

# 5장

## CSS 선택자 다루기

5.1 기본 선택자 사용하기

5.2 조합 선택자 사용하기

5.3 가상 요소 선택자 사용하기

5.4 가상 클래스 선택자 사용하기

5.5 다양한 선택자 조합하기

## 04월 16일 과제1

1. 앞의 자식 선택자 예제코드를 응용하여  
 lorem2만 빨간색으로 되도록 코드를 작성

lorem 1

lorem 2

lorem 3

# 04월 16일 과제1

```
1  <!DOCTYPE html>
2  <html lang="ko">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>자식 선택자(Child Selector)</title>
8      <style>
9          .box > div > p {
10              color: red;
11          }
12      </style>
13 </head>
14 <body>
15     <div class="box">
16         <p>lorem 1</p>
17         <div>
18             <p>lorem 2</p>
19         </div>
20         <p>lorem 3</p>
21     </div>
22 </body>
23 </html>
```



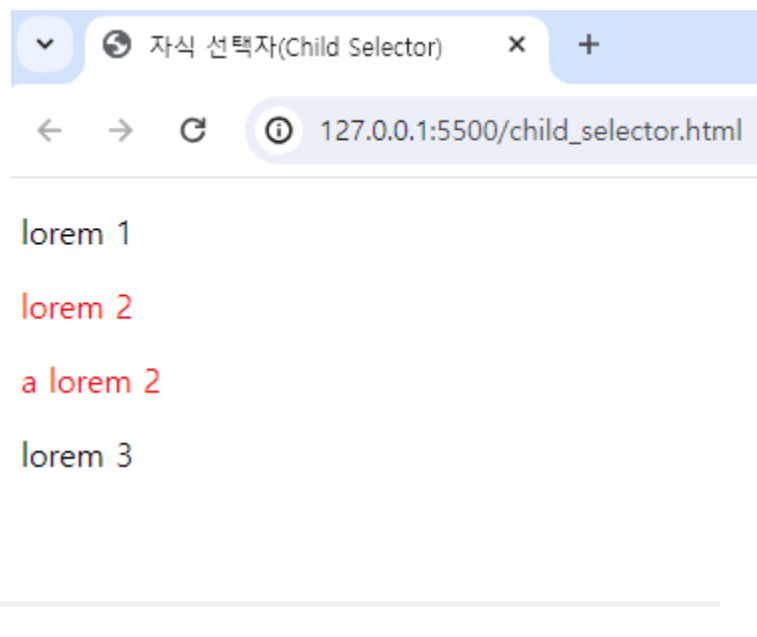
lorem 1

lorem 2

lorem 3

# 04월 16일 과제1

```
1  <!DOCTYPE html>
2  <html lang="ko">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>자식 선택자(Child Selector)</title>
8      <style>
9          .box > div {
10             color: red;
11         }
12     </style>
13 </head>
14 <body>
15     <div class="box">
16         <p>lorem 1</p>
17         <div>
18             <p>lorem 2</p>
19             <a>a lorem 2</a>
20         </div>
21         <p>lorem 3</p>
22     </div>
23 </body>
24 </html>
```



## 5.4 가상 클래스 선택자 사용하기

- 요소의 상태를 이용해 선택자를 지정하는 방법
- 가상 클래스 선택자 앞에 : 기호를 붙임

형식 기준 요소:가상 클래스 선택자{ /\* CSS 코드 \*/ }

가상클래스 선택자가 약 60개정도이지만

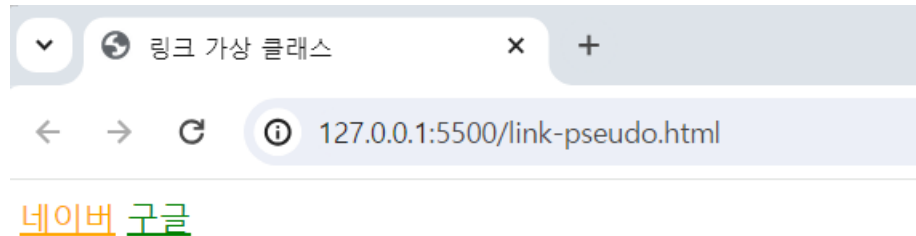
실무에서 자주 사용하는 선택자만 알아보자

### 1. 링크 가상 클래스 선택자

- a태그에서 발생할 수 있는 링크상태를 이용해 선택하는 방법
- :link - 링크를 한 번도 방문한 적 없는 상태
- :visited - 링크를 한 번 이상 방문한 적이 있는 상태

## 5.4 가상 클래스 선택자 사용하기 예시

```
<!DOCTYPE html>
<html lang="ko">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>링크 가상 클래스</title>
    <style>
      a:link {
        color: ■ orange;
      }
      a:visited {
        color: ■ green;
      }
    </style>
  </head>
  <body>
    <a href="https://www.naver.com">네이버</a>
    <a href="https://www.google.com">구글</a>
  </body>
</html>
```



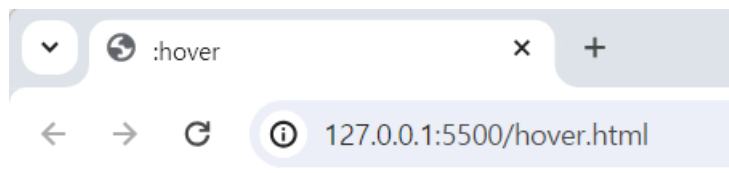
## 5.4 가상 클래스 선택자 사용하기

### 2. 동적 가상 클래스 선택자

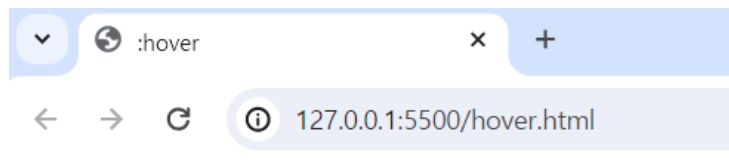
- 사용자의 행동에 따라 동적으로 변하는 상태를 이용해 선택자 지정
- :hover - 마우스를 올린 상태
- :active - 마우스로 클릭한 상태

## 5.4 가상 클래스 선택자 사용하기 예시, 실습

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>:hover</title>
8   <style>
9     p:hover{
10       color: ■ red;
11     }
12   </style>
13 </head>
14 <body>
15   <p>hover</p>
16 </body>
17 </html>
```



hover



hover



## 5.4 가상 클래스 선택자 사용하기

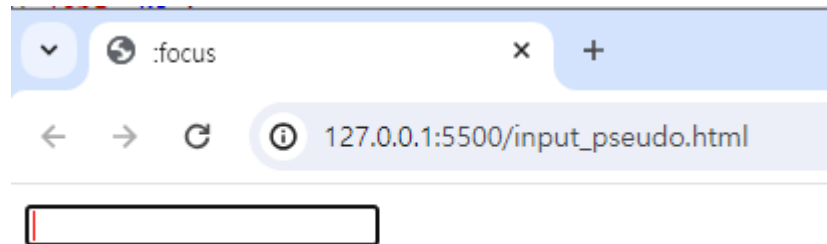
### 3. 입력 요소 가상 클래스 선택자

- 입력요소의 특정상태를 이용하여 선택자로 지정
  - :focus - 입력 요소에 커서가 활성화된 상태
  - :checked - 체크박스 요소에 체크한 상태
  - :disabled - 상호작용 요소가 비활성화된 상태
  - :enabled - 상호작용 요소가 활성화된 상태

## 5.4 가상 클래스 선택자 사용하기 예시

- :focus

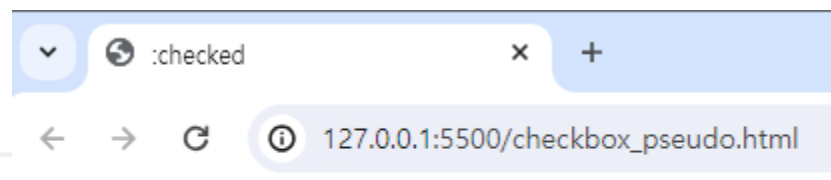
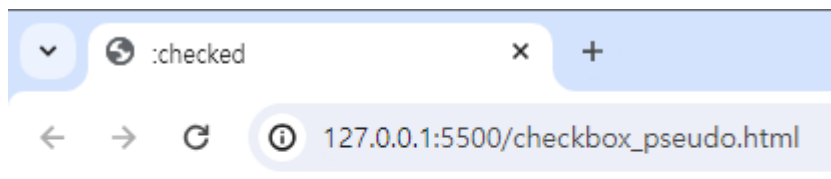
```
1  <!DOCTYPE html>
2  <html lang="ko">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>:focus</title>
8      <style>
9          input:focus{
10             color: ■ red;
11         }
12     </style>
13 </head>
14 <body>
15     <input type="text">
16 </body>
17 </html>
```



## 5.4 가상 클래스 선택자 사용하기 예시, 실습

- :checked

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta http-equiv="X-UA-Compatible" content="IE=edge">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <title>:checked</title>
8  <style>
9    input:checked + label{
10      color: red;
11    }
12  </style>
13 </head>
14 <body>
15   <input type="checkbox" id="userid">
16   <label for="userid">아이디</label>
17 </body>
18 </html>
```

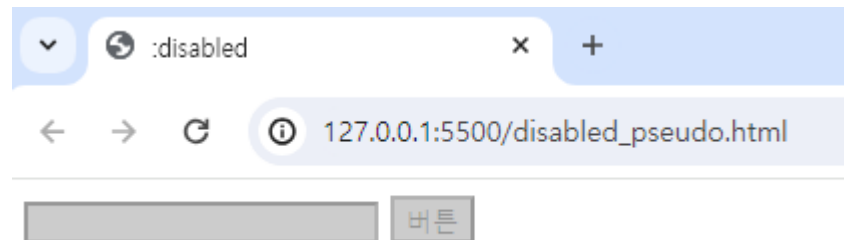


아이디

## 5.4 가상 클래스 선택자 사용하기 예시

- :disabled

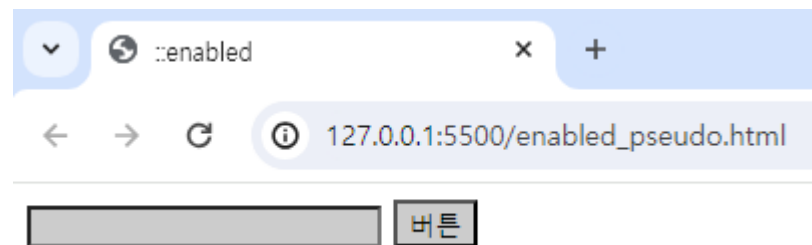
```
1  <!DOCTYPE html>
2  <html lang="ko">
3    <head>
4      <meta charset="UTF-8" />
5      <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7      <title>:disabled</title>
8      <style>
9        input:disabled,
10        button:disabled {
11          background-color: #ccc;
12        }
13      </style>
14    </head>
15    <body>
16      <input type="text" disabled />
17      <button disabled>버튼</button>
18    </body>
19  </html>
20
```



## 5.4 가상 클래스 선택자 사용하기 예시

- :enabled

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8" />
5    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7    <title>::enabled</title>
8    <style>
9      input:enabled,
10     button:enabled {
11       background-color: #ccc;
12     }
13   </style>
14 </head>
15 <body>
16   <input type="text">
17   <button>버튼</button>
18 </body>
19 </html>
20
```



## 5.4 가상 클래스 선택자 사용하기

### 4. 구조적 가상 클래스 선택자

- HTML 태그의 사용위치, 다른 태그와의 관계에 따라 요소 선택

종류	설명
:first-child	첫 번째 자식 태그
:last-child	마지막 자식 태그
:nth-child(n)	n번째 자식 태그
:nth-last-child(n)	끝에서 n번째 자식 태그
:nth-of-type(n)	n번째 특정 자식 태그
:nth-last-of-type(n)	끝에서 n번째 특정 자식 태그
:first-of-type	부모의 첫 번째 특정 자식 태그
:last-of-type	부모의 마지막 특정 자식 태그

## 5.4 가상 클래스 선택자 사용하기 예시, 실습

- :first-child 와 :last-child

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>:first-child, :last-child</title>
8   <style>
9     li:first-child{
10       color: red;
11     }
12     li:last-child{
13       color: red;
14     }
15   </style>
16 </head>
17 <body>
18   <ul>
19     <li>lorem 1</li>
20     <li>lorem 2</li>
21     <li>lorem 3</li>
22     <li>lorem 4</li>
23   </ul>
24 </body>
25 </html>
```

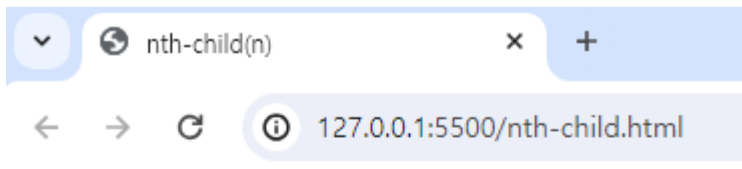


- lorem 1
- lorem 2
- lorem 3
- lorem 4

## 5.4 가상 클래스 선택자 사용하기 예시, 실습

- :nth-child(n)

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>nth-child(n)</title>
8   <style>
9     p:nth-child(1){
10       color: red;
11     }
12   </style>
13 </head>
14 <body>
15   <p>lorem 0</p>
16   <div>
17     <p>lorem 1</p>
18     <p>lorem 2</p>
19   </div>
20   <div>
21     <p>lorem 3</p>
22     <p>lorem 4</p>
23   </div>
24 </body>
25 </html>
```

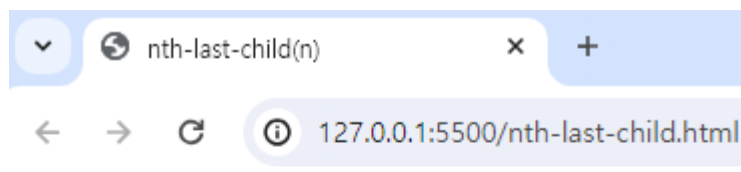




## 5.4 가상 클래스 선택자 사용하기 예시, 실습

- :nth-last-child(n)

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>nth-last-child(n)</title>
8   <style>
9     p:nth-last-child(2){
10       color: red;
11     }
12   </style>
13 </head>
14 <body>
15   <p>lorem 0</p>
16   <div>
17     <p>lorem 1</p>
18     <p>lorem 2</p>
19   </div>
20   <div>
21     <p>lorem 3</p>
22     <p>lorem 4</p>
23   </div>
24 </body>
25 </html>
```



lorem 0

lorem 1

lorem 2

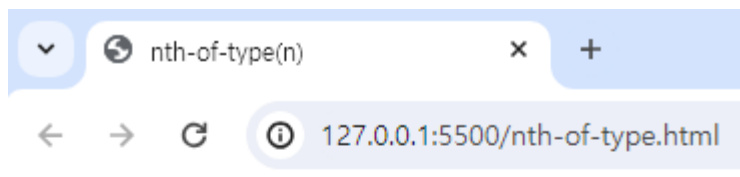
lorem 3

lorem 4

## 5.4 가상 클래스 선택자 사용하기 예시, 실습

- :nth-of-type(n)

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>nth-of-type(n)</title>
8   <style>
9     p:nth-of-type(2){
10       color: red;
11     }
12   </style>
13 </head>
14 <body>
15   <p>lorem 0</p>
16   <div>
17     <p>lorem 1</p>
18     <span>span 1</span>
19     <p>lorem 2</p>
20   </div>
21   <div>
22     <p>lorem 3</p>
23     <span>span 2</span>
24     <span>span 3</span>
25     <p>lorem 4</p>
26   </div>
27 </body>
28 </html>
```



lorem 0

lorem 1

span 1

lorem 2

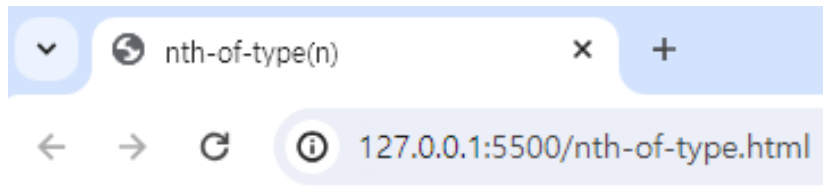
lorem 3

span 2 span 3

lorem 4

# 과제

- :nth-of-type(n)을 이용하여 span 3만 파란색으로 나오도록 수정 코딩하면 캡처, 결과화면 캡처 과제물 제출



lorem 0

lorem 1

span 1

lorem 2

lorem 3

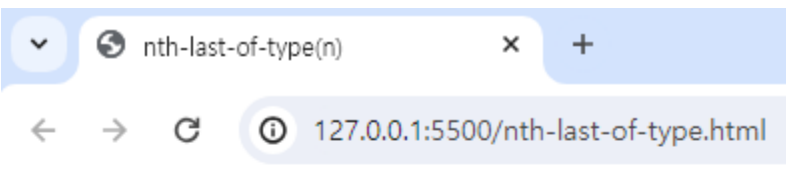
span 2 span 3

lorem 4

## 5.4 가상 클래스 선택자 사용하기 예시, 실습

- :nth-last-of-type(n)

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>nth-last-of-type(n)</title>
8   <style>
9     p:nth-last-of-type(1){
10       color: red;
11     }
12   </style>
13 </head>
14 <body>
15   <p>lorem 0</p>
16   <div>
17     <p>lorem 1</p>
18     <span>span 1</span>
19     <p>lorem 2</p>
20   </div>
21   <div>
22     <p>lorem 3</p>
23     <span>span 2</span>
24     <span>span 3</span>
25     <p>lorem 4</p>
26   </div>
27 </body>
28 </html>
```



lorem 0

lorem 1

span 1

lorem 2

lorem 3

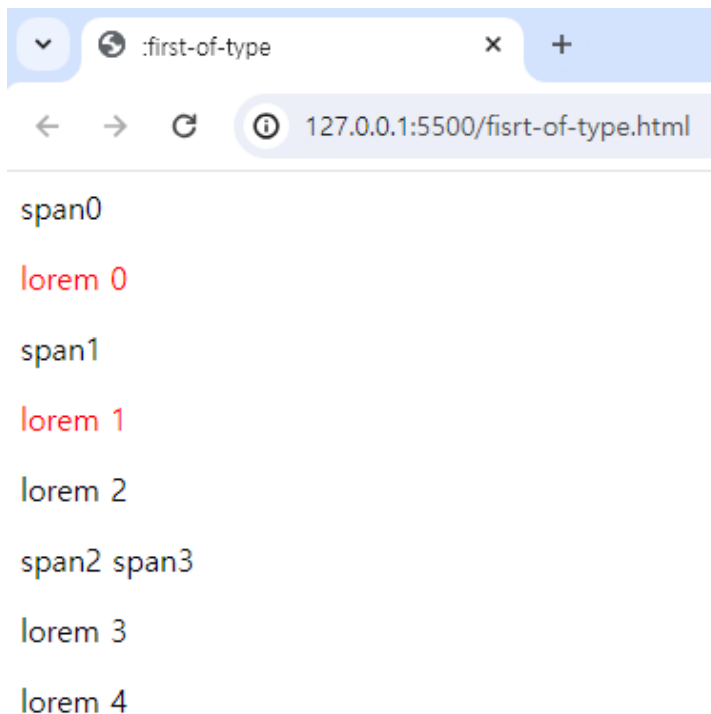
span 2 span 3

lorem 4

## 5.4 가상 클래스 선택자 사용하기 예시, 실습

- :first-of-type

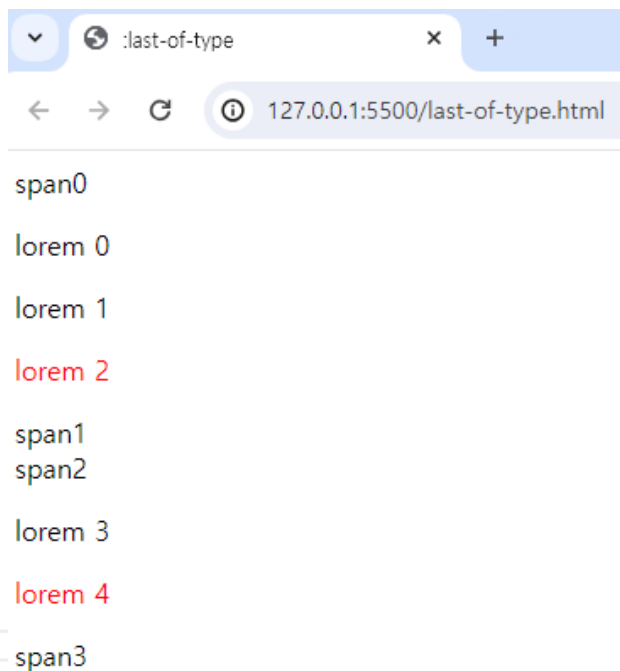
```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>:first-of-type</title>
8   <style>
9     p:first-of-type{
10       color: red;
11     }
12   </style>
13 </head>
14 <body>
15   <span>span0</span>
16   <p>lorem 0</p>
17   <div>
18     <span>span1</span>
19     <p>lorem 1</p>
20     <p>lorem 2</p>
21   </div>
22   <span>span2</span>
23   <span>span3</span>
24   <p>lorem 3</p>
25   <p>lorem 4</p>
26 </body>
27 </html>
```



## 5.4 가상 클래스 선택자 사용하기 예시, 실습

- :last-of-type

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>:last-of-type</title>
8   <style>
9     p:last-of-type{
10       color:red;
11     }
12   </style>
13 </head>
14 <body>
15   <span>span0</span>
16   <p>lorem 0</p>
17   <div>
18     <p>lorem 1</p>
19     <p>lorem 2</p>
20     <span>span1</span>
21   </div>
22   <span>span2</span>
23   <p>lorem 3</p>
24   <p>lorem 4</p>
25   <span>span3</span>
26 </body>
27 </html>
```



## 5.5 다양한 선택자 조합하기

- 선택자는 조합해서 사용할 수 있음

예

아래 2가지는 실무에서 흔하게 사용

```
div.box{} /* class 속성값이 box인 div 태그 */
```

```
section#main /* id 속성값이 main인 section 태그 */
```

일반적이지는 않지만 아래와 같은 형태도 가능

```
#main.box{}
```

```
/* id 속성값이 main이고, class 속성값이 box인 요소 */
```

## 5.5 다양한 선택자 조합하기

가상 클래스와 하위선택자, 자식선택자를 조합한 형태도 가능

```
div:hover button{}
```

/\* div 태그에 마우스를 올린(hover) 상태일 때, 해당 div 태그 하위에 있는 button 태그 선택 \*/

```
div:hover > button{}
```

/\* div 태그에 마우스를 올린(hover) 상태일 때, 해당 div 태그와 자식 관계에 있는 button 태그 선택 \*/



# 6장

## CSS 필수 속성 다루기

6.1 CSS의 특징 살펴보기

6.2 텍스트 속성으로 텍스트 꾸미기

6.3 박스 모델을 구성하는 속성 다루기

6.4 배경 속성으로 요소의 배경 설정하기

6.5 위치 속성으로 HTML 요소 배치하기

6.6 전환 효과 속성 적용하기

6.7 애니메이션 속성으로 전환 효과 제어하기

6.8 변형 효과 적용하기

6.9 웹 폰트와 아이콘 폰트 사용하기

## 6.1 CSS의 특징 살펴보기

### 1. 기본 스타일 시트

- 웹 브라우저에 기본으로 내장된 스타일 시트

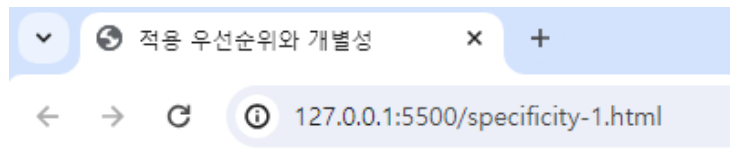
### 2. 적용 우선순위와 개별성

- 단계적 적용 : 같은 태그에 여러 스타일이 적용되더라도 단계적으로 적용되어 결국 마지막에 영향을 주는 하나의 스타일만 적용
- 적용 우선순위 : 같은 태그에 스타일 속성이 중복으로 작성됐을 때, 어느 속성을 적용할지 결정하는 기준
- 개별성 규칙의 점수에 따라 계산

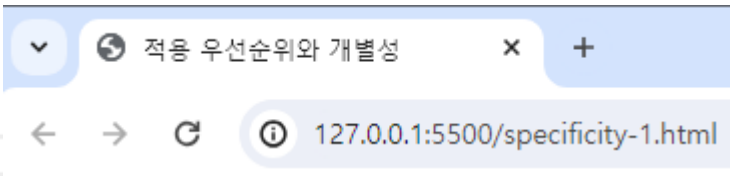
# 6.1 CSS의 특징 살펴보기

## 2. 적용 우선순위와 개별성 예시 -1

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>적용 우선순위와 개별성</title>
8   <style>
9     h1{
10       font-size:16px;
11       font-weight:normal;
12     }
13   </style>
14 </head>
15 <body>
16   <h1>h1 tag</h1>
17 </body>
18 </html>
```



**h1 tag**

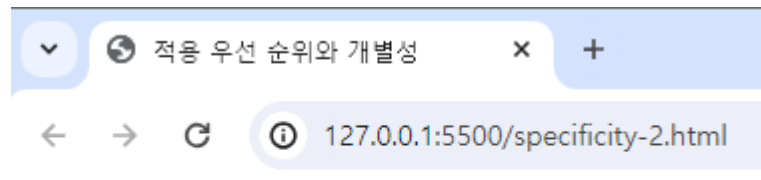


h1 tag

# 6.1 CSS의 특징 살펴보기, 실습

## 2. 적용 우선순위와 개별성 예시 -2

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>적용 우선 순위와 개별성</title>
8      <style>
9          p{color: red;}
10         p{color: blue;}
11         p{color: orange;}
12     </style>
13 </head>
14 <body>
15     <p>specificity</p>
16 </body>
17 </html>
```



specificity

## 6.1 CSS의 특징 살펴보기

### 2. 적용 우선순위와 개별성 -3

선택자	예	점수
전체 선택자	*	0
태그 선택자	div, p, h1	1
가상 요소 선택자	::before, ::after	1
클래스 선택자	.box, .title	10
가상 클래스 선택자	:hover, :visited, :link	10
아이디 선택자	#title, #main	100
인라인 스타일	style="color:red"	1,000

## 6.1 CSS의 특징 살펴보기

### 2. 적용 우선순위와 개별성 -3 예시

```
<style>
  .title{color:blue} /* 0, 0, 1, 0 */
  h1{color:red} /* 0, 0, 0, 1 */
</style>
</head>
<body>
  <h1 class="title">h1</h1>
</body>
```

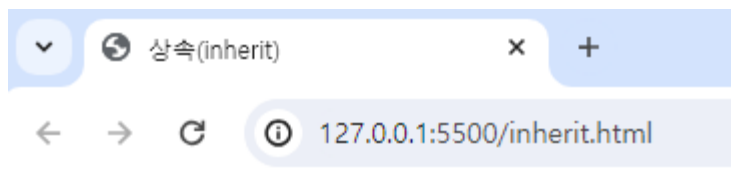
```
<style>
  nav > h2{} /* 0, 0, 0, 2 */
  nav .title{} /* 0, 0, 1, 1 */
  #main .sub{} /* 0, 1, 1, 0 */
  a:hover{} /* 0, 0, 1, 1 */
  a:hover span::first-letter{} /* 0, 0, 1, 3 */
</style>
```

## 6.1 CSS의 특징 살펴보기

### 3. 상속

- 부모 요소에 적용된 스타일을 자식 요소가 그대로 물려받는 것

```
1 <!DOCTYPE html>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>상속(inherit)</title>
8   <style>
9     div{
10       color: red;
11     }
12   </style>
13 </head>
14 <body>
15   <div>
16     <p>inherit</p>
17   </div>
18 </body>
19 </html>
```



inherit

## 6.1 CSS의 특징 살펴보기

### 4. 단위

- 절대 단위(CSS에서 사용할 수 있는 가장기본단위)
  - px : 모니터 화면을 구성하는 사각형 1개의 크기
- 상대 단위
  - % : 부모 요소의 속성값에 따라 상대적인 크기
  - em : 부모 요소의 텍스트 크기에 상대적인 크기
  - rem : html 태그의 텍스트 크기에 상대적인 크기
  - vw : 뷰포트의 너비를 기준으로 상대적인 크기
  - vh : 뷰포트의 높이를 기준으로 상대적인 크기



## 6.1 CSS의 특징 살펴보기

### 4. 단위 예시

- 상대 단위
  - %

---

```
.parent{
  font-size:16px;
}
.child{
  font-size:80%; /* 16px(부모 font-size 크기) * 0.8 = 12.8px */
}
.child-to-child{
  font-size:80%; /* 12.8px(부모 font-size 크기) * 0.8 = 10.24px */
}
```

---

## 6.1 CSS의 특징 살펴보기

### 4. 단위 예시

- 상대 단위
  - em

---

```
.parent{
  font-size:16px;
}
.child{
  font-size:2em; /* 16px(부모 font-size 크기) * 2em = 32px */
}
.child-to-child{
  font-size:2em; /* 32px(부모 font-size 크기) * 2em = 64px */
}
```

---

## 6.1 CSS의 특징 살펴보기

### 4. 단위 예시

- 상대 단위
  - rem

---

```
html{
  font-size:1rem;
}
.parent{
  font-size:1rem; /* 16px * 1rem = 16px */
}
.child{
  font-size:2rem; /* 16px * 2rem = 32px */
}
```

---

## 6.1 CSS의 특징 살펴보기

### 4. 단위 예시

- 상대 단위
  - Vw 와 vh

---

```
/* 기준 뷰포트 너비 900px */  
.parent{  
  font-size:1vw; /* 900px * 1/100 = 9px */  
}  
.child{  
  font-size:2vw; /* 900px * 2/100 = 18px */  
}
```

---

---

```
/* 기준 뷰포트 높이 400px */  
.parent{  
  font-size:1vh; /* 400px * 1/100 = 4px */  
}  
.child{  
  font-size:2vh; /* 400px * 2/100 = 8px */  
}
```

---

## 6.1 CSS의 특징 살펴보기

### 5. 색상 표기법

- 키워드 표기법 : 색상의 영문명을 속성값으로 사용
- RGB 표기법 : Red, Green, Blue, alpha 값을 0~255로 표기
- HEX 표기법 : Red, Green, Blue에 해당하는 값을 각각 16진수로 변환해 00~ff로 표기

그림 6-7 RGB 색상 표기법의 rgb와 rgba 형식

`rgb(red, green, blue)`

`rgba(red, green, blue, alpha)`

그림 6-9 HEX 표기법

`#RRGGBB`

## 6.1 CSS의 특징 살펴보기

### 5. 색상 표기법

색상 팔레트

#FF00DD

조희


색상 코드

RGB

R

255

G

0

B

221

HSB

H

308

°

S

100

%

B

100

%

CMYK

C

0

%

M

100

%

Y

13

%

K

0

%

\* 직접입력 또는 색상팔레트에서 색을 선택하세요.

조희