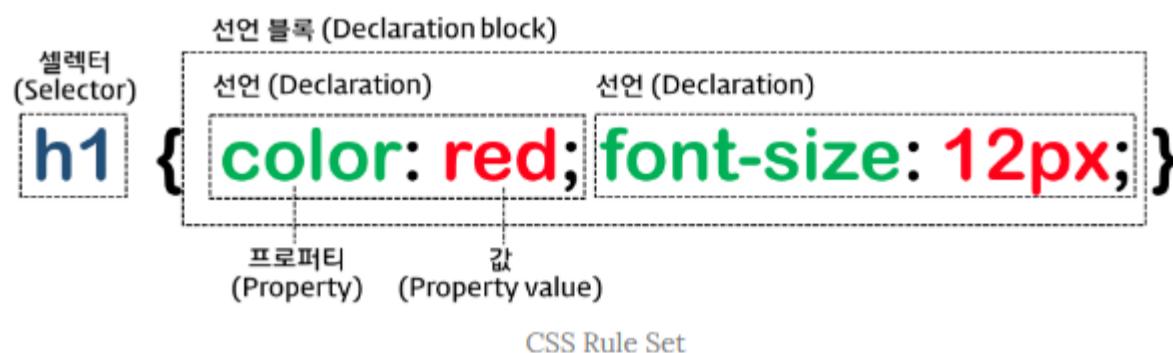




2. Selector

[made by 이우주]

style을 적용하고자 하는 HTML 요소를 셀렉터로 특정하고 선택된 요소에 스타일을 정의



전체 / 태그 / ID / 클래스 / 어트리뷰트 / 복합 /
가상 클래스 / 가상 요소 “셀렉터”

1. 전체 셀렉터(Universal Selector) : [*]

```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* 모든 요소를 선택 */
    * { color: red; }
  </style>
</head>
<body>
  <h1>Heading</h1>
  <div>
    <p>paragraph 1</p>
    <p>paragraph 2</p>
  </div>
  <p>paragraph 3</p>
</body>
</html>
```

Heading

paragraph 1

paragraph 2

paragraph 3

2. 태그 셀렉터(Type Selector) : [태그명]

```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* 모든 p 태그 요소를 선택 */
    p { color: red; }
  </style>
</head>
<body>
  <h1>Heading</h1>
```

Heading

paragraph 1

paragraph 2

paragraph 3

```
<div>
  <p>paragraph 1</p>
  <p>paragraph 2</p>
</div>
<p>paragraph 3</p>
</body>
</html>
```

3. ID 셀렉터(ID Selector) : [#id 어트리뷰트 값]

```
<!DOCTYPE html>
<html>
<head>
<style>
  /* id 어트리뷰트 값이 p1인 요소를 선택 */
  #p1 { color: red; }
</style>
</head>
<body>
  <h1>Heading</h1>
  <div class="container">
    <p id="p1">paragraph 1</p>
    <p id="p2">paragraph 2</p>
  </div>
  <p>paragraph 3</p>
</body>
</html>
```

Heading

paragraph 1

paragraph 2

paragraph 3

4. 클래스 셀렉터(Class Selector) : [.class 어트리뷰트 값]

```
<!DOCTYPE html>
<html>
<head>
<style>
  .container { color: red; }
  /* not supported in IE */
  #p2 { color: initial; }
</style>
</head>
<body>
  <h1>Heading</h1>
  <div class="container">
    <p id="p1">paragraph 1</p>
    <p id="p2">paragraph 2</p>
  </div>
  <p>paragraph 3</p>
</body>
</html>
```

Heading

paragraph 1

paragraph 2

paragraph 3

HTML 요소에 class 어트리뷰트 값은 공백으로 구분하여 여러개 지정할 수 있음

아래와 같이 클래스 셀렉터를 사용하여 미리 스타일을 정의해 두고,

HTML 요소는 이미 정의되어 있는 클래스를 지정하는 것으로 필요한 스타일 지정 가능

5. 어트리뷰트 셀렉터 : 셀렉터[어트리뷰트 조건]

셀렉터 내에서 명시된 어트리뷰트 조건에 해당하는 요소를 선택

패턴	Description
셀렉터[어트리뷰트]	지정된 어트리뷰트를 갖는 모든 요소를 선택한다.

패턴	Description
셀렉터[어트리뷰트="값"]	지정된 어트리뷰트를 가지며 지정된 값과 어트리뷰트의 값이 일치하는 모든 요소를 선택한다.

패턴	Description
셀렉터[어트리뷰트~="값"]	지정된 어트리뷰트의 값이 지정된 값을 (공백으로 분리된) 단어로 포함하는 요소를 선택한다.

패턴	Description
셀렉터[어트리뷰트 = "값"]	지정된 어트리뷰트의 값과 일치하거나 지정 어트리뷰트 값 뒤 연이은 하이픈("값-")으로 시작하는 요소를 선택한다.

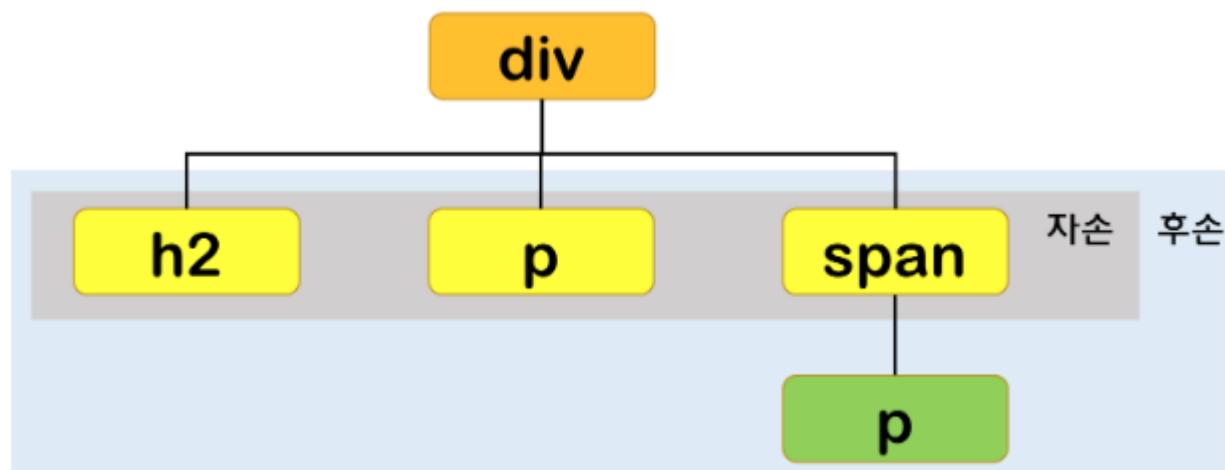
패턴	Description
셀렉터[어트리뷰트 ^= "값"]	지정된 어트리뷰트 값으로 시작하는 요소를 선택한다.

패턴	Description
셀렉터[어트리뷰트\$= "값"]	지정된 어트리뷰트 값으로 끝나는 요소를 선택한다.

패턴	Description
셀렉터[어트리뷰트*= "값"]	지정된 어트리뷰트 값을 포함하는 요소를 선택한다.

6. 복합 셀렉터 (Combinator)

1) 후손 셀렉터(Descendant Combinator)



셀렉터A 셀렉터B

후손 셀렉터는 셀렉터 A의 모든 **후손(하위) 요소** 중 셀렉터 B와 일치하는 요소를 선택

```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* div 요소의 후손요소 중 p 요소 */
    div p { color: red; }
  </style>
</head>
<body>
  <h1>Heading</h1>
  <div>
    <p>paragraph 1</p>
    <p>paragraph 2</p>
    <span><p>paragraph 3</p></span>
  </div>
  <p>paragraph 4</p>
</body>
</html>
```

Heading

paragraph 1

paragraph 2

paragraph 3

paragraph 4

2) 자식 셀렉터(Child Combinator)

셀렉터A > 셀렉터B

자손 셀렉터는 셀렉터 A의 모든 **자식 요소** 중 셀렉터B와 일치하는 요소를 선택

```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* div 요소의 자식요소 중 p 요소 */
    div > p { color: red; }
  </style>
</head>
<body>
  <h1>Heading</h1>
  <div>
    <p>paragraph 1</p>
    <p>paragraph 2</p>
    <span><p>paragraph 3</p></span>
  </div>
  <p>paragraph 4</p>
</body>
</html>
```

Heading

paragraph 1

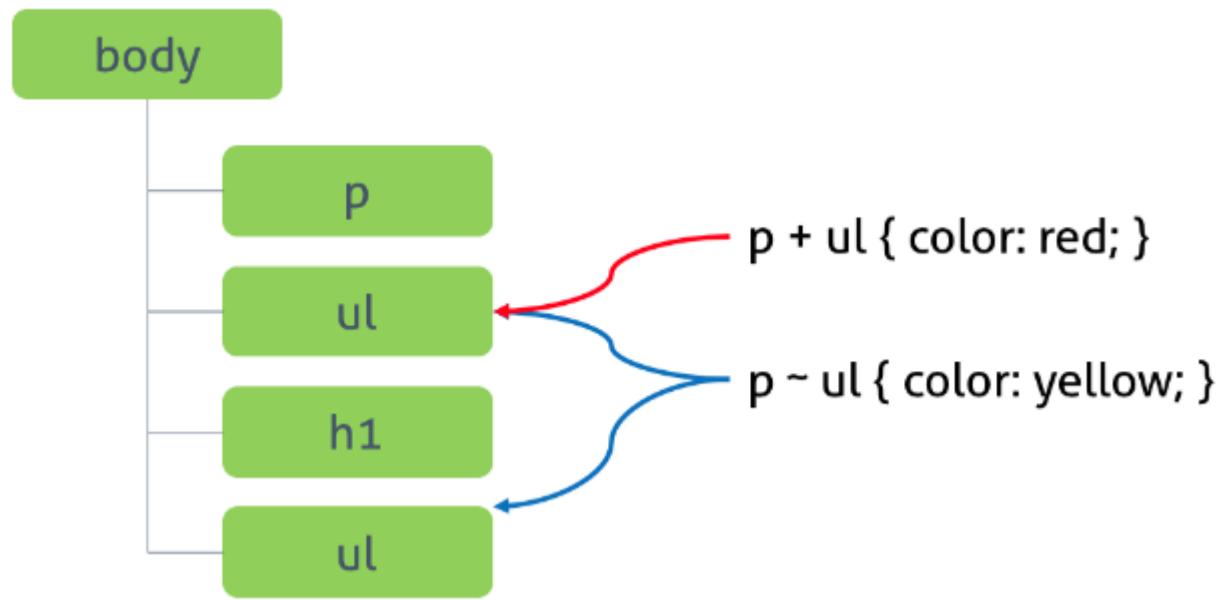
paragraph 2

paragraph 3

paragraph 4

3) 형제(동위) 셀렉터(Sibling Combinator)

형제(동위)셀렉터는 형제 관계(동위 관계)에서 뒤에 위치하는 요소를 선택할 때 사용



- 인접 형제 셀렉터(Adjacent Sibling Combinator)

셀렉터A + 셀렉터B

⇒ 셀렉터 A의 형제 요소 중 셀렉터A 바로 뒤에 위치하는 셀렉터B 요소를 선택
(A와 B 사이에 다른 요소가 존재하면 선택되지 않는다)

- 일반 형제 셀렉터(General Sibling Combinator)

셀렉터A ~ 셀렉터B

⇒ 셀렉터 A의 형제 요소 중 셀렉터A 뒤에 위치하는 셀렉터B 요소를 모두 선택

7. 가상 클래스 셀렉터 (Pseudo-Class Selector)

요소의 특정 상태에 따라 스타일을 정의할 때 사용

콜론(:)을 사용

```

CSS
selector:pseudo-class {
    property: value;
}

```

[특정 상태]

- 마우스가 올라와 있을 때 (:hover)
- 링크를 방문했을 때와 아직 방문하지 않았을 때 (:link, :visited)
- 포커스가 들어와 있을 때 (:focus)

```

<!DOCTYPE html>
<html>
<head>
    <style>
        /* a 요소가 hover 상태일 때 */
        a:hover { background-color: red; }
        /* input 요소가 focus 상태일 때 */
        input:focus { background-color: yellow; }
    </style>
</head>

```

```

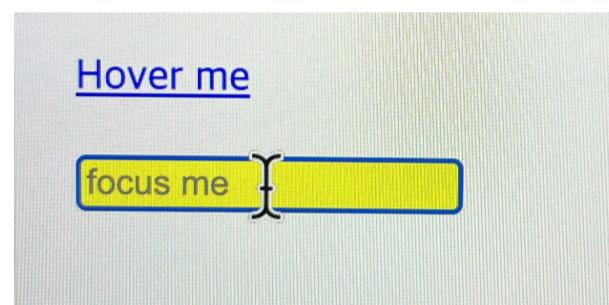
<body>
  <a href="#">Hover me</a><br><br>
  <input type="text" placeholder="focus me">
</body>
</html>

```

[Hover me](#)

전) 기본 상태

후) a 태그에 마우스 올라와 있을 때



후) input 태그에 마우스 포커스 했을 때

* 위 예시의 링크 셀렉터와 동적 셀렉터는 상대적으로 사용 빈도가 높은 셀렉터입니다!
 그 외의 셀렉터는 눈으로 한번 훑어보고 이런 셀렉터도 있구나~~ 정도로만 넘어가시고
 나중에 실제 코딩을 하다가 실제 필요할 때 적용하여 자주 사용하다 보면 많이 쓰는 셀렉터들은
 저절로 암기 될 것이라고 생각합니다 😊

-링크 셀렉터, 동적 셀렉터

pseudo-class	Description
:link	셀렉터가 방문하지 않은 링크일 때
:visited	셀렉터가 방문한 링크일 때
:hover	셀렉터에 마우스가 올라와 있을 때
:active	셀렉터가 클릭된 상태일 때
:focus	셀렉터에 포커스가 들어와 있을 때

-UI 요소 상태 셀렉터

pseudo-class	Description
:checked	셀렉터가 체크 상태일 때
:enabled	셀렉터가 사용 가능한 상태일 때
:disabled	셀렉터가 사용 불가능한 상태일 때

-구조 가상 클래스 셀렉터

pseudo-class	Description
:first-child	셀렉터에 해당하는 모든 요소 중 첫번째 자식인 요소를 선택한다.
:last-child	셀렉터에 해당하는 모든 요소 중 마지막 자식인 요소를 선택한다.

pseudo-class	Description
:nth-child(n)	셀렉터에 해당하는 모든 요소 중 앞에서 n번째 자식인 요소를 선택한다.
:nth-last-child(n)	셀렉터에 해당하는 모든 요소 중 뒤에서 n번째 자식인 요소를 선택한다.

pseudo-class	Description
:first-of-type	셀렉터에 해당하는 요소의 부모 요소의 자식 요소 중 첫 번째 등장하는 요소를 선택한다.
:last-of-type	셀렉터에 해당하는 요소의 부모 요소의 자식 요소 중 마지막에 등장하는 요소를 선택한다.
:nth-of-type(n)	셀렉터에 해당하는 요소의 부모 요소의 자식 요소 중 앞에서 n번째에 등장하는 요소를 선택한다.
:nth-last-of-type(n)	셀렉터에 해당하는 요소의 부모 요소의 자식 요소 중 뒤에서 n번째에 등장하는 요소를 선택한다.

-부정 셀렉터

pseudo-class	Description
:not(셀렉터)	셀렉터에 해당하지 않는 모든 요소를 선택한다.

-정합성 체크 셀렉터

pseudo-class	Description
:valid(셀렉터)	정합성 검증이 성공한 input 요소 또는 form 요소를 선택한다.
:invalid(셀렉터)	정합성 검증이 실패한 input 요소 또는 form 요소를 선택한다.

8. 가상 요소 셀렉터 (Pseudo-Element Selector)

두 개의 콜론(:)을 사용

CSS

```
selector::pseudo-element {  
    property:value;  
}
```

pseudo-element	Description
::first-letter	콘텐츠의 첫글자를 선택한다.
::first-line	콘텐츠의 첫줄을 선택한다. 블록 요소에만 적용할 수 있다.
::after	콘텐츠의 뒤에 위치하는 공간을 선택한다. 일반적으로 content 프로퍼티와 함께 사용된다.
::before	콘텐츠의 앞에 위치하는 공간을 선택한다. 일반적으로 content 프로퍼티와 함께 사용된다.
::selection	드래그한 콘텐츠를 선택한다. iOS Safari 등 일부 브라우저에서 동작 않는다.