

Getting started with HTML and CSS

Margit Tennosaar

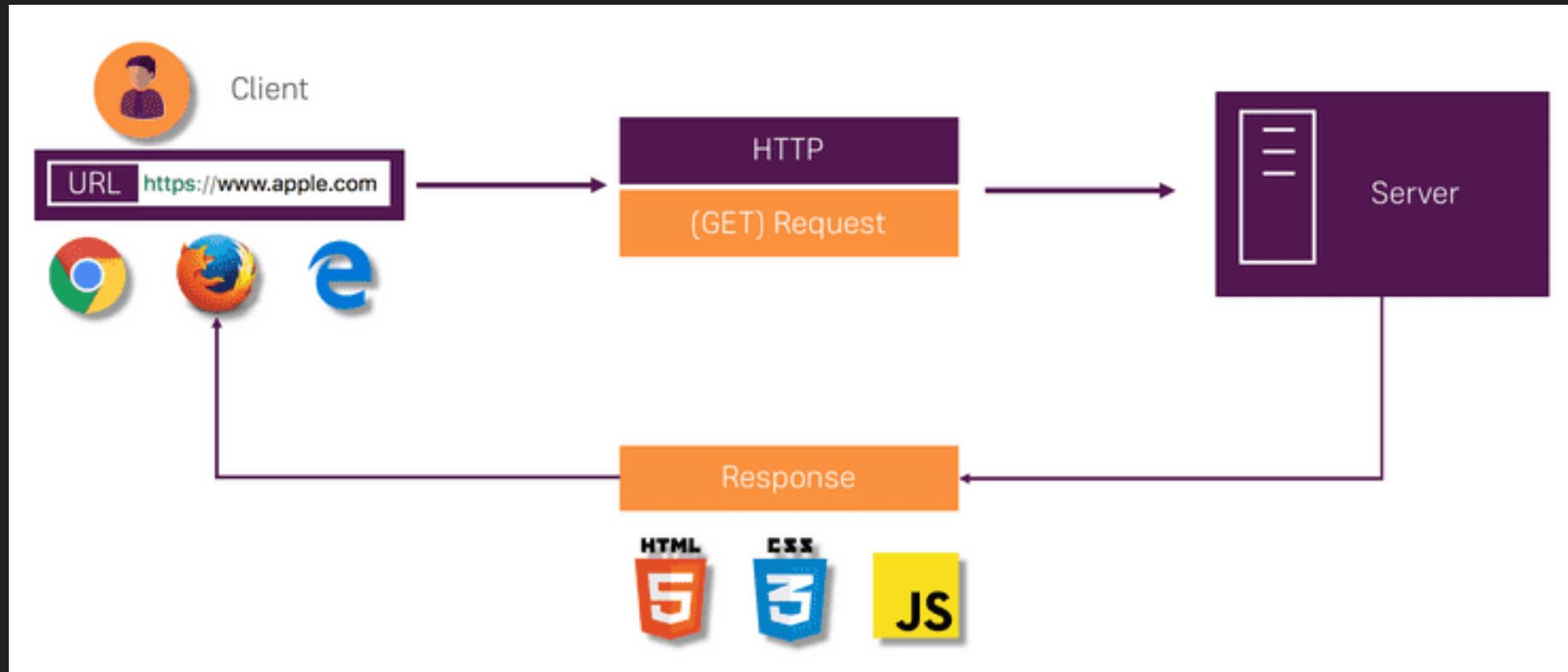
In this session

- Setting up your project folder structure.
- Understanding the basic HTML document structure.
- Practicing typography, lists, links, and images.
- Including CSS and JavaScript files.
- Testing and reflecting on your project.

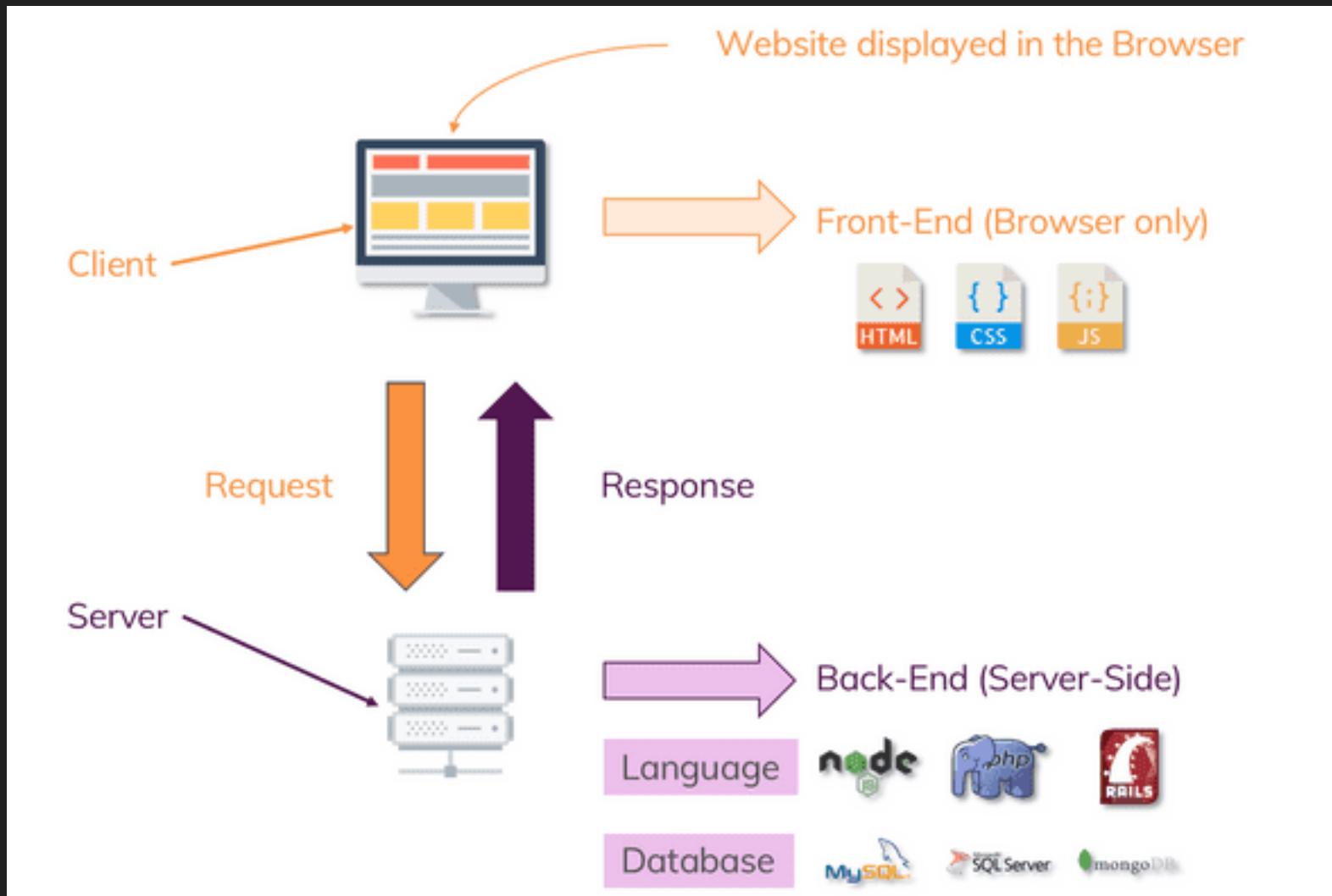
How web works?

https://github.com/margittennosaar/HTML-CSS.REACT25K/blob/main/01_week/how_web_works.md

How web works?



Front-End and Back-End



The Client-Server model:

- Client: Your browser requesting a web page.
- Server: A computer sending the requested files.

HTTP/HTTPS:

- HTTP: Communication protocol.
- HTTPS: Secure version with encryption.

DNS (Domain Name System):

- Converts URLs to IP addresses.

HTML

https://github.com/margittensoar/HTML-CSS.REACT25K/blob/main/01_week/html_base.md

HTML

HyperText Markup Language



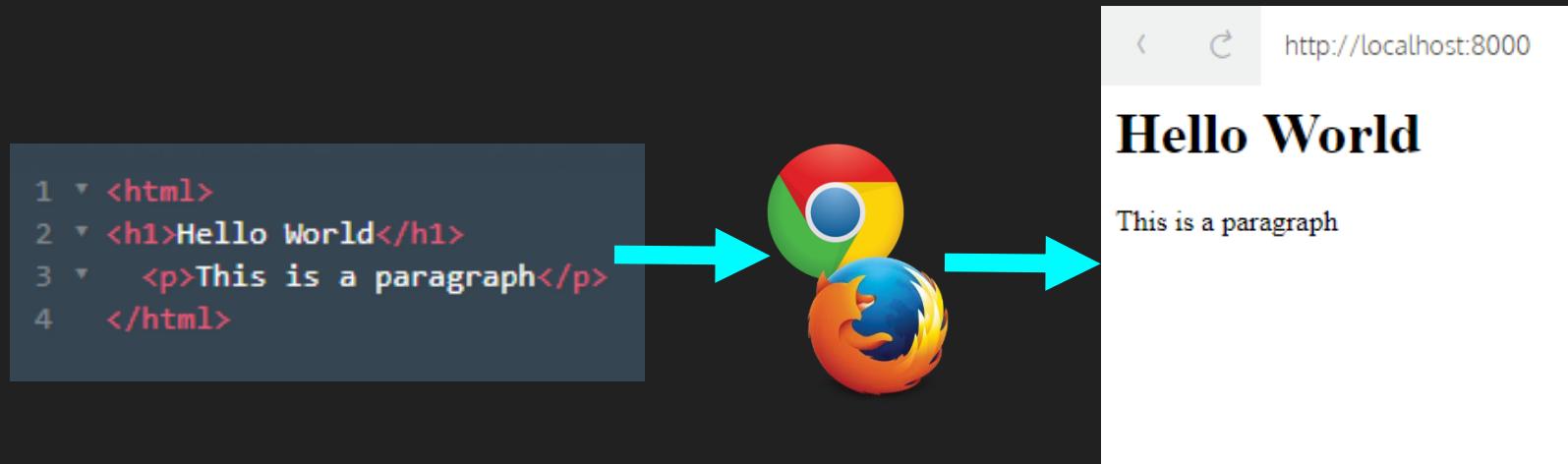
Markup language! = computer language that defines the structure and presentation of raw text. Markup surrounds the raw text with information the computer can interpret, "marking it up" to be processed.

Tags represent HTML elements. Semantic HTML elements names describe what that element does "heading", "paragraph", "table", and so on.

Browsers do not display the HTML tags but use them to render the page's content.

HTML

Browser reads the HTML file and transforms it into an interactive document



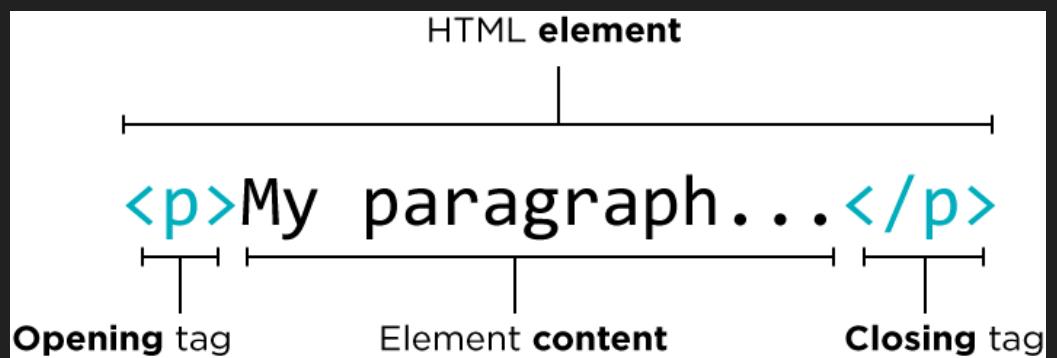
How it works?

The browser can interpret the text that is wrapped in HTML elements. These can be nested inside one another, each containing information about the type and structure of information displayed in the browser.

HTML tags usually come in pairs like `<p>` and `</p>`.

The first tag in a pair is the start tag, and the second tag is the end tag.

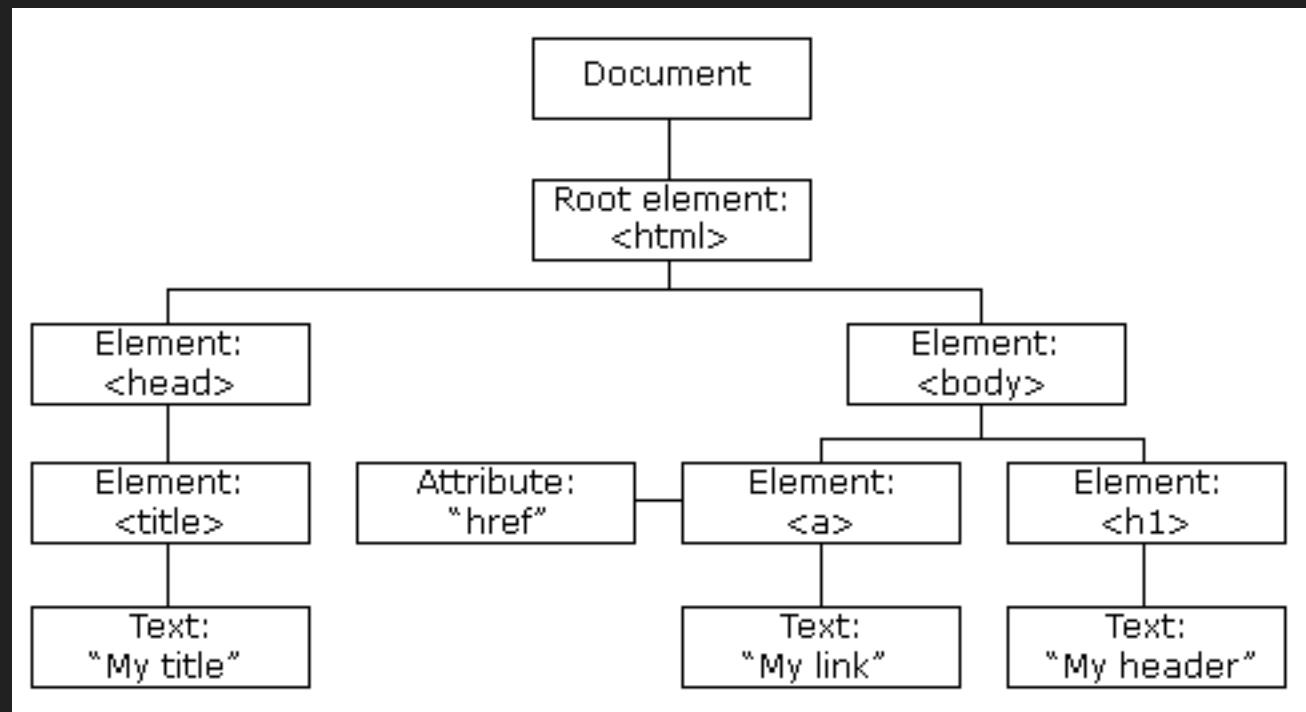
Some HTML tags **close themselves**.



HTML DOM

HTML Document Object Model –

HTML page is constructed as a tree
of Objects



Learn by doing

- Launch and **set up VSc**
 - Prettier
 - Material Icon Theme
 - Live Server
 - Emmet <https://docs.emmet.io/cheat-sheet/>
 - Default browser

Step 1

- Create a course folder named html_course.
- Inside html_course, create a folder named 01_week.
- Inside 01_week, create subfolders:
 - css
 - js
 - images
- Create an index.html file inside 01_week.

HTML document structure

`<!DOCTYPE html>`: Defines document type.

`<html>`: Root element.

`<head>`: Metadata (title, CSS links, etc.).

`<body>`: Visible content.

```
<!DOCTYPE html>
<html>
<head>
  <title>Example</title>
</head>
<body>
  <h1>Hello, World!</h1>
</body>
</html>
```

Semantic HTML

Tags that describe content meaning

<header>, <nav>, <main>, <section>, <footer>

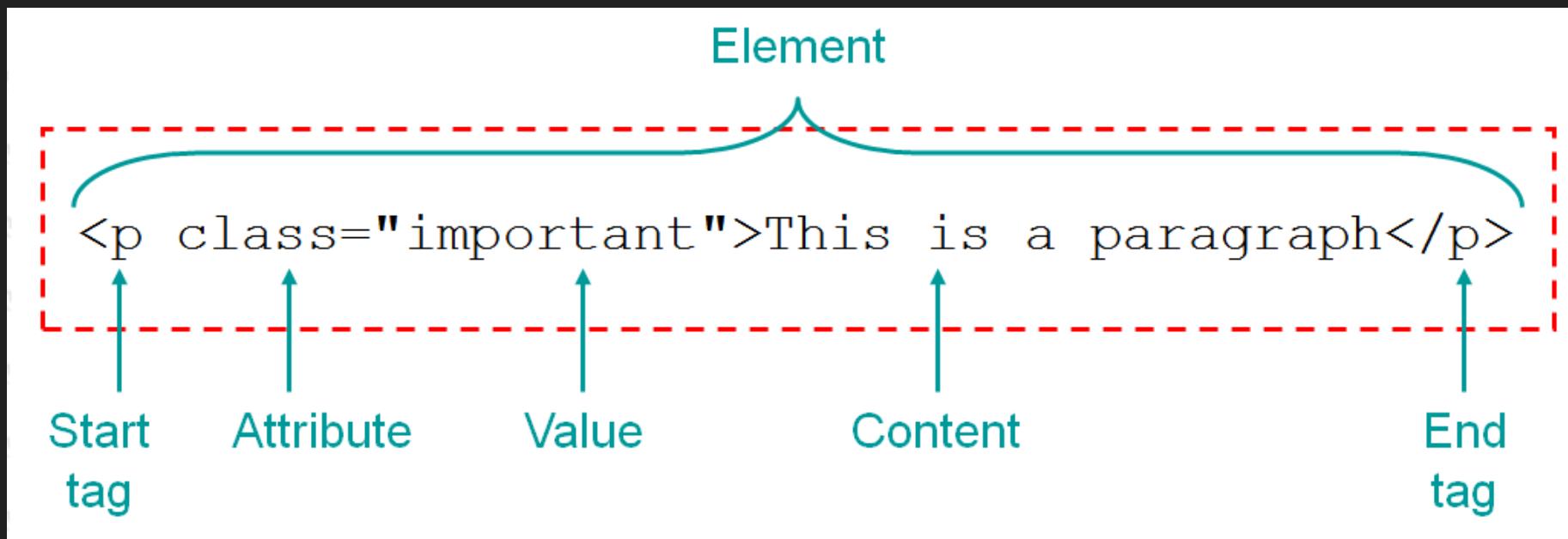
Improves accessibility

Easier for search engines to understand

HTML Attributes

Attributes provide **additional information about an element**

name="value"



Step 2

- Add the basic structure to index.html with a title, <head>, and <body> sections.
- Write it yourself first and then check it with Emmet

HTML Typography

https://github.com/margittennosaar/HTML-CSS.REACT25K/blob/main/01_week/html_typography.md

HTML typography

Headings

<h1>Heading 1</h1> should be used for main headings (**only once on page**)

<h2>Heading 2</h2> then second level headings

<h3>Heading 3</h3> following third level

<h4>Heading 4</h4> and so on....

<h5>Heading 5</h5>

<h6>Heading 6</h6>

HTML typography

`<p>This is a paragraph.</p>`

`` - Important text

`` - Emphasized text

`<mark>` - Marked text

`<small>` - Small text

`` - Deleted text

`<ins>` - Inserted text

`<sub>` - Subscript text

`<sup>` - Superscript text

`<pre>` - pre-formatted text

`` - Bold text (don't use it, instead use CSS)

`<i>` - Italic text (don't use it, instead use CSS)

HTML Quotations

<abbr title=""> - Defines an abbreviation or acronym

<address> - Defines contact information for the author/owner of a document

<bdo dir="rtl"> - Defines the text direction

<blockquote> - Defines a section that is quoted from another source

<cite> - Defines the title of a work

<q> - Defines a short inline quotation

HTML breaks

**
 line break**

<hr> horizontal rule

Task 3

- Add a paragraph below the heading.
- **Bold Text:** Use **** and ****.
- **Italic Text:** Use **<i>** and ****.
- **Underline Text:** Use **<u>**.
- **Superscript and Subscript:** Add a scientific formula (e.g., $E=mc^2$).
- **Quotation:** Add a short inline quote using **<q>** and a blockquote using **<blockquote>**.

Block and inline elements

https://github.com/margittennosaar/HTML-CSS.REACT25K/blob/main/01_week/html_block_inline.md

Block and inline elements



INLINE ELEMENTS FLOW WITH TEXT

PELLentesque habitant morbi tristique senectus
et netus et malesuada fames ac turpis egestas.
Vestibulum **INLINE ELEMENT** vitae, ultricies
eget, tempor sit amet, ante. Donec eu libero sit
amet quam egestas semper. Aenean ultricies mi
vitae est. Mauris placerat eleifend leo.

HTML Block Elements

```
<address><article><aside><blockquote><canvas><dd><div><dl><dt><fieldset>
    ><figcaption><figure><footer>
<form><h1><h6><header><hr><li><main><nav>
<noscript><ol><p><pre><section><table><tfoot><ul><video>
```

HTML **Inline** Elements

```
<a><abbr><acronym><b><bdo><big><br><button>
    <cite><code><dfn><em><i><img><input><kbd>
    <label><map><object><output><q><samp><script>
<select><small><span><strong><sub><sup><textarea><time><tt><var>
```

But what if I want to use something else...

<div> - Defines a section in a document (block-level)

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

HTML links

https://github.com/margittennosaar/HTML-CSS.REACT25K/blob/main/01_week/html_links.md

HTML links

Anchors/links connect two different documents or locations.

Usually, you can recognise it by hovering the link, and the mouse arrow turns to the cursor.

```
<a href="url" target="“_blank”>link text</a>
```

HTML Links - The target Attribute

The **target** attribute specifies **where to open the linked document**.

The target attribute can have **only one** of the following **values**:

_blank - Opens the linked document in a new window or tab

_self - Opens the linked document in the same window/tab as it was clicked (this is the default)

_parent - Opens the linked document in the parent frame

_top - Opens the linked document in the whole body of the window

framename - Opens the linked document in a named frame

Links

External links

```
<a href="http://google.com">Click For Google</a>
```

Internal links

```
<a href="/about.html">Typography</a>
```

```
<a href="#footer">Footer</a>
```

Images HTML

Internal image

```

```

External image

```

```

Step 4

- Add a link to your favourite website and set it to open in a new tab.
- Add an image inside the images folder.
- Display the image in your HTML file with an appropriate alt text.

Lists

https://github.com/margittennosaar/HTML-CSS.REACT25K/blob/main/01_week/html_lists.md

Lists

Unordered lists

- first item
- first item first subitem
- first item second subitem
- second item
- third item

Ordered lists

1. first item
 - 1.1. first item first subitem
 - 1.2. first item second subitem
2. second item
3. third item

Lists

```
<ul>
  <li>First item
    <ul>
      <li>first item first subitem</li>
      <li>first item second subitem</li>
    </ul>
  </li>
  <li>second item</li>
  <li>third item</li>
</ul>
```

Use of lists

An unordered list is one common method for creating a **navigation**

Step 5

- Add an **unordered list** with three items of your choice. (for example, your hobbies)
- Add an **ordered list** with three personal goals for this course.

CSS

https://github.com/margittennosaar/HTML-CSS.REACT25K/blob/main/01_week/css_base.md



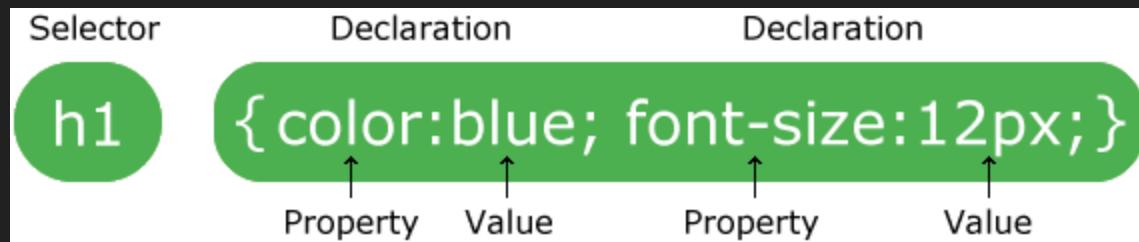
What is CSS?

CSS describes how HTML elements are to be displayed on the screen, paper, or in other media.

CSS saves a lot of work. It can control the layout of multiple web pages all at once.

External stylesheets are stored in **CSS files**.

CSS Syntax and Selectors



Selectors can be:

- HTML elements
- IDs #
- Classes .

CSS selectors

HTML element

p {color:blue}

id

#my_id {color:blue}

class

.my_class {color:blue}

The difference between an **ID** and a **class** is that an **ID** can identify one element, whereas a **class** can identify more than one.

Including CSS

1. External style sheet

2. Internal style sheet

3. Inline style

1

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

2

```
<head>
<style>
body {background-color: linen;}
h1 {color: maroon; margin-left: 40px;}
</style>
</head>
```

3

```
<h1 style="color:blue;margin-left:30px;">This is a heading</h1>
```

CSS

style.css

```
body {  
    background-color: lightblue;  
}  
  
#my_id {  
    color: white;  
    text-align: center;  
}  
  
.my_class {  
    font-family: verdana, sans-serif;  
    font-size: 20px;  
}
```

Comments

<!-- Write your HTML comments here -->

/* Write your CSS comments here */

Step 6

- Create a `styles.css` file in the `CSS` folder and link it to your `HTML` file.
- Add basic styling for the body, heading, and paragraph.
- Create a `script.js` file in the `js` folder and link it to your `HTML` file.
- Add a simple `console.log()` message in the script file.

Testing your project

Step 7

- Open index.html in your browser.
- Ensure all elements (heading, paragraph, lists, link, image) are displayed correctly.
- Verify that styles are applied.
- Check the console for your JavaScript message.

Manuals

<https://developer.mozilla.org/en-US/docs/Web/HTML>

<https://devdocs.io/>

<https://zealdocs.org/>