Challenge

Style a Web Page Using CSS Properties and CSS Box Model

Challenge

- Style a Professional Portfolio

Points to Remember

- Give meaningful ID and class attribute names to HTML elements.
- Use appropriate CSS selectors to select HTML elements for styling them.
- Use shorthand properties wherever possible that make CSS code concise.
- Make sure that the CSS code is well-indented.
- Add proper comments wherever required.

Style a Professional Portfolio

Utilize CSS3 style properties to style a user-friendly web page that displays a professional portfolio.

In the boilerplate, the sample output is provided in the form of an image file.

Click this link for the sample output.

CHALLENGE

Challenge: Tasks

- The boilerplate for this challenge contains **index.html** with HTML5 tags.
- As part of the solution, the HTML elements should be styled using CSS3 style properties.
- The solution for this practice can be developed in 4 steps:
  - **Step 1:** Define styles in **style.css** file located in CSS folder of the boilerplate.
  - **Step 2:** Link the **style.css** file with index.html using <link> tag.
  - **Step 3:** Link the predefined stylesheets for fonts in **index.html** file.
  - **Step 4:** Apply the styles defined using class or id selector to the page elements.

Pre-Defined Styles

- Below is the list of predefined stylesheets that need to be added inside the **index.html** for formatting the fonts.
  - `<link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.1.1/css/all.min.css">`
  - `<link rel="preconnect" href="https://fonts.gstatic.com">`
  - `<link href="https://fonts.googleapis.com/css2?family=Caveat:wght@700&display=swap" rel="stylesheet">`
  - `<link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Dosis:400,700%7CBitter:400,400italic,700&subset=latin,latin">`

Let's look at CSS Values and Units

CSS Values and Units

Every property used in CSS has a value type defining the set of values that are allowed for the respective property.

Data Type	Description
<integer>	An <integer> is a whole number such as 1024 or -55.
<number>	A <number> represents a decimal number — it may or may not have a decimal point with a fractional component. For example, 0.255, 128, or -1.2.
<dimension>	A <dimension> is a <number> with a unit attached to it. For example, 45deg, 5s, or 10px. <dimension> is an umbrella category that includes the <length>, <angle>, <time>, and <resolution> types.
<percentage>	A <percentage> represents a fraction of some other value. For example, 50%. Percentage values are always relative to another quantity.

Absolute Length Units

Unit	Name	Equivalent to
cm	Centimeters	1cm = 38px = 25/64in
mm	Millimeters	1mm = 1/10th of 1cm
Q	Quarter-millmetes	1Q = 1/40th of 1cm
in	inches	1in = 2.54cm = 96px
pc	picas	1pc = 1/6th of 1in
pt	points	1pt = 1/72th of 1in
px	pixels	1px = 1/96th of 1in

Relative Length Unit

Unit	Relative to
em	Font size of the parent, in case of typographical properties like font-size, and font size of the element itself, in the case of other properties like width.
ex	x-height of the element's font
ch	The advance measure (width) of the glyph "0" of the element's font
rem	Font size of the root element
lh	Line height of the element
vw	1% of the viewport's width
vh	1% of the viewport's height
vmin	1% of the viewport's smaller dimension
vmax	1% of the viewport's larger dimension

BEM Syntax: Naming Conventions for Class Names

Check this link for better understanding.