Front End Technologies CSS - Day 12

Agenda

- **SPECIFICITY**
- **INHERITANCE**
- **CASCADE**

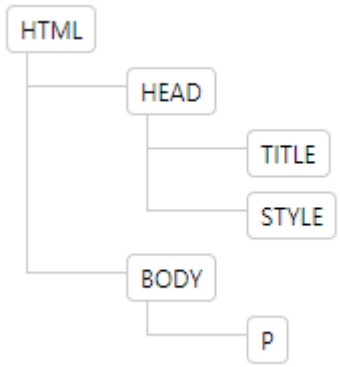
SPECIFIFICITY: Specificity is the means by which browsers decide which CSS property values are the most relevant to an element and, therefore, will be applied. That mean When more than one set of CSS rules apply to the same element, the browser will have to decide which specific set will be applied to the element. The rules the browser follows are collectively called Specificity.

Example:

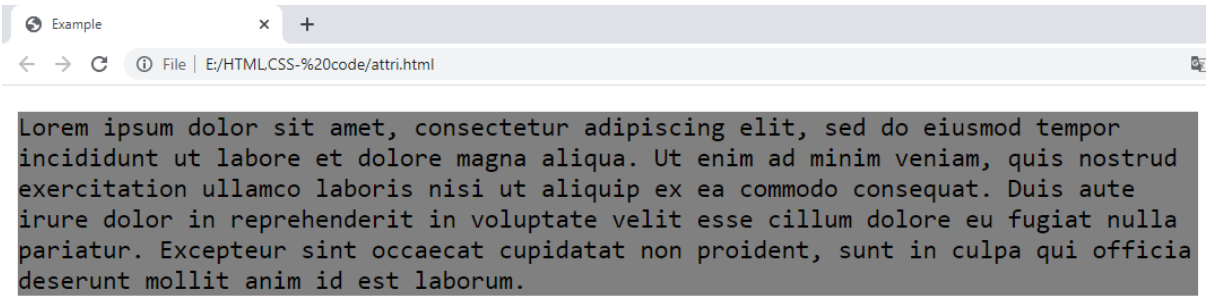
Index.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Example</title>
    <style type="text/css">
        p{
            background-color: orange;
            font-family: cursive;
        }
        p.para1{
            background-color: grey;
            font-family: monospace;
        }
        body p{
            background-color: cyan;
            font-family: sans-serif;
        }
    </style>
</head>
<body>
    <p class="para1">Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
    aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure
    dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident,
    sunt in culpa qui officia deserunt mollit anim id est laborum.</p>
</body>
</html>
```

Document Object Model:



Output:



In this example, if you observe the code and output 3 styling is applied to one p element but in the output only one styling is reflected based on specificity calaculator shown below.

Specificity Calculator.

Selector	Specificity		
	ID's	Classes	elements
p	0	0	1
p.para1	0	1	1
Body p	0	0	2

If you observe the specificity calculator, the selector containing only p element has a specificity 001, the selector containing p element and

class has a specificity 011 and selector containing 2 elements has a specificity 002. Specificity values are sorted from left to right thus the specificity for the above example is 011 and styling given to that selector is applied.

Example:

Index.html

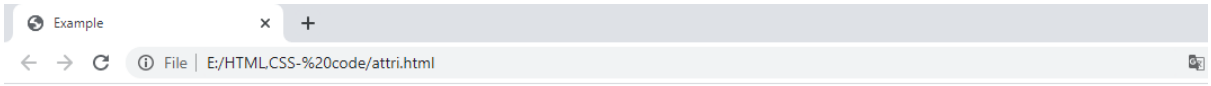
```
<!DOCTYPE html>
<html>
<head>
    <title>Example</title>
    <style type="text/css">
        p{
            background-color: orange;
            font-family: cursive;
        }
        p.para1{
            background-color: grey;
            font-family: monospace;
        }
        body p{
            background-color: cyan;
            font-family: sans-serif;
        }
        p#p1{
            background-color: violet;
            font-family: fantasy;
        }
    </style>
</head>
<body>
    <p id="p1" class="para1">Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore
    magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute
    irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non
    proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>
</body>
</html>
```

Specificity Calculator.

Selector	Specificity		
	ID's	Classes	elements
P	0	0	1
p.para1	0	1	1
Body p	0	0	2
P#p1	1	0	1

In this example, if you observe the specificity calculator from left to right all the selector expect the last selector id is 0. So specificity for this example is 101 and styling given to that selector is applied.

Output:



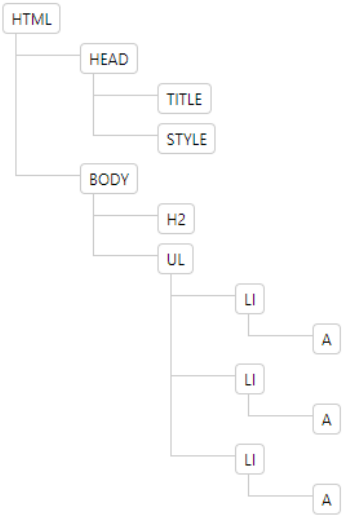
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Example:

index.html

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <style type="text/css">
        ul li a[href]{
            background-color: orange;
        }
        li a[href ^= "https"]:first-child{
            background-color: violet;
        }
    </style>
</head>
<body>
    <h2>Best website for front-end development</h2>
    <ul>
        <li>
            <a href="https://www.w3schools.com">W3schools</a>
        </li>
        <li>
            <a href="https://www.codepen.io">Codepen</a>
        </li>
        <li>
            <a href="https://www.cssbattle.dev">Cssbattle</a>
        </li>
    </ul>
</body>
</html>
```

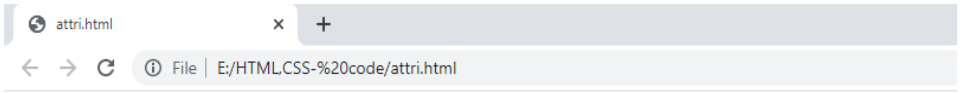
Document Object Model:



Specificity Calculator.

Selector	Specificity		
	ID's	Classes and attribute	elements
ul li a[href]	0	1	3
li a[href ^= "https"]:first-child	0	2	2

Output:



Best website for front-end development

- W3schools
- Codepen
- Cssbattle

In this example, if you observe the specificity calculator from left to right second selector has 2 attributes. So specificity for this example is 022 and styling given to that selector is applied.

Example:

Index.html

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <style type="text/css">
        ul li a[href]{
            background-color: orange;
        }
        li a[href ^= "https"]:first-child{
            background-color: violet;
        }
        body ul > li > a{
            background-color: cyan;
        }
    </style>
</head>
<body>
    <h2>Best website for front-end development</h2>
    <ul>
        <li>
            <a href="https://www.w3schools.com">W3schools</a>
        </li>
        <li>
            <a href="https://www.codepen.io">Codepen</a>
        </li>
        <li>
            <a href="https://www.cssbattle.dev">Cssbattle</a>
        </li>
    </ul>
</body>
</html>
```

Specificity Calculator.

Selector	Specificity		
	ID's	Classes and attribute	elements
ul li a[href]	0	1	3
li a[href ^= "https"]:first-child	0	2	2
body ul > li > a	0	0	4

Output:

Example: when Specificity of the selectors are same

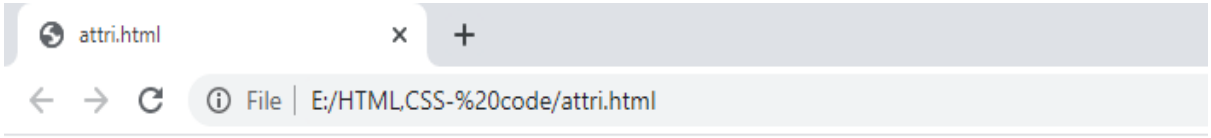
Index.html

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <style type="text/css">
        a.firsta{
            background-color: orange;
        }
        a[href="https://www.w3schools.com"]{
            background-color: cyan;
        }
    </style>
</head>
<body>
    <h2>Best website for front-end development</h2>
    <ul>
        <li>
            <a class="firsta" href="https://www.w3schools.com">W3schools</a>
        </li>
        <li>
            <a href="https://www.codepen.io">Codepen</a>
        </li>
        <li>
            <a href="https://www.cssbattle.dev">Cssbattle</a>
        </li>
    </ul>
</body>
</html>
```

Specificity Calculator.

Selector	Specificity		
	ID's	Classes and attribute	elements
a.firsta	0	1	1
a[href="https://www.w3schools.com"]	0	1	1

Output:



Best website for front-end development

- W3schools
- Codepen
- Cssbattle

Here, the specificity of both the selctor are same so the selctor which is at the last is given priority and styling given to that selector is applied.

Example: Calculating specificity with inline css

Index.html

```html
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <style type="text/css">
        a.firsta{
            background-color: orange;
        }
        a[href="https://www.w3schools.com"]{
            background-color: cyan;
        }
    </style>
</head>
<body>
    <h2>Best website for front-end development</h2>
    <ul>
        <li>
            <a class="firsta" href="https://www.w3schools.com" style="background-color: violet;">W3schools</a>
        </li>
        <li>
            <a href="https://www.codepen.io">Codepen</a>
        </li>
        <li>
            <a href="https://www.cssbattle.dev">Cssbattle</a>
        </li>
    </ul>
</body>
</html>
```

Specificity Calculator.

Selector	Specificity			
	Inline	ID's	Classes and attribute	elements
a.firsta	0	0	1	1
a[href="https://www.w3schools.com"]	0	0	1	1
	1	0	0	0

Output:

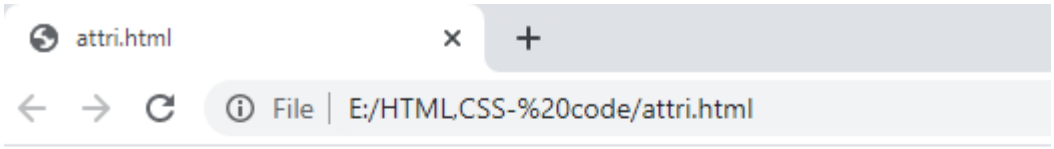
In this example, inline css is used whenever there is an inline css priority is given to that styling given to that is applied.

Example:

Index.html

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <style type="text/css">
        a.firsta{
            background-color: orange;
        }
        a[href="https://www.w3schools.com"]{
            background-color: cyan;
        }
        a{
            background-color: red !important;
        }
    </style>
</head>
<body>
    <h2>Best website for front-end development</h2>
    <ul>
        <li>
            <a class="firsta" href="https://www.w3schools.com" style="background-color: violet;">W3schools</a>
        </li>
        <li>
            <a href="https://www.codepen.io">Codepen</a>
        </li>
        <li>
            <a href="https://www.cssbattle.dev">Cssbattle</a>
        </li>
    </ul>
</body>
</html>
```

Output:



Best website for front-end development

- W3schools
- Codepen
- Cssbattle

Whenever you want to apply styling inspite whatever the specifity the selector have you must mention it as **!important.** The moment you mention that selctor has !important as shown in the example then styling given to that selector is applied.

Example: Multiple important selector

Index.html

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <style type="text/css">
        a.firsta{
            background-color: orange;
        }
        a[href="https://www.w3schools.com"]{
            background-color: cyan;
        }
        a{
            background-color: red !important;
        }
        a{
            background-color: grey !important;
        }
    </style>
</head>
<body>
    <h2>Best website for front-end development</h2>
    <ul>
        <li>
            <a class="firsta" href="https://www.w3schools.com" style="background-color: violet;">W3schools</a>
        </li>
        <li>
            <a href="https://www.codepen.io">Codepen</a>
        </li>
        <li>
            <a href="https://www.cssbattle.dev">Cssbattle</a>
        </li>
    </ul>
</body>
</html>
```

Specificity Calculator.

Selector	Specificity		
	ID's	Classes and attribute	elements
a	0	0	1
a	0	0	1

In this example, both selectors are important and both have specificity thus the selector which is at the last is selected.

Output: