CSS (cascading style sheets).

CSS is used for adding a visual style to a website. With CSS you can add color to your website, manipulate the position and size of your content such as fonts, headers, etc.

CSS takes a plain website and turn it into a colorful page that is much better and easier to view and see. You can define design, layout, and different displays for devices and screen sizes.

CSS files are saved with the extention .css, example would be ==> stylesheet.css

With CSS you can save a lot of work, because you can use one css file to change and manipulate one or all of the web pages on your website.

The way CSS works is, the browser reads the css file/document and then displays the page according to the information in the style sheet (css file).

There are 3 main ways to implement CSS,

1-Externel
2-Internal
3-Inline

External >>

With the External method, you can change the entire website (multiple pages) with just one css file. Example, each website page will have a link to the css file in <head></head> section of the HTML document (observe below, highlighted in red).

<!doctype html>
<html>
<head>
        <link rel="stylesheet" type="text/css" href="stylesheet.css">
        <title>siteB for all</title>
</head>

The stylesheet can be written in any text editor no matter which method you use.

Internal>>

You would use the Internal method if you have one website (1 website/page) that will being using one style, such as the example below highlighted in red-

```
<!doctype html>
<html>
<head>
        <title>siteB for all</title>
                <style>
                        CSS code would go here
                </style>
</head>
```

Inline >>

The inline method is used for when you want to embed CSS code right into the website page like this below highlighted in red –

```
<tag=”CSS code here”>content on website</tag>
```

CSS code >>
CSS syntax code is made up of selectors and declarations, example code in red-

Selector   Declaration
```
selector {property:value;}
```

OR

Selector   Multiple Declarations
```
selector {
      property:value;
      property:value;
}
```

A selector points to an HTML element/tag you want to style. Selectors can either be based on the html tag, id (#), class (.), or an attribute.

Example – the selector html <p> tag as a CSS id- in red

```
#p {
      text-align: center;
      color: red;
}
```

==

Example – the selector html <p> tag as a CSS class- in red

```
.p {
      text-align: center;
      color: red;
}
```

==

Example – the selector html <img> tag as a CSS id- in red

```
#img {
      border-width: 0px;
      float: left;
}
```

Declaration block contains one or more declarations separated by semicolons

A more direct example would be if you wanted all the text in your website to be say red and be in the center, see example below –

```
p {
        text-align: center;
        color: red;
}
```

Colors can be added by either name or by it's hexadecimal code equivalent. For a list of color hex codes, go to w3schools link –
https://www.w3schools.com/colors/colors_hex.asp

===============================

declarations are actually used by CSS to manipulate all the elements/tags on the website page, such as magins, borders, padding, content.

Look at the next examples below to see how to use → magins, borders, padding, content.

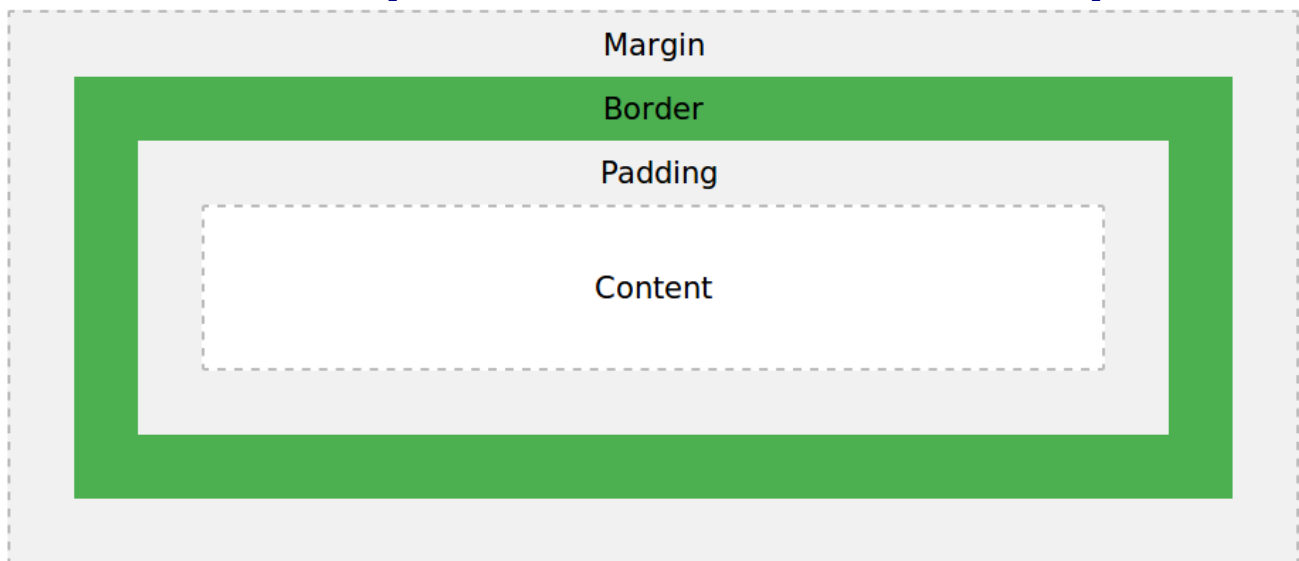**Content** - The content of the box, where text and images appear

Padding - Clears an area around the content. The padding is transparent

Border - A border that goes around the padding and content

Margin - Clears an area outside the border. The margin is transparent

Reference w3schools - https://www.w3schools.com/css/css_boxmodel.asp

Example –

```
.tag {
      width: 320px;
      padding: 10px;
      border: 5px solid gray;
      margin: 0;
}
```

A website page is made up basically of the following – headers, menus, content, and a footer.