1. How to write CSS code

* CSS code inside an HTML tag.

Ex : <p style=”font-size: 120%”>

* CSS code inside the html document  
  <head>

<style>

P{

font-size:120%

}

</style>

</head>

* CSS code in an external file

1. How do you link html and css

* We use the <link> tag in the head section.
* <link rel="stylesheet" type="text/css" href='./style.css'>
* We tell that we want to link a style sheet and it’s a CSS document and its location.

1. CSS for text formatting.

* CSS is made up of rules
* Each rules contains a ‘selector(h1,p,h2 img etc…..)’ and a ‘declarative ({….})’ block
* h1{
* font-size: 2em;
* color: green;
* }
* For the h1 selector we apply the font-size and color properties with their respective values. Notice the semi colon.

1. Grouping selectors + text-align

* h1,h2 {
* color: green;
* font-family: Arial, Helvetica, sans-serif;
* }
* h1{
* font-size: 40px;
* }
* h2{
* font-size: 20px;
* }
* p {
* font-size: 3em;
* text-align: justify
* }
* text-align : justify means it stretches the lines so that each line has equal width like in news papers.
* text-align : left means that text will be aligned to the left (margin ?)

1. An html Anchor tag.

* <a href="https://www.pluralsight.com" target="\_blank">plural sight</a>

1. Global rule : as everything is inside a body this rule applies to everything inside body.

Inheritance : child elements inherit their properties from their parent elements, unless we override them

* body{
* font-family: Arial, Helvetica, sans-serif;
* }

1. COLORS

* colors are defined in the color table using the hexadecimal notation.
* R - ff0000
* G – 00ff00
* B – 0000ff

ff in hexadecimal is 255.

Yellow = REG + GREEN

* Yellow – ffff00

For Grey color : 8b8b8b R,G,B will have same values

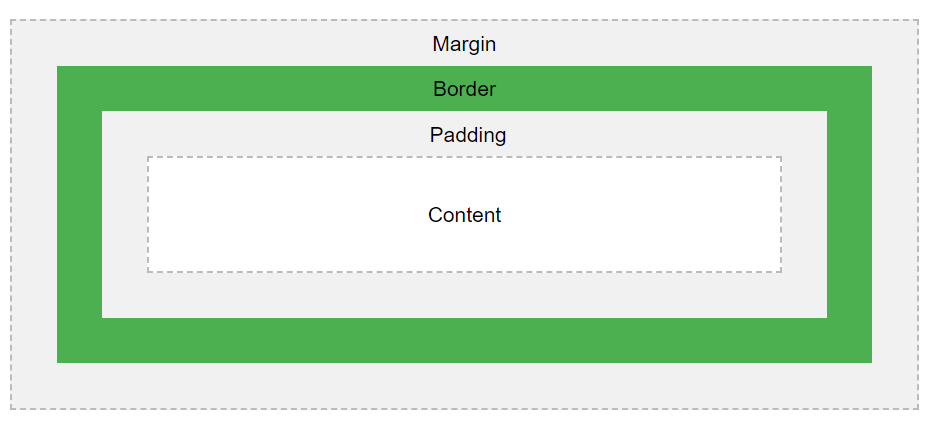
CSS colors have transparencies.

* In this case we do not use the hexadecimal notation
* Rgba(29,167,23,0.75) : 75 percent transparency

1. NEW SELECTORS : classes and ids

* We can attribute class or id to html elements. And we can use the same to select them in the css code.
* ID can be used only once in a html document but class can be used multiple times.
* We select the **class** using the dot . in css
* We select the **id** using the hash # in css.
* <p class="main-text">.....</p>
* <p id="author-text">T....
* .main-text{….}
* #author-txt {…}
* Using classes is a good idea than ID – as this can be used only once.

1. BOX MODEL

* All html elements can be seen as boxes.
* With the help of these you can define space between elements and add border around the elements.
* 
* Content: it is where text and images appear.
* **Padding**: it like a transparent area around the content, but inside of the box.
* **Border**: border goes around the content and padding, may be transparent or not.
* **Margin**: it is the space between boxes.
* Padding , border, margin are css properties and can be specified for entire box or individual sides – top bottom left right.
* We can set the height and width of an element can be set using the box model.
* Note that we can set the height and width of the content – not of the entire box itself.
* This means that padding margin and border will be added to the height we specify- not good. As it will hard to add padding margin and border to your imagination.
* Solution : **box-sizing** : border-box if this CSS property is set 🡪 we can define the height and width for an entire box rather than just the content.

1. **BLOCK** AND **INLINE** elements

* Block elements: block elements (LIKE A STACK OF BOXES) always start on a new line and occupy the entire space of its parent element. Heading and paragraphs
* Inline elements: images, links, strong, and em elements don’t do any of that. You can neither set their height or width.

1. \* selector : all elements are affected.

* \* {
* margin:0;
* padding: 0;
* box-sizing: border-box;
* }
* /\*if you want to add some space below a header h1 and a succeeding paragraph\*/
* h1{
* margin-bottom: 20px;
* }

1. DIV.

* Using this we divide our page into sections by creating boxes where we put our contents in.

1. How to center some content to the page

* {  
  left-margin : auto;

right-margin : auto;  
width:1140px;  
}  
The auto means – the left and right margin of the element are adjusted automatically according to the context of the element – which is the browser window.

1. FLOAT

* Put elements side by side
* With float an element can be pushed to the left or to the right.
* The float property specifies how an element should float.
* This float property is used for positioning and formatting content Ex : Float an image ‘Left’ to the text in a container, Or wrap text around an image

1. CLEAR

* The clear property specifies what elements ‘can float’ ‘beside the cleared element’ ‘and on which side’
* The most common way to use a clear property is after you have used a float property on an element.
* If an element is floated to the left, then you should clear to the left. Your floated element will continue to float, but the cleared element will appear below it on the web page.

1. Select html elements under a class [instead of assigning them another id or class]

* .author-box img {
* width : 180px;
* height : 200px;
* border-radius: 50%;
* }
* HTML CODE IS BELOW
* <div class='author-box'>
* <img src="./rowling.jpg" alt="JK ROWLING ">
* <p class="author-text">JK ROWLING</p>
* </div>

1. While adjusting images we say   
   height : your pixels  
   width : auto   
   In this case the image will be automatically scaled appropriately.
2. When you use margin , padding , border.---------?
3. **Element Positioning / Layout. [Absolute / relative / fixed / sticky / static.]**Relative : Their position on the webpage is determined by other elements

Absolute : These elements can be positioned anywhere inside their parent elements.

Parent Element :   
.blog-post {

**position : relative;**

}

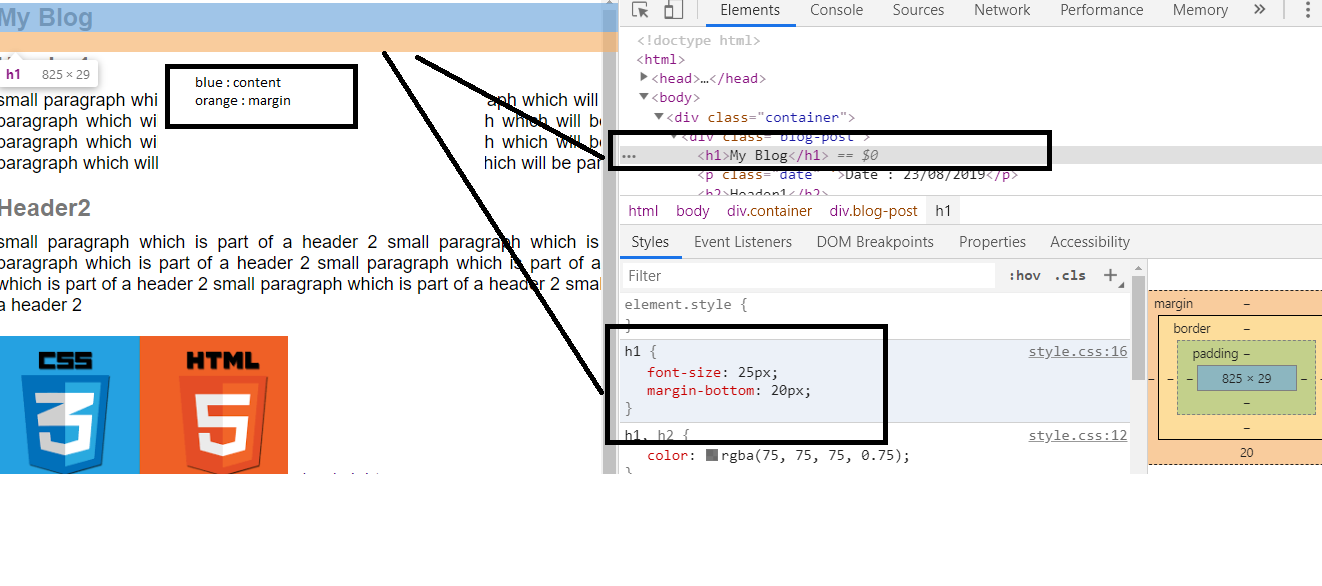
----------------------  
Child Element :   
.date {

position: absolute;

top : 10px;

right:30px;

}

1. Chrome developer tools  
   
2. TYPOGRAPHY  
   body text : 15-25 pixels

Head lines : 60 pixels/90 pixels ..depends. but use less font weight.

Line spacing : 120-150 % of font size.

No of characters in a line : 70-80  
good fonts : san serif : they look clean and professional

Serif : traditional and story telling  
google fonts : lato.

1. COLORS  
   - use any one single base color i.e any different color other than black,white,or some shade of grey.

* Stick to one main color through out the website.
* <https://flatuicolors.com/> online resource.
* <https://www.0to255.com/> for color palette
* Don’t use black in the design
* Which colors ?   
  RED : Power, passion,strength,excitement  
  orange : draws attention : cheerfulness,confidence,courage,creativity.  
  yellow : happiness and loviness,- curiosity brightness,intelligence

Green : harmony,health,nature,money – balancing effect

Blue : patience,peace,trustworthiness,stability,professionalism , trust , honor.

1. IMAGES  
   - learn how to put text over images.

* You can put text –directly on an image : the problem is image should be dark and text should be light .. else you will not have enough contrast.
* Overlay the image with another colors
* Put a text on a box..which is opaque..white color wth transparency
* Blur the image
* Floor fade

1. ICONS  
   -