# HTML Document explained

## Introduction

HTML stands for HyperText Markup Language which is read by your browser and displayed as the webpages we are all familiar with. You could consider HTML as the constitution of web pages. Hyper Text is everything on a webpage that you can interact with, text, pictures and other things you can click that results in the webpage showing you something new. This can be a new webpage or a popup. Markup Language is basically telling the computer how to format things, this regards fonts, sizes, outlines, picture sizes and everything else you may want to edit from just plain text in a document.

## Tags

There are plenty of different tags in the HTML language, they may lok different but all of them serve the same purpose, make your elements look different or do something different that just being plain text or a standard element.

```
<div>Block element</div>
```

This line of code divides your elements into different sections in the webpage, making the document for organized for you to read and for different commands to apply only where you want them to.

```
<h1>Page title</h1>
<h2>Subheading</h2>
<h3>Tertiary heading</h3>
<h4>Quaternary heading</h4>
```

These lines are different headers in your document, the different tags have the of header that belongs to it written inside of them.

```
<p style="text-align: center;">text</p>
```

Paragraphs are the text you want to have under headers, just as in a normal document you write in word. The difference here is that you can style the paragraphs with the different commands inside the first <p>. The `>text<` in Between the closing and opening angled brackets is where your text goes, this is a good rule to always remember when styling an element like this.

```
<img src="/demo.jpg" alt="description" height="48" width="100" />
```

This is the tag for images, just like the `<p>` element this has different style commands inside the first angle brackets. The most important one is `src=` which is what refers to where the image you want to display is located. Without this command there will be no picture displayed.

```
<a href="https://htmlg.com/" target="_blank" rel="nofollow">Click here</a>
```

This tag creates an outbound link in your document, just like the `<img>` this tag needs a referring command so it knows where the link should lead to. In this tag the command is called `href=`. The text inside this tag does not need to be the link but will be highlighted in the web page so the user can see where to click to open the link.

```
<!-- HTML
Comment -->
```

Comments are something only displayed to those who view the backend code that creates a web page. This tag will not edit anything in the displayed web page but serves the purpose to add comments into the code to make it easier to read for the one programming or for others who want to either view or put some of their own code into the page and need to know what does what. The `Comment` is where you will write whatever you want to explain what is going on here.

```
<br />
```

Command for text to jump down a line when displayed in the web page.

```
<table><caption>Phone numbers</caption>
    <thead>
        <tr>
            <th>Name</th>
            <th colspan="2">Phone</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <td>John</td>
            <td>577854</td>
            <td>577855</td>
        </tr>
        <tr>
            <td>Jack</td>
            <td>577856</td>
            <td>577857</td>
        </tr>
    </tbody>
    <tfoot>
        <tr>
            <td> </td>
            <td>Personal</td>
            <td>Office</td>
        </tr>
    </tfoot>
</table>
```

This is a table, is works a lot like all other Markup Language but instead of just formatting the text it adds borders between the different elements of text or images for purposes where you might want to format a part of your web page in such a manner.

**How to insert an element**

```
<!DOCTYPE html>
<html>
<body style="background-color:powderblue;">

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Here you see a normal HTML structured page containing a header and a paragraph just like I showed earlier. This page has one unique style command that we have not showed yet. Inside the body there are style commands you can set just like in other tags. This exact style command defines a background color for your page and can be very usefull for creating your first web page. As you see in the example there is a defined `<body>` with a `</body>` later on in the code. These two commands defines where the body starts and ends which means everything that comes after the first body tag but before the ending body tag will be a part of the body. Therefore all the space that is created between the text is also a part of the body and is then styled to the color picked when you write the style command inside the first angle brackets of the opening body tag.

To fill text into this document you want to edit the text already displayed between `<h1>` and `</h1>` you can do this with all text tags like <p>, <h1> - <h4> and so on.

```
<img src="/demo.jpg" alt="description" height="48" width="100" />
```

This command as said earlier is used to insert an image into your web page. The `src="/demo.jpg"` is used to locate the image on your computer. The easiest way to do this is to put the picture in a different subfolder than the HTML document is in, but in the same folder. If you for example create a folder named “Web Page” and then create “index.html” to put your html code in and “Pictures” To put your pictures in inside the “Web Page” folder your code can look like this `src="../demo.jpg"`. The two dots tells the computer to look inside the folder that this document is located in but back out once to where the “Web Page” folder that contains all these folders are. `alt="description"` is used to put some text under the picture to describe what you are showing the reader. `height="48"` `width="100"` are styling commands used to define how many pixels wide and high the picture will be displayed as in the document. For best results you should always stick to the same aspect ratio (if the picture is 5px high and 10px wide the new size you define it as should always make the picture twice as wide as it is long e.g 10px high - 20px wide) to avoid stretching the photo.

```
<video width="200" height="150" controls>
      <source src="vid.mp4" type="video/mp4">
      <source src="vid.ogg" type="video/ogg">
      No video support.
</video>
```

This is how you insert a video into your web page. The `src=` command works the same here as in the other tags, you define the location of the vide file so the document can locate it and place it where it belongs in the page. The styling options is also the same as with images and should also be kept at the same aspect ratio as the original.

## CSS

CSS servers a lot of the same purposes as the styling you can do inside the angle brackets in HTML code. The difference between doing that and using CSS is that CSS is an external styling feature. This means that the styling you do in CSS is not written inside the index.html but rather in a different document normally called “style.css”. For the CSS to work you will need to link the two documents togehter, this is done by declaring the styling to be done by CSS at the top of your index.html like this

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="styles.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

As you see the fourth line in this code is `<link rel="stylesheet" href="..\styles.css">` which declares that this document will be styled by the CSS we linked it to.

The way CSS works is that it tells the HTML document “Hey, all pictures will have a light gray outline 10px wide and all headers are gonna be light green.” The advantage to doing this is that you won’t have to manually go and set all <h1> elements to the color light green because CSS just tells the document that they should all have it.

## Different types of CSS

### Inline CSS

Inline CSS is one of three types of CSS, this one and the next one we are going to be looking at are written inside the same document as the html code. The way inline CSS works is how we have already styled elements earlier on in the document, by setting an unique style to one element by typing the styling inside the angle brackets likte this.

```
<h1 style="color:blue;">This is a Blue Heading</h1>
```

### Internal CSS

Internal CSS is also written inside the HTML document and will therefore only apply to one HTML document, this is why we would choose this one over external CSS. The difference between this type of styling over Inline CSS is that this will apply to everything in the index and not just one element. You set this CSS type in the `<head>` section within a `<style>` element. With Internal CSS we can set all `<h1>` to be the color blue like this.

```
<!DOCTYPE html>
<html>
<head>
<style>
body {background-color: powderblue;}
h1   {color: blue;}
p    {color: red;}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

### External CSS

External CSS is what is most commonly used by web designers as the one file you create and edit can be used for all different pages of a website to make the styling match for all of them. To declare the external CSS file as the one to be used for an HTML document you will have to link them inside the `<head>` section at the top like this.

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="styles.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

### How to set up a menu for your web page

1) Add the code for a menu bar

```
<div class="topnav">
  <a class="active" href="#home">Home</a>
  <a href="#news">News</a>
  <a href="#contact">Contact</a>
  <a href="#about">About</a>
</div>
```

2) Style the menubar like this

```
/* Add a black background color to the top navigation */
.topnav {
  background-color: #333;
  overflow: hidden;
}

/* Style the links inside the navigation bar */
.topnav a {
  float: left;
  color: #f2f2f2;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 17px;
}
```

```
/* Change the color of links on hover */
.topnav a:hover {
  background-color: #ddd;
  color: black;
}

/* Add a color to the active/current link */
.topnav a.active {
  background-color: #4CAF50;
  color: white;
}
```

3) View in the web page by opening the index.html file with a browser.