**CSS**

* Cascading Style Sheets 3
* Released in 1996 and maintained by W3C Org
* Used to change look/feel of html elements (makeover)

Like color, background, border, alignment, wallpaper, animations, size of element, padding, margin, opacity etc…

* CSS provides only styles but not tags
* Style is group of properties or is a rule

Width=val height=val 🡸 html attributes

Width:val height:val 🡸 css properties

**Where we can define styles?**

We can define styles in 3 places (scope), those are:

* Inline styles
* Internal styles
* External styles

Different ways to implement css:

**Inline (1st Approch)**

Html tag and css properties both are defined within the same line

Syn:-

**<tag style=”property:value; property:value; … “**>

Unique style or personal styles

**Internal (2ndApproch):**

Html tags and css styles are designed in the same program, but not in same line.

Internal css should be implements in Style tag, style tag must be sub tag “head” tag.

Syn:-

**<style>**

**tag{**

property:value;   
 property:value;

…….

**}**

**tag{**

property:value;  
 property:value;

…….

**}**

Etc…

**</style>**

Personal style but repeated applying

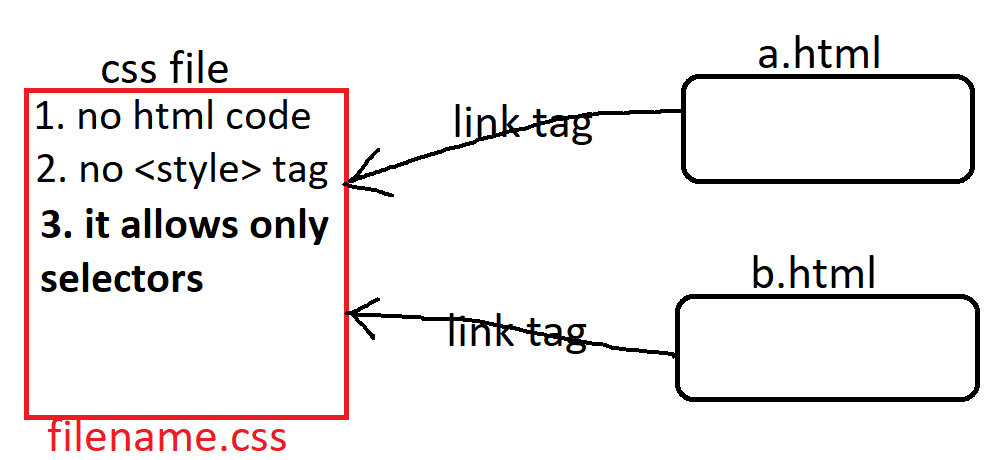
**External (3rdApproch)**

Css styles are defined in separate file and should be save with “**.css**”,

html code designed in separate file should be save with saved with **“.html**”

Use link tag for mapping css file to html file

Syn: **<link rel=”stylesheet” href=”filename1.css”/>**

****

**note:**

* css attributes we can't use in place of html attributes.
* html attributes we can't in place of css attributes.

**html colors**

html supports 3types of patterns, those are

> named colors

> RGB colors

> Hexadecimal colors

**named colors:**

>it supports to write direct color name

>we have some limited colors

ex: white, black, red, green etc...

>Color names are not case-sen

**RGB colors:**

>RGB model specifies that the composition of 3 basic colors (Red, Green, Blue)

>RGB produces 16millions colors.

Syn: **rgb**(red,green,blue)

red => 0 - 255

green => 0 - 255

blue => 0 – 255

ex: **rgb**(10, 45, 201) 401%255 🡺146

**Hexadecimal number colors:**

>Hexadecimal model is the shortcut for rgb model

>Hexadecimal system ranges from 0 - 15

0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f

**Syn: #**RRGGBB 1,2 red 3,4 green 5,6 blue

ex: #1a4b68

**#RGB**

**e**x: #3d7

**Note:** in realtime "Hexadecimal model" is recommended.

these colors we can use for foreground color, background color, border color

etc..

for setting colors we have some attributes, those are

**color** 🡺 to set/to change foreground color (text color)

**background-color** 🡺 to set/to change background color

**border-color** 🡺 to set/to change border color (line color)

**box-shadow** 🡺 to set/to change shadow color

**text-shadow** 🡺 to set/to change text shadow color

**Note:** all these are CSS attributes. Support by Most of html tags

**Gradient colors**

background: #FC466B; /\* fallback for old browsers \*/

background: -webkit-linear-gradient(to bottom, #3F5EFB, #FC466B); 🡸 Chrome 10-25, Safari 5.1-6

background: **linear-gradient**(to bottom, #3F5EFB, #FC466B); 🡸W3C, IE 10+/ Edge, Firefox 16+, Chrome 26+, Opera 12+, Safari 7+

**linear-gradient**(direction, color1,color2,…color-n)

dir: to left (r=>l)

to right (l=>r)

to top (b=>t)

to bottom (t=>b)

background: **linear-gradient**(to bottom, #3F5EFB 40%, #FC466B 60%);

-webkit-linear-gradient(to left, #3F5EFB, #FC466B);

linear-gradient(to left, #3F5EFB, #FC466B);

**background: radial-gradient(circle, rgba(2,0,36,1) 0%, rgba(38,38,162,1) 60%, rgba(0,212,255,1) 100%);**

**radial-gradient(shape, color1, color2, …color-n)**

radial-gradient(circle, rgb(131,58,180) 0%, rgb(29,166,65) 50%, rgb(252,176,69) 100%);

radial-gradient(circle, rgba(166,29,142,1) 57%, rgba(100,180,111,1) 78%, rgba(69,252,96,1) 100%);

**Note:** while applying gradient colors we have to use “**background**” property in place of “**background-color**”.

opacity: 0.5;

filter: blur(5px);

brightness(125%)

contrast(135%)

grayscale(100%)

invert(100%)

hue-rotate(180deg)

saturate(8)

sepia(100%)

drop-shadow(8px 8px 10px green)