

# Web Standard Publishing

## 04. CSS – introduce

# CSS의 이해

---

1. CSS란?
2. CSS의 과거와 현재
3. CSS 사용의 의의

# 1. CSS란?

---

1. CSS 또는 캐스케이딩 스타일 시트(Cascading Style Sheet)는 마크업 언어가 실제로 표시되는 방법을 기술하는 언어로, HTML과 XHTML에 주로 쓰이며, XML에서도 사용할 수 있음.
2. CSS는 W3C의 표준이며, 레이아웃과 스타일을 정의할 때의 자유도가 높은 언어임.
3. CSS는 웹 사이트에서 사용되는 스타일을 지정하기 위한 언어로, 스타일은 웹의 독창적인 개념이 아니라 웹 이전부터 워드프로세스 등에 사용했던 개념임.

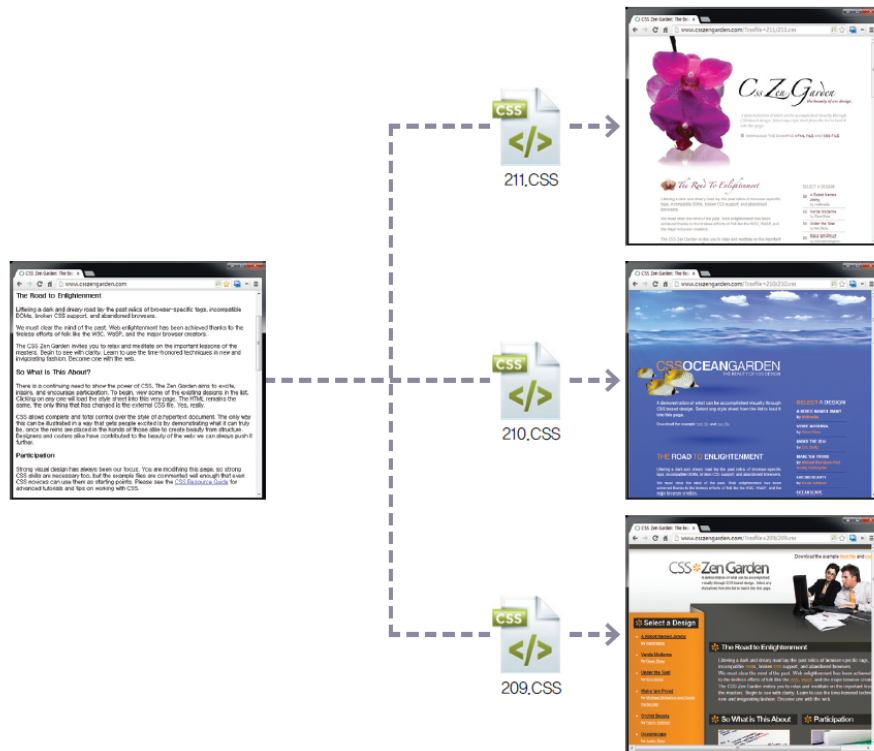
## 2. CSS의 과거와 현재

---

1. 1996년 W3C의 주도하에 첫 번째 CSS 버전인 CSS Level 1이 발표.
2. 1998년에 CSS Level 2가 등장하면서 대부분의 웹 브라우저들이 CSS Level 2를 지원하기 시작
3. 그 후 CSS Level 2의 버그를 수정한 CSS Level 2.1이 2006년에 발표되면서 현재까지 표준으로 사용되고 있음.

### 3. CSS사용의 의의

1. 스타일 사용의 중요한  
의의는 문서의 구조와  
표현을 분리할 수 있다는  
점을 들 수 있으며 이는  
구조와 표현을  
분리함으로써 문서  
구조의 수정 없이  
스타일의 변경만으로  
다양한 표현을 할 수



### 3. CSS사용의 의의

---

2. 웹 문서에 있어 구조와 표현의 분리는 워드프로세서 문서보다 중요할 수 있음. 그 이유 중 하나는 웹 문서가 기계적으로 해석될 수 있는 가능성이 높아지기 때문임.
3. 기계적으로 해석되는 것은 CSS가 아니라 HTML이지만, 구조와 무관한 표현적 요소를 CSS로 분리함으로써 HTML은 간결해지고 더욱 구조 화될 수 있기 때문임.

# CSS 서식

---

1. 기본 문법 (Rule Set)
2. 적용

# 1. 기본 문법 (Rule Set)

---

**selector** { **property : value ; property : value ;** }

선택자                      선언부

## 1. 선택자(selector)

- 스타일을 적용시킬 엘리먼트, id, class 이름을 지정

## 2. 선언부(declaration block)

- 속성과 속성값을 이용하여 스타일을 선언



# 1. 기본 문법 (Rule Set)

---

**selector** { **property** : **value** ; **property** : **value** ; }

속성                      속성값                      속성                      속성값

1. 속성(property)
  - 스타일의 종류
2. 속성 값(value)
  - 속성이 가질 수 있는 값
3. 선언부는 세미콜론(;)으로 속성과 속성 값을 구분하여 여러 개의 선언(declaration)을 지정할 수 있음.

## 2. 적용

---

### 1. External

- CSS파일을 외부에 생성하여 HTML 문서에 연결하는 방식

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="utf-8" />
  <link rel="stylesheet" type="text/css" href="css/style.css" />
  <style>
    @import url(css/style.css);
  </style>
</head>
<body>
  ~~~
```

CSS 2.1

## 2. 적용

---

### 2. Embedded

- 특정 웹페이지 내에 CSS코드를 직접 포함하여 스타일이 적용되도록 하는 방법

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="utf-8" />
  <style>
    h1 {color:#000;}
    p {margin:10px 0;}
  </style>
</head>
<body>
  ~~~
```

CSS 2.1

## 2. 적용

---

### 3. Inline

- 특정 HTML 요소에 style 속성을 사용하여 CSS 코드를 선언하는 방식

```
~~
</head>
<body>
  <h1 style="color:#000">Inline 방식</h1>
  <p style="margin:10px 0;">style 요소 내에 직접 CSS 코드를 포함한다</p>
</body>
</html>
```

CSS 2.1

# CSS 선택자

---

1. 전체 선택자
2. 요소 선택자
3. class, id 선택자
4. 가상요소, 가상 클래스
5. 선택자 조합
6. 선택자 그룹화

# 1. 전체 선택자

---

- 모든 요소를 선택하는 방법으로 "\*"를 선택자로 선언한다
- 유니버설 선택자

```
* {  
    margin:0;  
    padding:0;  
}
```

CSS 2.1

## 2. 요소 선택자

---

- HTML 요소를 선택하는 방법
- type선택자, 태그 선택자
- h1, p, div 등

```
div {margin:0;padding:0;}  
p {color:#333;font-weight:bold;text-decoration:underline;}
```

CSS 2.1

### 3. class, id 선택자

---

- XHTML 요소에 class나 id 속성을 부여한 경우 해당 class명이나 id명으로 접근하여 선택
- class명은 여러 번 사용할 수 있기 때문에 여러 가지 요소들을 한꺼번에 선택하여 일관된 스타일을 지정할 때 사용
- 특정 id값을 가지고 있는 유일한 요소에 접근하여 스타일을 지정할 때는 id 선택자 형식을 사용



### 3. class, id 선택자

---

- class 선택자

```
.box {font-size:14px;color:#FFF;background-color:#000;}  
h1.tit {font-size:20px;font-weight:normal;letter-spacing:-1px;}
```

CSS 2.1

- id 선택자

```
#box {font-size:14px;color:#FFF;background-color:#000;}  
h1#tit {font-size:20px;font-weight:normal;letter-spacing:-1px;}
```

CSS 2.1

## 4. 가상 요소, 가상 클래스

---

- 실제로 존재하지 않는 요소를 마치 존재하는 것 처럼 이용하는 방법
- 가상 요소 선택자
  - 콘텐츠 영역의 앞, 뒤에 가상 콘텐츠 추가
  - :before, :after, :first-line, :first-letter
- 가상 클래스 선택자
  - 상황에 따라 구분
  - :first-child, :link, :visited, :hover, :active, :focus

## 4. 가상 요소, 가상 클래스 - 가상 요소 선택자

---

- :first-line
  - 요소의 첫번째 줄에 있는 콘텐츠만 선택하여 스타일을 지정
  - 크기에 따라 유동적
- :first-letter
  - 요소의 첫 글자만 선택하여 스타일을 지정

```
p:first-line {color:red;}  
p:first-letter {color:blue;}
```

CSS 2.1

## 4. 가상 요소, 가상 클래스 - 가상 요소 선택자

---

- :before, :after
  - 가상의 콘텐츠를 이전이나 이후에 추가할 수 있는 선택자
  - content 속성에 원하는 문자열을 지정
  - 스타일을 위해서 사용. 문서에 내용을 추가하는 용도로 사용하지 않도록 주의

```
p:before {content:"시작";color:red;}  
p:after {content:"끝";color:blue;}
```

CSS 2.1

## 4. 가상 요소, 가상 클래스 - 가상 클래스 선택자

---

- :first-child
  - 첫 번째 자식 요소에만 스타일을 적용할 수 있는 선택자
  - 두 번째 자식요소부터는 스타일이 적용되지 않음

```
ul:first-child {color:red;}
```

CSS 2.1

## 4. 가상 요소, 가상 클래스 - 가상 클래스 선택자

---

- :link, :visited, :hover, :active, :focus
  - :link 방문하지 않은 링크
  - :visited 방문한 링크
  - :hover 마우스 오버
  - :active 활성화
  - :focus 포커스가 생긴 경우

```
a {text-decoration:none;}
a:link {color:blue;}
a:visited {color:purple;}
a:hover {color:orange;}
a:active {color:red;}
a:focus {background-color:yellow;}
```

CSS 2.1

## 5. 선택자 조합

---

- 하위 선택자
  - 상위 요소가 포함하고 있는 모든 하위 요소에 스타일을 지정할 때
  - 선택자와 선택자를 공백으로 구분하여 선언

```
div p {color:gray;}
```

CSS 2.1

- 자식 선택자
  - 부모 요소의 바로 다음에 있는 자식 요소에 스타일을 적용
  - 선택자와 선택자를 왼쪽 부등호로 구분하여 선언

```
Div > p {color:gray;}
```

CSS 2.1

## 6. 선택자 그룹화

---

- 다수의 요소에 똑같은 선언을 지정할 경우
  - 반복해서 작성하는 대신 선택자를 그룹화하여 표현
  - 콤마(,)를 사용하여 구분

```
div p, #tit, .box {background-color:gray;}
```

CSS 2.1



# CSS 개념 및 단위

---

1. 상속
2. 겹침과 개별성
3. 주석
4. 단위
5. 색상

# 1. 상속(inherit)

---

- 상위요소에 적용한 스타일이 하위 요소까지
  - 모든 스타일이 똑같이 상속되는 것이 아니라 일부는 상속의 대상에서 제외
  - 콤마(,)를 사용하여 구분

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="utf-8" />
  <title>개별성 (specificity)</title>
  <style type="text/css">
    body { font-size:62.5%; }
    div { font-size:2em; border : 1px solid red ; padding: 1em; }
    p { font-size:1.5em; }
  </style>
</head>
<body>
  <div>
    <p>상속 (inherit)이라고 합니다. CSS의 속성은 기본적으로 하위 요소로 상속되는
속성과 상속되지 않는 속성이 있습니다.</p>
    <p>border, margin, padding, float, position 등의 속성은 하위 요소에 상속되지
않습니다.</p>
  </div>
  <p>em 단위는 부모 크기를 기준으로 상대적으로 변하는 단위를 말하며, 부모 요소에 영향을
받기 때문에 상속의 개념이 적용됩니다. 글자 크기를 em 단위로 지정한 경우, 상속으로 인해 예상한
것보다 글자 크기가 작게 나오거나 크게 나올 수 있으므로 주의해야 합니다.</p>
</body>
</html>
```

## 2. 겹침(cascading)과 개별성(specificity)

---

- 하나의 콘텐츠에 여러 가지의 스타일이 중복 적용되어 충돌이 발생했을 때

선택자	사용 예	특정도
Type 선택자	p	1
Class 선택자	.note	10
Id 선택자	#gnb	100
Inline 스타일	style="color:red"	1000
가상 요소	:first-line	1
가상 클래스	:hover	10

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <title>개별성 (specificity)</title>
  <meta charset="utf-8" />
  <style type="text/css">
    h2      {color:red;}
    .cascade {color:green;}
    #specifi {color:blue;}
  </style>
</head>
<body>
  <h2>단계적으로 적용되는 스타일</h2>
  <h2 class="cascade">단계적으로 적용되는 스타일</h2>
  <h2 id="specifi">단계적으로 적용되는 스타일</h2>
</body>
</html>
```

### 3. 주석

---

- `/* ~~~ */`
  - 불필요한 주석 사용 지양
  - 버전 표시, 최종 갱신일 등

```
~~~  
<style type="text/css">  
/*개별성 규칙 (specificity)*/  
h2      { color:red; }  
.cascade { color:green; }  
#specifi { color:blue; }  
</style>  
~~~
```

CSS 2.1

## 4. 단위

---

- 절대단위
  - pt, cm, mm, pc, in
- 상대 단위
  - px, ex, em, %

```
~~~  
<style type="text/css">  
body { font-size:12px; }  
  
/*px을 사용한 경우*/  
.px { font-size:14px; }  
  
/*em을 사용한 경우*/  
.em { font-size: 1.5em; }  
</style>
```

~~~

## 5. 색상

---

- 색상명
  - black, silver, gray, white, maroon, red, purple, teal, fuchsia, green, lime, olive, yellow, navy, blue, aqua
- 16진수 코드
  - 빨간색, 녹색, 파란색의 상대적 양. 3개의 16진수로 구성
  - #FF0000 > #F00

```
~~~  
<style type="text/css">  
  .keyword { color:red; }  
  .hexa { color:#0000ff; }  
  .decimal { color:rgb(255,0,0); }  
  .percent { color:rgb(0%,0%,100%); }  
</style>  
~~~
```



# Thank you.