



# CSS3 스타일 속성 기본

모던 웹 디자인을 위한 HTML5 + CSS3 입문

# 4.1 CSS3 단위

1. Css적용과 단위 (1~6)
2. 선택자 (\*)
3. 폰트 (36~43)
4. 디스플레이 (7~11)
5. 백그라운드 (23~35)
6. 패딩마진 (12~17)
7. 박스속성 (53~55)
8. 포지션 (44~52)
9. 그림자 (56~58)



# 4.1 CSS3 단위

## ❖ CSS는 각각의 스타일 속성에 다양한 값을 입력한다.

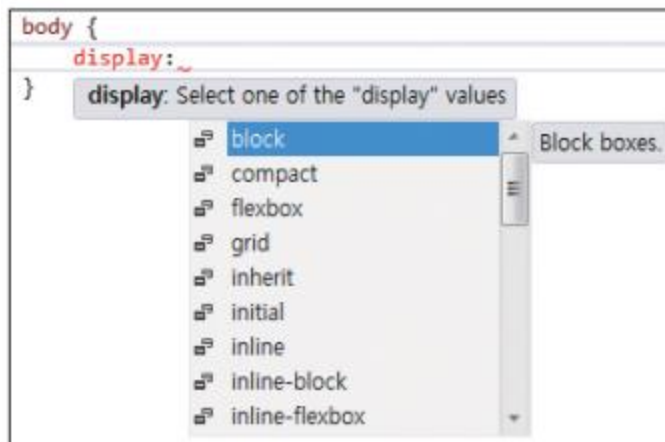
그림 3-1 CSS 블록

```
h1 { color: red; }  
      선택자      스타일 속성      스타일 값
```

## ❖ 키워드 단위

- 키워드는 CSS를 개발할 때에 이미 지정된 단어들을 의미한다.
- 통합 개발 환경을 사용할 때에 다음과 같이 확인할 수 있다.

그림 4-3 키워드



# 4.1 CSS3 단위

## ❖ 크기 단위

- CSS3에서 사용하는 크기 단위는 %, em, cm, mm, inch, px이다.

표 4-1 크기 단위

단위	설명
%	백분율 단위
em	배수 단위
px	픽셀

- 크기 단위는 다음과 같은 방법으로 사용한다.

코드 4-3 퍼센트 단위

```
<style>
  p:nth-child(1) { }
  p:nth-child(2) { font-size: 100%; }
  p:nth-child(3) { font-size: 150%; }
  p:nth-child(4) { font-size: 200%; }
</style>
```



## ❖ 색상 단위

- 색상을 적용하는 가장 기본 방법은 키워드를 사용하는 것이다.

코드 4-8 키워드를 사용한 색상 적용

```
<style>
  h1 { background-color: red; }
  h2 { background-color: orange; }
  h3 { background-color: blue; }
  h4 { background-color: green; }
  h5 { background-color: brown; }
  h6 { background-color: purple; }
</style>
```

- CSS3는 다음과 같은 색상 단위를 사용해 색상을 지정한다.

표 4-2 색상 단위

단위 형태	설명
#000000	HEX 코드 단위
rgb(red, green, blue)	RGB 색상 단위
rgba(red, green, blue, alpha)	RGBA 색상 단위
hsl(hue, saturation, lightness)	HSL 색상 단위
hsla(hue, saturation, lightness, alpha)	HSLA 색상 단위



# 4.1 CSS3 단위

## ❖ URL 단위

- 파일을 지정할 때는 URL 단위를 사용한다.
- URL 단위는 다음과 같은 방법으로 사용한다.

코드 4-14 URL 경로 표시 방법

```
/* 현재 폴더의 Desert.jpg */  
background-image: url('Desert.jpg');  
  
/* 현재 폴더 내부의 Other 폴더의 Desert.jpg */  
background-image: url('Other/Desert.jpg');  
  
/* 루트 폴더의 Desert.jpg */  
/* 루트 폴더의 개념은 서버를 알아야 합니다. */  
background-image: url('/Desert.jpg');
```



### ❖ display 속성

- display 속성은 태그의 영역 표현 방식을 지정하는 속성이다.
- 다음 키워드를 입력할 수 있다.

표 4-3 display 속성에 적용 가능한 키워드

스타일 이름	설명
none	태그를 화면에서 보이지 않게 만듭니다.
block	태그를 block 형식으로 지정합니다.
inline	태그를 inline 형식으로 지정합니다.
inline-block	태그를 inline-block 형식으로 지정합니다.

- 다음과 같은 방법으로 사용한다.

코드 4-19 inline 키워드와 inline-block 키워드

```
<style>
  #box {
    display: inline;

    background-color: red;
    width: 300px; height: 50px;
    margin: 10px;
  }
</style>
```

```
<style>
  #box {
    display: inline-block;

    background-color: red;
    width: 30px; height: 50px;
    margin: 10px;
  }
</style>
```



## 4.2 가시 속성

- none 키워드를 사용하면 태그가 완전히 화면에서 제거된다.

그림 4-12 none 키워드

Dummy Dummy

- block 키워드를 사용하면 태그가 한 줄을 차지한다.

그림 4-13 block 키워드 적용

Dummy
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Dummy

- inline 키워드 또는 inline-block 키워드를 사용하면 한 줄에 들어간다.

그림 4-14 inline 키워드와 inline-block 키워드 적용

Dummy	Lorem ipsum dolor sit amet, consectetur adipiscing elit.	Dummy
-------	--	-------





## 4.2 가시 속성

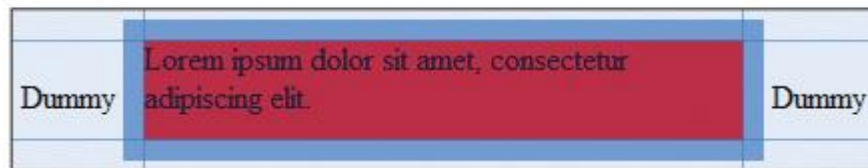
- inline 키워드를 사용할 때와 inline-block 키워드를 사용할 때의 차이점은 margin 속성과 padding 속성을 적용할 때에 알 수 있다.
- inline 키워드를 사용하면 margin 속성과 padding 속성이 좌우로만 적용된다.

그림 4-15 inline 키워드 적용



- inline-block 키워드를 사용하면 margin 속성과 padding 속성이 상하 좌우로 적용된다.

그림 4-16 inline-block 키워드 적용



### ❖ visibility 속성

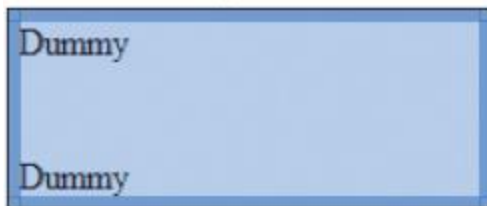
- 대상을 보이거나 보이지 않게 지정하는 속성이다.

표 4-4 visibility 속성에 적용 가능한 키워드

스타일 이름	설명
visible	태그를 보이게 만듭니다.
hidden	태그를 보이지 않게 만듭니다.
collapse	table 태그를 보이지 않게 만듭니다.

- display 속성을 none 키워드로 지정하는 것과 달리 영역을 유지한 채로 보이지 않게만 만든다.

그림 4-18 visibility 속성에 hidden 키워드 적용



### ❖ opacity 속성

- 대상의 투명도를 지정하는 속성이다.
- 0부터 1 사이의 숫자를 입력한다. (0은 투명이고 1은 불투명이다.)
- 다음과 같은 방법으로 사용한다.

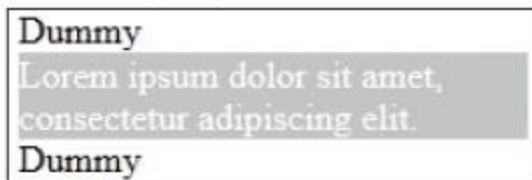
코드 4-22 opacity 속성

```
<style>
  #box {
    background-color: black;
    color: white;

    opacity: 0.2;
  }
</style>
```

- 코드를 실행하면 다음과 같이 출력한다.

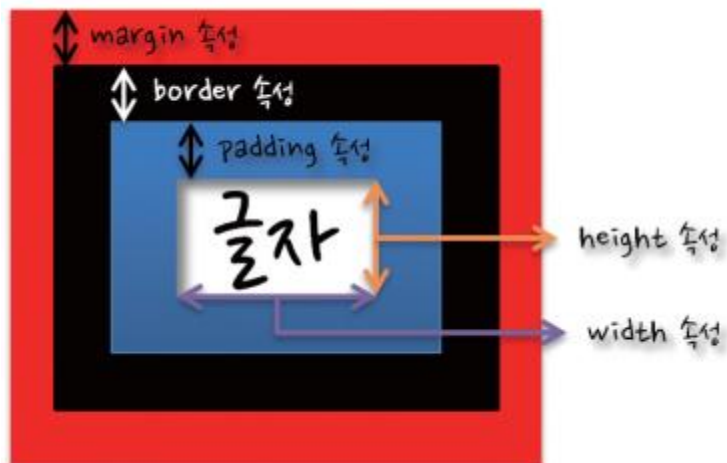
그림 4-21 opacity 속성



## 4.3 박스 속성

- ❖ 박스 속성은 웹 페이지의 레이아웃을 구성할 때 가장 중요한 속성이다.
- ❖ 다음 속성을 모두 합쳐 박스 속성이라고 이야기한다.

그림 4-22 박스 속성



### ❖ width 속성과 height 속성

- width 속성과 height 속성은 글자를 감싸는 영역의 크기를 지정하는 속성이다.
- 다음과 같은 방식으로 사용한다.

코드 4-23 width 속성과 height 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS Property Basic</title>
  <style>
    div {
      width: 100px; height: 100px;
      background-color: red;
    }
  </style>
</head>
<body>
  <div></div>
</body>
</html>
```

그림 4-23 width 속성과 height 속성



### ❖ margin 속성과 padding 속성

- margin 속성은 마진의 너비를 지정하는 속성이고 padding 속성은 패딩의 너비를 지정하는 속성이다.

코드 4-24 margin 속성과 padding 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS Property Basic</title>
  <style>
    div {
      width: 100px; height: 100px;
      background-color: red;

      border: 20px solid black;
      margin: 10px; padding: 30px;
    }
  </style>
</head>
<body>
  <div></div>
</body>
</html>
```

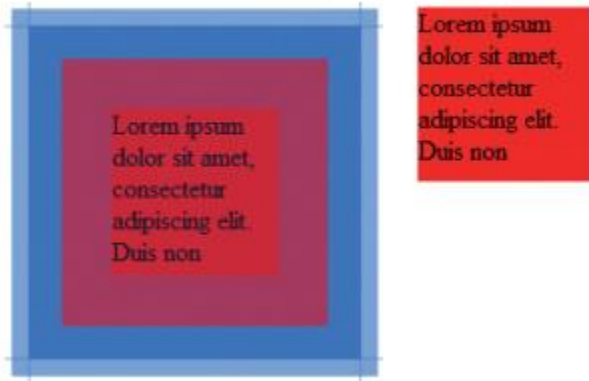
그림 4-24 margin 속성과 padding 속성



## 4.3 박스 속성

- margin 속성과 padding 속성은 width 속성과 height 속성과 별도로 적용된다.

그림 4-25 width 속성과 height 속성의 적용 범위



- 박스 크기는 다음과 같은 공식을 갖는다.

전체 너비 =  $\text{width} + 2 \times (\text{margin} + \text{border} + \text{padding})$

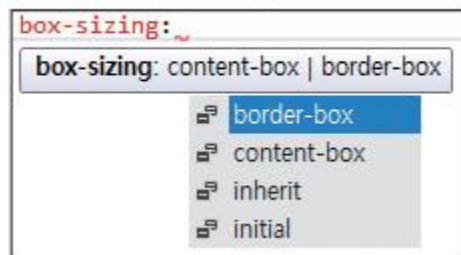
전체 높이 =  $\text{height} + 2 \times (\text{margin} + \text{border} + \text{padding})$



### ❖ box-sizing 속성

- width 속성과 height 속성은 글자를 감싸는 영역의 크기를 지정하는 속성이다. box-sizing 속성은 이러한 공식을 변경할 수 있는 CSS3 속성이다.
- 다음과 같은 키워드를 입력할 수 있다.

그림 4-29 box-sizing 속성에 적용할 수 있는 키워드

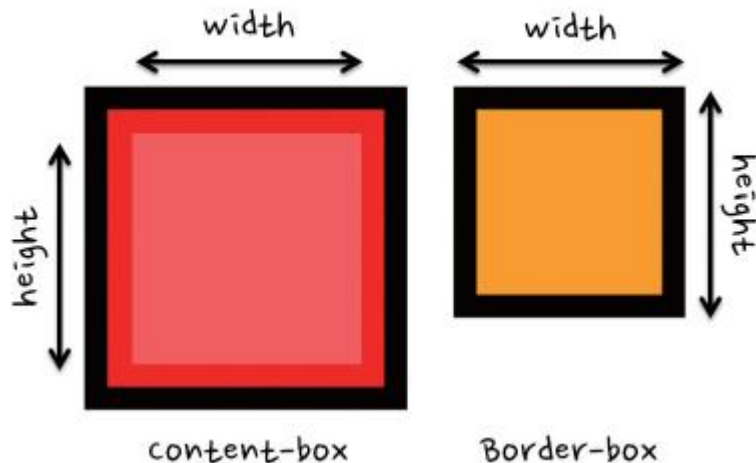




## 4.3 박스 속성

- 각각의 키워드를 적용할 때에 width 속성과 height 속성의 적용 범위가 다음과 같이 변경된다.

그림 4-31 content-box 키워드와 border-box 키워드



- content-box 키워드의 경우 박스의 크기를 다음 공식으로 구한다.

박스 너비 = width 속성 + 2 × (margin 속성 + border 속성 + padding 속성)

박스 높이 = height 속성 + 2 × (margin 속성 + border 속성 + padding 속성)

- border-box 키워드의 경우 박스의 크기를 다음 공식으로 구한다.

박스 너비 = width 속성 + 2 × margin 속성

박스 높이 = height 속성 + 2 × margin 속성

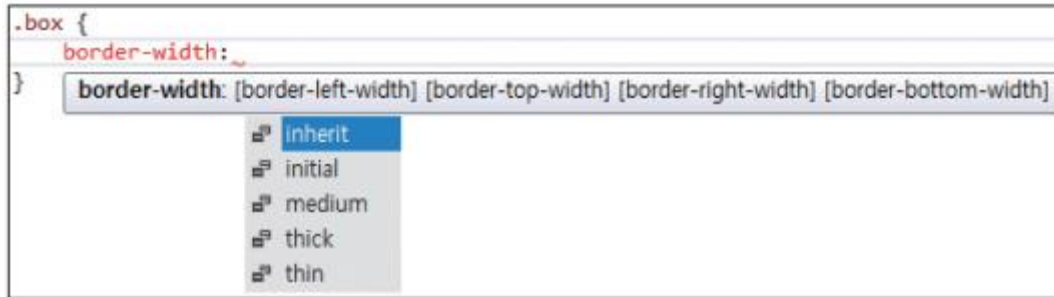


## 4.4 테두리 속성

### ❖ border-width 속성

- 테두리의 너비를 지정하는 속성이다.
- 다음 키워드를 적용한다.

그림 4-32 border-width 속성에 적용 가능한 키워드

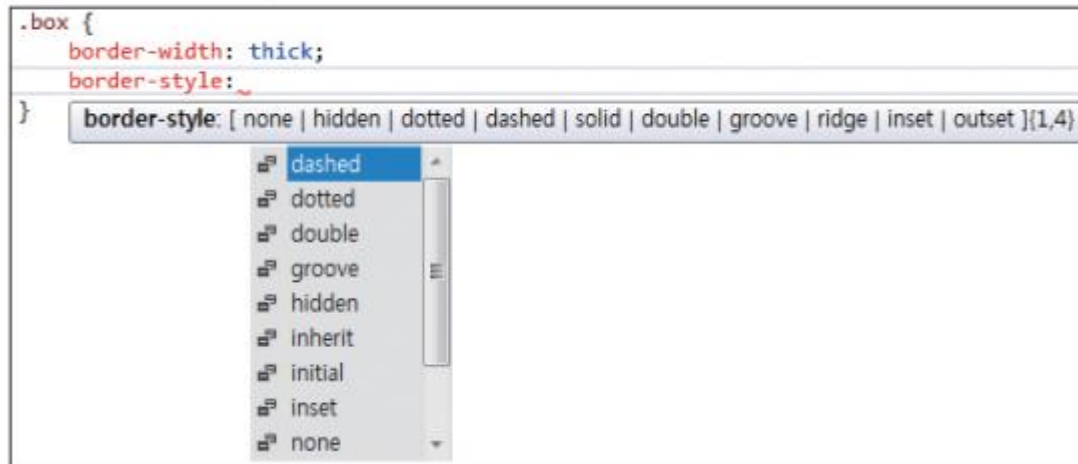


## 4.4 테두리 속성

### ❖ border-style 속성

- 테두리의 형태를 지정하는 속성이다.
- 다음 키워드를 적용한다.

그림 4-33 border-style 속성에 적용 가능한 키워드



## 4.4 테두리 속성

### ❖ border-width, border-style, border-color 속성을 사용하여 테두리를 생성할 수 있다.

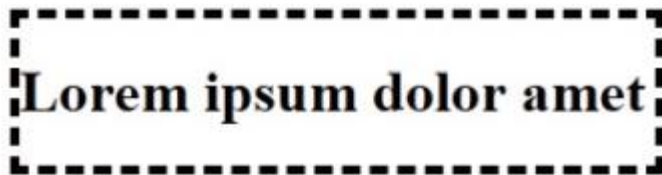
- 각각의 속성은 다음과 같은 방법으로 사용한다.

코드 4-28 테두리 속성

```
<style>
  .box {
    border-width: thick;
    border-style: dashed;
    border-color: black;
  }
</style>
```

- 코드를 실행하면 다음과 같이 출력한다.

그림 4-34 테두리 속성



## 4.4 테두리 속성

### ❖ 방금 살펴본 3가지 테두리 속성은 border 속성으로 한번에 입력할 수 있다.

코드 4-29 border 속성

```
<style>
  .box {
    border: thick dashed black;
  }
</style>
```

### ❖ border-radius 속성

- border-radius 속성을 사용하면 테두리가 둥근 사각형 또는 원을 만들 수 있다.
- 다음과 같은 방식으로 사용한다.

코드 4-30 border-radius 속성

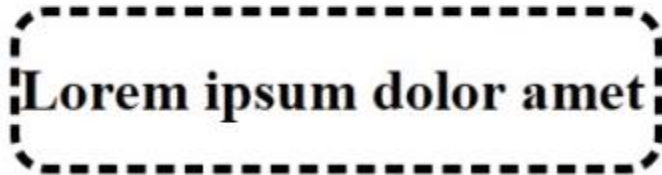
```
<style>
  .box {
    border: thick dashed black;
    border-radius: 20px;
  }
</style>
```



## 4.4 테두리 속성

- 코드를 실행하면 다음 그림처럼 출력한다.

그림 4-38 border-radius 속성



- 각각의 테두리의 둥글기를 설정할 수도 있다.

코드 4-31 각각의 모서리에 다른 border-radius 속성 적용

```
<style>
  .box {
    border: thick dashed black;

    /* border-radius: 왼쪽위 오른쪽위 오른쪽아래 왼쪽아래 */
    border-radius: 50px 40px 20px 10px;
  }
</style>
```

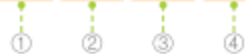
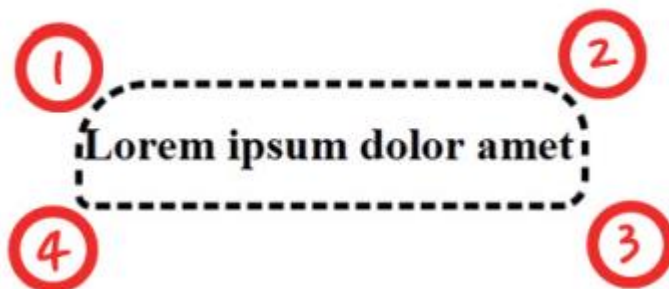


그림 4-39 각각의 모서리에 다른 border-radius 속성 적용



### ❖ background-image 속성

- 배경에 넣을 그림을 지정하는 속성이다.
- 다음과 같은 방법으로 사용한다.

코드 4-32 background-image 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Property Basic</title>
  <style>
    body {
      background-image: url('BackgroundFront.png');
    }
  </style>
</head>
<body>

</body>
</html>
```

그림 4-41 폴더 구성



## 4.5 배경 속성

- 코드를 실행하면 다음과 같이 출력한다.

그림 4-42 background-image 속성





## 4.5 배경 속성

- 이미지를 중첩해서 사용할 수 있다.(CSS3부터 지원하므로 구 버전의 웹 브라우저에서는 사용 불가능)

코드 4-33 이미지 중첩 사용

```
<style>
  body {
    background-image: url('BackgroundFront.png'), url('BackgroundBack.png');
  }
</style>
```

- 코드를 실행하면 다음과 같이 출력한다.

그림 4-43 이미지 중첩



- 왼쪽에 위치하는 그림이 앞에 위치한다.



### ❖ background-size 속성

- 그림 크기를 조절할 때에 사용하는 속성이다.
- CSS3에서 추가된 속성이므로 인터넷 익스플로러 8 이하에서는 사용할 수 없다.
- 다음과 같은 크기 단위를 넣어 사용한다.

코드 4-34 background-size 속성의 너비

```
<style>
  body {
    background-image: url('BackgroundFront.png'), url('BackgroundBack.png');
    background-size: 100%;
  }
</style>
```



## 4.5 배경 속성

- 코드를 실행하면 다음과 같이 출력한다.

그림 4-45 background-size 속성의 너비



- 크기 단위를 2개 입력하면 두 번째 크기 단위는 높이를 의미한다.

코드 4-35 background-size 속성의 높이

```
<style>
  body {
    background-image: url('BackgroundFront.png'), url('BackgroundBack.png');
    background-size: 100% 250px;
  }
</style>
```



## 4.5 배경 속성

- 심표를 사용해 구분하면 여러 개의 배경 이미지에 크기를 각각 적용한다.

코드 4-36 중첩 이미지의 크기 조정

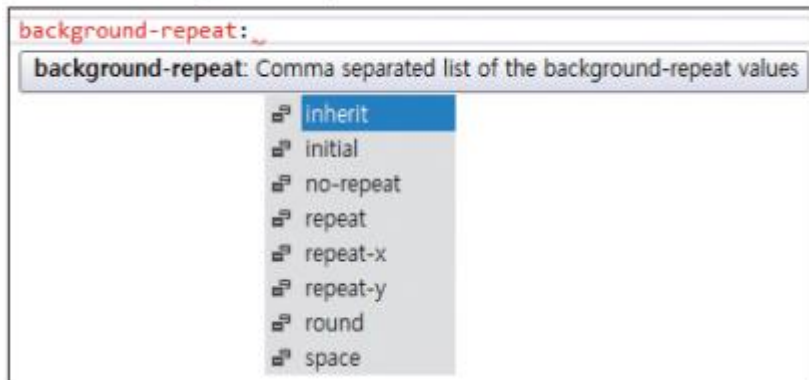
```
<style>
  body {
    background-image: url('BackgroundFront.png'), url('BackgroundBack.png');
    background-size: 100%, 200%;
  }
</style>
```



### ❖ background-repeat 속성

- 배경 패턴 방식을 지정하는 속성이다.
- 다음 키워드를 입력할 수 있다.

그림 4-49 background-repeat 속성



- 다음과 같은 방식으로 사용한다.

코드 4-37 background-repeat 속성에 no-repeat 키워드 적용

```
<style>
  body {
    background-image: url('BackgroundFront.png'), url('BackgroundBack.png');
    background-size: 100%;
    background-repeat: no-repeat;
  }
</style>
```



## 4.5 배경 속성

- 코드를 실행하면 다음과 같이 출력한다.

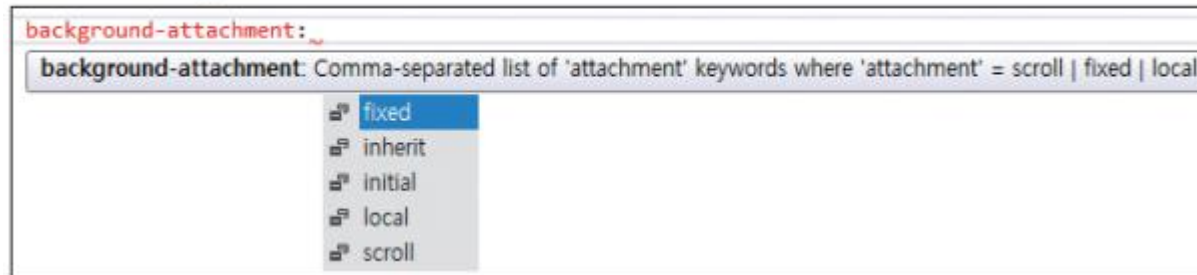
그림 4-50 background-repeat 속성에 no-repeat 키워드 적용



### ❖ background-attachment 속성

- 배경 이미지를 어떠한 방식으로 화면에 붙일 것인지를 지정하는 속성이다.
- 다음 키워드를 적용할 수 있다.

그림 4-51 background-attachment 속성에 적용 가능한 키워드



# 4.5 배경 속성

## ■ 다음과 같은 방식으로 사용한다.

코드 4-39 background-attachment 속성에 fixed 키워드 적용

```
<style>
  body {
    background-color: #E7E7E8;
    background-image: url('BackgroundFront.png'), url('BackgroundBack.png');
    background-size: 100%;
    background-repeat: no-repeat;
    background-attachment: fixed;
  }
</style>
```

## ■ 코드를 실행하면 다음과 같이 출력한다.

그림 4-52 background-attachment 속성(1)



그림 4-54 background-attachment 속성에 fixed 키워드 적용

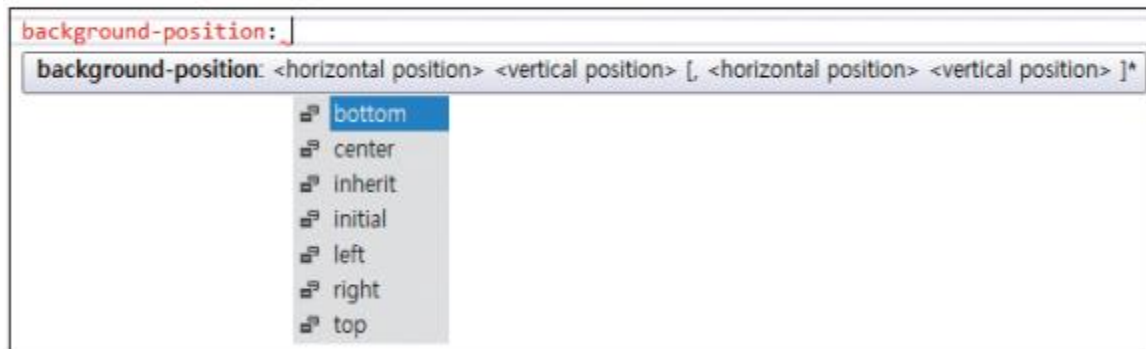




### ❖ background-position 속성

- 배경의 위치를 지정하는 속성이다.
- 다음과 같은 형식으로 값을 입력한다.
  - background-position: 키워드;
  - background-position: X축 크기;
  - background-position: X축 크기 Y축 크기;
- 다음 키워드를 사용한다.

그림 4-55 background-position 속성



## 4.5 배경 속성

- 다음과 같은 방식으로 사용한다.

코드 4-40 background-position 속성에 bottom 키워드 적용

```
<style>
  body {
    background-color: #E7E7E8;
    background-image: url('BackgroundFront.png'), url('BackgroundBack.png');
    background-size: 100%;
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-position: bottom;
  }
</style>
```



- 코드를 실행하면 다음과 같이 출력한다.

그림 4-56 background-position 속성에 bottom 키워드 적용



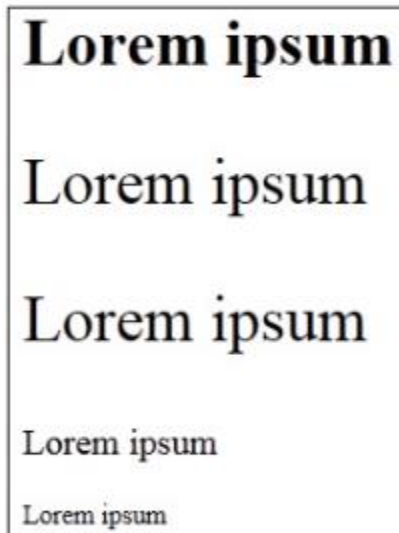
## ❖ font-size 속성

- 글자의 크기를 지정할 때에 사용하는 속성이다.
- 다음과 같은 방법으로 사용한다.

코드 4-44 font-size 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Font Property</title>
  <style>
    .a { font-size: 32px; }
    .b { font-size: 2em; }
    .c { font-size: large; }
    .d { font-size: small; }
  </style>
</head>
<body>
  <h1>Lorem ipsum</h1>
  <p class="a">Lorem ipsum</p>
  <p class="b">Lorem ipsum</p>
  <p class="c">Lorem ipsum</p>
  <p class="d">Lorem ipsum</p>
</body>
</html>
```

그림 4-59 font-size 속성 적용



### ❖ font-family 속성

- 폰트를 지정하는 속성이다.
- 다음과 같이 폰트 이름을 입력해서 사용한다.

코드 4-45 font-family 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Font Property</title>
  <style>
    .font_arial { font-family: Arial }
    .font_roman { font-family: 'Times New Roman' }
  </style>
</head>
<body>
  <h1 class="font_arial">Lorem ipsum</h1>
  <p class="font_roman">Lorem ipsum</p>
</body>
</html>
```



## 4.6 폰트 속성

- 사용자의 컴퓨터에 폰트가 없으면 폰트가 적용되지 않는다.
- 만약을 대비해 여러 개의 폰트를 연속적으로 입력한다.
- 하지만 다국적 웹 페이지를 제공할 경우 사용자에게 무슨 폰트가 있는지 일일이 확인할 수 없다.
- 이러한 경우에는 generic-family 폰트를 사용한다.
- generic-family 폰트는 웹 브라우저에서 미리 지정하고 있는 폰트이다.

그림 4-64 크롬 폰트 설정

<b>Serif 체</b>	Adobe Arabic ▼	16: Lorem ipsum dolor sit amet, consectetur adipiscing elit.
<b>Sans-serif 체</b>	Adobe Arabic ▼	16: Lorem ipsum dolor sit amet, consectetur adipiscing elit.
<b>고정폭 글꼴</b>	Adobe Arabic ▼	Lorem ipsum dolor sit amet, consectetur adipiscing elit.



## 4.6 폰트 속성

- Serif 폰트(명조체), Sans-serif 폰트(고딕체), Mono space 폰트(고정 폭 글꼴)를 사용할 수 있다.
- 다음과 같은 방법으로 사용한다.

코드 4-47 generic-family 폰트

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Font Property</title>
  <style>
    .font_arial { font-family: '없는 폰트', sans-serif }
    .font_roman { font-family: '없는 폰트', serif }
  </style>
</head>
<body>
  <h1 class="font_arial">Lorem ipsum</h1>
  <p class="font_roman">Lorem ipsum</p>
</body>
</html>
```

인터넷 익스플로러의 경우  
sans-serif 폰트와 serif 폰  
트에 따옴표를 적용하면 안  
됩니다.

그림 4-65 generic-family 폰트 사용



### ❖ font-style 속성과 font-weight 속성

- 폰트의 기울기와 두께를 지정하는 속성이다.
- 각각의 속성에는 다음 키워드를 입력한다.

그림 4-66 font-style 속성

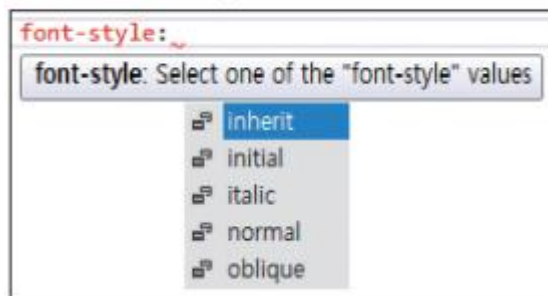
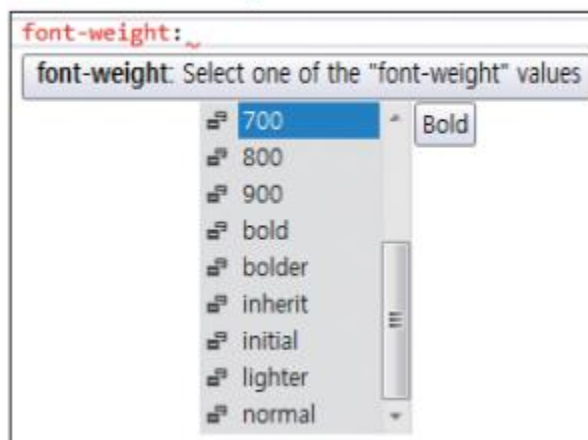


그림 4-67 font-weight 속성





## 4.6 폰트 속성

- 각각의 속성은 다음과 같은 방법으로 사용한다.

코드 4-48 font-style 속성과 font-weight 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Font Property</title>
  <style>
    .font_big { font-size: 2em }
    .font_italic { font-style: italic }
    .font_bold { font-weight: bold }
  </style>
</head>
<body>
  <p class="font_big font_italic font_bold">Lorem ipsum dolor amet</p>
</body>
</html>
```

그림 4-68 font-style 속성과 font-weight 속성

***Lorem ipsum dolor amet***

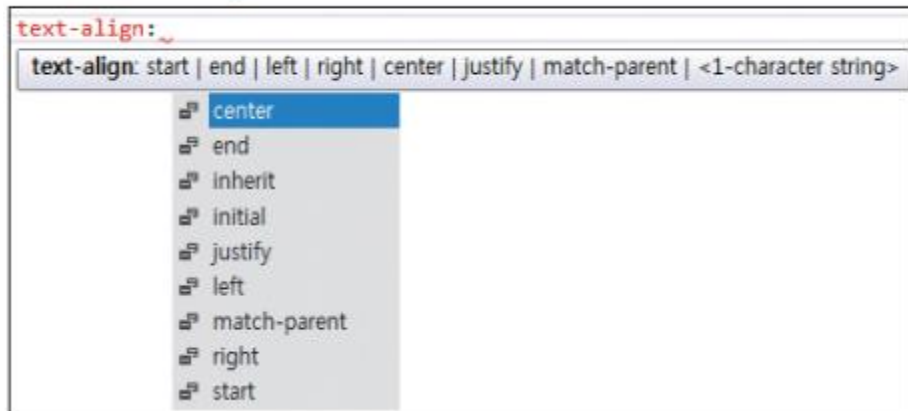


## 4.6 폰트 속성

### ❖ text-align 속성

- 글자의 정렬과 관련된 스타일 속성이다.
- 다음 키워드를 입력한다.

그림 4-72 text-align 속성에 적용 가능한 키워드



### ❖ text-decoration 속성

- a 태그에 href 속성이 지정되면 다음과 같이 밑줄이 표시된다.

그림 4-76 링크 표시

[Lorem ipsum dolor amet](#)

- 이를 제거하려면 다음과 같이 text-decoration 속성에 none 키워드를 지정한다.

코드 4-53 text-decoration 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Font Property</title>
  <style>
    a { text-decoration: none }
  </style>
</head>
<body>
  <h1>
    <a href="#">Lorem ipsum dolor amet</a>
  </h1>
</body>
</html>
```

그림 4-77 text-decoration 속성

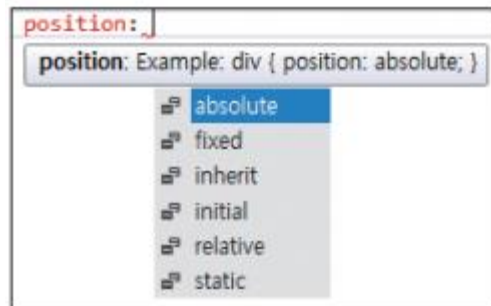
[Lorem ipsum dolor amet](#)



## ❖ position 속성

- 태그의 위치 설정 방법을 변경할 때 사용한다.
- 다음 키워드를 입력할 수 있다.

그림 4-78 position 속성에 적용할 수 있는 키워드



- 각각의 키워드는 다음과 같은 의미를 갖는다.

표 4-5 position 속성에 적용할 수 있는 키워드

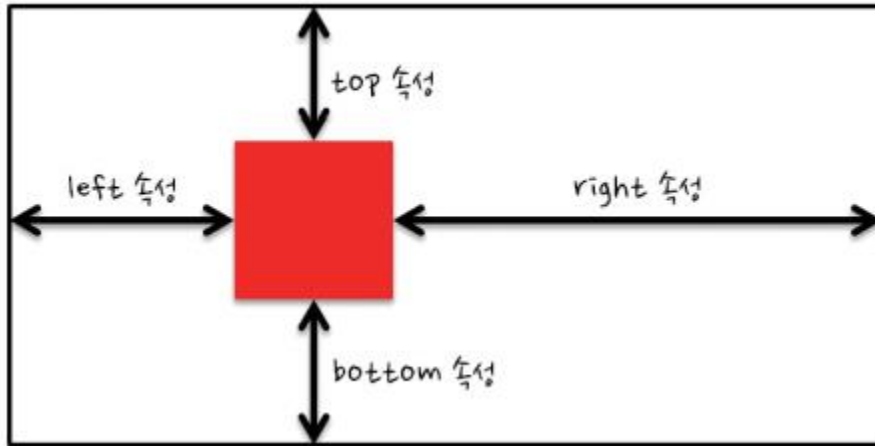
키워드	설명
static	태그가 위에서 아래로 순서대로 배치됩니다.
relative	초기 위치 상태에서 상하좌우로 위치를 이동합니다.
absolute	절대적 위치 좌표를 설정합니다.
fixed	화면을 기준으로 절대적 위치 좌표를 설정합니다.



## 4.7 위치 속성

- position 속성은 반드시 다음 스타일 속성과 함께 사용한다.

그림 4-80 position 속성과 함께 사용하는 스타일 속성



## 4.7 위치 속성

- 다음과 같은 방식으로 사용한다.

코드 4-55 position 속성과 함께 사용하는 스타일 속성

```
<style>
  .box {
    width: 100px; height: 100px;
    position: absolute;
  }
  .box:nth-child(1) {
    background-color: red;
    left: 10px; top: 10px;
  }
  .box:nth-child(2) {
    background-color: green;
    left: 50px; top: 50px;
  }
  .box:nth-child(3) {
    background-color: blue;
    left: 90px; top: 90px;
  }
</style>
```

그림 4-81 position 속성을 사용한 위치 지정



### ❖ z-index 속성

- HTML 태그는 아래 입력한 태그가 위로 올라온다.
- 이러한 순서를 변경할 때에 z-index 속성을 사용한다.
- 큰 값을 입력할 수록 위로 올라온다.
- 다음과 같은 방법으로 사용한다.

코드 4-56 z-index 속성

```
<style>
    .box {
        width: 100px; height: 100px;
        position: fixed;
    }
    .box:nth-child(1) {
        background-color: red;
        left: 10px; top: 10px;

        z-index: 100;
    }
    .box:nth-child(2) {
        background-color: green;
        left: 50px; top: 50px;

        z-index: 10;
    }

```



## 4.7 위치 속성

```
.box:nth-child(3) {  
    background-color: blue;  
    left: 90px; top: 90px;  
  
    z-index: 1;  
}  
</style>
```

- 코드를 실행하면 다음과 같이 출력한다.

그림 4-82 z-index 속성





## ❖ 위치 속성과 관련된 공식

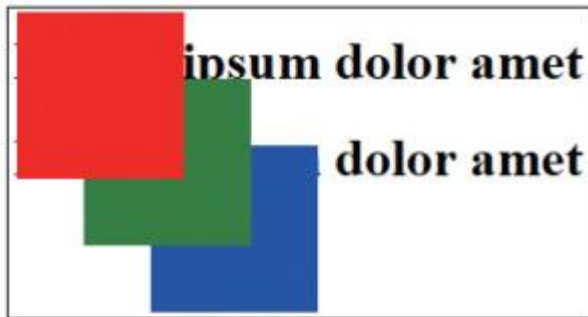
- 현재 만들고 있는 예제에서 다음과 같은 코드의 실행 결과를 예측해보자

코드 4-57 body 태그 구성

```
<body>
  <h1>Lorem ipsum dolor amet</h1>
  <div>
    <div class="box"></div>
    <div class="box"></div>
    <div class="box"></div>
  </div>
  <h1>Lorem ipsum dolor amet</h1>
</body>
```

- 코드를 실행하면 다음과 같이 출력한다.

그림 4-83 실행 결과



## 4.7 위치 속성

- 다음과 같은 문제를 발견할 수 있다.
  - h1 태그 두 개가 붙어 있다(div 태그가 영역을 차지하지 않는다).
  - 색상이 적용된 상자가 자신의 부모를 기준으로 위치를 잡지 않는다.
- 이를 해결할 때는 다음과 같은 공식을 사용한다.
  - 자손에게 position 속성을 absolute 키워드로 적용하면 부모에게 height 속성을 입력한다.
  - 자손의 position 속성을 absolute 키워드로 적용하면 부모의 position 속성을 relative 키워드로 적용한다.
- 이를 사용하면 다음과 같이 코드를 적용할 수 있다.

코드 4-59 position 속성에 relative 키워드 적용

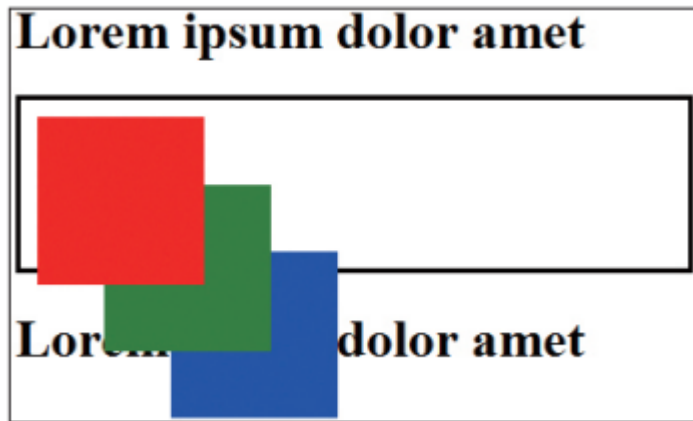
```
body > div {  
    width: 400px; height: 100px;  
    border: 3px solid black;  
  
    position: relative;  
}
```



## 4.7 위치 속성

- 코드를 실행하면 다음 그림처럼 출력한다.

그림 4-84 position 속성에 relative 키워드 적용



### ❖ overflow 속성

- 내부의 요소가 부모의 범위를 벗어날 때 어떻게 처리할지 지정하는 속성이다.
- 다음과 같은 키워드를 입력한다.

표 4-6 overflow 속성에 적용 가능한 키워드

스타일 이름	설명
hidden	영역을 벗어나는 부분을 보이지 않게 만듭니다.
scroll	영역을 벗어나는 부분을 스크롤로 만듭니다.



## 4.7 위치 속성

- 다음과 같은 방식으로 사용한다.

코드 4-60 overflow 속성에 hidden 키워드 적용

```
body > div {  
    width: 400px; height: 100px;  
    border: 3px solid black;  
  
    position: relative;  
    overflow: hidden;  
}
```

- 각각의 키워드를 적용할 때에 다음과 같이 실행된다.

그림 4-85 overflow 속성에 hidden 키워드 적용

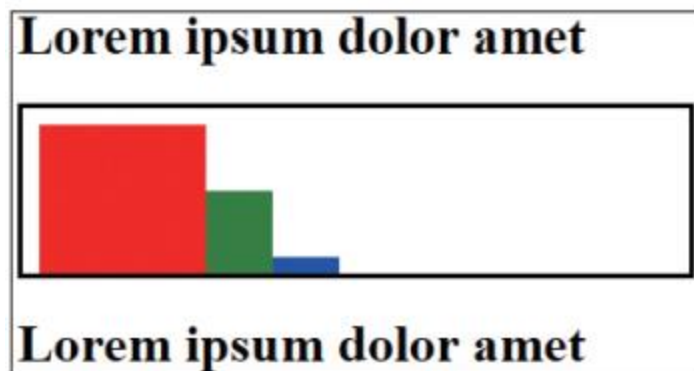
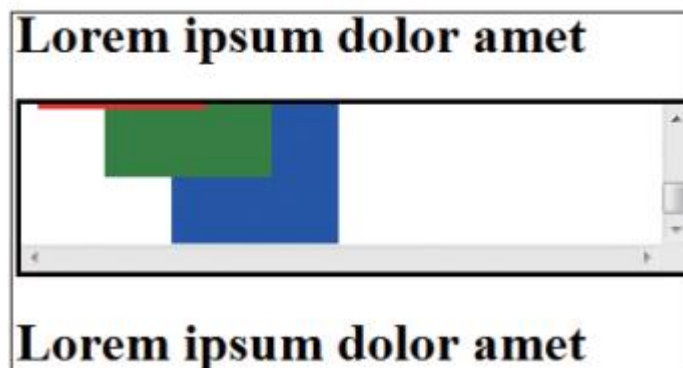


그림 4-86 overflow 속성에 scroll 키워드 적용



### ❖ float 속성 개요

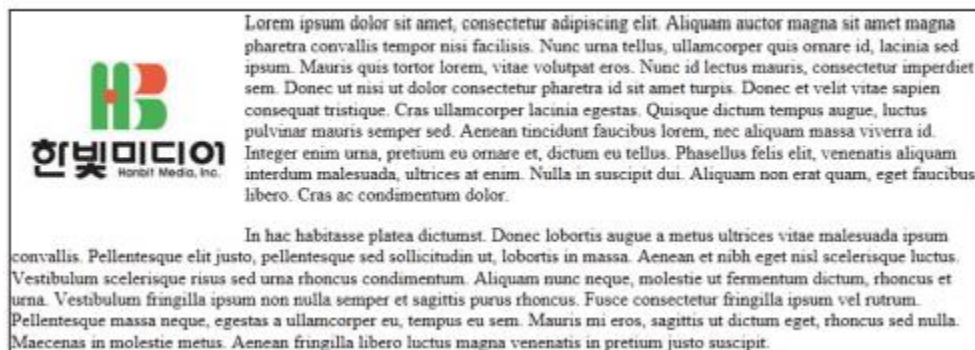
- 웹 페이지 레이아웃을 구성할 때에 반드시 사용하는 속성
- 부유하는 대상을 만들 때에 사용하는 속성이다.
- 다음 키워드를 입력한다.

표 4-7 float 속성에 적용 가능한 키워드

키워드	설명
left	태그를 왼쪽에 붙입니다.
right	태그를 오른쪽에 붙입니다.

- float 속성을 사용하면 다음과 같이 이미지가 글자 위를 부유하게 만들 수 있다.

그림 4-89 float 속성



## 4.8 float 속성

### ❖ float 속성을 사용한 수평 정렬

- 동위 위치에 있는 태그에 모두 float 속성을 적용하면 수평 정렬된다.
- 다음 코드에서 div.box 태그는 동위 위치에 있다.

코드 4-65 body 태그 구성

```
<body>
  <div class="box">1</div>
  <div class="box">2</div>
</body>
```

- 이러한 태그에 다음과 같은 스타일을 적용한다.

코드 4-66 float 속성에 적용 가능한 키워드

```
<style>
  .box {
    width: 100px; height: 100px;
    background-color: red;
    margin: 10px; padding: 10px;

    /* 태그를 왼쪽으로 붙입니다. */
    float: left;
  }
</style>
```

```
<style>
  .box {
    width: 100px; height: 100px;
    background-color: red;
    margin: 10px; padding: 10px;

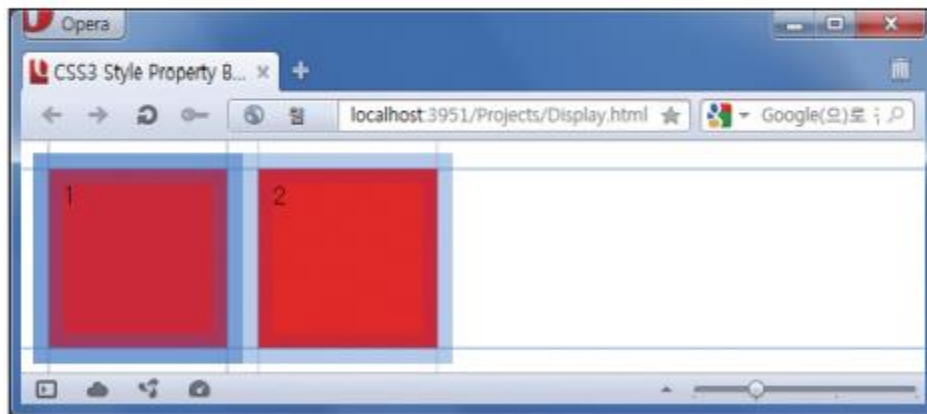
    /* 태그를 오른쪽으로 붙입니다. */
    float: right;
  }
</style>
```



## 4.8 float 속성

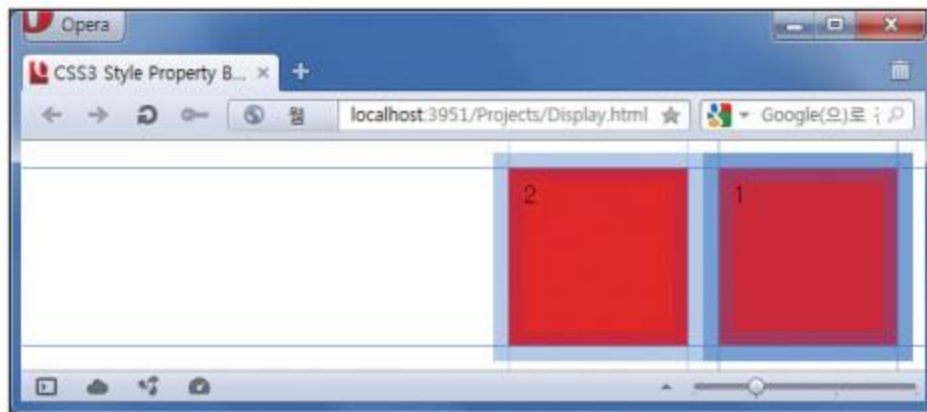
- 코드를 실행하면 다음과 같이 수평 정렬된다.

그림 4-90 float 속성에 left 키워드 적용



- right 키워드를 적용했을 경우에는 위에 위치한 태그가 오른쪽에 붙는다는 것을 주의한다.

그림 4-91 float 속성에 right 키워드 적용



### ❖ text-shadow 속성

- 글자에 그림자를 부여하는 속성이다.
- 다음과 같은 형태로 값을 입력한다.

그림 4-97 text-shadow 속성

text-shadow: 5px 5px 5px black

오른쪽 아래 흐림도 색상

- 다음과 같이 사용한다.

```
<style>
  h1 {
    text-shadow: 5px 5px 5px black;
  }
</style>
```

그림 4-98 text-shadow 속성 적용

Lorem ipsum dolor amet





### ❖ box-shadow 속성

- 박스에 그림자를 부여하는 속성이다.
- 다음과 같은 형태로 값을 입력한다.

그림 4-99 box-shadow 속성

box-shadow: 5px 5px 5px black

오른쪽    아래    흐림도    색상

- 다음과 같이 사용한다.

```
<style>
  div {
    border: 3px solid black;
    box-shadow: 10px 10px 30px black;
    text-shadow: 5px 5px 5px black;
  }
</style>
```

그림 4-100 box-shadow 속성 적용



### ❖ CSS3 Generator

- text-shadow 속성과 box-shadow 속성은 CSS3 Generator를 사용해 쉽게 만들 수 있다.

그림 4-102 CSS3 Generator(<http://css3generator.com/>)



### ❖ 개요

- 벤더 프리픽스는 웹 브라우저 공급 업체(마이크로소프트, 구글, 모질라, 애플, 오페라)에서 제공하는 실험적인 기능을 사용할 때 사용한다.
- 예를 들어 변환 속성은 CSS3 표준에 있지만 아직 완벽하게 제정된 상태가 아니다. 따라서 웹 브라우저 업체가 무턱대고 속성을 추가할 수 없다.
- 하지만 다른 웹 브라우저를 이기려면 새로운 기능을 모두 제공해야 하므로 벤더 프리픽스를 사용해 지원한다.
- 각 웹 브라우저마다 다음과 같은 벤더 프리픽스를 갖는다.

그림 4-103 벤더 프리픽스



## 4.10 벤더 프리픽스

### ❖ 다음과 같은 방식으로 벤더 프리픽스를 사용한다.

코드 4-75 벤더 프리픽스

```
<style>
  input[type=checkbox]:checked + div {
    height: 0px;
  }

  div {
    overflow: hidden;
    width: 650px; height: 300px;

    /* 변환 효과를 적용합니다. */
    -ms-transition-duration: 1s;
    -webkit-transition-duration: 1s;
    -moz-transition-duration: 1s;
    -o-transition-duration: 1s;
    transition-duration: 1s;
  }
</style>
```



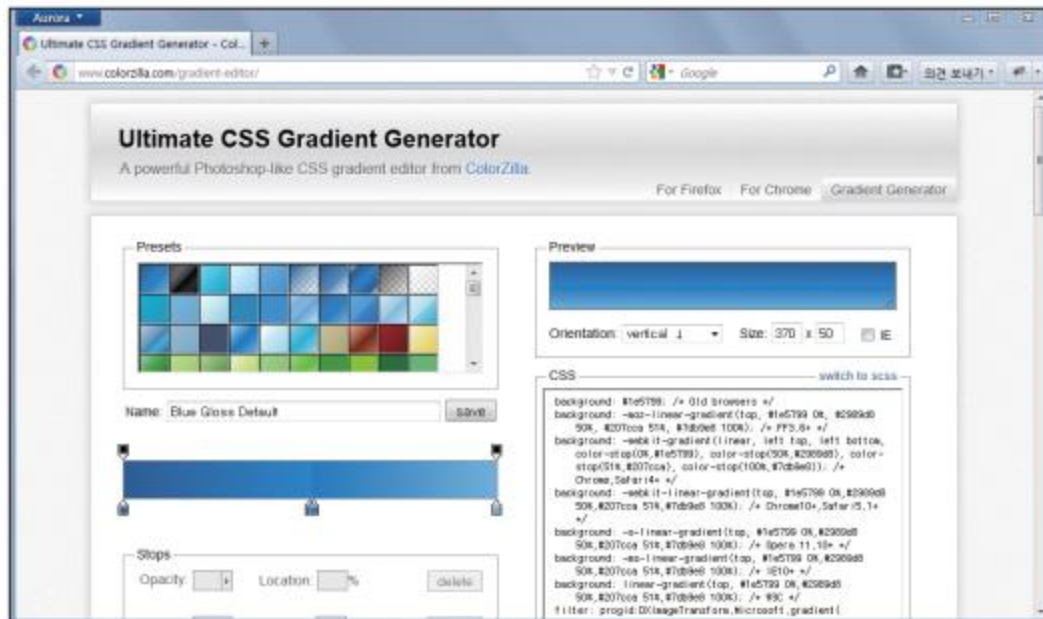
## ❖ 개요

- 그레이디언트는 2가지 이상의 색상을 혼합한 색을 만들어 채색하는 기능이다.

## ❖ Ultimate CSS Gradient Generator

- CSS3 그레이디언트를 손쉽게 생성할 수 있다.

그림 4-109 Ultimate CSS Gradient Generator



## ❖ linear-gradient() 함수

- 선형 그레이디언트를 만들 때에 사용한다.
- 다음과 같은 형태로 사용한다.

그림 4-111 linear-gradient() 함수(1)

linear-gradient(<sup>각도</sup>45deg, <sup>색상 블록</sup>#f85032 0%, #e73827 100%)  
<sub>색상 위치</sub>

- 위의 그레이디언트 함수는 다음과 같은 그레이디언트를 생성한다.

그림 4-112 linear-gradient() 함수(2)





# Thank You !

모던 웹 디자인을 위한 HTML5 + CSS3 입문