

Class 3 - intro to CSS

CSS

1. Recall that html is the structure and css is the style or design that we apply to the structure
2. CSS stands for cascading style sheets
3. CSS is a rule-based language — you define the rules by specifying groups of styles that should be applied to particular elements or groups of elements on your web page.
- 4.
5. For example, you can decide to have the main heading on your page to be shown as large red text. The following code shows a very simple CSS rule that would achieve the styling described above:
- 6.
7. Create index.html in class-3 folder

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <h1>I am heading 1</h1>
```

```
</body>  
</html>
```

8. There are multiple ways to include css in html document
9. Lets start by first discussing what is called as internal styles.
 - a. Here it means that css is part of the same html document
10. Add the style tag inside head tag

```
<style>  
    h1{  
        color: red;  
        font-size:32px;  
    }  
</style>  
</head>
```

11. In the above example, the CSS rule opens with a selector. This selects the HTML element that we are going to style. In this case, we are styling level one headings (h1).
12. We then have a set of curly braces { }.
13. Inside the braces will be one or more declarations, which take the form of property and value pairs. We specify the property (color in the above example) before the colon, and we specify the value of the property after the colon (red in this example).
14. This example contains two declarations, one for color and the other for font-size. Each pair specifies a property of the element(s) we are selecting (h1 in this case), then a value that we'd like to give the property.
15. What we see here is an example of **element selector**
 - a. They target HTML elements and apply styles to them
 - b. when we use an element selector, we are essentially telling the browser: Apply these styles to every instance of this particular HTML element on the page.

Different ways to add css

1. We saw the internal style declarations
2. Lets learn another way to add css
 - c. Every html element has this style attribute where we can define the style for that element

d. `<h2 style="color:blue">I am heading 2</h2>`

e. Here the styling is in the same line as the element. Hence this is known as inline style

16. The third way to add css is called external style

a. If internal styling has css in the same file, where will the external style code be

b. In another file with .css extension

c. Add a h3 tag

```
<h3>I am heading 3 with external styling</h3>
```

d. In style.css, add below

```
h3{  
    color:blueviolet  
}
```

e. nothing happens just with this setup

f. Nothing happens because we are not telling our browser where this external css is for our html

g. So in our html we need to link this external css that we created

h. `<link rel="stylesheet" href="./style.css">`

i. `</head>`

j. Link tag

i. The `<link>` tag is used to define a relationship between an HTML document and an external resource. It's

most commonly used to link external style sheets, like CSS files.

- ii. rel stands for relationship and specifies the type of relationship that the document has with the linked resource.
- iii. In our example, `<link rel="stylesheet" href="./style.css">`, the rel attribute has the value stylesheet, indicating that the linked resource (style.css) is a style sheet that should be applied to the HTML document.

User-agent-stylesheet

1. Browsers themselves have certain predefined styles for some of the html elements
2. Your anchor tags, your para and other tags look a certain way
3. There is slight variation in how checkboxes and focus rings look in one browser than other
4. https://source.chromium.org/chromium/chromium/src/+/main:third_party/blink/renderer/core/html/resources/html.css

Class selector in css

1. We have styling instructions that we create for a class and then apply or add that class to any element

```
<h3 class="blue">A blue heading</h3>
  <p class="blue">a blue para</p>
  <p >a default para</p>
  <p class="blue">a blue para</p>
  <h3 class="red">A red heading</h3>
  <p class="red">a red para</p>
  <p >a default para</p>
  <p class="red">a red para</p>
  <p class="blue">a blue para</p>
```

2. Define css

```
.blue{
    color: blue;
}
.red{
    color: red;
}
```

3. If we want that our red color heading should also yellow background color

```
<style>
  h1{
    color: red;
  }
  .blue{
    color: blue;
  }
  .red{
    color: red;
  }
  .yellowBackground{
```

```
        background-color: yellow;
    }
</style>
```

4. Add one more class to the red para

```
<h3 class="red yellow">A red heading</h3>
```

ID in html / css

1. We sometimes need to style one particular element differently than others
2. There we can leverage the ids and apply specific styles

```
<p class="red" id="splPara">a red para with some spl  
style</p>
```

```
#splPara{
    color:lightcoral;
    font-size: 30px;
}
```

Conflicting properties - which one gets applied

1. Like if we have the same style from the external style sheet and internal and maybe also inline then which one takes precedence
2. Or if we have the same rule in the class and also in the id - which one gets applied
3. There is an order to how this is evaluated. We will take this in one of the coming class

Quiz 1 - Which CSS selector is used to select all elements with the class "highlight"?

```
#highlight  
highlight  
*highlight*  
.highlight
```

CSS selectors questions

1. add css styles to all h1 tags making it blue

```
h1{  
    color : blue  
}
```

2. Descendant selectors

- a. Anything down the order can be termed as a descendent.
For example, a child is the descendent of their Father as well as their GrandFather.
- b. In the second question, the task is to select the li with Select and Me text
- c. Bring out the difference with the above lis where they are directly under body and this one is under div
- d. The idea is in order to apply styles selectively, you have to figure out a rule which identifies the tags that you wish to select
- e. Here the difference is that they are present under a div

- f. a descendant selector in CSS allows you to target an element that is nested inside another specific element, regardless of the level of nesting.
- g. Understand that it's not just about direct child elements but any descendant in the hierarchy.

```
<style>
    /* descendant selector */

    div li{
        color: blue;
    }

</style>
```

- h. So the above way of writing style is useful when there is some difference in the structure that you can capitalise to apply styling selectively
- i. When you select `ol li` as the selector then all the elements will get blue.
- j. below also works

```
div ol li{
    color: blue;
}
```

- k. There can be multiple ways to solve a problem in html css . this is one way

3. Child selectors

1. Descendants can be anyone in the lineage. a child can be descendant but not every descendant will be child
2. Open q3.html
3. what rule we can write to identify the direct son

```
<body>
  <!-- select "I'm the direct son" and make it blue -->

  <section>
    <h1>
      <span>I'm the direct son</span>
    </h1>
  </section>

  <section>
    <div>
      <h1>
        <span>I'm the indirect son</span>
      </h1>
    </div>
  </section>
</body>
```

4. with descendant selector it will not work
5. We need something that tells that pick the immediate descendant or the child of the element
6. So we have the child selectors

```
/* children selector */
section > h1 > span {
```

```
color: blue;

}
```

4. Class selectors

1. Q4.html our goal is to style p tags differently

```
<body>
  <!-- Select the obvious p tags and make class 1 blue and class 2 red -->
  <!-- <p>Random</p>
  <p>Random</p>
  <p>Random</p>
  <p>Random</p>
  <p>Random</p>
  <p>Random</p>
  <p>Random</p>
  <p>Random</p> -->

  <p class="class1">Make me Blue</p>
  <p class="class2">Make me Red</p>

</body>
```

2. With the help of class attribute we can write rules to select each one and style them as per our goal

3. How do we select elements with a particular class

```
.class1{
    color: blue;
}

.class2{
    color: red;
}
```

a. Let us create one more class with a background color

```
<p class="class1">Make me Blue</p>
```

```
<p class="class2 class3">Make me Red</p>
```

b. Add css rule

```
.class3{  
    background-color: yellow;  
}
```

5. In this question, we need to write a selector to select an element which has two class on it

```
.m1.m2{  
    color:blue  
}
```

6. Q6-> Here we need to select a div

```
<body>  
  <!-- Select "I'm here. find me" and make it blue -->  
  
  <div class="c1">  
    <div>  
      <div>  
        <span>  
          <div class="c2">I'm here. find me</div>  
        </span>  
      </div>  
    </div>  
  </div>  
  
  <div>  
    <div class="c2">Dont select me</div>  
  </div>
```

```
</body>
```

- a. Here the difference lies that what we want to select has a class c2 which is inside another class c1
- b. This is descendant like scenario

```
<style>
  .c1 .c2{
    color: blue;
  }
</style>
```

7. Q7 Here we need to write a selector for p which is a direct child of a div with class c1

```
<style>
  /* class with children combinator */
  .c1>p{
    color: blue;
  }
</style>
```

8. Id selector

```
<style>
  #the-one{
    color: blue;
  }
</style>
```

9. We can target any attributes and write selectors

```
input[value='Select me']{  
    background-color: blue;  
}
```

- a. What if we create an input text field and want to select it

```
<input type="text" value="Click Me">
```

```
input[type='text']{  
    background-color: yellow;  
}
```

Pseudo classes

1. pseudo-classes are a special kind of selector in CSS.
2. They allow you to define a special state of an element. For example, styling an element when it is hovered over or when it has focus.
3. :hover
 - a. Activates when the user hovers their mouse over an element.
 - b. Example: Change the color of a button when the mouse is over it.

```
button{  
    background-color: blueviolet;  
}
```

```
button:hover{  
    background-color: lightblue;  
}
```

```
a:hover{  
    color : blueviolet  
}
```

A brief Intoduction to Basic CSS properties on background , text and fonts

Background color in CSS.

1. It works the same as the color.
2. You can add CSS code for background color as :

```
<style>  
    .container {  
        height: 50vh;  
        width: 50vw;  
        background-color: lightpink;  
    }  
</style>  
</head>  
<body>  
    <div class="container">  
        <h1>I am a heading</h1>  
        <p >  
            Lorem, ipsum dolor sit amet consectetur adipisicing elit. Corporis,  
            eaque?  
        </p>  
    </div>
```

```
</body>
```

3. Background image

4. Search for background image -> copy image address

```
.container {  
    height: 50vh;  
    width: 50vw;  
    background-color: lightpink;  
    background-image: url("");  
}
```

5. Discuss other background properties

```
background-size: 800px;  
background-repeat: no-repeat;
```

Font Family

1. What is Font-Family?

- a. font-family is a CSS property used to specify the typeface for text in an HTML element.
- b. It defines how the text will look in terms of its font style.

2. The font-family property can hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font.

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


Font weight

1. Font weight
2. This font property is used to decide the intensity of the font.
3. It's value ranges from lightest 100 to boldest 800.
- 4.

```
p{  
  
    font-weight: 800;  
}
```

Font size

1. It refers to the size of the text.

```
p{  
  
    font-weight: 800;  
    font-size: x-small;  
}
```

2. We can give fixed values
3. We can give relative values

```
<style>  
    .container {  
        height: 50vh;  
        width: 50vw;  
        background-color: lightpink;  
        background-image: url("https://source.unsplash.com/random/800x800");  
        /* background-size: 800px; */  
        background-repeat: no-repeat;  
        font-size: 32px;  
    }
```

```

    }
    body {
        font-family: Arial, Helvetica, sans-serif;
    }
    p {
        font-weight: 800;
        font-size: 50%;
    }

```

Google Font

1. If you are not unable to get the font requirement from the above font properties, then you can use the Google font option.
2. You just need to go to the Google search and search "google font". Refer website: <https://fonts.google.com/>.
3. Search for fonts and select 400
4. Add it in the head tag

```

<link rel="preconnect" href="https://fonts.googleapis.com"
/>
    <link rel="preconnect" href="https://fonts.gstatic.com"
crossorigin />
    <link
href="https://fonts.googleapis.com/css2?family=Protest+Revo
lution&display=swap"
rel="stylesheet"
/>

```

5. Change the font family

```

body {

```

```
/* font-family: Arial, Helvetica, sans-serif; */  
font-family: 'Protest Revolution', sans-serif;  
}
```

Text-align

1. For divs, paras, headings, if we want to align the text inside them
2. It specifies how inline content like text is aligned within an element.

```
.container {  
    height: 50vh;  
    width: 50vw;  
    background-color: lightpink;  
    background-image:  
url("https://source.unsplash.com/random/800x800");  
    /* background-size: 800px; */  
    background-repeat: no-repeat;  
    font-size: 32px;  
    text-align: center;  
}
```

Text Decoration

1. It is used to make your text attractive using some properties.
2. These are oval-line, line-through, underline, etc.

Word Spacing

1. It defines how many spaces are there between any two consecutive words

2. `word-spacing : 10px;`

3.

Line Height

1. It defines the height between two consecutive lines.

2. `line-height : 50px;`

3.

Colors (self read)

rgb

1. The color set has some finite set of colors to choose from
2. We can create colors and shades of our own

3. What are the three primary colors using which we can derive many other colors - red, green , blue
4. Create h1, h2, h3 headings
5. For h2 add blue color and then use rgb

```
h2{  
    /* color: blue; */  
    color:rgb(0, 0, 255)  
}
```

6. Why 255 ? 8 bits range from 0 to 255

Hex codes

1. Hexadecimal color codes in web design provide a way to specify colors using a hexadecimal (base-16) format
2. Hexadecimal is a base-16 number system. It uses 16 symbols: the numbers 0 to 9 and the letters A to F (or a to f).
3. A hex color code typically starts with a hash symbol (#) followed by six hexadecimal digits. These six digits are actually three two-digit pairs, corresponding to the Red, Green, and Blue components.
4. Each pair of digits can range from 00 to FF in hexadecimal, which is equivalent to 0 to 255 in decimal. This represents the intensity of the color component.
5. Lets take the color red
6. Search for hex code of red - #FF0000
7. Hex code for green - #00FF00

8. Hex code for blue - #0000FF
9. Hex code for white - #FFFFFF
10. Color palette generator - <https://coolors.co/>

HSL

1. Hue:
 - a. hue represents the color itself, the base pigment without any tint or shade.
 - b. Hue is represented as a degree from 0 to 360. For example, 0 is red, 120 is green, 240 is blue.
2. Saturation:
 - a. Saturation refers to the intensity or purity of the color.
 - b. Understand it in terms of water added to paint - less water (higher saturation) means the color is more pure and vivid, while more water (lower saturation) makes it more washed out.
 - c. It's represented as a percentage, where 0% is a shade of gray, and 100% is the full color.
3. Lightness:
 - a. Lightness (or luminance) is about how light or dark the color is.
 - b. A lightness of 0% is black, 50% is the pure color (neither shaded nor tinted), and 100% is white.

```
h1{  
    color:hsl(240, 100%, 50%)  
}
```

- c. This CSS rule sets the color of the element with the h1 tag to a bright blue. The 240 defines the hue (blue), 100% for saturation (full color), and 50% for lightness (neither dark nor light, just the pure hue).
- d. If we want a lighter shade of the same blue, we can increase the lightness:
- e. If we want a less vibrant, more desaturated blue:

```
.desaturated-blue {  
    color: hsl(240, 50%, 50%);  
}
```

- f.