

1. CSS (Cascading Style Sheets)의 등장

일반적인 CSS 적용시 하드코딩은 유지보수에 불합리하다.

```
<h1><a href="index.html"><font color="red">WEB</font></a></h1>
<ol>
  <li><a href="1.html"><font color="red">HTML</font></a></li>
  <li><a href="2.html"><font color="red">CSS</font></a></li>
  <li><a href="3.html"><font color="red">Javascript</font></a></li>
</ol>
```

1) 일괄 적용

<style> 태그안에서 CSS 를 작성한다.

<style> 태그안에서 a { } 라고 한다면,

선택자(selector)인 a 는 모든 <a> 태그를 의미한다.

선언(Declaration)인 color:red; 는 선택자에게 적용될 기능을 의미한다.

속성(Property)는 color 에 해당하고,

값(Value)는 red가 해당된다.

```
<style>
  a {
    color:red;
  }
</style>
<body>
  <h1><a href="index.html">WEB</a></h1>
  <ol>
    <li><a href="1.html">HTML</a></li>
    <li><a href="2.html">CSS</a></li>
    <li><a href="3.html">JavaScript</a></li>
  </ol>
```

2) 개별 적용

필요한 부분에 직접 style="color:red" 작성해서 사용

```
<h1><a href="index.html">WEB</a></h1>
<ol>
  <li><a href="1.html">HTML</a></li>
  <li><a href="2.html" style="color:red">CSS</a></li>
  <li><a href="3.html">JavaScript</a></li>
</ol>
```

3) 정리

```
<!doctype html>
<html>
<head>
  <title>WEB1 - JavaScript</title>
  <meta charset="utf-8">
</head>
<style>
  a {
    color:blue;
  }
</style>
<body>
  <h1><a href="index.html">WEB</a></h1>
  <ol>
    <li><a href="1.html">HTML</a></li>
    <li><a href="2.html" style="color:red">CSS</a></li>
    <li><a href="3.html">JavaScript</a></li>
  </ol>
  <h2>JavaScript</h2>
  <p>
    JavaScript
  </p>
</body>
</html>
```

WEB

1. [HTML](#)
2. [CSS](#)
3. [JavaScript](#)

JavaScript

JavaScript

2. CSS 의 속성(Property) 은 개별 공부로

<https://developer.mozilla.org/ko/docs/Web/CSS/Reference>

<https://www.w3schools.com/cssref/index.php>



3. CSS 의 선택자(selector)

```
<!doctype html>
<html>
<head>
  <title>WEB1 - JavaScript</title>
  <meta charset="utf-8">
</head>
<style>
  a {
    color:black;
    text-decoration: none;
  }
  .saw {
    color:gray;
  }
  .active{
    color: red;
  }
  h1 {
    font-size: 45px;
    text-align: center;
  }
</style>
<body>
  <h1><a href="index.html">WEB</a></h1>
  <ol>
    <li><a href="1.html" class="saw">HTML</a></li>
    <li><a href="2.html" class="saw active">CSS</a></li>
    <li><a href="3.html">JavaScript</a></li>
  </ol>
  <h2>JavaScript</h2>
  <p>JavaScript</p>
</body></html>
```



1) 태그 선택자 : 가장 기본적인 선택자로 태그명 { } 의 형태를 가진다. ex) a { }

2) class 선택자 : . dot 은 class 를 의미하며 .className { } 형태이다. ex) .saw { } , .active { }
.saw 는 **class="saw"** 가 적용된 모든 곳에 프로퍼티와 벨류인 **color:gray;** 를 적용한다.

2-1) 다중 class 선택자 가능 : class 는 여러개를 적용할 수 있다. 띄워쓰기로 구분된다.

```
class="saw active"
```

Cascading의 특징상 .saw 보다 후술된 .active가 우선순위가 높다.

3) id 선택자 : #idName{ }

id는 가장 높은 우선순위를 가진다.

.saw 보다 앞에 작성되었지만 id로 선언되면 최우선순위다.

```
<!doctype html>
<html>
<head>
  <title>WEB1 - JavaScript</title>
  <meta charset="utf-8">
</head>
<style>
  a {
    color:black;
    text-decoration: none;
  }
  #active{
    color: red;
  }
  .saw {
    color:gray;
  }
  h1 {
    font-size: 45px;
    text-align: center;
  }
</style>
<body>
  <h1><a href="index.html">WEB</a></h1>
  <ol>
    <li><a href="1.html" class="saw">HTML</a></li>
    <li><a href="2.html" class="saw" id="active">CSS</a></li>
    <li><a href="3.html">JavaScript</a></li>
  </ol>
  <h2>JavaScript</h2>
  <p>
    JavaScript
  </p>
</body>
</html>
```

WEB

1. HTML
2. CSS
3. JavaScript

JavaScript

JavaScript

우선순위는 다음과 같다.

- 태그 선택자 < class 선택자 < id 선택자
- 동일한 등급의 선택자라면 후술되는 선택자가 높다.
- 잘 생각해보면 태그가 가장 포괄적으로 적용되고, class 는 그 다음, id 는 단일로만 적용이다.

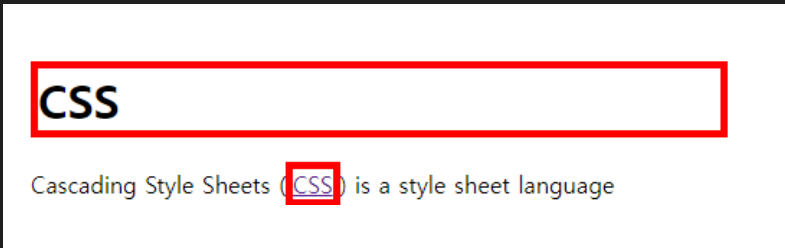
4. 박스모델

1) 태그의 기본 적용 범위

각 태그마다 기본값으로 적용 범위, 영역의 차이가 있다.

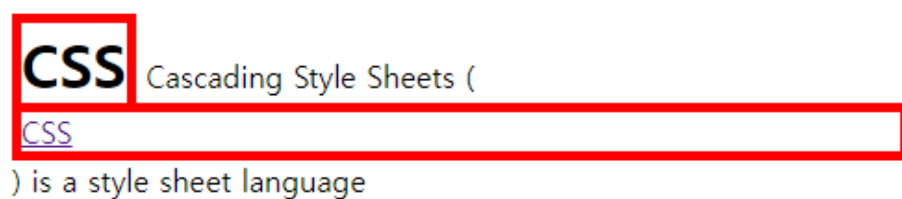
전체를 쓴다면 **block level element**, 자기크기만큼을 사용하면 **inline element** 이다.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <style>
    /* block level element : h1 태그는 블록 단위(화면 전체)를 사용 */
    h1 {
      border-width: 5px;
      border-color: red;
      border-style: solid;
    }
    /* inline element : a 태그는 자산의 크기만큼을 사용 */
    a {
      border-width: 5px;
      border-color: red;
      border-style: solid;
    }
  </style>
</head>
<body>
  <h1>CSS</h1>
  Cascading Style Sheets is a style sheet language
</body>
</html>
```



2) 태그의 적용 범위 변경

속성으로 **display:inline;** 이나 **display:block;** 를 사용하면 적용범위가 달라진다.



3) 동일한 속성과 값을 가진 태그라면 , 콤마로 나열

```
<style>
  h1 , a {
    border-width: 5px;
    border-color: red;
    border-style: solid;
    display:inline;
  }
</style>
```

4) 동일한 속성명은 나열이 가능

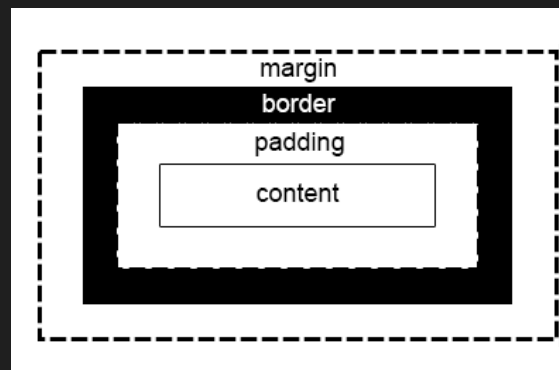
width, color, style 등의 순서는 상관없이 나열할 수 있다.

```
<style>
  h1 , a {
    border: 5px solid red;
  }
</style>
```

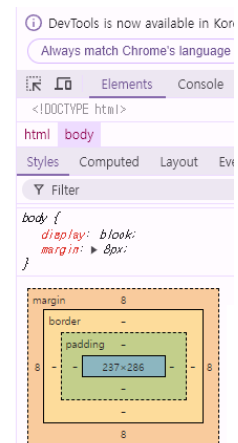
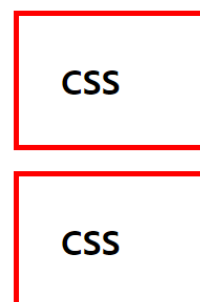
5) padding, margin

```
<style>
  h1 , a {
    border: 5px solid red;
    padding: 40px; /* 내부여백*/
    margin: 20px; /* 외부여백*/

    /* h1의 기본 display는 block 이다.
    여기에 넓이를 지정하면 박스형태가 된다.*/
    display:block;
    width: 100px;
  }
</style>
```



개발자 도구(F12)에서 Elements를 활용하자.



5. 박스모델 활용



1) Web 아래의 선은 **border-bottom: 1px solid gray;**로 밑줄만 남긴 것이다.

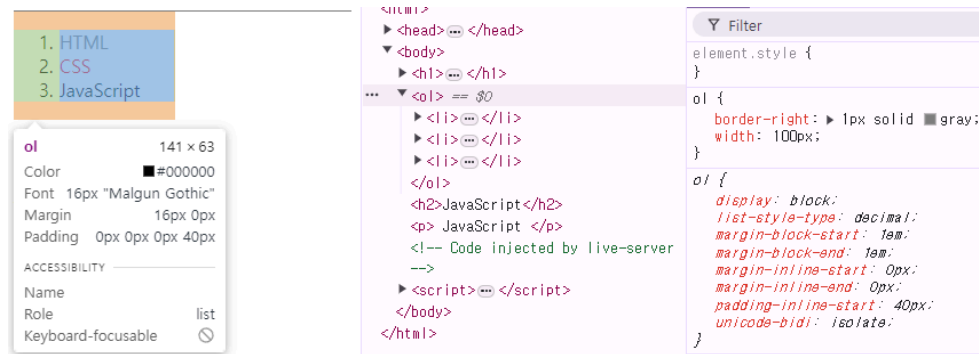
2) ol에 **border-right**만 준다면 우측끝에 선이 생기므로 **width**로 박스크기를 제한한다.

```
ol {
  border-right: 1px solid gray;
  width: 100px;
}
```

3) ol의 기본 속성에는 패딩이 적용되어 있다. 이를 세부 수정하면 더 깔끔해진다.

```
ol {
  border-right: 1px solid gray;
  width: 100px;
  margin: 0;
  padding: 20px;
}
```

3-1) 기본 ol 의 값



3-2) 변경해서 조절



6. 그리드

div 태그에 id 를 부여, display 속성도 grid 값을 준다.
grid 속성값은 1fr 을 주면 프레임별로 나눌 수 있게 된다.



7. 그리드의 활용

1) 시작전

WEB

1. HTML
2. CSS
3. JavaScript

CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

```
<!doctype html>
<html>
<head>
  <title>WEB1 - JavaScript</title>
  <meta charset="utf-8">
</head>
<style>
  body {
    margin:0;
  }
  a {
    color:black;
    text-decoration: none;
  }
  #active{
    color: red;
  }
  .saw {
    color:gray;
  }
  h1 {
    font-size: 45px;
    text-align: center;
    border-bottom: 1px solid gray;
    margin: 0px;
    padding: 20px;
  }
  ol {
    border-right: 1px solid gray;
```

```

width: 100px;
margin:0;
padding: 20px;
}
</style>
<body>
<h1><a href="index.html">WEB</a></h1>
<ol>
<li><a href="1.html" class="saw">HTML</a></li>
<li><a href="2.html" class="saw" id="active">CSS</a></li>
<li><a href="3.html">JavaScript</a></li>
</ol>
<h2>CSS</h2>
<p> Cascading Style Sheets (CSS) is a style sheet language</p>
</body>
</html>

```

2) div 태그로 그룹을 묶고, 상단 div 태그에 id 를 부여하여 속성을 작성한다.

```

<body>
<h1><a href="index.html">WEB</a></h1>
<div id="grid">
<ol>
<li><a href="1.html" class="saw">HTML</a></li>
<li><a href="2.html" class="saw" id="active">CSS</a></li>
<li><a href="3.html">JavaScript</a></li>
</ol>
<div>
<h2>CSS</h2>
<p>Cascading Style Sheets (CSS) is a style sheet language</p>
</div>
</div>
</body>

```

```

#grid {
display : grid;
grid-template-columns: 150px 1fr;
}

```

WEB

1. HTML
2. CSS
3. JavaScript

CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation

3) 세부조정

padding (녹색 부분)을 더블 클릭하면 미세 조정으로 원하는 위치를 찾을 수 있게 된다.
자동 적용은 아니므로 원하는 값을 입력해야 적용된다.
원하는 부분마다 id를 부여한다.

1. HTML

2. CSS

3. JavaScript

CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body>
    <h1>...</h1>
    <div id="grid">
      <ol style="padding-left: 24px;">
        <li>...</li>
        <li>...</li>
        <li>...</li>
      </ol>
    </div>
  </body>
</html>
```

margin

border

padding 20

24 100x467.812 20 1

3-1) 정리

모든 `li` 에 적용되는 내용이 있고,
`grid` 안에서의 `li`이 있을 것이다. 이를 구분하기 위해, 상위격을 먼저 써주고 하위격을 작성해준다.

```
li {
  border-right: 1px solid gray;
  width: 100px;
  margin:0;
  padding: 20px;
}
#grid {
  display : grid;
  grid-template-columns: 150px 1fr;
}
#grid li {
  padding-left: 33px;
}
#grid #article {
  padding-left: 25px;
}
```

3-2) 완료된 소스

```
<!doctype html>
<html>
<head>
  <title>WEB1 - JavaScript</title>
  <meta charset="utf-8">
</head>
<style>
  body {
    margin:0;
  }
  a {
    color:black;
    text-decoration: none;
  }
  h1 {
    font-size: 45px;
    text-align: center;
    border-bottom: 1px solid gray;
    margin: 0px;
    padding: 20px;
  }
  ol {
    border-right: 1px solid gray;
    width: 100px;
    margin:0;
    padding: 20px;
  }
  #grid {
    display : grid;
    grid-template-columns: 150px 1fr;
  }
  #grid ol {
    padding-left: 33px;
  }
  #grid #article {
    padding-left: 25px;
  }
</style>
<body>
  <h1><a href="index.html">WEB</a></h1>
  <div id="grid">
    <ol>
      <li><a href="1.html">HTML</a></li>
```

```
<li><a href="2.html">CSS</a></li>
<li><a href="3.html">JavaScript</a></li>
</ol>
<div id="article">
  <h2>CSS</h2>
  <p>
```

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

```
</p>
</div>
</div>
</body>
</html>
```

8. 반응형 디자인 - 미디어쿼리

화면의 크기에 따라 미디어들이 적용되는 디자인이 달라지는 것

screen 의 width > 800 px 라는 조건은,

`@media(min-width:800px){ }` 는 800px 이하면 { } 이 실행된다.

`@media(max-width:800px){ }` 는 800px 초과면 { } 이 실행된다.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <style>
    div {
      border: 10px solid green;
      font-size: 60px;
    }
    @media(min-width:800px){
      div{
        display: none;
      }
    }
  </style>
</head>
<body>
  <div>
    Responsive
  </div>
</body>
</html>
```

Responsive

정리

```
<!doctype html>
<html>
<head>
  <title>WEB1 - JavaScript</title>
  <meta charset="utf-8">
</head>
<style>
  body {
    margin:0;
  }
</style>
```

```

a {
  color:black;
  text-decoration: none;
}
h1 {
  font-size: 45px;
  text-align: center;
  border-bottom: 1px solid gray;
  margin: 0px;
  padding: 20px;
}
ol {
  border-right: 1px solid gray;
  width: 100px;
  margin:0;
  padding: 10px;
}
#grid {
  display : grid;
  grid-template-columns: 150px 1fr;
}
#grid ol {
  padding-left: 33px;
}
#grid #article {
  padding-left: 25px;
}
@media(max-width:800px){
  #grid{
    display: block;
  }
  ol {
    border-right: none;
  }
  h1 {
    border-bottom: none;
  }
}
</style>
<body>
<h1><a href="index.html">WEB</a></h1>
<div id="grid">
  <ol>
    <li><a href="1.html">HTML</a></li>

```

```

</li><a href="2.html">CSS</a></li>
<li><a href="3.html">JavaScript</a></li>
</ol>
<div id="article">
  <h2>CSS</h2>
  <p>
    Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation
    of a document written in a markup language. Although most often used to set the visual style of web
    pages and user interfaces written in HTML and XHTML, the language can be applied to any XML
    document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other
    media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to
    create visually engaging webpages, user interfaces for web applications, and user interfaces for many
    mobile applications.
  </p>
</div>
</div>
</body>
</html>

```

9. style.css 파일로 쪼개기

<style> </style> 내용만 이동

4,5,6.html 에 복붙

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

사실 효율은 각 html 에 CSS 부분이 있는 것은 속도 측면에서 좋을 수 있다. 그러나 페이지가 계속 불러올 때마다 같은 내용을 중복적으로 가져오니 트래픽 낭비가 심할 것이다.

그래서 캐싱 덕분에 한번 CSS를 다운 받았다면 파일이 바뀌기전까지는 웹브라우저는 우리 컴퓨터에 CSS를 가져오기 때문에 속도를 높일 수 있다. 네트워크 를 안쓰기 때문에 트래픽감소효과가 있다.

추가 강의

- WEB1 - HTML & Internet : <https://opentutorials.org/module/3135>
- WEB2 - JavaScript : <https://opentutorials.org/course/3085>
- WEB2-CSS : <https://opentutorials.org/course/3086>