



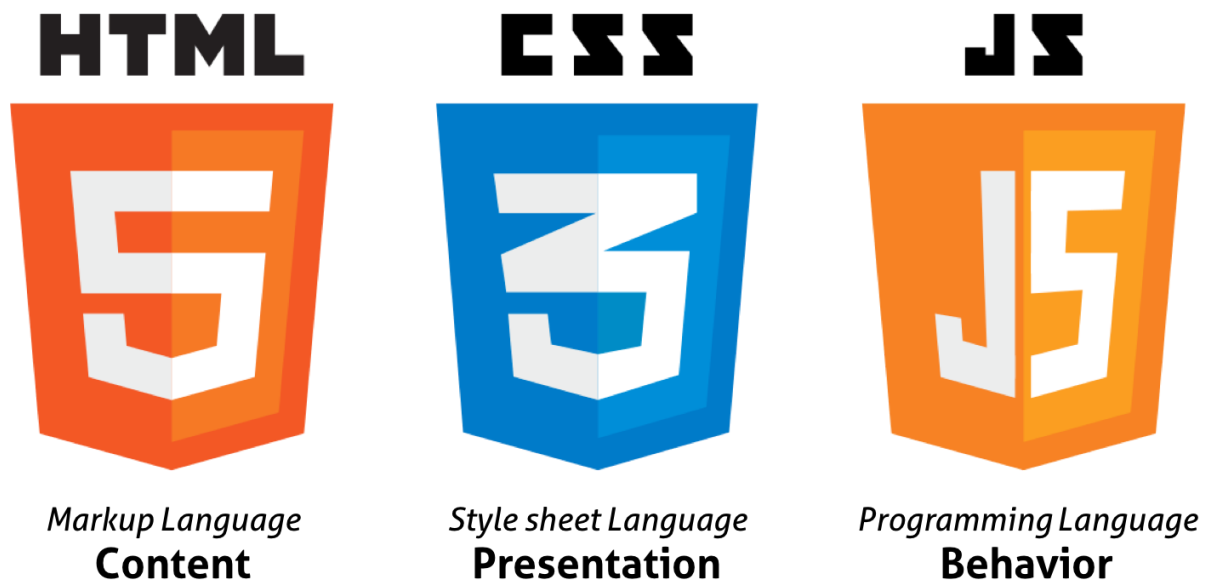
CSS 기본 문법

CSS(Cascading Style Sheets)는 HTML이나 XML과 같은 구조화 된 문서(Document)를 화면, 종이 등에 어떻게 렌더링할 것인지를 정의하기 위한 언어이다. 즉, CSS는 HTML의 각 요소(Element)의 **style**(design, layout etc)을 정의하여 화면(Screen) 등에 어떻게 렌더링하면 되는지 브라우저에게 설명하기 위한 언어이다.



HTML5 이전 버전의 HTML에는 style을 컨트롤할 수 있는 태그(font, center)가 존재하여 CSS가 없이도 어느 정도의 스타일 표현이 가능하였으나 정보와 구조를 담당하는 HTML의 본연의 역할과 동떨어진 기능까지 추가됨으로서 복잡하고 혼란스러운 언어가 되어 버렸다.

HTML5에서는 **HTML**는 정보와 구조화, **CSS**는 styling의 정의라는 본연의 임무에 충실한 명확한 구분이 이루어졌다.



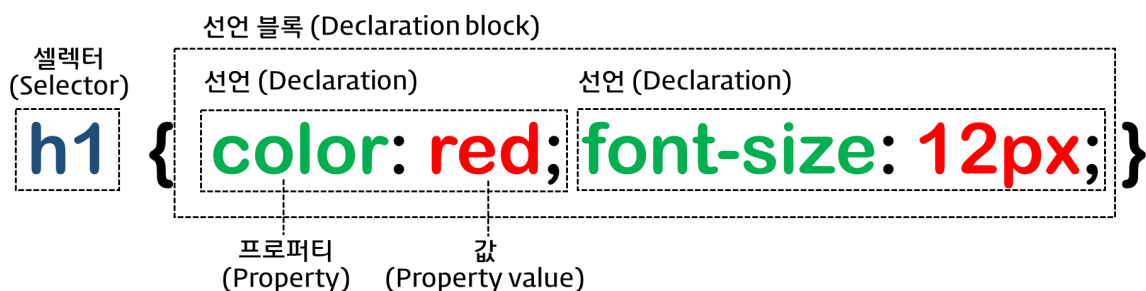
HTML과 CSS는 **각자의 문법을 갖는 별개의 언어**이며 HTML은 CSS를 포함할 수 있다. 그러나 HTML없이 단독으로 존재하는 CSS는 의미가 없다.

새로운 언어를 학습할 때에는 먼저 용어에 익숙해져야 할 필요가 있다. CSS3의 용어에 대해 알아보도록 하자.

셀렉터 (Selector, 선택자)

CSS는 HTML 요소의 style(design, layout etc)을 정의하는데 사용된다. 이를 위해서 선행되어야하는 것은 **스타일을 적용하고자 하는 HTML 요소를 선택**할 수 있어야 한다.

셀렉터는 스타일을 적용하고자 하는 HTML 요소를 선택하기 위해 CSS에서 제공하는 수단이다.



위와 같은 구문을 Rule Set(또는 Rule)이라 하며 셀렉터에 의해 선택된 특정 HTML 요소를 어떻게 렌더링(Rendering)할 것인지 브라우저에 지시하는 역할을 한다. 위의 CSS Rule set은 HTML 문서에 속해 있는 셀렉터를 통해 모든 h1 요소를 선택한 후 선택된 h1 요소의 스타일을 선언 블록에서 정의하고 있다.

이와 같은 Rule Set의 집합을 스타일시트(Style Sheet)라 한다.

프로퍼티 (Property / 속성)

셀렉터로 HTML 요소를 선택하고 {} 내에 프로퍼티(속성)와 값을 지정하는 것으로 다양한 style을 정의할 수 있다. 프로퍼티는 **표준 스펙**으로 이미 지정되어 있는 것을 사용하여야 하며 사용자가 임의로 정의할 수 없다. 여러 개의 프로퍼티를 연속해서 지정할 수 있으며 세미 콜론(;)으로 구분한다.

```
p {  
  color: ...;  
  font-size: ...;  
}
```

값 (Value / 속성값)

셀렉터로 지정한 HTML 요소에 style을 적용하기 위해 프로퍼티를 사용했다. 프로퍼티의 값은 해당 프로퍼티에 사용할 수 있는 값을 "키워드"나 "크기 단위" 또는 "색상 표현 단위" 등의 **특정 단위**로 지정하여야 한다.

```
p {  
  color: orange;  
  font-size: 16px;  
}
```

HTML과 CSS의 연동

HTML은 CSS를 포함할 수 있다. CSS를 가지고 있지 않은 HTML은 브라우저에서 기본으로 적용하는 CSS(user agent stylesheet)에 의해 렌더링된다.

CSS와 HTML을 연동하는 방법은 다음과 같다.

Link style

HTML에서 외부에 있는 CSS 파일을 로드하는 방식이다. 가장 일반적으로 사용된다.

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="css/style.css">
  </head>
  <body>
    <h1>Hello World</h1>
    <p>This is a web page.</p>
  </body>
</html>
```

```
h1 { color: red; }
p { background: blue; }
```

Embedding style

HTML 내부에 CSS를 포함시키는 방식이다. 매우 간단한 웹페이지의 경우는 문제될 것이 없겠지만 Link style을 사용하는 편이 좋다. HTML과 CSS는 서로 역할이 다르므로 다른 파일로 구분되어 작성하고 관리되는 것이 바람직하다.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      h1 { color: red; }
```

```

    p { background: aqua; }
  </style>
</head>
<body>
  <h1>Hello World</h1>
  <p>This is a web page.</p>
</body>
</html>

```

Inline style

HTML요소의 style 프로퍼티에 CSS를 기술하는 방식이다. JavaScript가 동적으로 CSS를 생성할 때 사용하는 경우가 있다. 하지만 일반적인 경우 Link style을 사용하는 편이 좋다.

```

<!DOCTYPE html>
<html>
  <body>
    <h1 style="color: red">Hello World</h1>
    <p style="background: aqua">This is a web page.</p>
  </body>
</html>

```

Reset CSS 사용하기

모든 웹 브라우저는 디폴트 스타일(브라우저가 내장하고 있는 기본 스타일)을 가지고 있어 CSS가 없어도 작동한다. 그런데 웹브라우저에 따라 디폴트 스타일이 상이하고 지원하는 tag나 style도 제각각이어서 주의가 필요하다.

Reset CSS는 기본적인 HTML 요소의 CSS를 초기화하는 용도로 사용한다. 즉, 브라우저 별로 제각각인 디폴트 스타일을 하나의 스타일로 통일시켜 주는 역할을 한다.

자주 사용되는 Reset CSS는 다음과 같다.

- [Eric Meyer's reset](#)
- [normalize.css](#)

다음은 Eric Meyer's reset css이다. 이것을 기준으로 사용자의 CSS를 완성해 나가는 방법은 매우 유용하다.

```
/* http://meyerweb.com/eric/tools/css/reset/
   v2.0 | 20110126
   License: none (public domain)
*/

html, body, div, span, applet, object, iframe,
h1, h2, h3, h4, h5, h6, p, blockquote, pre,
a, abbr, acronym, address, big, cite, code,
del, dfn, em, img, ins, kbd, q, s, samp,
small, strike, strong, sub, sup, tt, var,
b, u, i, center,
dl, dt, dd, ol, ul, li,
fieldset, form, label, legend,
table, caption, tbody, tfoot, thead, tr, th, td,
article, aside, canvas, details, embed,
figure, figcaption, footer, header, hgroup,
menu, nav, output, ruby, section, summary,
time, mark, audio, video {
    margin: 0;
    padding: 0;
    border: 0;
    font-size: 100%;
    font: inherit;
    vertical-align: baseline;
}
/* HTML5 display-role reset for older browsers */
article, aside, details, figcaption, figure,
footer, header, hgroup, menu, nav, section {
    display: block;
}
body {
    line-height: 1;
}
ol, ul {
    list-style: none;
```

```
}  
blockquote, q {  
    quotes: none;  
}  
blockquote:before, blockquote:after,  
q:before, q:after {  
    content: ' ';  
    content: none;  
}  
table {  
    border-collapse: collapse;  
    border-spacing: 0;  
}
```