

# 221011 수업 -1

☰ 유형	
☰ 키워드	

## 셀렉터

### 전체 셀렉터

\* : HTML 문서 내의 모든 요소를 선택한다. html 요소를 포함한 모든 요소가 선택된다

### 태그 셀렉터

- 지정된 태그명을 가지는 요소를 선택한다.

### ID 셀렉터

#id 어트리뷰트 값 : id 어트리뷰트 값을 지정하여 일치하는 요소를 선택

- id 어트리뷰트 값은 중복될 수 없는 유일한 값

### 클래스 셀렉터

.class 어트리뷰트 값 : class 어트리뷰트 값을 지정하여 일치하는 요소를 선택

- class 어트리뷰트 값은 중복될 수 있다.

### 속성 셀렉터 (Attribute Selector)

#### 셀렉터[속성]

- 지정된 어트리뷰트를 갖는 모든 요소를 선택

```
<!DOCTYPE html>
<html>
<head>
  <style>
    a[href] { color: red; }
  </style>
</head>
<body>
  <a href="http://www.poiemaweb.com">poiemaweb.com</a><br>
```

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

### 셀렉터[어트리뷰트="값"]

- 지정된 어트리뷰트를 가지며 지정된 값과 어트리뷰트의 값이 일치하는 모든 요소를 선택

```
<!DOCTYPE html>
<html>
<head>
  <style>
    a[target="_blank"] { color: red; }
  </style>
</head>
<body>
  <a href="http://www.poiemaweb.com">poiemaweb.com</a><br>
  <a href="http://www.google.com" target="_blank">google.com</a><br>
  <a href="http://www.naver.com" target="_top">naver.com</a>
</body>
</html>
```

→ \_blank : 새 탭

### 셀렉터[어트리뷰트~="값"]

- 지정된 어트리뷰트의 값이 지정된 값을 (공백으로 분리된) 단어로 포함하는 요소를 선택

```
<!DOCTYPE html>
<html>
<head>
  <style>
    h1[title~="first"] { color: red; }
  </style>
</head>
<body>
  <h1 title="heading first">Heading first</h1>
  <h1 title="heading-first">Heading-first</h1>
  <h1 title="heading second">Heading second</h1>
  <h1 title="heading third">Heading third</h1>
</body>
</html>
```

→ h1 요소 중에 title 어트리뷰트 값에 "first"를 단어로 포함하는 요소

### 셀렉터[어트리뷰트="값"]

- 지정된 어트리뷰트의 값과 일치하거나 지정 어트리뷰트 값 뒤 연이은 하이픈("값-")으로 시작하는 요소를 선택

```

<!DOCTYPE html>
<html>
<head>
  <style>
    p[lang="en"] { color: red; }
  </style>
</head>
<body>
  <p lang="en">Hello!</p>
  <p lang="en-us">Hi!</p>
  <p lang="en-gb">Ello!</p>
  <p lang="us">Hi!</p>
  <p lang="no">Hei!</p>
</body>
</html>

```

→ p 요소 중에 lang 어트리뷰트 값이 "en"과 일치하거나 "en-"로 시작하는 요소  
 셀렉터[어트리뷰트^="값"]

- 지정된 어트리뷰트 값으로 시작하는 요소를 선택

```

<!DOCTYPE html>
<html>
<head>
  <style>
    a[href^="https://"] { color: red; }
  </style>
</head>
<body>
  <a href="https://www.test.com">https://www.test.com</a><br>
  <a href="http://www.test.com">http://www.test.com</a>
</body>
</html>

```

→ a 요소 중에 href 어트리뷰트 값이 "https://"로 시작하는 요소  
 셀렉터[어트리뷰트\$="값"]

- 지정된 어트리뷰트 값으로 끝나는 요소를 선택

```

<!DOCTYPE html>
<html>
<head>
  <style>
    a[href$=".html"] { color: red; }
  </style>
</head>
<body>
  <a href="test.html">test.html</a><br>
  <a href="test.jsp">test.jsp</a>
</body>
</html>

```

→ a 요소 중에 href 어트리뷰트 값이 ".html"로 끝나는 요소

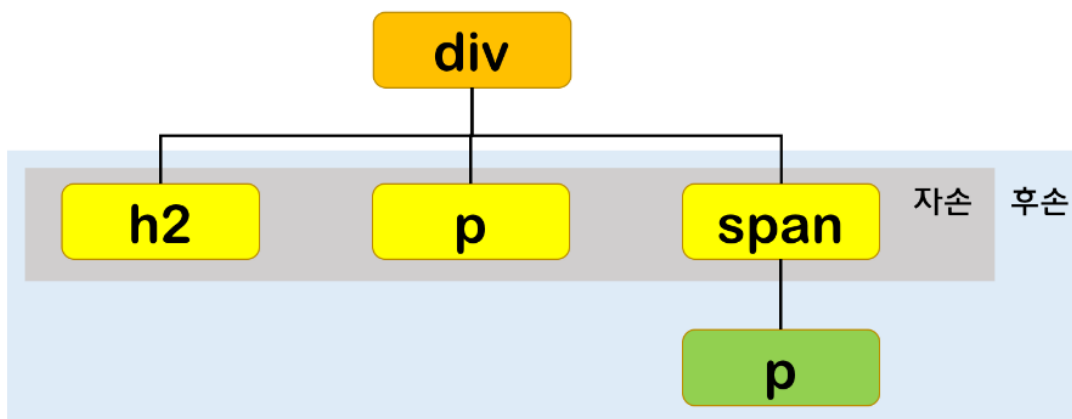
셀렉터[어트리뷰트\*="값"]

- 지정된 어트리뷰트 값을 포함하는 요소를 선택

```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* div 요소 중에서 class 어트리뷰트 값에 "test"를 포함하는 요소 */
    div[class*="test"] { color: red; }
    /* div 요소 중에서 class 어트리뷰트 값에 "test"를 단어로 포함하는 요소 */
    div[class~="test"] { background-color: yellow; }
  </style>
</head>
<body>
  <div class="first_test">The first div element.</div>
  <div class="second">The second div element.</div>
  <div class="test">The third div element.</div>
  <p class="test">This is some text in a paragraph.</p>
</body>
</html>
```

복합 셀렉터 (Combinator)

- 후속 셀렉터 (Descendant Combinator)



→ 부모 요소 : 자신의 1 level 상위에 속하는 요소

→ 자손 요소(자식 요소) : 1 level 하위에 속하는 요소

→ 후손 요소(하위 요소) : 자신보다 n level 하위에 속하는 요소

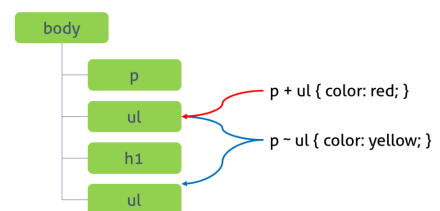
- 자식 셀렉터 (Child Combinator)

- 셀렉터A의 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>
```

- 형제(동위) 셀렉터 (Sibling Combinator)

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



```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* p 요소의 형제 요소 중에 p 요소 바로 뒤에 위치하는 ul 요소를 선택한다. */
    p + ul { color: red; }
  </style>
</head>
<body>
  <div>A div element.</div>
  <ul>
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
  </ul>

  <p>The first paragraph.</p>
  <ul>
    <li>Coffee</li>
    <li>Tea</li>
  </ul>
```

```

    <li>Milk</li>
  </ul>

  <h2>Another list</h2>
  <ul>
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
  </ul>
</body>
</html>

```

- 일반 형제 선택터(General Sibling combinator)

- 선택터A의 형제 요소 중 선택터A 뒤에 위치하는 선택터B 요소를 모두 선택

```

<!DOCTYPE html>
<html>
<head>
  <style>
    /* p 요소의 형제 요소 중에 p 요소 뒤에 위치하는 ul 요소를 모두 선택한다.*/
    p ~ ul { color: red; }
  </style>
</head>
<body>
  <div>A div element.</div>
  <ul>
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
  </ul>

  <p>The first paragraph.</p>
  <ul>
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
  </ul>

  <h2>Another list</h2>
  <ul>
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
  </ul>
</body>
</html>

```

## 가상 클래스 선택터 (Pseudo-Class Selector)

- 요소의 특정 상태에 따라 스타일을 정의할 때 사용

ex\_ 마우스가 올라와 있을 경우, 링크를 방문했을 때와 아직 방문하지 않은 경우

```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* a 요소가 hover 상태일 때 */
    a:hover { 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>
```

링크 셀렉터(Link pseudo-classes), 동적 셀렉터(User action pseudo-classes)

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

```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* a 요소가 방문하지 않은 링크일 때 */
    a:link { color: orange; }

    /* a 요소가 방문한 링크일 때 */
    a:visited { color: green; }

    /* a 요소에 마우스가 올라와 있을 때 */
    a:hover { font-weight: bold; }

    /* a 요소가 클릭된 상태일 때 */
    a:active { color: blue; }

    /* text input 요소와 password input 요소에 포커스가 들어와 있을 때 */
    input[type=text]:focus,
    input[type=password]:focus {
      color: red;
    }
  </style>
</head>
<body>
  <a href="#">Link</a>
  <a href="#">Visited</a>
  <a href="#">Hover</a>
  <a href="#">Active</a>
  <input type="text" value="Text Input Focus" />
  <input type="password" value="Password Input Focus" />
</body>
</html>
```

```

    }
  </style>
</head>
<body>
  <a href="#" target="_blank">This is a link</a><br>
  <input type="text" value="I'll be red when focused"><br>
  <input type="password" value="I'll be red when focused">
</body>
</html>

```

## UI 요소 상태 셀렉터

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

```

<!DOCTYPE html>
<html>
<head>
  <style>
    /* input 요소가 사용 가능한 상태일 때,
       input 요소 바로 뒤에 위치하는 인접 형제 span 요소를 선택 */
    input:enabled + span {
      color: blue;
    }
    /* input 요소가 사용 불가능한 상태일 때,
       input 요소 바로 뒤에 위치하는 인접 형제 span 요소를 선택 */
    input:disabled + span {
      color: gray;
      text-decoration: line-through;
    }
    /* input 요소가 체크 상태일 때,
       input 요소 바로 뒤에 위치하는 인접 형제 span 요소를 선택 */
    input:checked + span {
      color: red;
    }
  </style>
</head>
<body>
  <input type="radio" checked="checked" value="male" name="gender"> <span>Male</span><br>
  <input type="radio" value="female" name="gender"> <span>Female</span><br>
  <input type="radio" value="neuter" name="gender" disabled> <span>Neuter</span><hr>

  <input type="checkbox" checked="checked" value="bicycle"> <span>I have a bicycle</span><br>
  <input type="checkbox" value="car"> <span>I have a car</span><br>
  <input type="checkbox" value="motorcycle" disabled> <span>I have a motorcycle</span>
</body>
</html>

```



## 구조 가상 클래스 셀렉터 (Structural pseudo-classes)

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.

```
<!DOCTYPE html>
<html>
<head>
  <style>
    /* p 요소 중에서 첫번째 자식을 선택 */
    p:first-child { color: red; }

    /* p 요소 중에서 마지막 자식을 선택 */
    /* body 요소의 두번째 p 요소는 마지막 자식 요소가 아니다.
       body 요소의 마지막 자식 요소는 div 요소이다. */
    p:last-child { color: blue; }
  </style>
</head>
<body>
  <p>This paragraph is the first child of its parent (body).</p>

  <h1>Welcome to My Homepage</h1>
  <p>This paragraph is not the first child of its parent.</p>

  <div>
    <p>This paragraph is the first child of its parent (div).</p>
    <p>This paragraph is not the first child of its parent.</p>
  </div>
</body>
</html>
```