



DoiT Web Project Team

3. CSS

Cascading Style Sheets

- HTML 문서가 어떻게 보여질 지 정하기 위해 사용하는 언어.
- 요소에 스타일을 부여해 크기, 색깔, 위치, 애니메이션 등을 지정할 수도 있다.

구성

```
p {  
  font-size: 20px;  
  color: #ff0000;  
}
```

Hello, world!

- `p` 태그를 선택해 `font-size` 와 `color` 라는 속성을 사용.
- 속성 값으로 각각 `30px` 와 `#ff0000` 을 부여.

Inline Style

```
<!-- index.html -->
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Hello</title>
  </head>
  <body>
    <p style="font-size: 20px; color: #ff0000;">Hello, world!</p>
  </body>
</html>
```

Hello, world!

- HTML 코드와 스타일 코드가 섞여 난잡해짐.
- 진짜 불가피한 상황이 아니라면 **절대 이렇게 하지 마세요!**

Internal Style Sheet

```
<!-- index.html -->
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Hello</title>
    <style>
      p {
        font-size: 20px;
        color: #ff0000;
      }
    </style>
  </head>
  <body>
    <p>Hello, world!</p>
  </body>
</html>
```

Hello, world!

External Style Sheet

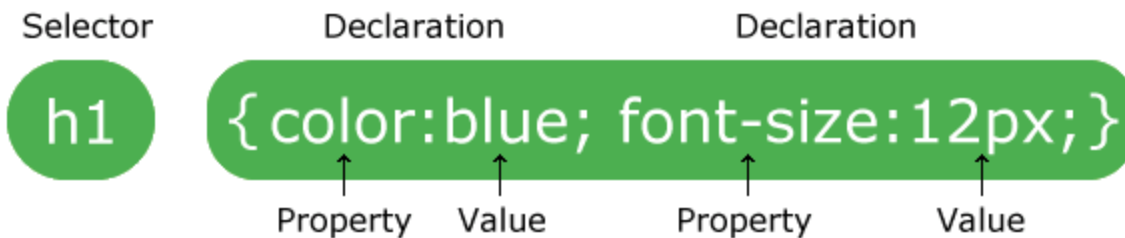
```
<!-- index.html -->
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Hello</title>
    <link rel="stylesheet" type="text/css" href="./index.css">
  </head>
  <body>
    <p>Hello, world!</p>
  </body>
</html>
```

```
/* index.css */
p {
  font-size: 20px;
  color: #ff0000;
}
```

Hello, world!



Anatomy



- Selector (선택자)
- Property (속성)
- Value (값)

Type Selector

```
p {  
  font-size: 20px;  
  color: #ff0000;  
}
```

- HTML 문서 상에서 스타일을 적용할 요소를 선택하는 구문.
- 위 코드에서는 `p` 태그를 선택.
- 문서 상의 모든 `p` 태그에 스타일이 적용된다.

ID Selector

```
...  
<body>  
  <p>Paragraph 1</p>  
  <p>Paragraph 2</p>  
  <p>Paragraph 3</p>  
</body>  
...
```

Paragraph 2에만 밑줄을 긋고 싶다면?

ID Selector

```
...  
<body>  
  <p>Paragraph 1</p>  
  <p id="main-text">Paragraph 2</p>  
  <p>Paragraph 3</p>  
</body>  
...
```

- 요소의 속성으로 `id` 를 추가. (요소에 이름을 붙이는 것)
- 하나의 HTML 문서 내에 동일한 id가 존재할 수 없다.

ID Selector

```
...  
<body>  
  <p>Paragraph 1</p>  
  <p id="main-text">Paragraph 2</p>  
  <p>Paragraph 3</p>  
</body>  
...
```

```
#main-text {  
  text-decoration: underline;  
}
```

- 특정 id를 선택하려면 `#id` 형태로 셀렉터를 사용.

Class Selector

```
...  
<body>  
  <p>Paragraph 1</p>  
  <p id="main-text">Paragraph 2</p>  
  <p>Paragraph 3</p>  
</body>  
...
```

Paragraph 1, Paragraph 3의 배경색을 노란색으로 바꾸고 싶다면?

Class Selector

```
...  
<body>  
  <p class="highlighted-text">Paragraph 1</p>  
  <p id="main-text">Paragraph 2</p>  
  <p class="highlighted-text">Paragraph 3</p>  
</body>  
...
```

- 요소의 속성으로 `class` 를 추가. (요소의 카테고리를 정하는 것)
- class는 중복 가능하다.

Class Selector

```
...  
<body>  
  <p class="highlighted-text">Paragraph 1</p>  
  <p id="main-text">Paragraph 2</p>  
  <p class="highlighted-text">Paragraph 3</p>  
</body>  
...
```

```
.highlighted-text {  
  background-color: #ffff00;  
}
```

- 특정 class를 선택하려면 `.class` 형태로 셀렉터를 사용.

ID & Class Selector

```
...  
<body>  
  <p>Paragraph 1</p>  
  <p id="main-text" class="highlighted-text">Paragraph 2</p>  
  <p class="highlighted-text">Paragraph 3</p>  
</body>  
...
```

```
#main-text { text-decoration: underline; }  
.highlighted-text { background-color: #ffff00; }
```

Paragraph 1

Paragraph 2

Paragraph 3

Combinators and groups of selectors

Group of selectors: A, B

```
h1, h2, h3, h4, h5, h6 {  
  font-family: '함초롱바탕', serif;  
}
```

- 요소의 특별한 상태를 지정.
- 셀렉터에 추가해서 사용.

Child selector: A > B

```
...  
<body>  
  <h1>Lorem ipsum <i>dolor sit</i> amet</h1>  
  <p>Lorem ipsum <i>dolor sit</i> amet</p>  
</body>  
...
```

```
h1 > i {  
  color: #ff0000;  
}
```

Lorem ipsum *dolor sit* amet

Lorem ipsum *dolor sit* amet

General sibling selector: A ~ B

```
...  
<body>  
  <h1>Lorem ipsum <i>dolor sit</i> amet</h1>  
  <p>Lorem ipsum <i>dolor sit</i> amet</p>  
  <p>Lorem ipsum <i>dolor sit</i> amet</p>  
</body>  
...
```

```
p ~ p {  
  color: #ff0000;  
}
```

Lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet

Pseudo-classes

```
a {  
  color: blue;  
}  
  
a:hover {  
  color: red;  
}  
  
a:active {  
  background-color: yellow;  
}
```

- 요소의 특별한 상태를 지정.
- 셀렉터에 추가해서 사용.

Inheritance

```
...  
<body>  
  <p id="text">Lorem ipsum <i>dolor sit</i> amet</p>  
</body>  
...
```

```
#text {  
  color: red;  
}
```

Lorem ipsum *dolor sit* amet

- 속성 값이 자식 요소로 상속된다.

Inheritance

```
...  
<body>  
  <p id="text">Lorem ipsum <i>dolor sit</i> amet</p>  
</body>  
...
```

```
#text {  
  border: 1px solid;  
}
```

Lorem ipsum *dolor sit* amet

- `border` 처럼 상속되지 않는 속성도 있다.

Cascade

- 중요도: 어디에 선언되었는지에 따라 우선순위가 달라진다.
- 명시도: 대상을 명확하게 특정할 수록 우선순위가 높아진다.
- 순서: 선언된 순서에 따라 우선 순위가 적용된다.



중요도

1. Author style sheets
2. User style sheets
3. Browser's default styles

위에서 아래로 스타일이 덮어쓰기된다.



명시도

1순위: Inline Style

2순위: ID Selector

3순위: Class Selector

4순위: Type Selector

5순위: 상속된 스타일 속성



순서

나중에 선언된 속성이 앞서 선언된 속성을 덮어쓴다.

```
...  
<body>  
  <p>Lorem ipsum dolor sit amet</p>  
</body>  
...
```

```
p { color: red; }  
p { color: blue; }
```

Lorem ipsum dolor sit amet

Cascade

!important

!important 를 붙여 무조건 적용되도록 할 수 있다.

```
...  
<body>  
  <p>Lorem ipsum dolor sit amet</p>  
</body>  
...
```

```
p { color: red; !important }  
p { color: blue; }
```

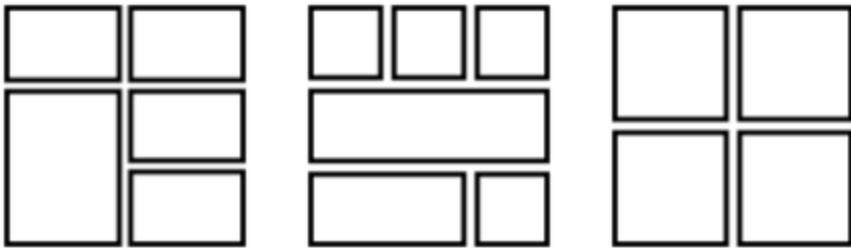
!important Lorem ipsum dolor sit amet



DoiT Web Project Team

3.5. Layout

Web Layout



- 페이지의 구역을 어떻게 나눌 것인가?
- 어디에 무엇을 어떻게 배치할 것인가?

<div>

div (header)

div (main)

div (footer)

Block element & Inline element

```
<div>...</div>  
<div>...</div>  
<div>...</div>
```

div 1

div 2

div 3

```
<span>...</span>  
<span>...</span>  
<span>...</span>
```

div 1div 2div 3

Block element & Inline element

Block element

```
p { display: block; }
```

test

test

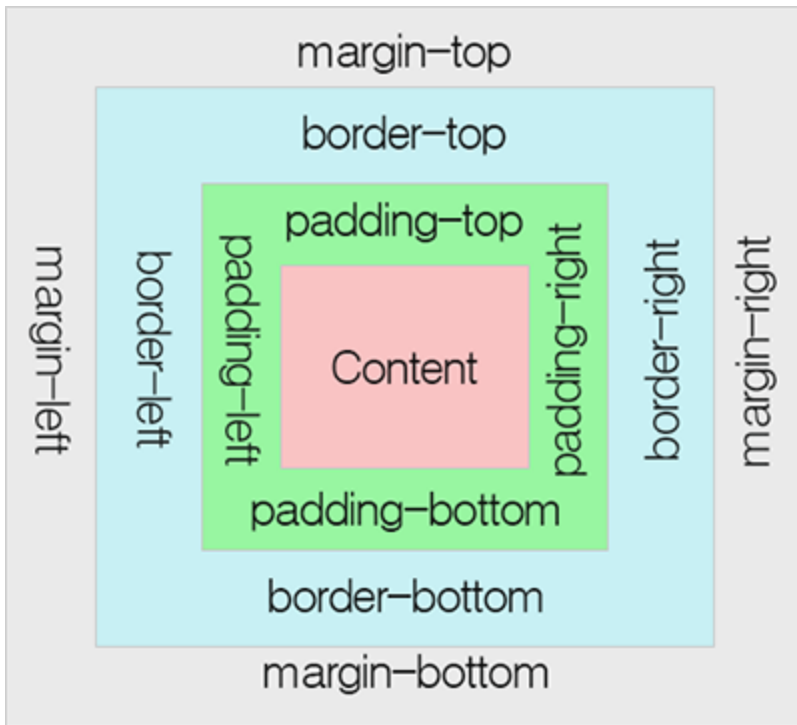
test

Inline element

```
p { display: inline; }
```

testtesttest

Box model



- HTML 요소가 가진 속성.

Layout

여기를 참고해보세요 👉 [CSS 레이아웃을 배웁시다](#)