

## CSS(Cascading Style Sheets)

### 1. css 기본문법

CSS(Cascading Style Sheets)를 정의하는 방법으로 내부 스타일 시트, 외부 스타일 시트, 인라인 스타일시트 등이 있다.

css에서의 주석은 /\* \*/로 표현한다.

#### 1) 내부 스타일시트

css를 HTML 문서 안에 <style> 태그로 기재하여 HTML 문서 안에 css가 함께 저장되도록 하는 형태이다.

#### 2) 외부 스타일시트

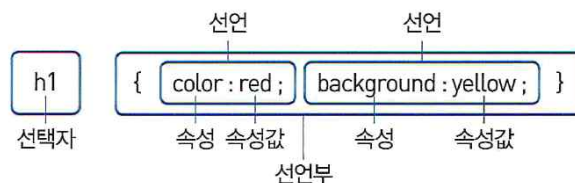
스타일 속성들을 따로 저장하여 HTML 문서에 파일명으로 연결할 수 있다. 이때 스타일은 \*.css 확장자를 가진 별도의 파일로 저장하고 HTML에서는 <link> 태그를 이용하여 선언한다. css 파일의 상단에는 @charset "utf-8"; 등 HTML 문서에서 지정한 것과 동일한 문자셋을 지정한다. 내부 스타일시트는 그 스타일이 기술된 페이지에만 적용되는 반면, 외부 스타일시트는 그 스타일을 선언한 모든 HTML페이지에 적용된다.

#### 3) 인라인 스타일

인라인 스타일은 태그에 직접 CSS를 정의해주는 기법으로, 해당 요소에 style=" " 형식으로 기술한다. 인라인 스타일은 내부스타일이나 외부스타일에서 기술된 속성보다 우선으로 적용되므로 우선순위가 절대적으로 높아야 할 경우에 사용할 수 있으나, 스타일을 공통 CSS로 수정할 수 없고 일일이 HTML을 열어서 편집해야 하므로 일반적으로 많이 사용하지는 않는다.

### 2. 선택자(selector)

선택자란 CSS로 UI의 어느 부분을 디자인할지, 즉 표현할 대상이 되는 부분을 말한다.



- (1) type 선택자
  - html 문서의 태그 이름을 선택자로 사용할 수 있다.
- (2) id 선택자
  - HTML 요소에 id로 이름을 붙일 때에는 유일한 이름을 부여해야 하며, 다른 요소에 같은 id명을 또 주어서는 안된다. CSS에서는 id 선택자 앞에 '#'을 붙여야 한다.
  - id명이나 class명은 숫자로 시작할 수 없으며, 주로 영문자로 시작한다.
- (3) class 선택자
  - HTML 문서의 여러 요소 중 같은 이름을 갖는 요소들만 모아 따로 속성을 부여할 수 있다.
  - HTML 요소에 같은 이름을 부여할 때에는 class로 이름을 붙인다. 즉 여러개의 요소에 같은 class 명을 부여할 수 있다. CSS에서는 class 선택자 앞에 '.'을 붙여야 한다.
- (4) 전체 선택자
  - 전체 선택자는 페이지의 모든 요소를 가리키는 선택자로서 '\*'로 표시한다.
- (5) 하위 선택자
  - 하위 선택자는 요소 내부에 있는 모든 해당 요소를 가리키며, 선택자 사이를 공백으로 분리한다.
- (6) 자식 선택자
  - 자식 선택자는 요소 내부에 있는 해당 요소를 가리키지만, 하위 요소의 하위 요소는 가리키지 않으며 선택자 사이를 '>'으로 분리한다.
- (7) 인접 선택자
  - 인접 선택자는 현재 요소의 바로 뒤에 나오는 요소만을 가리키는 선택자로, 선택자 사이를 '+'로 분리한다.
- (8) 형제 선택자
  - 형제 선택자는 현재 요소와 같은 계층에 있는 요소만을 선택할 수 있으며 '~'로 구분한다.
- (9) 그룹 선택자
  - 그룹 선택자는 여러 선택자들을 ','로 구분하여 함께 묶어 속성을 부여하는 것이다.
- (10) 속성 선택자
  - HTML 요소의 속성 유무 또는 속성값을 중괄호 [ ] 안에 넣어 선택자로 사용할 수 있다.

속성값	설명
h1[class]	class명을 가진 h1요소
img[alt]	alt속성을 가진 img요소
p[class="abc"]	class명이 유일하게 'abc'인 p요소
p[class~="abc"]	class명이 유일하게 'abc'이거나 여러 개의 class명 중 하나가 'abc'인 p요소
p[class="abc"]	class명이 'abc'이거나 'abc'로 시작하는 p요소
p[class^="abc"]	class명이 'abc'로 시작하는 p요소
p[class\$="abc"]	class명이 'abc'로 끝나는 p요소
p[class*="abc"]	class명이 'abc'가 포함되어 있는 p요소
a[href^="mailto"]	href 속성값이 'mailto'로 시작하는 a요소

(11) 가상클래스 선택자

- 가상클래스 선택자란 링크가 걸린 문자에 스타일을 부여하는 것이다.

속성 값	설명
a:link	링크가 걸린 문자에 속성을 부여하는 선택자
a:visited	링크를 클릭하여 해당 페이지에 갔다가 돌아온 경우의 속성을 부여하는 선택자
a:hover	링크가 걸린 문자에 마우스가 닿았을 경우의 속성을 부여하는 선택자
a:active	링크 걸린 글자가 활성화되었을 경우의 속성을 부여하는 선택자
a:focus	링크 걸린 글자에 포커스가 생길 경우의 속성을 부여하는 선택자

(12) 가상요소 선택자

- 가상요소 선택자는 요소에 id명이나 class명을 부여하지 않고도 위치를 찾아서 선택할 수 있는 선택자이다.

속성 값	설명
:first-letter	요소의 첫글자
:first-line	요소의 첫 줄
:first-child	같은 요소 중 첫 번째 요소
:last-child	같은 요소 중 마지막 요소
:nth-child(n)	같은 요소중 n번째 요소
:before	요소 안 맨 앞에 배치될 요소(마크업에는 없는 가상 요소)
:after	요소 안 맨 뒤에 배치될 요소(마크업에는 없는 가상 요소)

## All CSS Pseudo Classes

Selector	Example	Exampledescription
:linka:	link	방문하지 않은 모든 링크 선택
:visiteda	:visited	방문한 모든 링크 선택
:hoverta	:hover	마우스 오버 시 링크 선택
:activea	:active	활성링크선택
:checked	input:checked	체크된 모든 <input> 요소를 선택
:disabled	input:disabled	비활성화된 모든 <input> 요소를 선택
:emptytp	:empty	자식이 없는 모든 <p> 요소를 선택
:enabled	input:enabled	활성화된 모든 <input> 요소를 선택
:first-child	p:first-child	부모의 첫 번째 자식인 모든 <p> 요소를 선택
:first-of-type	p:first-of-type	부모의 첫 번째 <p> 요소인 모든 <p> 요소를 선택
:last-child	p:last-child	부모의 마지막 자식인 모든 <p> 요소를 선택
:last-of-type	p:last-of-type	부모의 마지막 <p> 요소인 모든 <p> 요소를 선택
:not(selector)	:not(p)<p>	요소가 아닌 모든 요소를 선택
:nth-child(n)	p:nth-child(2)	부모의 두 번째 자식인 모든 <p> 요소를 선택
:nth-last-child(n)	p:nth-last-child(2)	마지막 자식부터 계산하여 부모의 두 번째 자식인 모든 <p> 요소를 선택
:nth-last-of-type(n)	p:nth-last-of-type(2)	마지막 자식부터 부모의 두 번째 <p> 요소인 모든 <p> 요소를 선택
:nth-of-type(n)	p:nth-of-type(2)	부모의 두 번째 <p> 요소인 모든 <p> 요소를 선택
:only-of-type	p:only-of-type	부모의 유일한 <p> 요소인 모든 <p> 요소를 선택
:only-child	p:only-child	부모의 유일한 자식인 모든 <p> 요소를 선택
:target	#news:target	현재 활성 #news 요소 선택(해당 앵커 이름이 포함된 URL 클릭)
:focus	input:focus	포커스가 있는 <input> 요소를 선택
:in-range	input:in-range	지정된 범위 내의 값을 가진 <input> 요소를 선택
:invalid	input:invalid	유효하지 않은 값을 가진 모든 <input> 요소를 선택
:lang(language)	p:lang(it)	"it"으로 시작하는 lang 속성 값을 가진 모든 <p> 요소를 선택
:optional	input:optional	"필수" 속성이 없는 <input> 요소를 선택
:out-of-range	input:out-of-range	지정된 범위를 벗어난 값을 가진 <input> 요소를 선택
:read-only	input:read-only	"readonly" 속성이 지정된 <input> 요소를 선택
:read-write	input:read-write	"readonly" 속성이 없는 <input> 요소를 선택
:required	input:required	"필수" 속성이 지정된 <input> 요소를 선택
:root	root	문서의 루트 요소를 선택
:valid	input:valid	유효한 값을 가진 모든 <input> 요소를 선택

## All CSS Pseudo Elements

Selector	Example	Example description
::after	p::after	모든 <p> 요소 뒤에 내용 삽입
::before	p::before	모든 <p> 요소 앞에 내용 삽입
::first-letter	p::first-letter	모든 <p> 요소의 첫 글자를 선택
::first-line	p::first-line	모든 <p> 요소의 첫 번째 줄을 선택
::selection	p::selection	사용자가 선택한 요소의 부분을 선택(사용자가 드래그나 더블클릭을 하여 블록 처리된 글자)

요소에 직접 style=" " 형식으로 CSS를 기술하는 인라인 스타일(inline style)은 위의 모든 선택자보다 우선으로 실행된다. 지금까지 나온 방법보다 최우선으로 CSS를 적용해야 할 경우에는 선택자에 속성을 부여하는 마지막에 "!important"를 붙이면 된다.

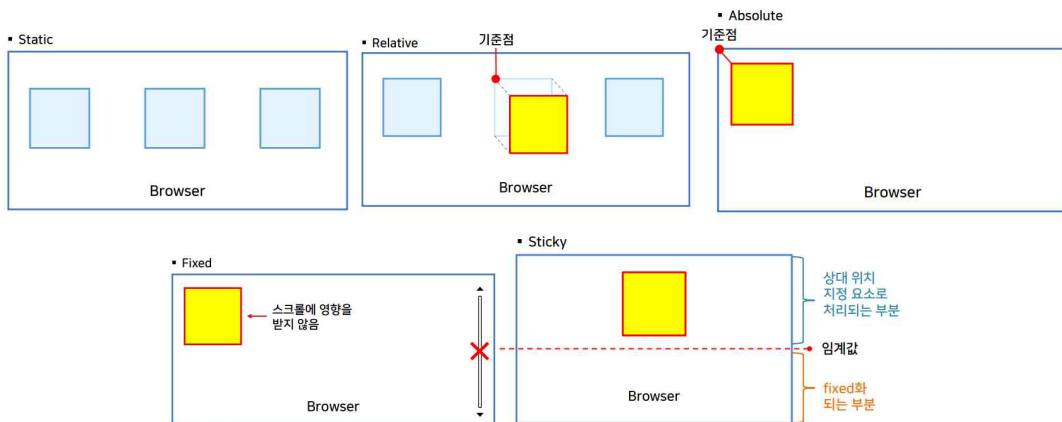
```
P { color: red !important; }
```

Type 선택자 < class 선택자 < id 선택자 < inline style < !important

### 3. 레이아웃

position : 레이아웃을 배치하거나 요소 위치를 지정하는 중요한 속성

속성 값	속성 설명
static	요소를 이동하거나 겹칠 수 없는 원래 그대로의 상태
relative	left, top, right, bottom 속성으로 위치 지정할 수 있음 position: absolute로 지정된 요소의 부모 요소 역할을 할수도 있음
absolute	원래 위치에서 따로 떼어내 독립적으로 새로운 위치를 설정함. left, top, right, bottom 속성으로 위치 지정할 수 있음 부모 박스를 기준으로 위치를 지정함
fixed	요소의 위치를 screen 기준으로 지정함 left, top, right, bottom 속성으로 위치 지정할 수 있음
sticky	position의 계산값이 sticky (끈적끈적한)인 요소 평소에는 상대 위치 지정 요소로 처리하지만, 자신의 플로우 루트에서 지정한 임계값을 넘으면 마치 fixed처럼 화면에 고정되는 요소



z-index : 겹쳐 있는 요소들의 계층을 변경할 때 사용하는 속성

float : block 요소는 원래 세로로만 나열되지만 float을 이용하여 가로로 배치할 수 있다.

속성값	설명
left	요소가 왼쪽으로 이동되도록 지정
right	요소가 오른쪽으로 이동되도록 지정
none	유동성이 없도록 지정, 기본값
inherit	부모 요소의 속성을 이어 받음

clear : float된 박스들의 바로 다음 박스에게 주변을 흐르지 않고 원래대로 아래에 배치되도록 float을 해제시켜주는 속성

그 외 float 해제하는 방법 : float된 요소를 감싸는 박스에 overflow: hidden;

#### 4. 문자 관련 스타일

font-family : 문자의 글꼴을 지정하는 속성

여러 단어이거나 한글 글꼴이 경우는 따옴표로 감싸고 여러 개의 글꼴을 쉼표로 구분한다.

font-family: "돋움", Dotum, Arial, sans-serif;

font-size : 문자의 글자 크기를 지정하는 속성

px : 해상도에 따라 상대적으로 기본 단위

% : 부모 요소의 글자 크기를 100% 기준으로 계산한 % 단위

em : 부모 요소의 글자 크기를 100% 기준으로 계산한 100분의 1 단위

rem : em과 비슷하지만 부모 요소가 아닌 최상위(html) 요소의 크기를 100% 기준으로 계산

vw : 뷰포트 너비값의 100분의 1단위

vh : 뷰포트 높이값의 100분의 1단위

font-weight : 문자를 굵게 하거나 굵은 문자를 보통으로 재설정하는 속성

font-style : 문자를 기울어지게 하거나 기울어지지 않게 재설정하는 속성

font-variant : 문자를 작은 대문자로 설정하거나 원래 문자로 재설정하는 속성

line-height : 줄간격을 px, %, em등의 단위로 지정

CSS font - Shorthand property

font: italic small-caps bold 12px/30px Georgia, serif;

font-style

font-variant

font-weight

font-size/line-height

font-family

글자 크기와 글꼴은 생략할 수 없다.

웹폰트

@font-face{

font-family: "Open Sans";

src: url(webfonts/OpenSans-Regular.eot);

src: url(webfonts/OpenSans-Regular.woff) format("woff"),

url(webfonts/OpenSans-Regular.ttf) format("ttf");

}

color : 글자의 색상을 지정하는 속성

letter-spacing : 글자 간의 간격

word-spacing : 단어 간의 간격

text-decoration : 글자에 밑줄, 윗줄, 가운데줄을 치거나 쳤던 줄을 없애준다.

text-transform : 대소문자 변경을 실행한다.

text-shadow : 글자에 그림자를 주는 속성

text-align : 블록 요소에만 적용되는 속성으로서 문단을 블록의 왼쪽(left), 가운데(center), 오른쪽(right), 양쪽(justify) 등으로 정렬시킨다.

vertical-align : 이미지나 폼 요소를 위(top), 가운데(middle), 아래(bottom)로 정렬시켜 주는 속성으로 img, input, select, label, 테이블의 th, td등에 사용한다.

text-indent : 문단의 첫머리를 들여쓰기 해주는 속성으로서, 블록 요소에만 적용된다.

text-shadow: 2px 3px 5px rgba(0,0,0,0.4);

속성 값	속성 설명
2px	그림자가 원본에서 떨어지는 가로 거리
3px	그림자가 원본에서 떨어지는 세로 거리
5px	그림자가 흐릿하게 퍼지는 정도
rgba(0,0,0,0.4)	그림자의 색상(red, green, blue, alpha)

white-space : 마크업한 문자들 사이의 공백을 처리하여 내보내는 속성

속성 값	속성 설명
normal	연속 공백들과 줄바꿈이 하나의 공백으로 처리->길면 줄바꿈 일어남
nowrap	연속 공백들과 줄바꿈이 하나의 공백으로 처리->길어도 줄바꿈 없이 박스 밖으로 나감
pre	연속 공백들과 줄바꿈이 소스 그대로 표현됨
pre-wrap	연속 공백들과 줄바꿈이 소스 그대로 표현되나, 길면 줄바꿈 일어남
pre-line	연속 공백들이 하나의 공백으로 처리, 줄바꿈은 소스 그대로 표현되나, 길면 줄바꿈 일어남

text-overflow : 긴 텍스트에 말줄임 기호를 사용하고자 할 때 사용

display : 요소의 성격을 바꿔주는 속성 블록, 인라인 요소 변경

visibility : 요소를 숨기거나 보이도록 해주는 속성

overflow : 요소 안의 내용이 요소의 크기보다 많을 경우 넘치는 부분을 처리하는 속성

opacity : 요소에 투명도를 부여할 수 있는 속성으로 속성값은 0(투명)~1(불투명)로 표현



## 5. 배경(background)

background-color : 요소의 배경 색상을 지정하는 속성

background-image : 요소의 배경에 들어갈 이미지를 지정하는 속성

background-repeat : 배경 이미지를 어떻게 반복시킬지를 지정하는 속성

background-position : 배경 이미지를 원하는 위치로 옮겨주는 속성

background-attachment : 배경 이미지를 요소 내에 고정시킬지 화면에 고정시킬지에 대한 속성

background-size : 배경이미지의 크기를 변경할 수 있는 속성

background-origin : 배경 이미지의 시작점을 정하는 속성

속성 값	속성 설명
border-box	배경 이미지가 테두리의 좌측 상단 모퉁이에서 시작함
padding-box	배경 이미지가 안여백의 좌측 상단 모퉁이에서 시작함 (기본값)
content-box	배경 이미지가 콘텐츠의 좌측 상단부터 시작함

CSS background - Shorthand property

background: #ffffff url("이미지") no-repeat right top;

background-color

background-image

background-repeat

background-attachment

background-position

배경에 gradient 적용하기

background: linear-gradient(red, yellow, green);

background: linear-gradient(to top, red, yellow, green);

background: linear-gradient(to right, red, yellow, green);

background: linear-gradient(to bottom right, red, yellow, green);

background: linear-gradient(to right, red, yellow, green);

background: linear-gradient(45deg, red, yellow, green);

background: radial-gradient(red, yellow, green);

background: radial-gradient(circle, red, yellow, green);

background: repeating-linear-gradient(red, green 10%, blue 20%);

background: repeating-radial-gradient(red, green 10%, blue 20%);

## 6. BOX 관련 속성

속성 값	속성 설명
width: 800px;	요소의 가로 폭을 800px로 지정함
height: 100px;	요소의 높이 값을 100px로 지정함
min-width: 1024px;	요소의 가로폭을 1024px 이상으로 지정함
min-height: 200px;	요소의 세로폭을 200px 이상으로 지정함
max-width: 400px;	요소의 너비 값을 400px 이하로 지정함
max-height: 500px;	요소의 높이 값을 500px 이하로 지정함

margin : 요소의 바깥 여백을 지정하는 속성

border : 박스의 테두리를 설정하는 속성

border-radius : 박스의 네 모서리를 둥글게 만들 수 있는 속성

box-sizing : width, height 값에 padding 값이나 border 값은 포함되지 않게 하는 속성

box-shadow : 요소에 그림자를 발생시키는 속성

box-shadow: 8px 15px 10px 7px inset rgba(255,255,255,0.4);

속성 값	속성 설명
8px	그림자가 원본과 가로로 떨어진 간격
15px	그림자가 원본과 세로로 떨어진 간격
10px	그림자의 흐릿한 정도 (blur)
7px	그림자의 확장(spread-양수는 확장, 음수는 축소)
inset	그림자가 요소의 안쪽에 나타남
rgba(255,255,255,0.4);	그림자의 색상

## 7. 기타 CSS3속성

### (1) CSS object-fit 속성

컨테이너에 맞게 <img> 또는 <video> 크기를 조정하는 방법을 지정하는 데 사용한다.

이 속성은 콘텐츠에 다양한 방법으로 컨테이너를 채우도록 지시한다

fill - 기본값. 이미지는 주어진 치수를 채우도록 크기가 조정됩니다. 필요한 경우 이미지에 맞게 늘어나거나 찌그러집니다.

contain- 이미지는 종횡비를 유지하지만 주어진 크기에 맞게 크기가 조정됩니다.

cover- 이미지는 종횡비를 유지하고 주어진 크기를 채웁니다. 이미지에 맞게 잘립니다.

none- 이미지의 크기가 조정되지 않습니다.

scale-down- 이미지가 가장 작은 버전 none또는 contain

CSS object-position속성은 컨테이너 내에서 <img> 또는 <video>를 배치하는 방법을 지정하는 데 사용한다.

```
img {  
  width: 200px;  
  height: 300px;  
  object-fit: cover;  
  object-position: 80% 100%;  
}
```

### (2) 2D transform

CSS transforms을 사용하면 요소를 이동, 회전, 비율 조정 및 기울일 수 있습니다.

translate()

rotate()

scaleX()

scaleY()

scale()

skewX()

skewY()

skew()

matrix()

translate()는 현재 위치에서 요소를 이동합니다(X축 및 Y축에 지정된 매개변수에 따라 요소를 이동).

translateX(x)는 X축 값만 사용하여 3D 변환을 정의합니다.

translateY(y)는 Y축 값만 사용하여 3D 변환을 정의합니다.

translateZ(z)는 Z축 값만 사용하여 3D 변환을 정의합니다.

rotate()는 주어진 각도에 따라 요소를 시계 방향 또는 시계 반대 방향으로 회전합니다. 음

수 값을 사용하면 요소가 시계 반대 방향으로 회전합니다.  
 rotateX()메서드는 지정된 각도에서 X축을 중심으로 요소를 회전합니다.  
 rotateY()메서드는 지정된 각도에서 Y축을 중심으로 요소를 회전합니다.  
 rotateZ()메서드는 주어진 각도에서 Z축을 중심으로 요소를 회전합니다.

scale()는 요소의 크기를 늘리거나 줄입니다(너비 및 높이에 대해 지정된 매개 변수에 따라  
 요소의 크기를 늘리거나 줄임).  
 scaleX()는 요소의 너비를 늘리거나 줄입니다.  
 scaleY()는 요소의 높이를 늘리거나 줄입니다.

skewX()는 지정된 각도만큼 X축을 따라 요소를 기울입니다.  
 skewY()는 주어진 각도만큼 Y축을 따라 요소를 기울입니다.  
 skew()는 지정된 각도만큼 X축과 Y축을 따라 요소를 기울입니다.  
 matrix()방법은 모든 2D 변환 방법을 하나로 결합합니다.  
 matrix()는 요소를 회전, 크기 조정, 이동(변환) 및 기울일 수 있는 수학 함수를 포함하는 6개  
 의 매개 변수를 사용합니다.  
 matrix(scaleX(), skewY(), skewX(), scaleY(), translateX(), translateY());

예)  
 div {  
   transform: matrix(1, -0.3, 0, 1, 0, 0);  
 }

#### CSS Transform Properties

Property	Description
transform	요소에 2D 또는 3D 변환을 적용합니다.
transform-origin	변환된 요소의 위치를 변경할 수 있습니다.
transform-style	중첩된 요소가 3D 공간에서 렌더링되는 방식을 지정합니다.
perspective	3D 요소가 표시되는 방식에 대한 관점을 지정합니다.

### CSS transform-origin 속성

Property Value	Description
x-axis	뷰가 x축에 배치되는 위치를 정의합니다 가능한 값 : left center right length %
y-axis	뷰가 y축에 배치되는 위치를 정의합니다. 가능한 값: top center bottom length %

### CSS transform-style 속성

변환된 자식 요소가 3D 변환을 유지하도록 합니다.

Property Value	Description
flat	하위 요소가 3D 위치를 유지하지 않도록 지정합니다. 기본값입니다
preserve-3d	하위 요소가 3D 위치를 유지하도록 지정합니다.

## (2) CSS Transitions

CSS Transitions을 사용하면 지정된 기간 동안 속성 값을 원활하게 변경할 수 있습니다.

transition

transition-property

transition-duration

transition-timing-function

transition-delay

Property	Description
transition	네 가지 전환 속성을 단일 속성으로 설정하기 위한 속기 속성(property duration timing-function delay)
transition-property	전환 효과에 대한 CSS 속성의 이름을 지정합니다.
transition-duration	전환 효과가 완료되는 데 걸리는 시간(초 또는 밀리초)을 지정합니다.
transition-timing-function	전환 효과의 속도 곡선을 지정합니다.
transition-delay	전환 효과에 대한 지연(초)을 지정합니다.

transition-timing-function속성은 전환 효과의 속도 곡선을 지정합니다.

transition-timing-function 속성은 다음 값을 가질 수 있습니다.

ease - 느린 시작, 빠르게 시작, 느리게 종료하는 전환 효과를 지정합니다(기본값).

linear - 처음부터 끝까지 동일한 속도로 전환 효과를 지정합니다.

ease-in - 느린 시작으로 전환 효과 지정

ease-out - 느린 끝으로 전환 효과를 지정합니다.

ease-in-out - 느린 시작과 끝으로 전환 효과 지정

cubic-bezier(n,n,n,n)- 큐빅-베지어 함수에서 자신만의 값을 정의할 수 있습니다.

### (3) CSS Animations

CSS는 JavaScript 또는 Flash를 사용하지 않고 HTML 요소의 애니메이션을 허용합니다. 애니메이션을 사용하면 요소가 한 스타일에서 다른 스타일로 점진적으로 변경됩니다. 원하는 만큼 CSS 속성을 원하는 만큼 변경할 수 있습니다.

@keyframes

animation-name

animation-duration

animation-timing-function

animation-delay

animation-iteration-count

animation-direction

animation-fill-mode

animation-play-state

animation

#### CSS Animation Properties

Property	Description
@keyframes	애니메이션 코드를 지정합니다.
animation-name	@keyframes 애니메이션의 이름을 지정합니다.
animation-duration	애니메이션이 한 주기를 완료하는 데 걸리는 시간을 지정합니다.
animation-timing-function	애니메이션의 속도 곡선을 지정합니다.
animation-delay	애니메이션 시작 지연을 지정합니다.
animation-iteration-count	애니메이션을 재생해야 하는 횟수를 지정합니다.
animation-direction	애니메이션을 앞으로, 뒤로 또는 대체 주기로 재생해야 하는지 여부를 지정합니다.
animation-fill-mode	애니메이션이 재생되지 않을 때(시작 전, 종료 후 또는 둘 다) 요소의 스타일을 지정합니다.
animation-play-state	애니메이션이 실행 중인지 또는 일시 중지되었는지 지정합니다.
animation	모든 애니메이션 속성을 설정하기 위한 속기 속성

CSS 애니메이션을 사용하려면 먼저 애니메이션에 대한 몇 가지 키프레임을 지정해야 합니다. 키프레임은 특정 시간에 요소가 가질 스타일을 유지합니다.

#### @keyframes 규칙

규칙 내부에 CSS 스타일을 지정하면 @keyframes 애니메이션이 특정 시간에 현재 스타일에서 새 스타일로 점진적으로 변경됩니다.

애니메이션이 작동하려면 애니메이션을 요소에 바인딩해야 합니다.

다음 예제는 "예제" 애니메이션을 <div> 요소에 바인딩합니다. 애니메이션은 4초 동안 지속되며 <div> 요소의 배경색이 "빨간색"에서 "노란색"으로 점차 변경됩니다.

예)

```
@keyframes example {  
  from {background-color: red;}  
  to {background-color: yellow;}  
}  
  
div {  
  width: 100px;  
  height: 100px;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
}
```

#### animation-name

animation-name은 @keyframes 애니메이션의 이름을 지정합니다.

#### animation-duration

animation-duration은 애니메이션을 완료하는 데 걸리는 시간을 정의합니다.

animation-duration을 지정하지 않으면 기본값이 0초(0초)이므로 애니메이션이 발생하지 않습니다.

"from" 및 "to"(0%(시작) 및 100%(완료)를 나타냄) 키워드를 사용하여 스타일이 변경되는 시기를 지정했습니다.

퍼센트를 사용하는 것도 가능합니다. 퍼센트를 사용하여 원하는 만큼 스타일 변경을 추가할 수 있습니다.



## animation-timing-function

animation-timing-function 속성은 애니메이션의 속도 곡선을 지정합니다.

animation-timing-function 속성은 다음 값을 가질 수 있습니다.

ease - 천천히 시작한 다음 빠르게, 그 다음에는 느리게 끝나는 애니메이션을 지정합니다(기본값).

linear - 처음부터 끝까지 동일한 속도로 애니메이션을 지정합니다.

ease-in - 느린 시작으로 애니메이션을 지정합니다.

ease-out - 느린 종료로 애니메이션을 지정합니다.

ease-in-out - 느린 시작과 끝으로 애니메이션을 지정합니다.

cubic-bezier(n,n,n,n) - 큐빅-베지어 함수에서 자신만의 값을 정의할 수 있습니다.

## animation-delay

animation-delay 속성은 애니메이션 시작 지연을 지정합니다.

음수 값도 허용됩니다. 음수 값을 사용하면 이미 N 초 동안 재생된 것처럼 애니메이션이 시작됩니다 .

다음 예제에서는 애니메이션이 이미 2초 동안 재생된 것처럼 시작됩니다.

```
div {  
  width: 100px;  
  height: 100px;  
  position: relative;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
  animation-delay: -2s;  
}
```

## animation-iteration-count

animation-iteration-count은 애니메이션이 실행되어야 하는 횟수를 지정합니다.

다음 예제에서는 애니메이션이 중지되기 전에 애니메이션을 3번 실행합니다.

```
div {  
  width: 100px;  
  height: 100px;  
  position: relative;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
  animation-iteration-count: 3;  
}
```

"infinite" 값을 사용하여 애니메이션이 영원히 계속되도록 합니다.

## animation-direction

animation-direction 속성은 애니메이션을 앞으로, 뒤로 또는 대체 주기로 재생해야 하는지 여부를 지정합니다.

animation-direction 속성은 다음 값을 가질 수 있습니다.

normal - 애니메이션이 정상적으로 재생됩니다(앞으로). 이것은 기본값입니다

reverse - 애니메이션이 역방향(backwards)으로 재생됩니다.

alternate - 애니메이션이 먼저 앞으로 재생된 다음 뒤로 재생됩니다.

alternate-reverse - 애니메이션이 먼저 뒤로 재생된 다음 앞으로 재생됩니다.

## animation-fill-mode

CSS 애니메이션은 첫 번째 키프레임이 재생되기 전이나 마지막 키프레임이 재생된 후에 요소에 영향을 주지 않습니다. animation-fill-mode 속성은 이 동작을 재정의할 수 있습니다.

animation-fill-mode 속성은 애니메이션이 재생되지 않을 때(시작 전, 종료 후 또는 둘 다) 대상 요소에 대한 스타일을 지정합니다.

animation-fill-mode 속성

1. none - 기본값. 애니메이션은 실행 전이나 후에 요소에 스타일을 적용하지 않습니다.
2. forwards - 요소는 마지막 키프레임에 의해 설정된 스타일 값을 유지합니다(애니메이션 방향 및 애니메이션 반복 횟수에 따라 다름).
3. backwards - 요소는 첫 번째 키프레임(애니메이션 방향에 따라 다름)에 의해 설정된 스타일 값을 가져오고 애니메이션 지연 기간 동안 이 값을 유지합니다.
4. both - 애니메이션은 정방향 및 역방향 모두에 대한 규칙을 따르며 애니메이션 속성을 양방향으로 확장합니다.

## animation-play-state

animation-play-state 속성은 애니메이션이 실행 중인지 또는 일시 중지되었는지 지정합니다.  
속성 값

Value	Description
paused	애니메이션이 일시 중지되었음을 지정합니다.
running	기본값. 애니메이션이 실행 중임을 지정합니다.

CSS Syntax

animation: name duration timing-function delay iteration-count direction fill-mode play-state;

#### (4) CSS Multiple Columns

CSS Multiple Columns 레이아웃을 사용하면 신문에서와 같이 텍스트의 여러 열을 쉽게 정의할 수 있습니다.

column-count  
column-gap  
column-rule-style  
column-rule-width  
column-rule-color  
column-rule  
column-span  
column-width

column-count속성은 요소를 나누어야 하는 열 수를 지정합니다.

column-gap속성은 열 사이의 간격을 지정합니다.

column-rule-style속성은 열 사이의 규칙 스타일을 지정합니다.

column-rule-width속성은 열 사이의 규칙 너비를 지정합니다.

column-rule-color속성은 열 사이의 규칙 색상을 지정합니다.

column-rule속성은 위의 모든 column-rule-\* 속성을 설정하기 위한 속기 속성입니다.

예)

```
div {  
    column-rule: 1px solid lightblue;  
}
```

column-span속성은 요소가 걸쳐야 하는 열 수를 지정합니다.

column-width속성은 열에 대해 제안된 최적의 너비를 지정합니다.