of another element.	aside p { color: red; }
selector > selector	<b>Child</b> combinator. Matches elements that	.warning > p { color: red; }

	are children of another element	
selector + selector	<b>Adjacent sibling</b> combinator. Matches elements that immediately follow another element.	h1 + * { color: red; }
selector ~ selector	<b>General sibling</b> combinator. Matches elements that follow another element.	h2 ~ p { color: red; }

**END**