**Cascading Style Sheet(CSS)**

* It is the style sheet language used to define the looks of the web elements
* By using we can change the looks of the elements and we can make web page responsive.

**Why do we need CSS?**

* It is used to add styles for text content (colours, font size, font style etc.)
* It is used for layout or element designs (positioning, flex, animations)
* It is used to make webpage responsive

**History of CSS.**

* It was introduced in 1993
* The first version of css is css1 in year 1994
* It was invented by Håkon Wium Lie
* The current version of css is CSS4

We can add CSS in 3 different ways,

* In-line CSS(within a html element, with highest priority)
* Internal CSS(within a html document, with middle priority)
* External CSS(return in a eaxternal .CSS file and linked with the html document, with least priority)

1. **Inline CSS**

we can apply inline css to the html document by using style attribute inside a html tags.

Syntax:

<tagname style=”property:value”></tagname>

Example:

<h1 style=”color:blue;border:5px Solid red”> Welcome</h1>.

1. **Internal CSS**

Here we can apply CSS for html elements by using style tag inside a head section.

Syntax: <style></style>

<head>

<style>

Selectors{

color:red;

border:5px Solid red;

}

</style>

</head>

1. **External CSS**

By creating separate CSS file with extension .CSS and linking to the html document

Advantage: code reusability.

To link the CSS file to the html document we use a link tag.

**link** tag

this tag is used to link css file in a html document, we should use this tag in head section.

Syntax:

<link rel=”stylesheet” type=”text/css” href=”path/to/CSS file>

Example:

Style.css

H1{

Color:red;

Border:5px solid green

}

P{

Color:white;

Back-ground-color:blue;

Border:black

}

B{

Color:violet;

}

**CSS Selectors**

**6) Pseudo Class selector**

It defines the state of an object or an element

It means it changes the style of default element to another style

It mainly used to for anchor or button tag

We have different types of Pseudo Class Selector

* + Link
  + Hover
  + Active
  + Visited

**Syntax:**

Selector: Pseudo class {

Property: value

}

**7)Adjacent Selector.**

By using the selector we can select the child elements with reference of parent element and we can provide styling.

**Syntax:**

Parent\_element child\_element{

Property:Value

}

**8)Attribute Selector**

By using this selector we can select the elements based on attribute name or attribute value to provide styling

Syntax 1:

Tag\_name[attribute] {

Property: value

}

Syntax2:

Tag\_name[attribute=”value”] {

Property: value

}

**Properties**

**1.Color Property.**

Using this tag we can provide color for html elements like,

Text color

Background color

Border color

And some other decorative purposes also.

Syntax:

color:value === property:value

values: built in color

rgb

rgba

hsl

hsla

hexadecimal

**color:built-in-colors**

by using this value we can provide direct color names as a value.

Example:

color: red;

green;

blue;

yellow;etc….

**RGB format: color:rgb(red,green,blue)**

It is a function format of color combination red,green and blue.

The value of range of each color starts from 0-255 range

0 -is lowest range.

255 - is the highest range.

Syntax:color:rgb(r,g,b);

Example:

color:rgb(255,0,0) # red

color:rgb(0,255,0) #green

color:rgb(0,0,255) # blue

color:rgb(255,255,255) # white

color:rgb(0,0,0) # Black

color:rgb(50,250,70) # light green

**RGBA format: color:rgba(Red,Green,Blue,Alpha)**

RGB stands same as above context, the meaning syntax all are same.

The additional ‘A’ stands for Alpha which means opacity or transparency

The alpha value ranges from 0 .0 to 1.0

Example:

Alpha: 0.0 means full transparency the background color will be displayed

Alpha: 1.0 means no transparency the color combination pointing color will be displayed.

For other values of alpha that much % of transperancy will be applied.

**Hexa Format: color:#------**

Hexa values: 0-9 => 0-9, 10-15=>A-F

Using these values we specify the colors.

#followed by 6 hexa decimal digits. Among(0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)

Syntax:

color:#------

--: first 2 characters represent red

--: middle 2 characters represent green

--: last 2 characters represent blue

Example:

color:#000000 //Black

color:#0000ff //blue

color:#00ff00 //green

color:#ff0000 //red

color:#aaaaaa //cement

HSL Format:color:(hue,saturation,lightness)

Hue value represents for colors in degree format, the range is 0`-360`

0-red, 120-green, 240-blue

The saturation and lightness will be given based on the color values.

The saturation value ranges from 0% to 100%

The lightness value ranges from 0% to 100%

Syntax: color:hsl(240,100,100)

**HSLA Format:color:(hue,saturation,lightness,alpha)**

All the HSL is same as above context, the only difference is additional alpha, which controls the transparency of the color.

The range is from 0.0 to 1.0

Syntax: color:hsl(240,100,100.1.0)

NOTE: to know these color values use Google color picker

background property:

using the property we can apply background effect for html elements…

the background properties have few sub properties

* + background-color
  + background-image
  + background-repeat
  + background-position
  + background-size
  + background-attachment
* background-color

to provide color to background

Syn: background-color:value using any one of 6 format.

* background-image

to insert an image to background

Syn: background-image: url (‘image address/ image name’)

Note: by default, sometimes we will get repeated bg image due to default height and width of image.

* background-repeat

we can control the repetition of bg-images.

Syn: Background-repeat: values;

Values: 1. No repeat: to stop repetition of images.

2. repeat-x: repeat the image in x-axis.

3. repeat-y: repeat the image in y-axis.

Note: to use bg-repeat property we need bg-image property.

* Background-position

We can set the position of bg-images

By default, the position of bg-image is top left

Syn: Background-position: values

Values: 1. Top-left

2. top-right

3. bottom-right

4. bottom-left

5.top (in center top)

6. bottom (in center bottom)

7. center

8. center-left

9. center-right

Note: 🡪to use background position we need background image and bg-repeat property.

🡪 We need height and width property to set background postion.

* Background-size

We can set size of background image.

Syn: Background-size: values;

Values: 1. Auto and Cover

2. height and width: 300 px and

* Background-attachment

we can place background-image in a fixed place even if we scroll the page the image should be in fixed place

Syn: Background-attachment: value

Values: fixed

Border property:

using the property we can apply border of html elements.

The border property has 3 sub properties:

1. Border-width
2. Border-style
3. Border-color

* Border-width:

By using this property, we can provide width for the border lines.

Syn: border-width: values;

Values: 🡪predefined values- 1. Thin, 2. Thick 3. Medium

🡪user defined values- 1. In px or %

Ex: border-width: 10 px (all sides equal size)

Border-width: 10 px, 20px (opposite sides equal size)

Border-width: 10px, 20px, 30px (all three sides as defined size and the extra side will be the opposite side size).

* Border-style:

By using this property, we can provide style for the border lines.

Syn: border-style: values;

Values: 1. Solid

2. dotted

3. dashed

4. double

5. groove

6. ridge

7. inset

8. outset

9. hidden

10. none

* Border-Color:

By using this property, we can apply color for the border lines.

Syn: border-color: values;

Values:1. Built-in-color

2. RGB 3. RGBA

4. HSL 5. HSLA

* Border-radius:

By using this property, we can apply radial shapes for all four border lines.

Syn: border-radius: values;

Values: 1. Border-top-left-radius

2. border-top-right-radius

3. border-bottom-left-radius

4. border-bottom-right-radius

Outline property:

By using this property, we can provide outline for the border.

Outline property has 3 sub properties:

1. Outline-width
2. Outline-color
3. Outline-style
4. Outline-offset

* Outline-offset:

We can define a space between outline and the border.

Syn: outline-offset: values;

CSS box model

The CSS box model is the rectangular shape container box. Where elements are wrapped around.

The box model consists of content, border, padding, and margin property.

🡪Content: it may be text or image.

🡪Border: the line around the content.

🡪Padding: the space between content and border.

🡪Margin: the space between border and viewport.