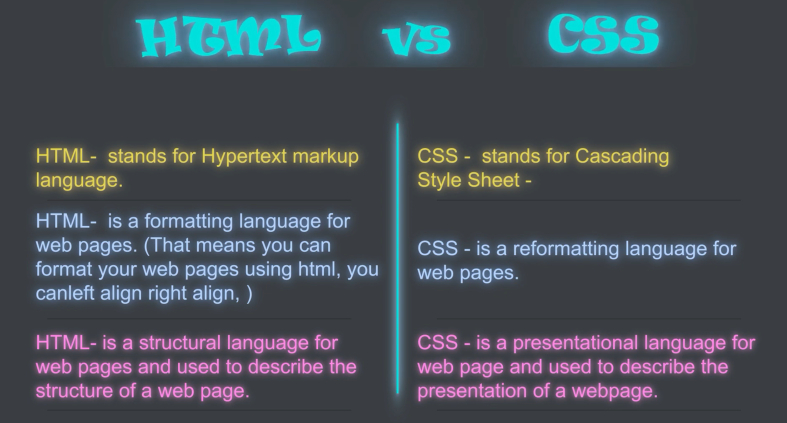
Presentation and layout = css

Cascading style sheet = css

Font color , size etc

Layout = placement.

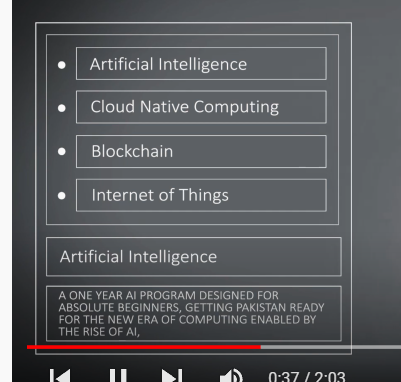
Html vs css :



Css describe how elements used to be displayed beautifully and attractive



We can style a skeleton into a content which can be visualized in terms of colors gradients placements and designs



Add borders , specify width and height , color size etc

Graphical user interface, text

Description automatically generated

Selector is paragraph declaraction is color name







Graphical user interface

Description automatically generated

Graphical user interface, application, website

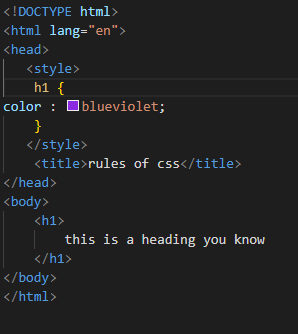
Description automatically generated

Applying styles in multiple elements

Graphical user interface, text

Description automatically generated

Adding a color in heading:

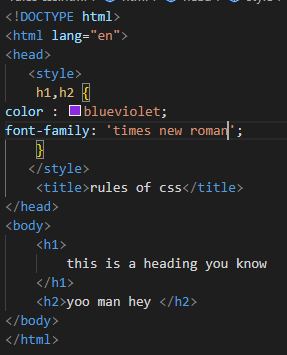


Adding multiple headings with same color :

A screenshot of a computer

Description automatically generated with medium confidence

Changing fonts



There are two types of css inline css and internal css

Inline css = css including into html code file.

Internal css means linking separate css code file with html source file.

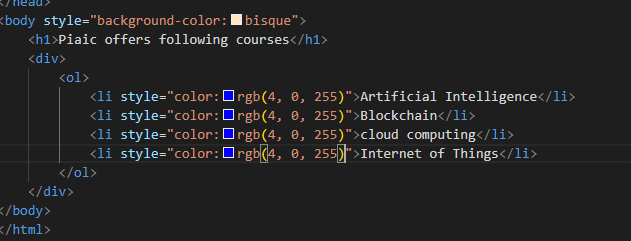
***Inline css:***

Styles used only single elements.



Three types of css we can apply.

Inline css



Text

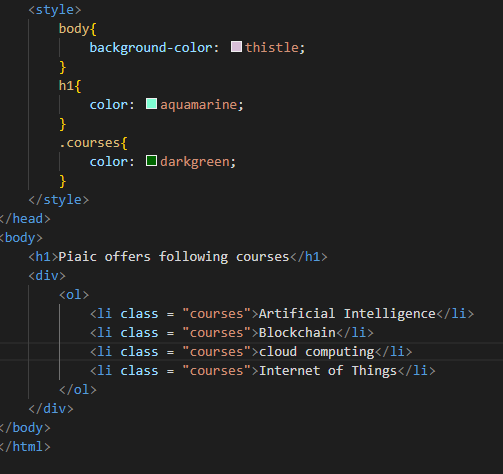
Description automatically generated

External css

Internal css

***Internal and embedded css:***

Written in head tag.



***External css:***

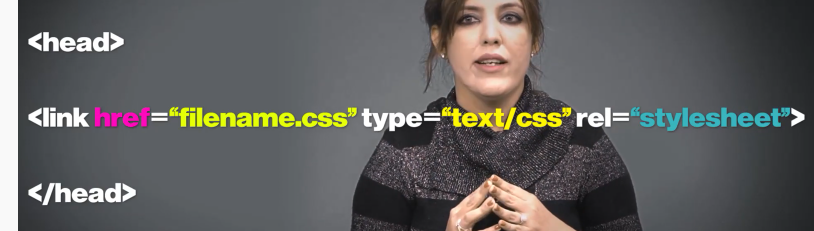


Specity file name with .css extension.

Text css we use with hyptertex reference.

Text

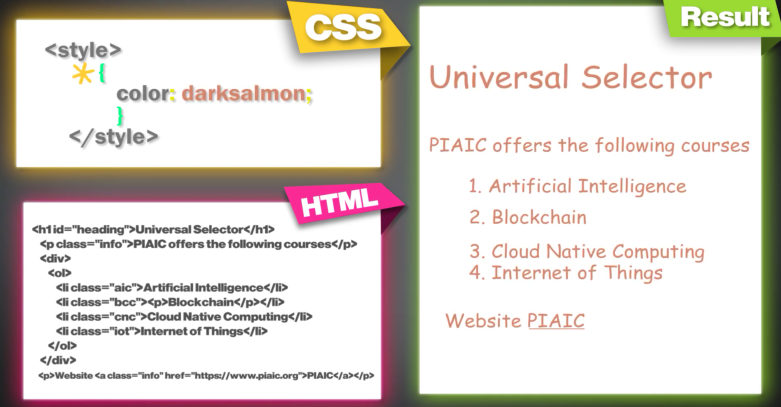
Description automatically generated with medium confidence



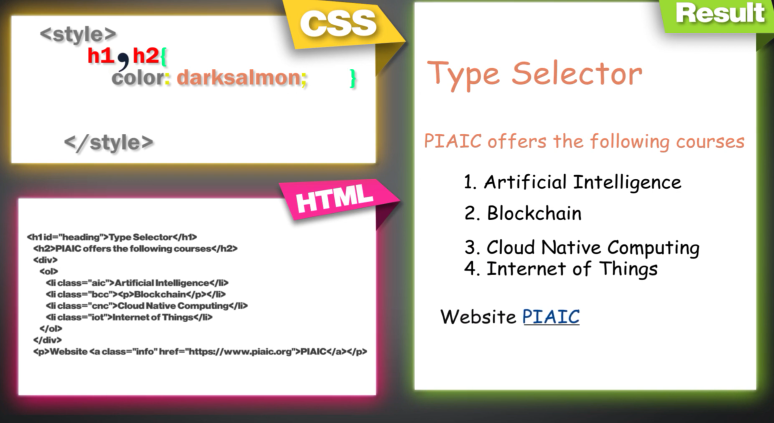
Fonts and colors and external stylesheet for that we use which can be applicable on all pages using just one stylesheet file.

***Css Selectors:***

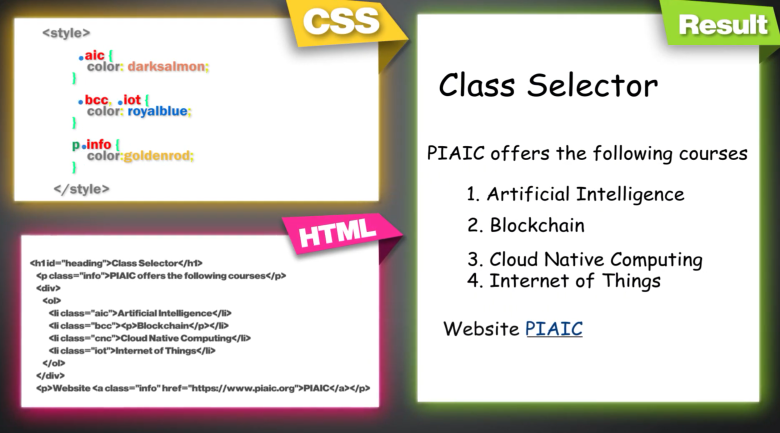
\*(universal selector) all elements on the page targets.applying same rule on the whole page.



,(type selector) element selector by giving element name.



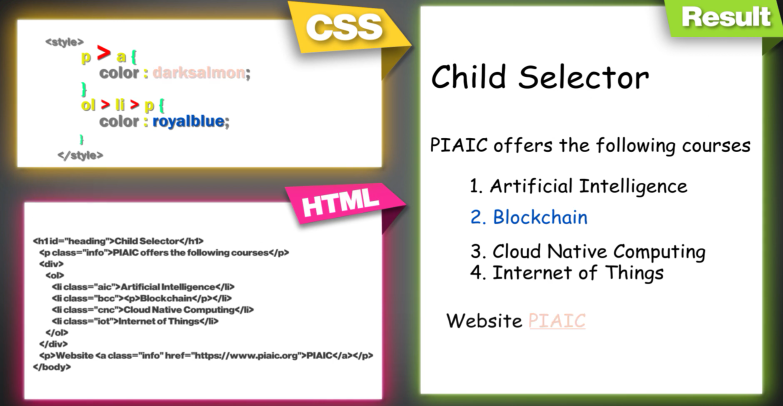
.(class selector) selects elements with specific class attributes.



#(id selector)



>(child selector)



<->(decendent selector)



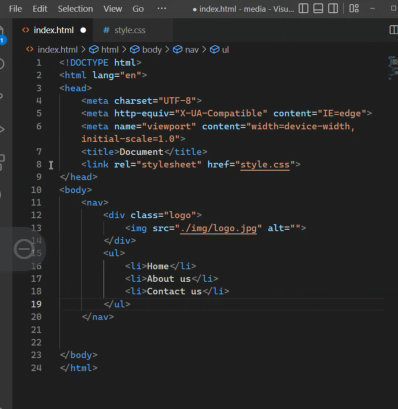
+(adjacent sibling selector) matches the element that is the next sibling of another.



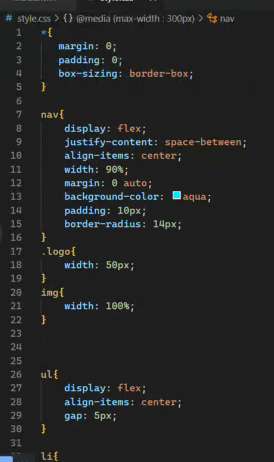
~(general sibling selector)

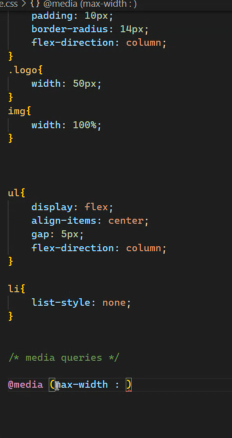
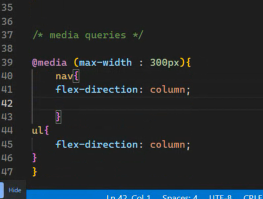


Responsiveness:



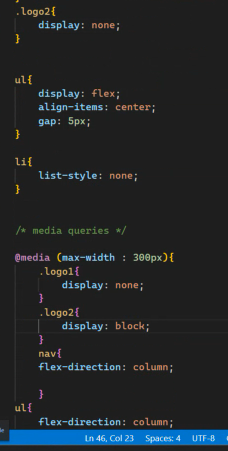




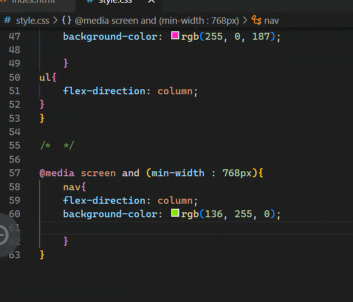
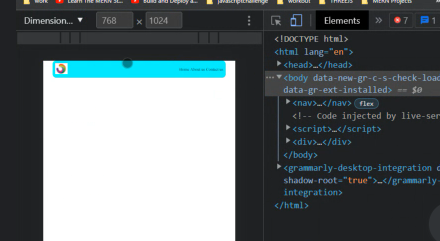
``

Text

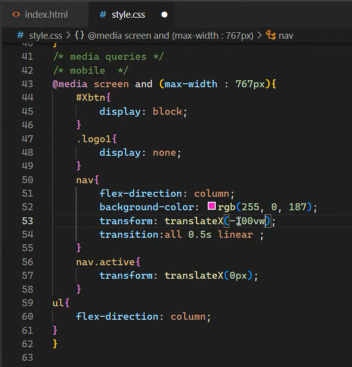
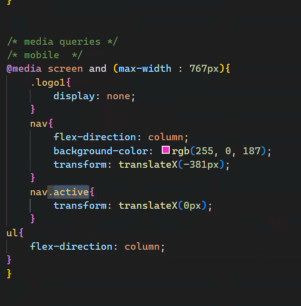
Description automatically generated



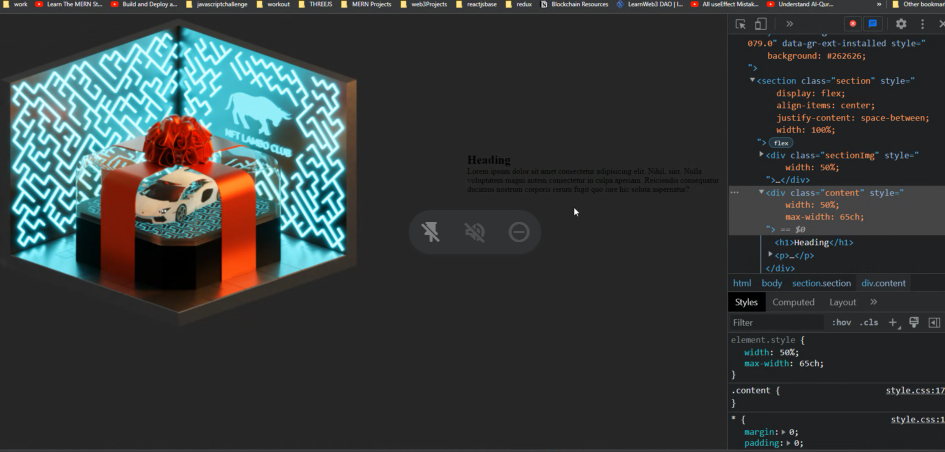
Checking and adding different mobile types dimensions into media queries

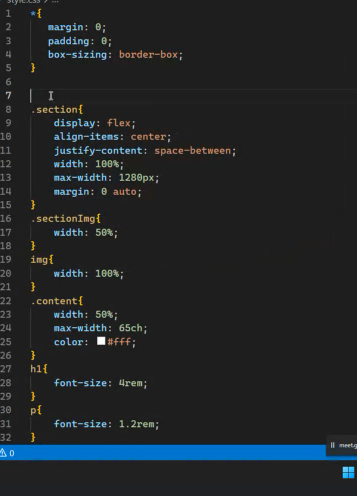
Text

Description automatically generated



Example 02 :





***HTML FILE FOR SELECTORS :***

***Text

Description automatically generatedText

Description automatically generated***

***Universal selector :***

***Graphical user interface, text

Description automatically generated***

***Table

Description automatically generated***

***Type selector :***

***Graphical user interface, application

Description automatically generated***

***Table

Description automatically generated***

***Type selector :***

***Text

Description automatically generated*** ***Table

Description automatically generated***

***ID selector :***

Id remains unique for each page.

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated Graphical user interface, text, application

Description automatically generated

***Child selector :***

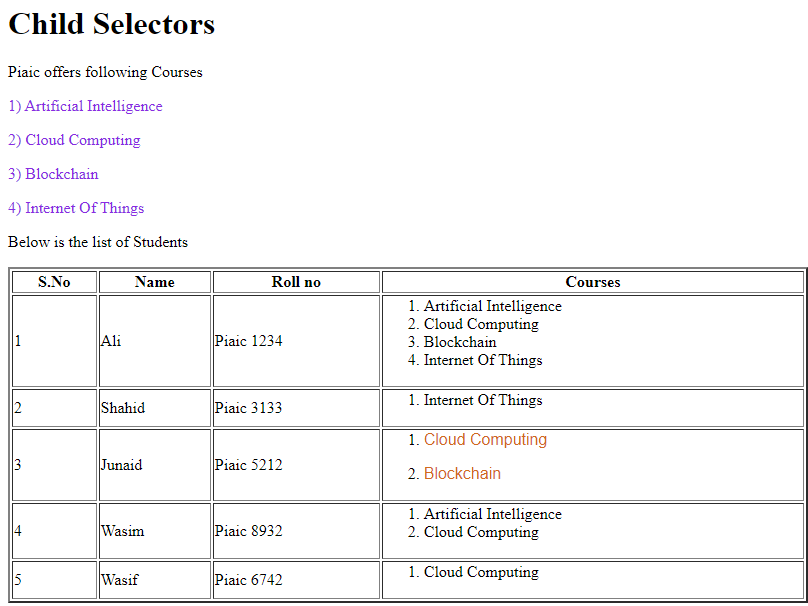
For instance target div tags for styling whose parent is div. any elements just associated with div.

Changes applied only those elements whomes are the child of parent “div”.

Target to direct child.

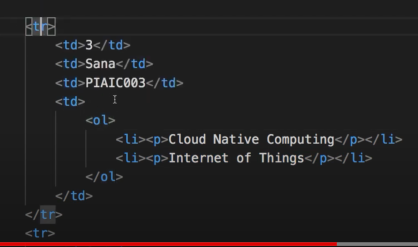
Text

Description automatically generated



***Decendent selector :***

No matter child is direct or indirect.

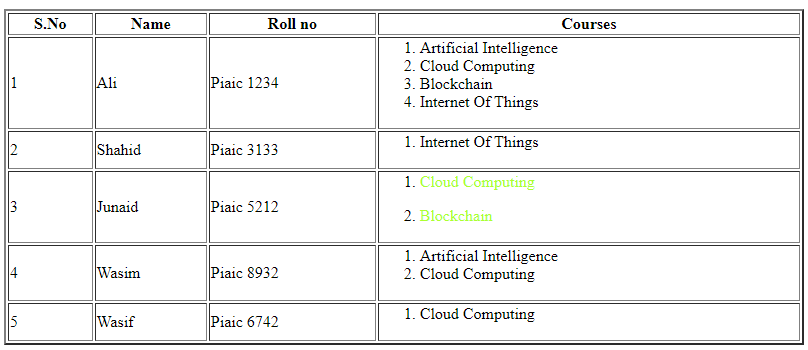


Since there are many elements of childs and parents to each other.

Targeting specific para form many paras already there of same child.

Graphical user interface, text

Description automatically generated

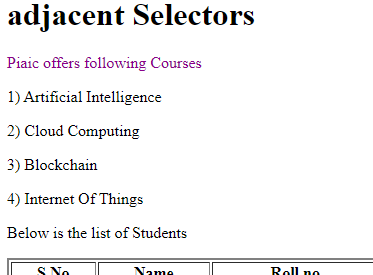


***Targetting sibling selector :***

Adjacent sibling , targets nearest sibling of first heading or element.

Graphical user interface, text

Description automatically generated



***General sibling selector :***

Same level elements just targeted.

H1’s general siblings are targeted like h2 h3 h4

Paras before h1 is ignored.

Targeting all div elements.

Graphical user interface, application

Description automatically generated

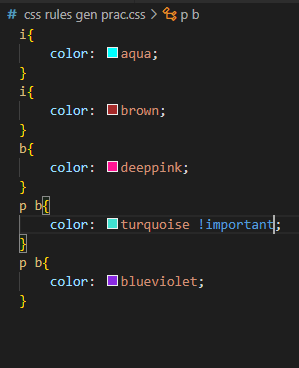


General Practice Examples :

There is rule to override to the past changes in css sytles. Use !important there.

Text

Description automatically generated



Graphical user interface, text, application, email

Description automatically generated

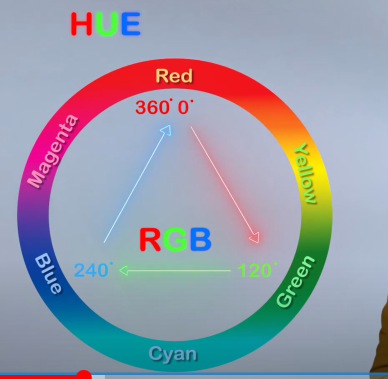
***Inheritance:***

Properties applied on parent to child , fetching data and properties of one parent to another child like fetching same color from some element to other elements etc.

Padding is used to give spaces from every side.

Every para have the same distance for instance 10 px from each other.

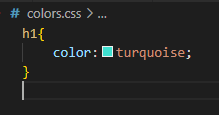
***Science of colors:***



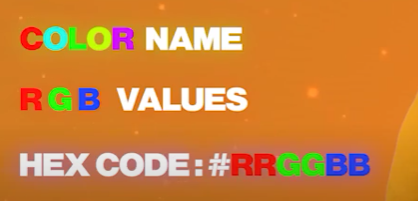
***Color property:***

There are three ways to specify color of elements:

* Give color name
* Rgb
* Hexcode
* Color names



…………………………………………………………….x…………………..x……………………….



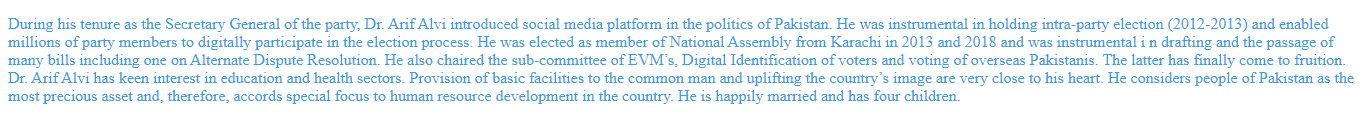
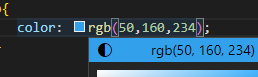
147 predefined color names.

RGB:

Mixing three colors to make new colors of our choice.

Graphical user interface, application

Description automatically generated



Hex code :

6 digit code. 

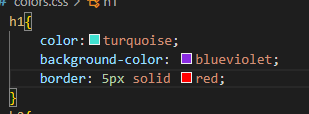


Graphical user interface, text

Description automatically generated

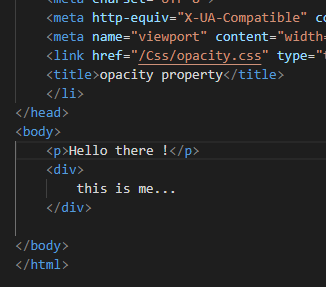


Working on border and background color:





Transparency how solid an element img or text looks.

Text

Description automatically generated

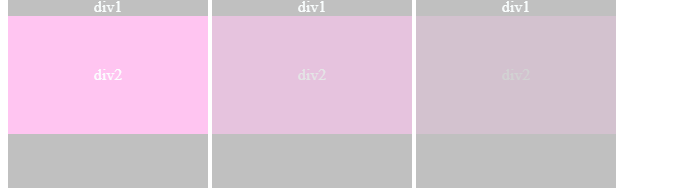
A picture containing background pattern

Description automatically generated

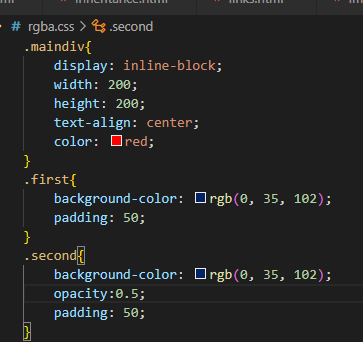
Example 02 :

Text

Description automatically generated



***RGBA:***

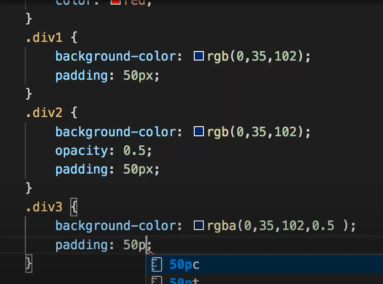
****** ***Chart

Description automatically generated with low confidence***

Since the child of the div1 which is div2 is also makes itself transparent although we just want background to be transparent.

For that see below code:

For that we use alpha property.

 A picture containing chart

Description automatically generated

***Padding:***

Highlighting area radius:



A picture containing graphical user interface

Description automatically generated

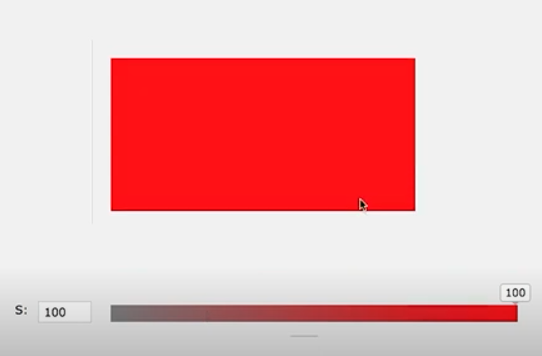
***HSL HSLA:***

Hsl = hue saturation lightness

Hsla = hue saturation lightness alpha

Alpha = fully transparent , fully opaque.

Hue can also we be shown as a slider.



A picture containing graphical user interface

Description automatically generated

Hui 0 = fully transparent

Hui 1 = fully opaque.

Alpha = opacity.

H = transperancy

S = color change

L = dark or light color

A = visibility

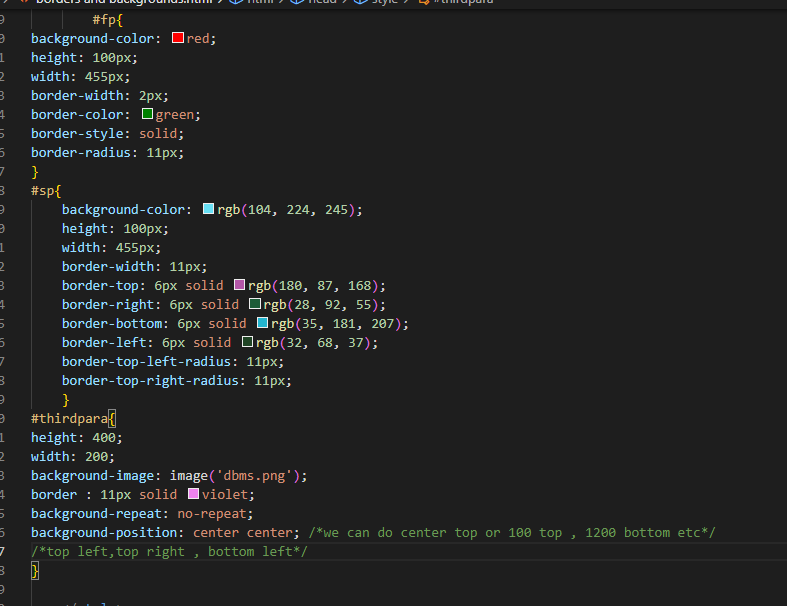
***Borders and backgrounds:***

Border radius = round border edges

Border width = thicker or thiner the boders from all four sides.

Border style = transparent , solid etc.

Example 1:



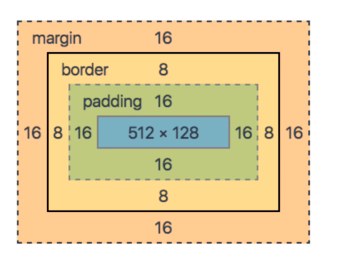
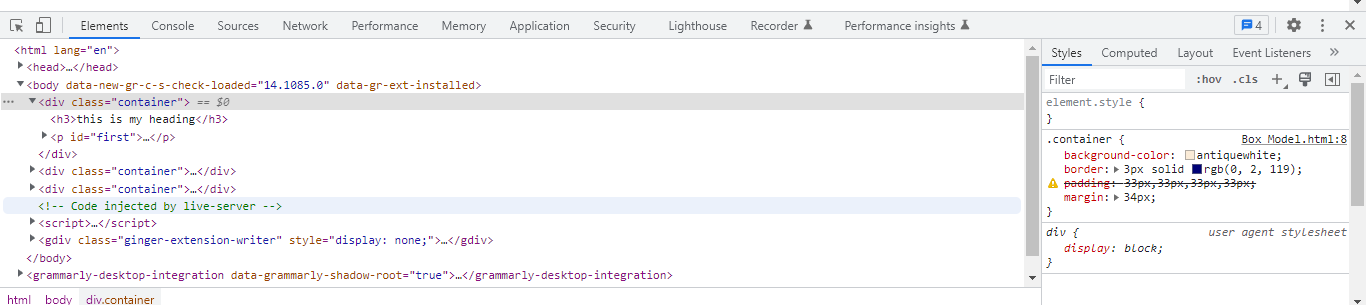
Output:

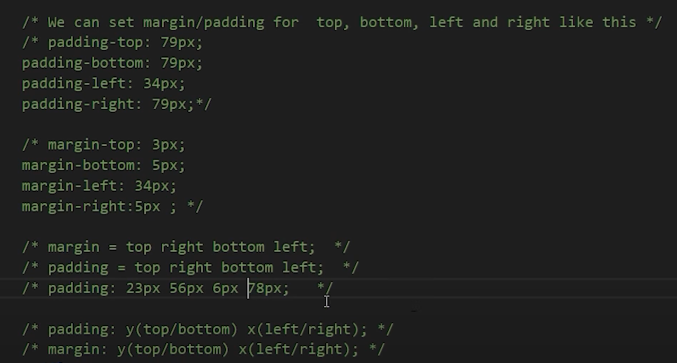
Graphical user interface

Description automatically generated with low confidence

***Box model margin padding:***

Box model

 Text

Description automatically generated Text

Description automatically generated