

CSS

김용만

1. CSS란?

CSS(Cascading Style Sheets)는 속성을 이용해 웹 디자인 요소를 제어하는 기능. CSS를 한 곳에 설정하면 특정 HTML 태그, 단일 웹 페이지, 전체 웹 사이트에 걸쳐 문서 외양에 스타일이 적용된다.

CSS는 기존 태그가 콘텐츠를 보여주는 방식을 재정의해(마크업 언어인) HTML의 기능을 개선하는 별도의 스타일시트 언어다.

1) CSS의 동작 방식

방문자가 주소창에 주소를 입력하거나 링크를 클릭해 웹 페이지를 열면(웹 페이지가 저장된 컴퓨터인)서버는 HTML 파일과 연결되거나 HTML 파일에 포함된 다른 파일과 함께 HTML 파일을 방문자의 컴퓨터로 전송한다. 브라우저는 HTML에 링크되어 있는 또는 삽입된 CSS코드를 해석하고 이를 HTML에 적용한다.

2) CSS의 장점

- 기능 확장성 : HTML 기능의 확장 가능
- 양식 모듈화 : 흐름이 같은 문서 양식으로 전체 구성 가능
- 간편성 : 문서 형식을 손쉽게 다양하게 구성 가능
- 일관성 : 사용 환경에 영향 받지 않음

3) CSS의 단점

- 브라우저의 호환성 문제

4) CSS의 발전 과정

- CSS 레벨 1 (CSS1)

W3C는 1996년에 첫 번째 CSS 공식 버전을 발표했다. 이 버전에는 텍스트 서식, 폰트 설정, 마진 설정 같은 CSS와 관련한 핵심 기능이 포함돼 있다.

- CSS 레벨 2 (CSS2)

CSS2 명세서는 1998년에 나왔고 현대 브라우저에서 가장 많이 채택되어 사용되고 있다.

국제적인 접근성 및 미디어 관련 **CSS**를 지정하는데 초점을 두었다.

2006년 W3C는 CSS 레벨 2.1이라는 수정 버전을 발표했다. 이 버전에서는 일부 에러가 수정되고 몇 가지 이슈를 명확히 정리했으며 다른 브라우저에서 이미 구현한 기능에 대한 명세를 추가했다.

- CSS 레벨 3 (CSS3)

CSS1,2와 달리 모든 명세를 포함한 단일 형태의 CSS3는 존재하지 않는다. CSS 작업 그룹에서는 한 번에 전체 명세를 발표하는 대신 일련의 모듈로 명세를 분할했다.

2. CSS 규칙의 구성요소

1) 스타일 형식

<pre>h { color : blue; }</pre>
<pre> </pre>
<pre> V</pre>
<pre> V 값</pre>
<pre> 속성</pre>
<pre>V</pre>
<pre>선택자</pre>

선택자(selector)	규칙을 식별하는 알파벳과 숫자로 된 문자다. 선택자는 HTML 태그 선택자, 클래스 선택자, ID 선택자, 공통 선택자 또는 맥락 기반 스타일 생성을 위해 이들 기본 선택자를 조합한 선택자를 사용할 수 있다.
속성(attribute)	정의하는 내용을 나타낸다. 각 속성은 페이지 콘텐츠의 동작 방식 및 외양을 책임진다.
값(value)	속성에 대입해 속성의 성질을 나타낸다.

2) CSS의 삽입방법 3가지

- 인라인 스타일 - HTML 태그 내의 스타일 추가

CSS를 사용한다는 말은 개별 태그의 외양을 일일이 설정할 필요가 없다는 뜻이기는 하지만 개별 태그 내에서 스타일을 설정하는 것도 얼마든지 가능하다. 이러한 스타일 설정을 인라인 스타일이라고 한다.

<pre><h1 style="color:red;">인라인 스타일</h1></pre>

- 문서내 지정형 (Embedding Style)

단일 엘리먼트(인라인)나 전체 웹 사이트(외부 스타일)가 아니라 단일 웹 페이지에 적용할 스타일을 추가하려면 **<style>** 태그를 사용해 웹 페이지에 스타일 규칙을 직접 기입한다. 이 때는 **<style>** 태그안에 모든 스타일 규칙이 들어간다.

문서의 헤드에서 **style** 태그를 입력한다. **type** 속성을 “text/css”로 정의해 임의 스타일이 아니라 **CSS**를 따르도록 스타일을 정의한다. 아울러 **media** 타입의 값으로 **all**을 지정한다. 이렇게 하면 페이지 결과를 렌더링하는 기기의 종류에 상관없이 스타일 시트가 페이지에 적용된다.

```
<style type="text/css" media="all">
  h1 {color:red;}
</style>
```

- 외부 경로 삽입형(Link Style)

CSS에는 한 번 만들어둔 스타일 시트를 단일 웹 페이지나 전체 웹 사이트에서 재활용할 수 있다는 큰 장점이다.

웹사이트에서 전체적으로 외부 스타일 시트를 사용하려면 먼저 모든 **CSS** 코드가 들어 있는 외부 파일을 생성해야 한다.

rel="stylesheet" : HTML 문서에 대한 링크 관계를 스타일시트(stylesheet)로 지정한다. **link** 태그는 다른 타입을 추가할 때도 사용할 수 있으므로 이 부분이 중요하다. 이부분을 빼먹으면 많은 브라우저에서 코드를 제대로 불러오지 못한다.

```
<link rel="stylesheet" href="style.css" type="text/css" media="all" />
```

3) @import의 사용

<style> 태그 내에서 다른 **CSS** 코드를 포함하기 전에 **@import url()**를 입력하고 괄호 사이에 임포트할 **CSS**문서의 **URL**을 입력한다.

```
<style type="text/css" media="all">
  @import url('style.css'); 또는 @import 'style.css';
  h1{
    color:gray;
  }
  h2{
    color:darkred;
  }
</style>
```

4) CSS에 주석 추가

```
h1{
  /* color:black;*/
  margin: 0 20px 20px 0;
}
```

5) 스타일의 우선순위

- 스타일은 상속된다.

웹문서에서 사용하는 여러 태그들은 서로 포함 관계를 가지고 있어 포함하는 태그는 부모 요소라고 하고, 포함된 태그는 자식 요소라고 한다. 스타일시트에서는 부모 요소에 있는 속성들이 자식 요소로 전달되는데, 이것을 '상속'이라고 한다.

엘리먼트 가계도

태그가 다른 태그에 둘러싸이면(한 태그가 다른 태그에 들어 있으면) 태그가 중첩된 것이다.

```
<article class="copy"> 부모
  <p> 자식
    <em>겨울 여행</em> 이전 형제
    <strong>행복한 나라</strong> 인접 형제
  </p>
</article>
```

- 스타일은 단계적으로 적용됨

CSS는 캐스케이딩 스타일시트(Cascading Style Sheets)의 줄임말이고, 단계적으로 스타일이 적용된다는 뜻. 여러 문서에 걸쳐 다양한 스타일을 적용하다보면 같은 요소에 여러 가지 스타일이 겹쳐지기도 하는데, 하나의 요소에 여러 스타일 규칙이 정의되어 있다면 가장 나중에 정의된 스타일을 적용.

- **!important** 적용

!important를 추가하면 해당 스타일 규칙이 가장 높은 우선순위를 갖게 됨

```
<style>
.accent{color:red !important;} ← 우선 적용
.accent{color:blue;}
</style>
```

3. 선택자

1) 전체 선택자

모든 요소에 스타일 적용

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

2) 태그 선택자

특정 태그에 스타일 적용하기

```
p{ font-size : 1em; }
```

3) 클래스 선택자

클래스 선택자를 사용하면 모든 HTML 태그에 적용할 수 있는 독립적인 스타일을 설정할 수 있다.

클래스명 지정하기

- 글자와 숫자, 하이픈, 밑줄 사용 가능
- 첫 글자로 숫자, 밑줄, 하이픈을 사용할 수 없다.
- 공백을 사용하지 않는다.

```
.authorName{ font-size:1em;}
```

4) ID 선택자

클래스 선택자와 마찬가지로 ID 선택자도 특정 HTML 태그와 무관한 고유 스타일을 만드는 데 사용할 수 있다.

클래스 선택자는 문서 안에서 여러 번 반복해서 적용할 수 있는 반면, id 선택자는 요소의 크기나 위치 등 레이아웃을 지정할 때 주로 사용하기 때문에 클래스 선택자와 달리 **문서 안에서 한 번만 적용**

```
#content{ position:relative;}
```

5) 태그 선택자와 클래스 선택자의 연계

```
p.authorName{ font-size:1em;}
```

6) 태그 선택자와 아이디 선택자의 연계

```
div#content{ position:relative;}
```

7) 그룹 지정 - 동일 스타일을 사용하는 엘리먼트 정의

둘 이상의 선택자가 동일 선언을 사용하게 하려면 콤마로 구분해 선택자를 그룹으로 지정할 수 있다.

```
h1,h2,.copy,#head{font-size:1em;}
```

8) 하위 선택자(descendant selector)

모든 하위 요소에 스타일 적용하기. 하위 선택자는 부모 요소에 포함된 모든 하위 요소에 스타일 적용.

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

9) 자식 선택자(child selector)

자식 요소에만 스타일 적용하기. 바로 밑의 하위 요소, 즉 자식 요소에만 스타일 적용

```
section > p {color:blue;}
```

10) 인접 형제 선택자(adjacent selector)

첫 번째 동생 요소에 스타일 적용하기

```
h1 + p {text-decoration:underline;}
```

11) 형제 선택자(sibling selector)

형제 요소에 스타일 적용하기

```
h1 ~ p {text-decoration:underline;}
```

12) 속성 선택자

- **[속성]** 형식

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

- **[속성 = 값] 형식**

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

- **[속성 ~= 값] 형식**

[속성 ~= 값] 선택자를 사용하면, 비교할 값이 여러 개일 경우 그중 하나만 일치해도 스타일을 적용.

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

- **[속성 ^= 값] 형식**

지정한 문자로 시작하는 속성 값에 대해서만 스타일을 적용

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

- **[속성 \$= 값] 형식**

지정한 문자로 끝나는 속성에 대해서만 스타일을 적용

```
a [href$=".zip"] {background:url (zip_icon.gif);}
```

- **[속성 |= 값] 형식**

속성이 지정한 값으로 시작하면 스타일을 적용.

<a> 태그로 링크를 건 부분 중 **class** 이름이 “red”로 시작하는 모든 요소에 **red.jpg** 그림을 배경 이미지로 넣은 것이다. 이 선택자는 **class="red"** 처럼 정확히 한 단어로 일치하거나, **class="red-bg"** 처럼 하이픈(-)으로 연결해 한 단어로 일치할 경우에 사용할 수 있다.

```
a [class |= "red"] {background:url (red.jpg);}
```

- **[속성 *= 값] 형식**

사용자가 지정한 속성 값의 앞이나 뒤, 또는 중간 등 어느 위치에 있든지 해당 값이 포함되어 있으면 스타일이 적용

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

13) 가상 클래스

가상 클래스는 슈도 클래스(**pseudo class**)라고도 하는데, 웹 문서의 소스에는 실제로 존재하지 않지만 필요에 의해 가상의 선택자를 지정하여 사용하는 것을 말함.

<a> 태그는 (평상시에는) 링크 상태, (방문자가 이미 링크 페이지를 방문한) 방문 상태,

(방문자가 링크에 마우스를 올려 놓은) 호버 상태, (방문자가 링크를 클릭한) 활성 상태 등을 갖는다. 이러한 네가지 상태에는 모두 개별 스타일을 적용할 수 있다.

```
a:link{color:red;}
```

링크 및 동적 의사 클래스

형식	이름	스타일 적용 엘리먼트
:link	링크	href의 값이 히스토리에 없는 경우
:visited	방문 링크	href의 값이 히스토리에 있는 경우
:active	활성	엘리먼트를 클릭한 경우
:hover	호버	마우스 포인터가 엘리먼트 위에 있는 경우
:focus	초점	해당 요소에 초점이 맞춰졌을 때 스타일을 지정

구조적/기타 의사 클래스

형식	이름	스타일 적용 엘리먼트
:root	루트	문서 안의 루트 요소에 스타일을 적용
:empty	공백	자식이 없는 엘리먼트
:only-child	유일 자식	형제가 없는 엘리먼트
:only-of-type	유일 타입	형제 가운데서 고유 선택자를 갖는 엘리먼트
:first-child	첫 번째 자식	다른 엘리먼트의 첫 번째 자식인 엘리먼트
:nth-of-type(n)	타입의 N번째	해당 선택자를 갖는 n번째 엘리먼트
:nth-last-of-type(n)	타입의 끝에서 N번째	해당 선택자를 갖는 마지막 엘리먼트로부터 시작해 같은 선택자를 갖는 n번째 엘리먼트
:last-child	마지막 자식	부모 엘리먼트의 마지막 자식 엘리먼트
:first-of-type	타입의 첫 번째	부모 엘리먼트에서 특정 선택자 타입을 갖는 첫 번째 자식 엘리먼트
:last-of-type	타입의 마지막	부모 엘리먼트에서 특정 선택자 타입을 갖는 마지막 자식 엘리먼트
:target	앵커의 목적지	앵커의 목적지가 되는 부분의 스타일을 지정

:lang()	언어	지정한 언어 코드를 갖는 엘리먼트
:not(s)	부정	특정 선택자를 사용하지 않는 엘리먼트

:enabled, :disabled, :checked 선택자

:enabled 와 :disabled는 요소 중에서 사용할 수 있는 상태(:enabled)와 사용할 수 없는 상태(:disabled)의 스타일을 지정할 수 있다.

:checked 클래스 선택자는 라디오 버튼이나 체크박스가 체크되었을 때 적용할 스타일을 지정.

::before, ::after 선택자

특정한 요소의 내용 앞(::before)이나 뒤(::after)에 지정한 내용을 만들 수 있음.

::selection 선택자

사용자가 마우스로 선택했을 때 커서가 드래그되어 선택되는 영역의 스타일을 지정.

4. 텍스트 관련 스타일

속성	설명
text-decoration	텍스트에 밑줄이나 취소선, 윗줄 등을 추가
text-align	텍스트 정렬
text-indent	텍스트의 들여쓰기와 내어쓰기
letter-spacing word-spacing	글자 간격이나 단어 간격 조절
text-transform	텍스트의 특정 부분에 대한 알파벳을 대문자 혹은 소문자로 전환
white-space	문서의 태그내에 생기는 공백의 처리 여부를 결정

1) text-decoration 속성

- 텍스트에 밑줄이나 취소선, 윗줄 등을 추가
- text-decoration 속성이 블록 레벨 요소에 적용된다면 하위 체계에도 모두 적용

속성	설명
none	형태가 없음
underline	밑줄 생성

overline	윗줄 생성
line-through	취소선 생성
color	텍스트의 색상을 적용

2) text-align 속성

- 텍스트를 정렬하는 프로퍼티

속성	설명
left	왼쪽 정렬
right	오른쪽 정렬
center	가운데 정렬
justify	좌우 정렬
<문자열>	테이블 행에만 적용되는 값으로, 지정된 문자열을 중심으로 정렬

3) text-indent 속성

- 텍스트의 첫 번째 행을 들여쓰기나 내어쓰기로 설정
- 내어쓰기 경우 음수값 지정

속성	설명
<길이>	고정된 길이로 지정
<백분율>	포함된 블록의 너비를 기준으로 지정

4) letter-spacing 속성 / word-spacing 속성

- 글자 간격이나 단어 간격을 조절

프로퍼티 값	설명
normal	웹 브라우저에 의존하도록 지정
고정 길이	기본 문자 간격에 추가적인 공간을 확보하는 경우에 사용

5) line-height 속성

- 줄간격 조절하기

p{font-size:10pt; line-height:1.2;}/*숫자를 사용*/
p{font-size:10pt; line-height:120%;}/*백분을 사용*/
p{font-size:10pt; line-height:12pt;}/*크기 값 사용*/

6) text-transform 속성

- 텍스트의 특정 부분의 알파벳을 대문자 혹은 소문자로 전환

속성	설명
capitalize	각 단어의 첫 글자를 대문자로 전환
uppercase	텍스트의 모든 글자를 대문자로 전환
lowercase	텍스트의 모든 글자를 소문자로 전환
none	대소문자를 전환하지 않음

7) white-space 속성

- 공백 처리 방법 결정

속성	설명
normal	웹 브라우저가 연속된 공백을 제거
pre	웹 브라우저가 연속된 공백을 제거하지 못하도록 함
nowrap	태그에 의한 줄바꿈(br 등)을 제외한 줄바꿈은 허용하지 않음

8) text-shadow 속성

- 텍스트에 그림자 효과 추가하기

text-shadow:h-shadow 값 v-shadow 값 blur 값 color 값;

속성 값	설명
h-shadow	그림자의 가로 옅셋 거리. 양수 값은 글자 오른쪽에 음수 값은 글자 왼쪽에 그림자를 만듦
v-shadow	그림자의 세로 옅셋 거리. 양수 값은 글자 아래쪽에 음수 값은 글자 위쪽에 그림자를 만듦
blur	그림자가 번지는 정도를 지정. 생략할 수도 있는데, 이 경우 그림자가 번지지

	않고 선명하게 표시됨.
color	그림자 색상을 지정. 16진수. 색상 이름 표기로 값을 입력.

9) text-overflow 속성

- 넘치는 텍스트 표기하기

text-overflow:속성 값;

속성값	설명
clip	넘치는 텍스트를 잘라버림
ellipsis	텍스트가 잘렸다는 의미로 말줄임표("...")를 표시

5. 글꼴 관련 스타일

- **font-family** 프로퍼티
- 텍스트를 표현할 때 어떤 글꼴을 사용할지를 지정
- 여러 가지 비슷한 스타일의 글꼴이 모여서 이루어진 일종의 글꼴 집단

body {font-family:Symbol, Serif;}

- **font-style** 프로퍼티
- 글꼴이 어떠한 형태를 가지고 있는가를 설명

프로퍼티 값	설명
normal	웹 브라우저에 설정되어 있는 기본 글꼴이 지정
oblique	normal 글꼴을 약간 기울인 형태로 지정
italic	보통 기울임체의 형태로 지정

- **font-variant** 프로퍼티
- 텍스트가 **small cap** 효과를 적용할지 결정
- 어떤 글꼴은 표준 모양만을 지원, **small cap** 형식만 지원도 가능

프로퍼티 값	설명
normal	웹 브라우저에 설정되어 있는 기본 글꼴이 지정

small-caps	소문자의 경우 대문자로 전환되면서 원래의 대문자보다 약간 작은 형태로 표시, 대문자와 한글의 경우 원래의 문자보다 약간 큰 형태로 표시
------------	-----------------------------------------------------------------------------

- **font-weight** 프로퍼티
- 같은 글꼴 가족에 있는 다른 글꼴에 비해 두껍거나 얇게 표시하는 경우에 사용
- “100-900”까지의 지정된 숫자 사용도 가능

프로퍼티 값	설명
normal	글꼴의 두께는 400
bold	글꼴의 두께는 700
bolder	상속된 두께보다 두꺼운 두께를 지정(600 이상은 똑같이 적용)
lighter	상속된 두께보다 얇은 두께를 지정(500 이하는 똑같이 적용)

- **font-size** 프로퍼티
- 글꼴 크기 조정

프로퍼티 값	설명
<절대적인 키워드>	Xx-small, x-small, small, medium, large, x-large, xx-large로 설정
<상대적인 키워드>	관련된 글꼴 크기나 상위 태그의 글꼴 크기에 연관되어 계산
<길이>	절대적인 글꼴의 크기를 지정
<백분율>	상위 태그의 글꼴 크기에 연관된 절대적인 글꼴 크기

- 크기 값 단위

단위	설명
em	해당 글꼴의 대문자 M의 너비를 기준으로 함
ex	X-height 해당 글꼴의 소문자 x의 높이를 기준으로 함
px	픽셀. 모니터에 따라 상대적인 크기가 됨
pt	포인트. 보통 문서에서 많이 사용하는 단위

- **font** 속성을 이용한 글꼴 관련 스타일 일괄 지정

아래와 같은 순서로 스타일을 한 번에 명시

font: font-style font-variant font-weight font-size/line-height font-family

@font-face 속성 - 웹 폰트 사용하기

인터넷 익스플로러 8 이하 버전에서는 Embedded Open Type 형태의 글꼴(*.eot 파일)을 지원하고, 인터넷 익스플로러 9 이상 및 파이어폭스와 크롬에서는 Web Open Font Format 형태의 글꼴(*.woff파일)을 지원.

```
@font-face {
  font-family:글꼴 이름;
  src:url(글꼴 파일 경로) format(파일 유형);
}
```

구글 웹 폰트 사용하기

<https://fonts.google.com/> 에서 사용 웹 폰트 확인

```
<style>
@import url(http://fonts.googleapis.com/earlyaccess/nanumbrushscript.css);
.new-font{
  font-family:'Nanum Gothic', 돋움;
}
</style>
```

6. 목록 스타일

1) list-style-type 속성

- 목록의 불릿과 번호 스타일 지정하기

list-style-type:속성 값;

순서 없는 목록에서 사용하는 불릿 유형

속성	설명
----	----

disc(●)	속이 찬 원
circle(○)	속이 빈 원
square(■)	속이 찬 사각형

순서 있는 목록에서 사용하는 번호 유형

decimal	1,2,3
decimal-leading-zero	01,02,03
upper-roman	I, II, III
lower-roman	i, ii, iii
upper-alpha 또는 upper-latin	A, B, C
lower-alpha 또는 lower-latin	a, b, c
lower-greek	α, β, γ
hebrew	히브리 숫자
armenian	아르메니아 숫자
georgian	조지 왕조 시대의 숫자 (예:an,ban,gan,...)
cjk-ideographic	단순한 표의 문자(ideographic)
hiragana	히라가나(예:a,i,u,e,o,ka,ki,...)
hiragana-iroha	히라가나 번호 붙이기 (예:i,ro,han,ni,ho,he,...)
katakana	가타가나(예:A,I,U,E,O,KA,KI,...)
katagana-iroha	가타가나 번호 붙이기(예:I,RO,HAN,NI,HO,HE,...)

2) list-style-position 속성

- 목록에 들어쓰는 효과 내기

list-style-position:속성값;

속성	설명
----	----

inside	불릿의 왼쪽 끝을 이어지는 텍스트의 왼쪽 끝에 맞춘다.
outside	텍스트의 첫 줄을 이어지는 텍스트와 정렬해 불릿 내어쓰기를 적용한다.

3) list-style-image 속성

- 불릿 대신 이미지 넣기

list-style-image:url(이미지 파일 경로)

4) list-style 속성

- 리스트 속성 한꺼번에 표시하기
- list-style-type과 list-style-position, list-style-image 속성을 한꺼번에 표시

7. 색상과 배경을 위한 스타일

1) color 속성

- 색상 지정하기

RGB 16진수 색상값 : 숫자 기호 - #FFFFFF, 숫자 기호 - #FFF
RGB 10진수 색상값 : rgb(255[적],255[녹],255[청])
RGB 10진수 색상값, 투명도 : rgba(255,255,255,.5[알파(투명도)])
RGB 백분율 색상값 : rgb(100%,100%,100%)
HSL 색상값 : hsl(0[색조],0[채도],100%[명도])
HSL 색상값, 투명도 : hsla(0,0,100%,.5[알파(투명도)])

2) 배경 관련 속성

프로퍼티	기능
background-color	문서 전체에 대한 배경 색상을 지정
background-image	배경으로 이미지를 사용할 경우에 사용
background-repeat	배경 이미지의 반복 형태를 결정
background-attachment	배경 이미지가 고정될 것인지, 문서와 함께 스크롤될 것인지를 결정

background-position	배경 이미지의 초기 위치를 지정
background	프로퍼티들을 한꺼번에 지정하기 위한 프로퍼티

3) background-color 속성

프로퍼티 값	설명
<색상값>	색상값 키워드 or RGB 값
transparent	투명한 색

4) background-image 속성

프로퍼티 값	설명
url(<URL>)	상대 경로 및 절대 경로를 사용
none	이미지 파일 없음

5) background-repeat 속성

프로퍼티 값	설명
repeat	이미지를 수평과 수직으로 반복 : 바둑판식
repeat-x	처음 위치에서 이미지가 수평으로 반복
repeat-y	처음 위치에서 이미지가 수직으로 반복
no-repeat	이미지가 반복되지 않고, 하나의 이미지만 표시

6) background-position 속성

프로퍼티 값	설명
<백분율>	% 단위로 위치 지정
<길이>	cm 단위로 위치 지정

7) background-attachment 속성

프로퍼티 값	설명
scroll	내용이 스크롤될 때 문서와 함께 스크롤되도록 지정
fixed	내용이 스크롤될 때 이미지는 계속 고정된 위치에 있도록 지정

8) background 속성

- Background-color, background-image, background-repeat, background-attachment, background-position) 프로퍼티들을 동시 지정

8. 그라데이션 효과

1) 선형 그라데이션

linear-gradient(각도 또는 위치, color-stop, [color-stop,...]);
linear-gradient(45deg, #ff0000, #ffffff) /*45도, 빨간색, 흰색*/

- 선형 그라데이션 위치 값

to top	아래에서 시작하여 위로 그라데이션이 만들어짐
to left	오른쪽에서 시작하여 왼쪽으로 그라데이션이 만들어짐
to right	왼쪽에서 시작하여 오른쪽으로 그라데이션이 만들어짐
to bottom	위에서 시작하여 아래로 그라데이션이 만들어짐

2) 원형 그라데이션

radial-gradient(위치,모양(및 크기), color-stop, [color-stop...])
radial-gradient(circle,white,blue); /*중앙에서 시작하여 흰색에서 파란색으로 번지는 원형 그라데이션*/

- 원과 타원의 크기를 조절하는 키워드 값

closest-side	원의 경우 그라데이션 가장자리가 그라데이션 중심에서 가장 가까운 요소의 모서리와 만나거나, 타원의 경우 그라데이션 중심에서 가장 가까운 요소의 수평축이나 수직축과 만나도록 함
closest-corner	그라데이션 가장자리가 그라데이션 중심에서 가장 가까운 요소의 코너에

	달도록 함.
farthest-side	원의 경우 그라데이션 가장자리가 그라데이션 중심에서 가장 멀리 떨어져 있는 모서리와 만나거나, 타원의 경우 그라데이션 가장자리가 그라데이션 중심에서 가장 멀리 떨어져 있는 모서리와 만나도록 함
farthest-corner	그라데이션의 가장자리가 그라데이션의 중심에서 가장 멀리 떨어져 있는 코너에 달도록 함. 기본 값.

9. 박스 모델



여백(margin)	부모 항목과 경계 사이의 간격
경계(border)	블록의 외곽 특성
채워넣기(padding)	경계와 블록 내용 사이의 간격
내용(content)	HTML 문서의 실제 내용

1) 박스 속성

속성	속성값	의미
width	숫자 또는 백분율	넓이
height	숫자 또는 백분율	높이
float	left, right, none	위치
margin, margin-top,	숫자 또는 백분율	여백

margin-bottom, margin-left, margin-right		
padding, padding-top, padding-bottom, padding-left, padding-right	숫자 또는 백분율	채워넣기
border, border-top, border-bottom, border-left, border-right	숫자 또는 백분율	경계

2) border-radius 속성

- 박스 모서리 둥글게 만들기

border-radius:속성값;

3) border-*radius 속성

- 박스 모서리 다르게 지정하기

border-*radius:속성값;

```
p{
  border-top-left-radius:10px; /*박스 모델의 왼쪽 윗부분을 10px만큼 둥글게*/
  border-top-right-radius:10px; /*박스 모델의 오른쪽 윗부분을 10px만큼 둥글게*/
}
```

3) box-shadow 속성

- 선택한 요소에 그림자 효과내기

box-shadow:수평거리 값 | 수직거리 값 | blur 값 | 확장거리 값 | 색상 값 | inset;

가로 오프셋 거리	그림자가 가로로 얼마나 떨어져 있는지 지정. 양수값은 요소의 오른쪽 쪽, 음수값은 요소의 왼쪽에 그림자 표시
세로 오프셋 거리	그림자가 세로로 얼마나 떨어져 있는지 지정. 양수값은 요소의 아래, 음수값은 요소의 위쪽에 그림자 표시
blur radius	그림자의 번지는 정도를 지정. 0이 최소값이며 이 값을 생략하면 번지는 효과를 사용하지 않음

그림자 색상	그림자 색상을 지정. 16진수로 된 색상값을 사용. 색상 이름이나 RGB 값 모두 사용.
--------	---------------------------------------------------

4) display 속성

- display 속성은 해당 요소가 화면에 어떻게 보여질 것인지 지정하는 것.

속성	설명
block	블록 레벨 박스로 만듦
inline	인라인 레벨 박스로 만듦
inline-block	블록 레벨 박스이지만 인라인 레벨 박스처럼 배치
none	박스를 만들지 않으며 시각적으로 표시하지 않음
inherit	상위 요소의 display 속성을 상속받음
table	블록 레벨의 표로 만듦
inline-table	인라인 레벨의 표로 만듦(HTML에서 table에 해당)
table-row	표의 행으로 만듦(HTML에서 tr에 해당)
table-row-group	표의 본체 행 그룹으로 만듦(HTML에서 tbody에 해당)
table-header-group	헤더 행 그룹으로 만듦(HTML에서 thead에 해당)
table-footer-group	푸터 행 그룹으로 만듦(HTML에서 tfoot에 해당)
table-column	표의 열로 만듦(HTML에서 col에 해당)
table-column-group	표의 열 그룹으로 만듦(HTML에서 colgroup에 해당)
table-cell	표에서 하나의 셀로 만듦(HTML에서 td나 th에 해당)
table-caption	표의 캡션을 만듦(HTML에서 caption에 해당)
list-item	목록의 항목을 표시할 수 있도록 기본적인 블록 박스와 표시자 박스를 만듦(HTML에서 li에 해당)

10. 위치 유형 설정

1) position 속성

속성	내용
----	----

static	<p>정적 위치 지정</p> <p>기본적으로 엘리먼트는 절대, 상대, 고정 위치를 지정하지 않으면 문서에서 정적으로 위치를 잡는다. 정적 엘리먼트는 하나씩 차례로 문서에 배치된다.</p>
relative	<p>상대 위치 지정</p> <p>상대 엘리먼트는 가만히 둘 경우 엘리먼트가 놓일 위치를 기준으로 엘리먼트의 위치를 정한다. 상대 위치를 지정한 엘리먼트는 문서의 정상 흐름 위치를 기반으로 위치를 조절한다. 하지만 상대 엘리먼트가 차지하는 공간은 빈 공간으로 보인다.</p>
absolute	<p>절대 위치 지정</p> <p>절대 위치 지정은 문서의 정상 흐름으로부터 엘리먼트를 가져오며 공백을 남기지 않는다. 이 방식으로 위치를 지정한 엘리먼트는 x와 y 좌표를 사용해 가장 최근에 위치를 지정한 부모 엘리먼트의 가장자리 또는 위치를 지정한 부모가 없을 경우 바디를 기준으로 정확한 위치를 잡는다.</p>
fixed	<p>고정 위치 지정</p> <p>엘리먼트의 위치를 (부모가 아니라)창에 고정하는 방식은 뷰포트 가장자리를 기준으로 항상 고정된 위치에 엘리먼트가 놓이는 점을 제외하고 절대 위치 지정과 동일하다. 뷰포트에서 문서가 스크롤되면 고정 엘리먼트는 초기 위치에 그대로 머물며 나머지 문서를 따라 스크롤되지 않는다. 이 방식을 활용하면 나머지 콘텐츠와 함께 스크롤되지 않고 화면에 고정적으로 보이는 엘리먼트를 만들 수 있다.</p>

2) visibility 속성

- 요소를 보이게 하거나 보이지 않게 하기

속성	설명
visible	요소를 화면에 표시함. 기본 값
hidden	요소가 화면에서 감춰짐. 하지만 크기는 그래도 유지하기 때문에 배치에 영향을 줌
collapse	표의 행이나 열, 행 그룹, 열 그룹 등에서 지정하면 서로 겹치도록 조절함
inherit	부모 요소의 visibility 속성을 상속함

3) z-index 속성

- 요소 쌓는 순서 정하기

z-index:숫자 값;

div{z-index:999;} /*무조건 화면 맨 앞에 표시하려고 할 때*/

4) float 속성

- 요소를 떠 있게 하기

float:속성 값;

5) clear 속성

- float 속성 무효로 만들기

속성	설명
left	왼쪽으로 떠 있지 않도록 함
right	오른쪽으로 떠 있지 않도록 함
both	왼쪽이나 오른쪽 어디에도 떠 있지 않도록 함
none	왼쪽과 오른쪽 모두에 뜰 수 있음. 기본 값

6) 다단으로 편집하기

column-count	단의 개수
column-width	단의 너비
column-gap	단 사이의 간격
column-rule	단 사이의 구분선

11. 표 관련 CSS 스타일

- 테이블 관련 프로퍼티
- 테이블 모델에 기초. 테이블은 제목, 행, 셀로 구성

프로퍼티 값	기능
display	테이블의 각 요소들에 문서 언어의 요소들을 맞추어 사용할 때 사용

table-layout	테이블의 각종 요소에 대한 너비 조절
border-collapse	테이블의 테두리 지정
border-spacing	인접 셀 테두리들의 간격 지정

- **display** 프로퍼티
- 테이블의 각 요소들에 문서 언어의 요소들을 일치시켜야 함.

프로퍼티 값	설명
table	요소를 블록 레벨 테이블로 생성
inline-table	요소를 인라인 레벨 테이블로 생성
table-row-group	요소를 한 개 이상의 열로 생성
table-header-group	table-row-group 프로퍼티와 같지만 인쇄 미디어에서 머릿글(header) 줄들은 분리된 각 페이지에 인쇄
table-footer-group	Table-row-group 프로퍼티와 같지만 인쇄 미디어에서 바닥글/footer) 줄들은 분리된 각 페이지에 인쇄
table-column	요소를 테이블의 행으로 생성
table-column-group	요소를 하나 이상의 행으로 생성
table-cell	요소를 테이블의 셀로 생성
table-caption	요소를 테이블의 캡션으로 생성

- **caption-size** 프로퍼티

프로퍼티 값	설명	프로퍼티 값	설명
top	위쪽에 위치(초기값)	left	왼쪽에 위치
bottom	아래쪽에 위치	right	오른쪽에 위치

- **table-layout** 프로퍼티
- 테이블의 각종 태그에 대한 너비 조절
- 테이블, 셀, 행, 칸들의 배치에 사용

프로퍼티 값	설명	프로퍼티 값	설명
--------	----	--------	----

auto	자동 테이블 지정	fixed	고정 테이블 지정
------	-----------	-------	-----------

- **border-collapse** 프로퍼티
- 테이블의 테두리를 지정

프로퍼티 값	설명
collapse	통합 테두리를 선택(초기값)
separate	분리된 테두리들의 테두리를 선택

- **border-spacing** 프로퍼티
- 셀 테두리들 간격을 지정
- 고정 길이를 하나만 지정하면 수평과 수직 간격은 지정한 값으로 설정
- 두개 지정시 첫번째는 수평 간격, 두번째는 수직 간격 지정
- 프로퍼티의 값은 <길이>로 표현, 음수 표현 불가

유형 선택자	프로퍼티	클래스	내용
table	border	없음	outset 10pt
table	border-collapse	없음	separate
table	border-spacing	없음	20pt
td	border	없음	Inset 10pt

12. 변형을 위한 스타일

변형을 적용하면 엘리먼트를 회전하고, 크기를 조절하고, 구부리고, 움직여서 사각형 레이아웃을 벗어난 디자인을 만들 수 있다.

1) 2D transform 속성값

rotate(<angle>), rotateX(<angle>), rotateY(<angle>), scale(<number>), scaleX(<number>), scaleY(<number>), skew(<angle>), skewX(<angle>), skewY(<angle>), translate(<length>), translateX(<length>), translateY(<length>), matrix(<various>)

```
-webkit-transform: rotate(15deg);
-moz-transform: rotate(15deg);
transform: rotate(15deg);
```

2) transform-origin 속성

- 변형 원점 설정하기

transform-origin:x축 y축 z축 값;

3) perspective

- 원근감 표현하기

perspective:속성값;

4) perspective-origin 속성

- 원근감 조절하기

perspective-origin:x축 값 | y축 값;

5) transform-style 속성

- 3D 변형 적용하기

transform-style:속성값;

flat	하위 요소를 평면으로 처리
preserve-3d	하위 요소들에 3D 효과를 적용

6) backface-visibility 속성

- 요소의 뒷면 표시하기

backface-visibility:속성 값;

7) transition

- 시간에 따른 변화를 만들어 주기

transition-property	기간 기간은 트랜지션이 시작부터 끝까지 걸리는 시간을 보통 초단위로 설정한다. 이 값은 선택 사항이며, 지정하지 않으면 기본값인 0이 지정된다.
transition-duration	시간 함수 시간 함수는 트랜지션 진행에 따른 트랜지션 속도를 정의하는 함수다.

	이 값은 필수 값은 아니며, 설정하지 않으면 기본값인 linear 이 지정된다.
transition-delay	지연 시간 지연 시간은 트랜지션이 시작하기 전에 얼마동안 멈출지를 지정한다. 이 값은 선택 사항이며, 설정하지 않으면 기본값인 0 이 지정된다.

transition-timing-function 속성값

시간 함수	동작방식
cubic-bezier(<number>)	X와 Y값을 0과 1사이로 지정해 정의한 3차 베지어곡선의 모양이 시간 함수로 사용된다.
linear	등속도
ease	점진적인 감속
ease-in	가속
ease-out	감속
ease-in-out	가속 후 감속

8) 애니메이션

- **animation-duration** 속성
- 애니메이션 실행 시간

```
animation-duration:1s; /*애니메이션 진행 시간 1초*/
```

- **animation-name** 속성
- **CSS**로 애니메이션을 만들 때 **@keyframes** 속성을 이용해 여러 개의 애니메이션을 정의할 수 있는데, 특정 요소에 어떤 애니메이션을 적용할 것인지 정의하는 속성

```
animation-name:ani; /*ani라는 애니메이션 정의*/
```

- **animation-iteration-count** 속성
- 반복 횟수 지정하기

```
Animation-iteration-count:infinite; /*애니메이션 반복 횟수 무제한 반복*/
```

- **animation-direction** 속성

- 애니메이션 반대 방향으로 실행하기

```
animation-direction:alternate; /*애니메이션 끝난 후 방향으로 실행*/
```

- animation 속성
- 애니메이션 관련 속성 한꺼번에 표기하기

```
Animation:name 값 | duration 값 | timing-function 값 | delay 값 | iteration-count direction 값;
```

13. 반응형 웹 디자인을 위한 미디어 쿼리

HTML이 모니터 화면에서 보이는지, 아니면 인쇄물, TV 또는 휴대용 기기나 기타 기기인지에 따라 스타일을 다르게 설정할 수 있다. CSS#에서는 너비, 높이, 화면, 비율, 사용 가능한 색상수 같은 일반적인 UI 속성을 활용해 스타일을 설정하는 새로운 기능이 추가됐다.

미디어 쿼리값

값	적용 대상
screen	컴퓨터 디스플레이
tty	텔레타이프, 컴퓨터 터미널, 오래된 휴대용 기기
tv	텔레비전 디스플레이
projection	프로젝션
handheld	휴대용 전화기 및 PDA
print	종이
braille	점자 리더기
speech	음성 신디사이저
all	모든 기기

미디어 쿼리에는 많은 속성이 있지만 다섯가지 형태로 요약할 수 있다.

1. 화면 비율은 16:9 같은 화면 비율로 기기의 상대적인 크기를 나타낸다.
2. 너비와 높이는 디스플레이 영역의 크기를 나타낸다. 이 값들은 최댓값 또는 최솟값으로 표현할 수 있다.
3. 방향은 가로(너비가 높이보다 큼)와 세로(높이가 너비보다 큼) 레이아웃을 나타낸다. 방향을

알면 가로, 세로로 방향을 전환할 수 있는 기기에 디자인을 맞출 수 있다.

4. 색상, 색상 인덱스, 모노크롬은 색상값 또는 색상 하나당 비트를 나타낸다. 이들 속성값을 알면 흑백 휴대용 기기에 디자인을 맞출 수 있다.

5. 해상도는 결과물의 픽셀 밀도를 나타낸다. 이 속성은 특히 72dpi 이상의 디스플레이 기기를 사용할 때 유용하다.

@media 규칙의 사용법

미디어 쿼리를 사용하면 <link> 및 <style> 태그의 media 속성에 스타일을 지정할 수 있다.

그런데 @media 규칙을 사용하면 스타일 시트에 미디어 쿼리를 직접 임베드할 수 있다.

```
@media screen and (width=480px){  
  h1{color:red;}  
}
```

14. 브라우저의 CSS 확장 속성

W3C에서 정한 CSS 속성을 지원할 뿐 아니라 브라우저 개발 업체는 종종 브라우저 관련 속성을 지원하기도 한다.

혼란을 피하고 CSS 코드의 지속적인 호환성을 보장하기 위해 각 렌더링 엔진에서는 해당 브라우저에서만 사용하는 확장 CSS 속성에 고유 접두어를 사용했다.

확장 속성	렌더링 엔진	브라우저	예제
-moz-	모질라	파이어폭스, 카미노	-moz-border-radius
-ms-	트라이던트	인터넷 익스플로러	-ms-layout-grid
-o-	프레스토	오페라	-o-border-radius
-webkit-	웹킷	크롬, 사파리	-webkit-border-radius

CSS3의 새 기능

- 보더 : 측면, 보더 이미지, 라운드 코너에 사용할 수 있는 여러 개의 보더 색상

- 배경 : 단일 엘리먼트에 여러 배경을 추가할 수 있고 배경을 더 정확히 위치시킬 수 있으며 배경을 확장하거나 보더 안팎에서 배경을 자를 수 있고 배경 크기를 조절 할 수 있다.

-색상 : 색상 투명 설정, 배경의 그라디언트, HSL 색상값

- 텍스트 : 텍스트 새도우, 텍스트 오버플로우, 워드랩

- 변형 : 2D, 3D 공간상에서의 엘리먼트의 스케일, 왜곡, 이동, 회전
- 트랜지션 : 단순한 동작 스타일 트랜지션
- 박스 : 드롭 새도우 추가, 사용자가 크기를 조절할 수 있는 박스 배치, 가로/세로 방향으로 오버플로우별도 설정, 외곽선과 보더 사이의 간격 설정을 위한 외곽선 위치 조절값(offset)사용, 박스에 적용할 너비와 높이를 설정하기 위한 박스 모델 명세 적용.
- 콘텐츠 : 스타일을 사용해 엘리먼트에 콘텐츠 추가 가능
- 불투명도 : 엘리먼트 투명 처리 가능
- 미디어 : 뷰포트 크기, 색상, 화면 비율, 해상도, 기타 주요 디자인 고려사항을 기반으로 한 스타일 시트 적용
- 웹 폰트 : 디자인에 사용된 폰트를 링크할 수 있는 기능 업데이트 및 확장