Css

**Background-Blend-Mode:**

It is just like the layer blend mode option in photoshop. The whole concept is that, you may want to join two object (color & image) together to achieve a single effect. It has all the blend options in photoshop like, screen, difference, color-dodge, color-burn, hard-light, soft-light, saturation, hue, unset, luminosity, initial, inherit

Just add a background image and a background color and decide which effect you want these two objects to have.

body {

    background-image: url("images/one.jpg"); //image url

    background-color: blue; // background color

    background-blend-mode:difference; //blend the above image with the above color to achieve a difference effect

}

**==================================================**

**Clip-Path:**

It used for generating shapes as backgrounds. The concept is that, you may want to create a shape like rectangle or star and make it as a background of an element like div.

There is no need to create it yourself. Head over here to generate one online https://bennettfeely.com

div {

    height: 300px;

    width: 100%;

    background-color: rgb(161, 194, 41);

    clip-path: polygon(0 0, 11% 0, 51% 17%, 85% 0%, 100% 0, 100% 8%, 100% 85%, 100% 100%, 85% 100%, 15% 100%, 0 100%, 0% 85%);

}

**==================================================**

**Google font:**

Is a way of using google fon’t in your project. The concept is that , you may be interested in some google font you want to use in your project.

// Just go to <https://fonts.google.com/>

// Select the font you want

//copy the link from google font and import it in your css

// assign the font family to an html element in your css with font-family: (you will see this on google font websit)

/\* import the font from google\*/

@import url('https://fonts.googleapis.com/css2?family=Fraunces:ital,wght@0,100;0,200;1,200&display=swap');

/\*assign the font to an html element\*/

h2{

    font-family: 'Fraunces', serif;

}

**==================================================**

**Css Gradient:**

Is a ways of applying multiple colors what is known as gradient to an element.

No need to great it yourself. Simple go to <https://cssgradient.io/> and generate one and copy the code to your project like below.

div {

    height: 300px;

    width: 100%;

    background: rgb(255, 113, 0);

    background: linear-gradient(240deg, rgba(255, 113, 0, 1) 0%, rgba(233, 207, 44, 1) 55%, rgba(255, 147, 0, 1) 100%);

}

**==================================================**

**:nth-child(n)**

Use to assign styles to multiple number of occurrence of a specific element. Just like chat application, you may want to change the background color of all the messages from the sender.

selectetor:nth-child(n) {  
  css declarations;  
}

<p>The first paragraph.</p>

<p>The second paragraph.</p>

<p>The third paragraph.</p>

<p>The first paragraph.</p>

<p>The second paragraph.</p>

<p>The third paragraph.</p>

<style>

// change the background color of the above paragraph element 2,4,6,8….. to blue even numbers

p:nth-child(odd) {

  background: red;

}

// Change the background of the above paragraphs to red by using odd numbers 1, 3, 5, 7, 9 (use it in chat application)

p:nth-child(even) {

  background: blue;

}

</style>

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

Tips For Creating Responsive UI ELEMENT

To create responsive element that will adjust it size based on the screen size. Forget about percentage (%) and pixels(px).

Use view width (vw) rather. View width will adjust the element width and height perfectly based on the size of the user screen. EX:

#top\_text {

    font-size: 3vw; // change the font-size perfectly based on the screen size (3 represent the size you want)

    padding-top: 5vw;

}

So use vm instead of pixels and width.

**==================================================**

**TIP TO ADD A RESPONSIVE IMAGE TO YOUR PROJECT**

#man\_image {

    /\*change this max-width if you want to change image size\*/

    max-width: 100%;

    width: 100%;

    height: auto;

}

**==================================================**

**Line-height property**

They are used to specify the vertical height of a div or paragraph.

Note= Use percentage “%” to specify line height properties instead of pixels.

Note= negative values are not allowed in the line-height property. Specifying a negative value will have no effect on the element.

**==================================================**

**Add Stroke** Around **Text**

Use the:

-webkit-text-stroke to set the thickness and color of the stroke.

-webkit-text-fill-color to set the main color of the text.

This two above will override the default color:xxxx property if any is set to it.

<p id="one\_week">HOME CALL</p>

#one\_week {

    -webkit-text-stroke: 2px white; stroke thickness and color \*/

    -webkit-text-fill-color: red; /\*text color \*/

}



**==================================================**

**Calc() function**

The calc() method is used to calculate numbers and assign the result to a value.

You can use these mathematical operators to work with the calc function. +, - , /, \*

**Example:**

Imagine I have this html tag: <div id=”check”> checking something </div>

 #checkk {

     /\*

    meaning 100% - 10% = 90%

    so the result 90% will be assigned to the width property below

     \*/

     width: calc( 100% - 10%);

 }

 #checkk {

     /\*

     Meaning, first 20% - 10% = 10%

     then the resulst 10% \* 5 = 50%;

     sow 50% will be assign to the height property below

     \*/

     height: calc( (20% - 10%) \* 5);

 }

**==================================================**

**How to use HTML tag name to access it own CLASS or ID attribute**

You can use the same html tag name to access it own class & id attribute.

Example:

<p class=”myp”> am paragraph </p>

If you want to access the above paragraph tag in css, you will use the “p” selector.

But if you want to access the class attribute, you can directly use “.myp”

But what if there is another html tag with the same class name like: <div class=”myp”> am the div </div>

In this case, if you access the class attribute “.myp” it will point to all the above two tags. But what if you want to point to only the <p> tag above?

You have to use each individual tag to access it class attribute like. P.myp

Note: you don’t have to add space between the tag name & it class or Id attribute in the css when referencing it

**Example:**

    /\*

    Below a.active {} means

    Access the class="active" attribute of the <a> input tag below

    <a href="#" class="active"> my Name</a>

    Does how you use html tag to access it own attribute.

    Remember not to add space between the "a" and the ".active"

    if you do, it will try to access something like this

    <a href="#" class="active"> my Name

    <span class="active"> access this child element </span>

    </a>

    \*/

    a.active {

        background-color: var(--color-light);

        color: var(--color-primary);

        margin-left: 0px;

    }

**====================================================================================================**

**::before & ::after pseudo element**

**::before =** it used to insert something before the content/text of the selected element. Use the “content” property to specify the content to insert, and apply other styles to the inserted content with any other html attribute.

**Example:**

<p>My name is Donald</p>

 <p>I live in Ducksburg</p>

    p::before {

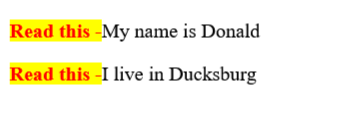
        content: "Read this -";

        background-color: yellow;

        color: red;

        font-weight: bold;

    }

****

**::after =** it used to insert something after the content/text of the selected element. Use the “content” property to specify the content to insert, and apply other styles to the inserted content with any other html attribute.

**Example:**

<p>My name is Donald</p>

 <p>I live in Ducksburg</p>

    p::before {

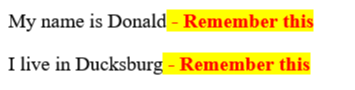
        content: "Read this -";

        background-color: yellow;

        color: red;

        font-weight: bold;

      }

****

**====================================================================================================**