

4. CSS 선택자

1. 선택자란

선택자란 html문서중 특정부분을 선택, 명시하여 해당 부분의 속성을 변경하는 목적으로 사용하는 것으로 CSS에서는 다양한 선택자가 존재한다.

1) 태그 선택자

타입 선택자라고도 하며 해당 태그 명이나 타입명을 가진 모든 태그에 영향을 준다. 순서나 특정위치가 아니라 해당되는 모든 태그을 전부 대상으로 한다.

1. tag 선택자란? 특정 태그를 선택하여 CSS속성을 적용할 수 있다

ex)

```
<style>
  div { /* 모든 div 태그에 영향을 준다.*/
    background-color : #ffd800; /*CSS 속성 : 속성값*/
  }
  a{ /*모든 a 태그에 영향을 준다.*/
    color: red;
  }
</style>
```

2) 전체(*) 선택자

해당 문서 전체에 영향을 미치는 선택자로 '*' 문자로 표시된다. 보통 웹 페이지를 초기화 할 때 사용하며 CSS 첫줄에 위치한다.

ex)

```
<style>
  *{ /*전체선택자 : 모든 태그에 적용 : 전체 페이지 디자인 초기화 용도*/
    margin : 0 ; /*밖 여백*/
    padding : 0 ; /*안 여백*/
    color : saddlebrown;
  }
</style>
```

3) id(#) 선택자

선택자 앞에 해시(#)기호로 id명을 대상으로 지정하는 선택자이다. 유일한 값을 가지는 id를 가지고 특정해서 해당 요소만 선택한다.

ex)

```
<style>
  #wrap { /*id 명이 wrap 인 태그만*/
    width: 600px;
    margin: 0 auto;
    background-color : aquamarine;
  }
  #content{ /*id 명이 content 인 태그만*/
    width : 350px;
    float : left;
    text-align : center;
  }
</style>
```

4) class(.) 선택자

id선택자와 유사하게 선택자앞에 ' .'을 붙여 해당 class명을 가진 태그를 대상으로 지정한다. class명은 중복이 가능하기 때문에 보다 광범위하게 사용할 수 있다. 다만 class선택자 앞에 특정 명칭이 있으면 해당 명칭의 태그의 일치하는 class만을 대상으로한다.

ex)

```
<style>
  .menu1 { /*class 명이 .menu1 인 태그 모두*/
    font-size : 1.5em;
  }
  .menu2, .menu4 { /*class 명이 .menu1 인 태그와 .menu4 인 태그 모두*/
    font-weight : bold;
    font-size: 1.5em;
  }
  li.menu1{ /*class 명이 .menu1 인 li 태그만 */
    color: red;
  }
</style>
```

5) 속성 선택자

속성 선택자는 태그가 가지고 있는 특정한 속성이나 그 속성값을 갖고 있는 요소를 대상으로 한다.

속성 선택자는 여러 속성 지정 방식이 있다.

- (1) 선택자[속성] : 지정한 속성을 가지고 있는 요소

```
img[alt] { /*img 태그중에서 alt 속성이 있는 요소 */
}
```

- (2) 선택자[속성=값] : 속성 값이 특정 값과 같은 요소

```
input[type='password'] { /*input 태그중에 type 속성이 password 인 요소*/
}
```

- (3) 선택자[속성^=값] : 속성 값이 지정한 값으로 시작하는 요소

```
input[type^='te'] { /*input 태그중에 type 속성이 'te'로 시작하는 요소*/
}
```

- (4) 선택자[속성\$=값] : 속성 값이 지정한 값으로 끝나는 요소

```
input[name$='me'] { /*input 태그중에 name 속성이 'me'로 끝나는 요소*/
}
```

- (5) 선택자[속성~=값] : 속성 값이 지정한 값으로 포함하는 요소

```
input[name~='el'] { /*input 태그중에 name 속성이 'el'가 있는 요소*/
}
```

6) 후손 선택자, 자손 선택자

하위 선택자라고도 하며 여기서 자손은 해당 태그 바로 아래에 위치하는 태그만을 말하며 후손은 자손의 자손, 즉 해당 태그의 아래 모든 태그를 말한다. 자손 선택자는 '>', 후손 선택자는 '*'(띄어쓰기)로 구분한다.

```
<style>
  div p { /*후손 선택자(div 밑에 있는 모든 p 에 영향)*/
    font-size : 20px;
  }
  div > p { /*자손 선택자(div 바로 밑에 있는 p 에만 영향)*/
    color : red;
    font-size : 15px;
  }
</style>
```

7) 동위 선택자

형제 선택자라고도 하며 특정 요소의 같은 위치에 있는 요소를 선택하는데

(1) 선택자A ~ 선택자B : 선택자A의 같은 위치에 있는 모든 선택자B

```
h3~div{ /*h3 태그 와 같은 위치의 모든 div 요소들 / h3 제외*/
  font-size : 1.5em;
  color : orangered;
  font-weight : bold;
}
```

(2) 선택자A + 선택자B : 선택자A의 같은 위치에 있는 첫번째 선택자B 하나

```
#msg+div{ /*id 가 msg 인 요소 바로 뒤에 오는 같은 위치의 div 요소 하나*/
  color : purple;
}
```

8) 반응 선택자와 상태 선택자

해당 태그에 마우스를 올려놓거나 클릭, 활성화 등의 일정 상태일 때 특정한 효과를 주는 선택자이다.

(1) 선택자:hover : 해당 선택자에 마우스를 올려놓으면 특정 효과를 준다.

```
<style>
  tr:hover{ /*tr(열)에 커서를 올려놓을때 색상이 바뀜*/
    background-color: green;
    color: wheat;
  }
  td:hover { /*td(행)에 커서를 올려놓을때 밑줄 생성*/
    text-decoration : underline;
  }
</style>
```

(2) 선택자:속성값 : 해당 선택자가 속성값의 상태일 때 특정 효과를 준다.

```
<style>
  input:focus{ /*input 태그가 포커스되었을때 해당 효과를 줌*/
    background-color : lightgoldenrodyellow;
    border : 2px solid lightgreen;
    outline : none;
    border-radius : 9px 9px 9px 9px;
  }
  input:disabled { /*input 태그가 disable 상태일때 색상 red*/
    color: mediumvioletred;
  }
</style>
```

9) 구조선택자

구성요소의 계층에서 특정 위치에 있는 요소를 선택할 때 사용된다.

- (1) **선택자:first-child** : 하위 선택자 요소 중 첫번째 요소

```
#content ul li:first-child{ /*ul 의 자식요소중 첫번째 li*/
}
```

- (2) **선택자:last-child** : 하위 선택자 요소 중 마지막 요소

```
#content ul li:last-child{ /*ul 의 자식요소중 마지막 li*/
}
```

- (3) **선택자:nth-child(n)** : 하위 선택자 요소 중 n번째 요소

```
#content ul li:nth-child(2){ /*ul 의 자식요소중 두번째 li*/
}
```

- (4) **선택자:nth-last-child(n)** : 하위 선택자 요소 중 마지막에서 n번째 요소

```
#content ul li:nth-last-child(2){ /*ul 의 자식요소중 뒤에서 두번째 li*/
}
```

- (5) **선택자: :empty** : 자식요소를 가지고 있지 않은 요소

```
#content ul li:empty{ /*ul 의 자식요소중 자식이 없는 li*/
}
```

10) 부정선택자

부정 선택자는 해당 선택자의 반대되는 요소를 지칭하는 선택자로 모든 다른 선택자들과 함께 사용 할 수 있다.

- **선택자:not(선택자)** : 괄호 안의 값에 해당하는 요소를 제외하고 선택한다.

```
<style>
#content ul li:not(.nate) a{
    /*li 중에 class 가 nate 인것을 제외한 요소에서 a 요소들만*/
}
#content ul input:not([type="password"]){
    /*input 요소중 type 이 password 인것을 제외한 요소들만*/
}
</style>
```