

05

기타 선택자와 가상클래스(p421)

❖ 교재 예제 참고

1. 연결 선택자
2. 속성 선택자
3. 가상클래스

❖ 속성 선택자 : 태그 안에 속성 선택자(p429)

[속성]

지정한 '속성'일 때 스타일 적용

`a[target] {background-color : skyblue;}`

[속성=값]

속성 값이 일치할 때

`a[target="_blank"] {background-color : skyblue;}`

[속성~=값]

여러 값 중 특정 값(단어)이 포함되어 있을 때

`[class ~="button"] {border-radius:10px;}`

[속성|=값]

특정 값이 포함되어 있을 때 (- 포함)

`a[href="en"] {font-style : italic;}`

[속성^=값]

캐럿(^) 지정한 값으로 시작할 때

`a[href^="http//"] {font-style : italic;}`

[속성\$=값]

지정한 값으로 끝날 때

`a[href$=".zip"] {font-style : italic;}`

[속성*=값]

해당 값이 속성값의 일부로 포함되어 있을 때

`p[class *="accent"] {font-style:italic;}`

❖ 가상 클래스(pseudo class)(p437)

- 웹 문서에 실제로 존재하지 않지만 필요에 의해 임의로 **가상의 선택자를 지정**하여 사용하는 것을 말한다.
- 웹 요소를 클릭하거나 마우스 커서를 올려 놓는 등 **특정 동작에 반응** 하도록 스타일 작업 시 사용

링크 관련

링크와 관련된 여러 가지 **상황 정의**를 위한 선택자

위치 기준

문서 안의 요소가 **몇 번째**에 있느냐에 따라 스타일을 다르게 지정할 때 지정하는 선택자

가상 요소

실제로 존재하지 않지만 존재하는 것처럼 스타일 지정하는 가상 선택자(내용의 일부에 적용)

❖ 가상 클래스(pseudo class)

링크 관련

위치 기준

기타


- ① `a:link {color : black; text-decoration : none;}` ← 방문 전 링크
- ② `a:visited {color: black; }` ← 방문 한 링크 스타일
- ③ `a:hover {color : red; text-decoration : underline;}`
← 마우스를 올려 놓을 때 스타일(rollover 효과)
- ④ `a:active {color: black;}` ← 마우스 누르고 있을 때 스타일
- ① ~ ④ 순서에 맞게 정의 해야 함
- ⑤ `a:focus {background-color : yellow;}` ← 초점이 맞춰졌을 때
(마우스 커서 위치, 키보드의 탭 키로 초점 이동 시 적용)

❖ 가상 클래스(pseudo class)

링크 관련

위치 기준

기타

- :root ← 문서 안에 전체적으로 적용할 스타일 지정(css3 부터 정의)
- :nth-child(n) : nth-last-child(n) ← 앞[뒤]에서 몇번째 요소 지정
- :nth-of-type(n) : nth-last-of-type(n) ← 앞[뒤]에서 몇번째 특정 태그 지정
- :first-child : last-child ← 첫 번째[마지막] 자식요소에 지정
- :first-of-type : last-of-type ← 형제관계에서 첫 번째[마지막] 요소에 적용
- :only-child : only-of-type ← 자식요소가 하나일 때, 해당 요소가 하나일 때(css3)
- :target ← 앵커의 목적지가 되는 부분의 스타일 지정
- :enabled, :disabled, :checked 
 - ← 여러 줄의 텍스트 입력 영역에서 사용자가 입력할 수 있도록[없도록] 상태 지정,
 - 체크 선택자는 라디오 버튼이나 체크박스가 체크 되었을 때 적용할 스타일 지정(css3)

❖ 가상 클래스(pseudo class)

링크 관련

위치 기준

기타

- ::before, ::after ← 요소의 앞이나 뒤에 텍스트나 이미지를 추가

```
p::before{content:"내용";}
```

- ::selection ← 마우스로 선택 시 커서가 드래그 되어 선택되는 스타일 지정

```
::selection{color:#00ff00;}
```

- :not ← '()'안에 있는 요소를 제외한' 모든 요소에 스타일 지정

```
:not(h1) {color:blue;}
```