

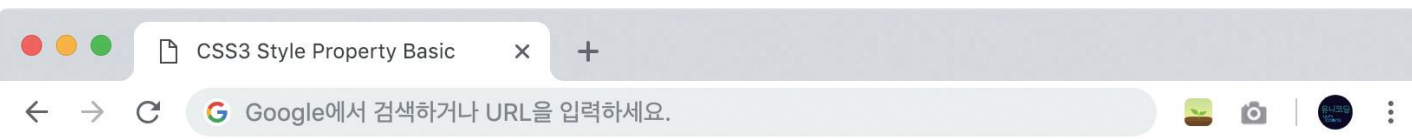
## 4.2 CSS3 단위 / 크기 단위

### 좌측의 코드를 구현한 사례

```
<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>
```

```
<style>
  p:nth-child(1) { }
  p:nth-child(2) { font-size: 1.0em; }
  p:nth-child(3) { font-size: 1.5em; }
  p:nth-child(4) { font-size: 2.0em; }
</style>
```

```
<style>
  p:nth-child(1) { }
  p:nth-child(2) { font-size: 16px; }
  p:nth-child(3) { font-size: 24px; }
  p:nth-child(4) { font-size: 32px; }
</style>
```



Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

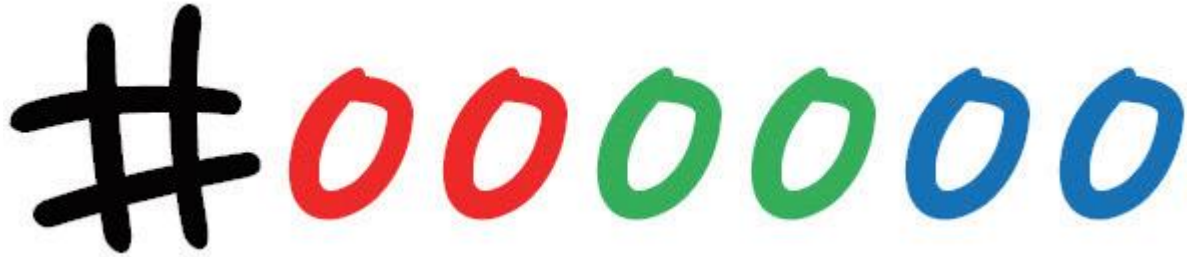
Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

단위	설명
%	백분율 단위
em	배수 단위
px	픽셀 단위

## 4.2 CSS3 단위 / 색상 단위

```
<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>
```



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

## 4.2 CSS3 단위 / URL 단위

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Property Basic</title>
  <style>

  </style>
</head>
<body>
  <h1>Lorem ipsum dolor amet.</h1>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
</body>
</html>

<style>
  body {
    background-image: url('Desert.jpg');
  }
</style>

/* 현재 폴더의 Desert.jpg */
background-image: url('Desert.jpg');

/* 현재 폴더 내부에 있는 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');
```

HTML 페이지 구성

background-image 속성에 URL 단위를 입력

url() 함수 내부에 경로를 입력

## 4.3 가시 속성 / display 속성

```
<style>
  #box {
    display: none;
  }
</style>
```

display 속성에 none 키워드 적용

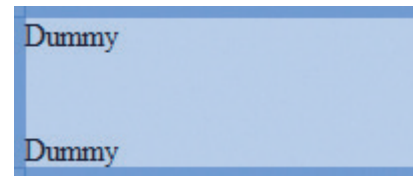
```
<style>
  #box {
    display: block;
  }
</style>
```

display 속성에 block 키워드 적용

단위 형태	설명
none	태그를 화면에서 보이지 않게 만듭니다.
block	태그를 block 형식으로 지정합니다.
inline	태그를 inline 형식으로 지정합니다.
inline-block	태그를 inline-block 형식으로 지정합니다.

## 4.3 가시 속성 / visibility 속성

```
<style>
  #box {
    visibility: hidden;
  }
</style>
```



**display** 속성을 사용하면 태그가 화면에서 제거되지만 **visibility** 속성을 사용하면 화면에서 보이지 않게 됨

단위 형태	설명
none	태그를 화면에서 보이지 않게 만듭니다.
block	태그를 block 형식으로 지정합니다.
inline	태그를 inline 형식으로 지정합니다.
inline-block	태그를 inline-block 형식으로 지정합니다.

## 4.3 가시 속성 / opacity 속성

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

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

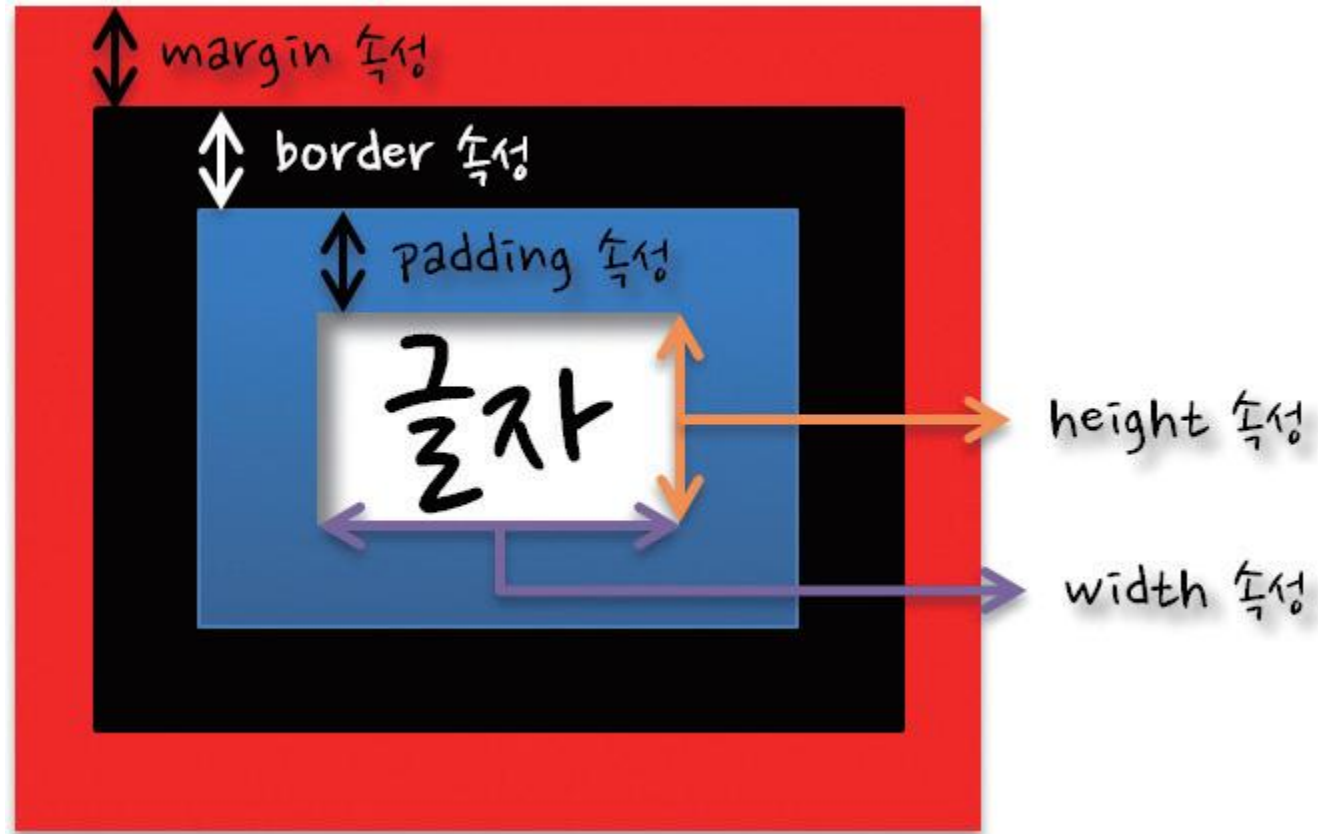
Dummy

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit.

Dummy

- opacity 속성은 태그의 투명도를 조절하는 스타일 속성
- 0.0부터 1.0 사이의 숫자를 입력할 수 있으며 0.0은 투명한 상태를 나타내고 1.0은 불투명한 상태

## 4.4 박스 속성



박스 속성 사례

## 4.4 박스 속성 / 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>
```



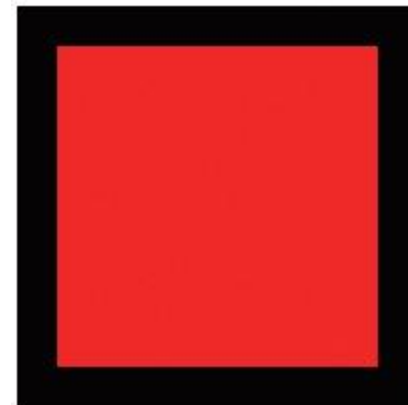
- width 속성: 박스의 너비
- height 속성: 박스의 높이
- div 태그의 너비와 높이가 100픽셀로 지정



## 4.4 박스 속성 / 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>
```



- margin 속성에 10픽셀 적용
- padding 속성에 30픽셀 적용

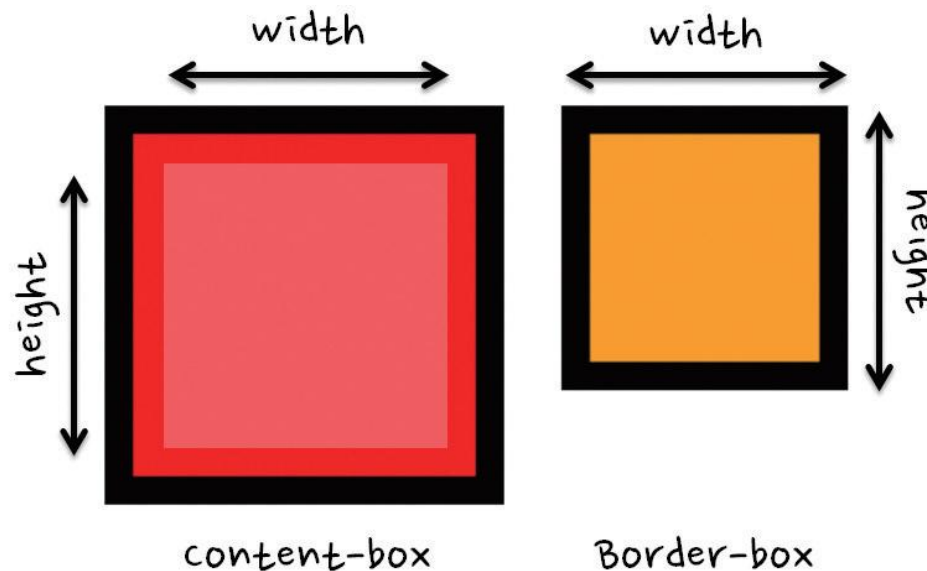
## 4.4 박스 속성 / box-sizing 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Property Basic</title>
  <style>
    div {
      margin: 10px; padding: 10px;
      width: 100px; height: 100px;
      border: 10px solid black;
    }

    div:first-child {
      background: red;
      box-sizing: content-box;
    }

    div:last-child {
      background: orange;
      box-sizing: border-box;
    }
  </style>
</head>

<body>
  <div></div>
  <div></div>
</body>
</html>
```



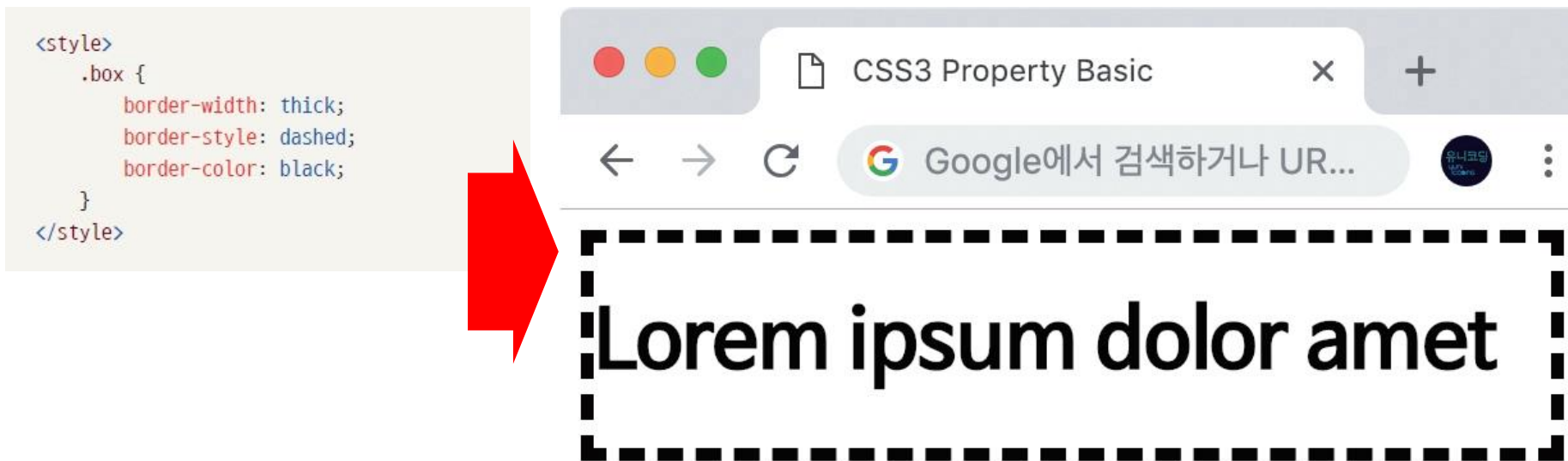
### Content-box

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

### Border-box

- 박스 너비 = width 속성 + 2×margin 속성
- 박스 높이 = height 속성 + 2×margin 속성

## 4.5 테두리 속성 / border-width 속성과 border-style 속성



- borderwidth 속성: 테두리의 너비를 지정하는 스타일 속성 / 크기 단위의 키워드 사용
- border-style 속성: 테두리의 형태를 지정하는 속성
- 좌측의 코드 출력: 두꺼운 dashed 형태의 검정색 테두리가 형성됨

## 4.5 테두리 속성 / border-radius 속성

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



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

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

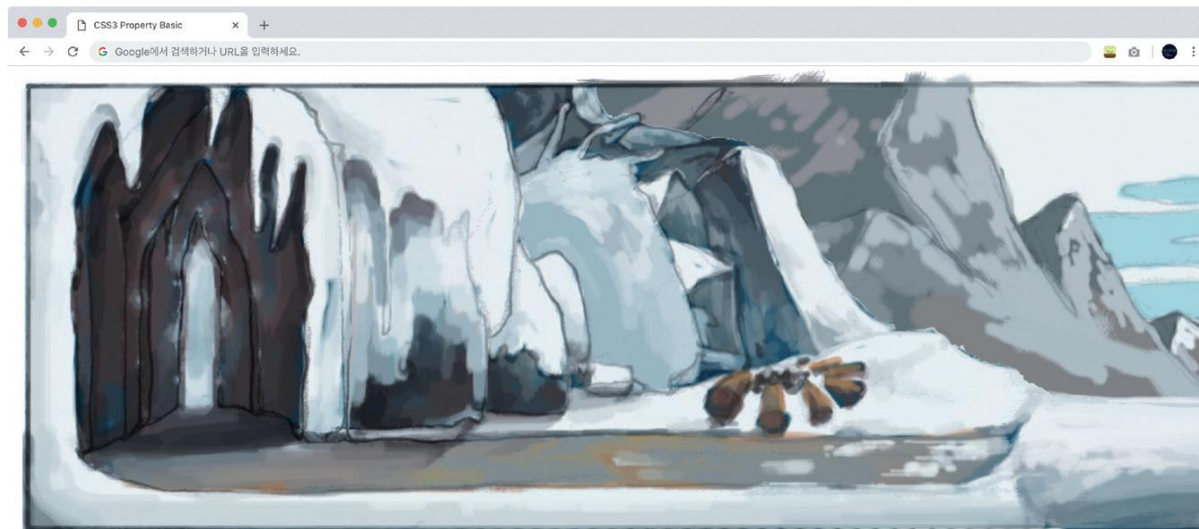


- border-radius 속성을 사용하면 테두리가 둥근 사각형 또는 원을 만들 수 있음
- 좌측 코드의 출력: 4개의 크기 단위를 각각 적용할 수 있음

## 4.6 배경 속성 / background-image 속성

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

</body>
</html>
```



- CSS3 이전에는 이렇게 배경 이미지를 1개씩만 적용할 수 있었음
- 그러나 CSS3부터는 여러 개의 배경 이미지 적용이 가능해짐
- 여러 개의 배경을 적용할 경우에는 왼쪽에 위치한 이미지가 앞으로 나오게 됨
- 코드를 실행하면 2개의 이미지가 층을 이루어 출력됨

## 4.6 배경 속성 / background-size 속성

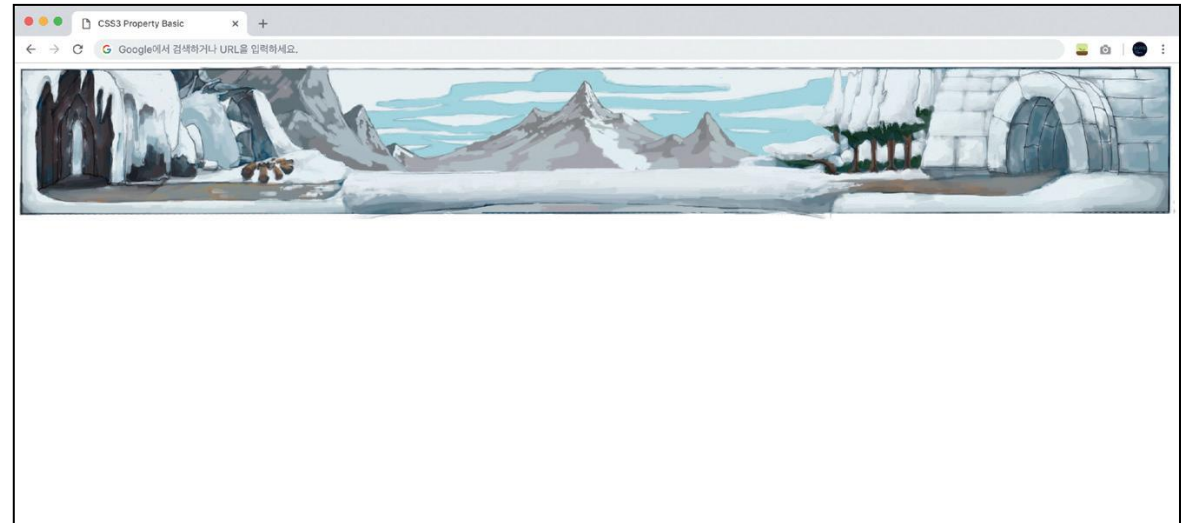
```
<style>
  body {
    background-image: url('BackgroundFront.png'), url('BackgroundBack.png');
    background-size: 100% 250px;
  }
</style>
```



- 그림 크기를 조절할 때는 background-size 속성을 사용: 인터넷 익스플로러 8 이하에선 사용 불가
- background-size 속성은 1개 또는 2개의 크기 단위를 적용하며 각각 너비와 높이를 의미
- 좌측 코드의 출력: 높이에 250픽셀 적용
- 쉼표를 사용해 크기 단위를 적용하면 BackgroundFront.png 파일은 너비가 100%로 적용

## 4.6 배경 속성 / background-repeat 속성

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

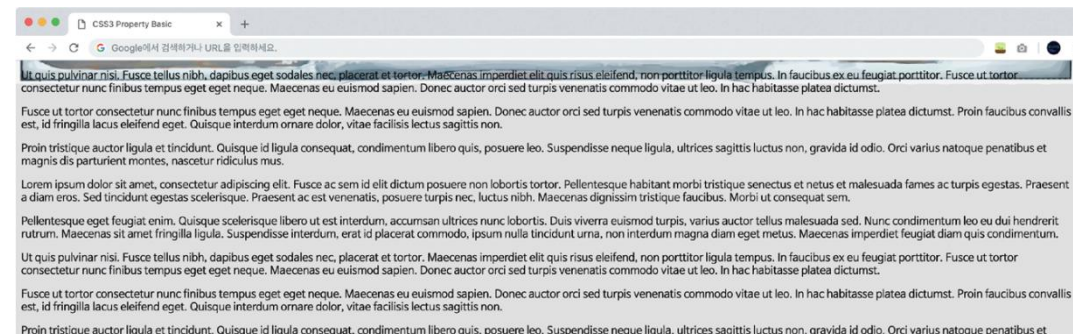
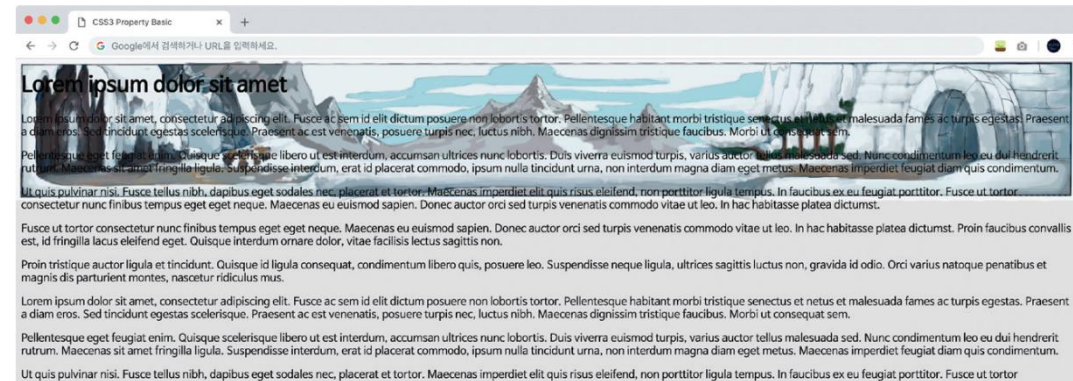


- background-repeat 속성의 기본 키워드가 repeat: 그림이 패턴을 이루어 여러 개 출력
- repeat 키워드를 적용하면 이미지가 패턴을 이루게 됨
- repeat-x 키워드를 적용하면 X축 방향으로 이미지가 반복
- repeat-y 키워드를 적용하면 Y축 방향으로 이미지가 반복



## 4.6 배경 속성 / background-attachment 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Property Basic</title>
  <style>
    body {
      background-color: #E7E7E8;
      background-image: url('BackgroundFront.png'), url('BackgroundBack.png');
      background-size: 100%;
      background-repeat: no-repeat;
    }
  </style>
</head>
<body>
  <h1>Lorem ipsum dolor sit amet</h1>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  <p>Pellentesque est velit, laoreet vel rhoncus sit amet.</p>
  <p>Proin vitae elit est, ut accumsan arcu. Sed consectetur.</p>
  <p>Donec molestie massa id lorem hendrerit eu bibendum augue vestibulum.</p>
  <p>Morbi ut lorem in purus facilisis vulputate. Sed purus nibh.</p>
</body>
</html>
```

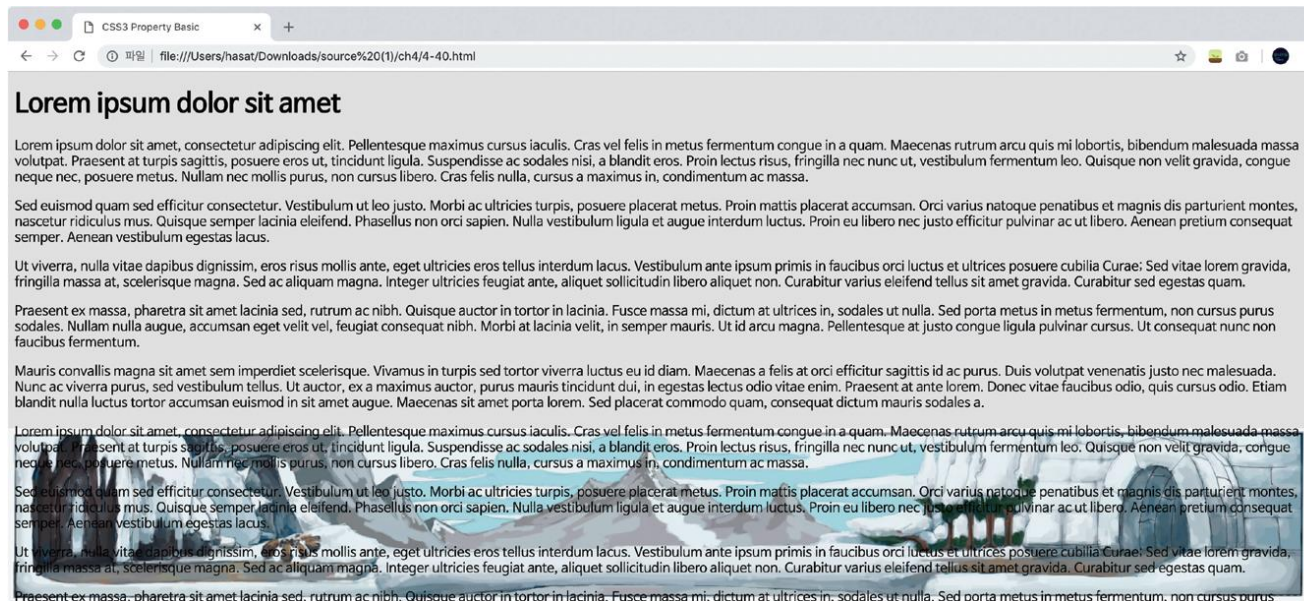


- background-attachment 속성: 배경 이미지를 어떠한 방식으로 화면에 붙일 것인지를 지정하는 스타일 속성
- background-attachment 속성의 기본 키워드는 scroll 키워드: 화면 스크롤에 따라 배경 이미지가 함께 이동한다는 뜻
- background-attachment 속성에 fixed 키워드를 적용 시 스크롤을 내려도 배경 이미지는 고정됨



## 4.6 배경 속성 / background-position 속성

```
<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>
```



- background-position: 키워드;
- background-position: X축크기;
- background-position: X축크기 Y축크기;
- 2개의 값을 입력하면 각각 X축 위치와 Y축 위치를 적용

## 4.7 폰트 속성 / 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>
```



Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

Lorem ipsum

- font-size 속성은 글자의 크기를 지정하는 스타일 속성
- 참고: h1 태그의 기본 크기는 32픽셀 / p 태그의 기본 크기는 16픽셀

## 4.7 폰트 속성 / font-size 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>
```



- font-family 속성은 폰트를 지정하는 스타일 속성
- 일반적으로 한 단어로 이루어진 폰트는 따옴표를 사용하지 않음
- 반두 단어 이상으로 이루어지는 단어는 따옴표를 반드시 사용해야 함
- 개발자의 컴퓨터에는 설치되어 있으나, 웹 페이지를 사용할 사용자에게는 폰트가 설치되어 있지 않을 수 있음
- 이러한 문제를 예방하고자 font-family 속성을 여러 개 사용
- 다국어 웹 페이지의 경우 font-family 속성의 가장 마지막 폰트에는 Serif 폰트(명조체), Sans-serif 폰트(고딕체), Mono space 폰트(고정폭 글꼴)를 적용

## 4.7 폰트 속성 / 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>
```



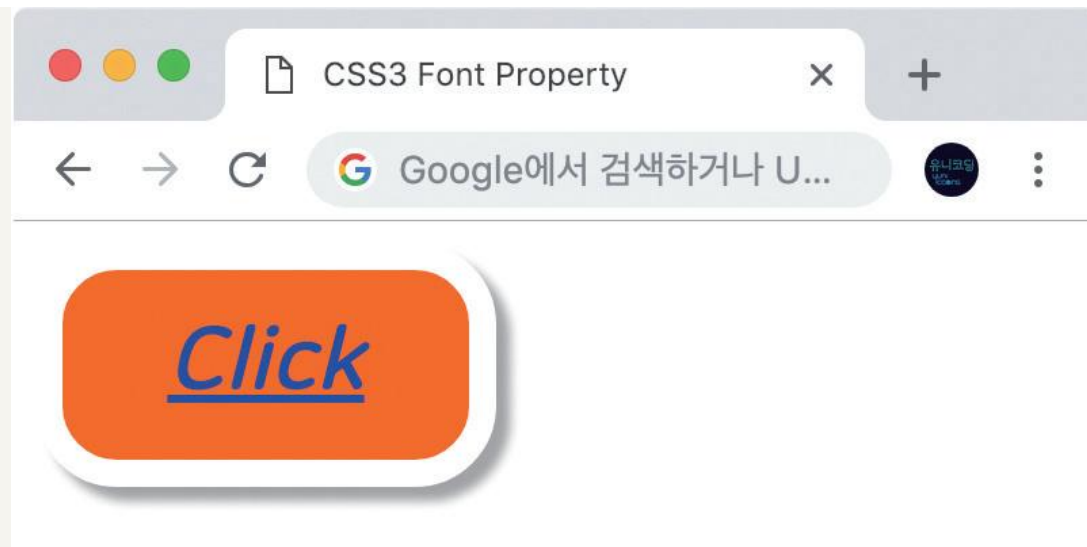
- font-style 속성과 font-weight 속성은 폰트의 기울기 또는 두께를 조정함
- 일반 폰트의 두께는 400 / 두꺼운 폰트의 두께는 700

## 4.7 폰트 속성 / line-height 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; }
    .font_center { text-align: center; }

    .button {
      width: 150px;
      height: 70px;
      background-color: #FF6A00;
      border: 10px solid #FFFFFF;
      border-radius: 30px;
      box-shadow: 5px 5px 5px #A9A9A9;
    }

    .button > a {
      display: block;
    }
  </style>
</head>
<body>
  <div class="button">
    <a href="#" class="font_big font_italic font_bold font_center">Click</a>
  </div>
</body>
</html>
```

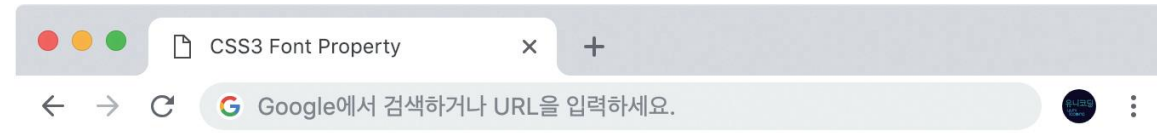


- line-height 속성은 글자의 높이를 지정
- CSS는 block 형식을 가지는 태그를 수직 정렬할 수 있는 스타일 속성이 없음
- 대체 방안으로 line-height 속성을 사용
- line-height 속성에 글자를 감싸는 박스의 높이와 같은 크기인 70픽셀을 적용하여 출력한 사례가 우측 상단의 사례



## 4.7 폰트 속성 / text-align 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; }
    .font_center { text-align: center; }
    .font_right { text-align: right; }
  </style>
</head>
<body>
  <p class="font_big font_italic font_bold font_center">Lorem ipsum dolor amet</p>
  <p class="font_bold font_right">2019.02.14</p>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
</body>
</html>
```



*Lorem ipsum dolor amet*

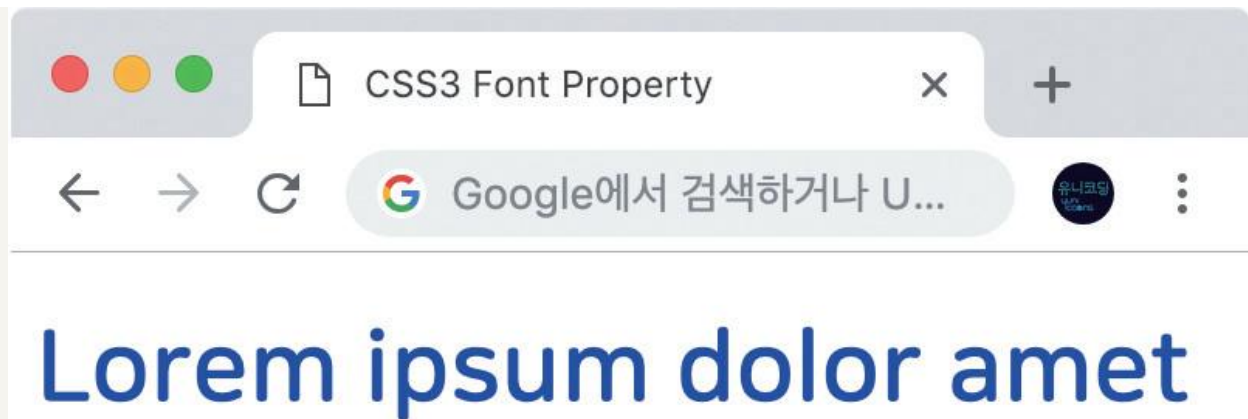
2019.02.14

Suspendisse vitae nunc dolor. Vivamus convallis sem tincidunt urna hendrerit mollis. Aliquam tincidunt velit ut tortor dapibus, vel iaculis tellus aliquam. Integer semper arcu eget lectus tincidunt maximus. In hac habitasse platea dictumst. Sed fringilla nulla auctor, vestibulum augue ac, ultrices metus. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Mauris imperdiet ullamcorper ipsum, et auctor turpis condimentum non. Maecenas id volutpat justo. Vestibulum porta eu erat ut ullamcorper. Integer malesuada pulvinar tempus. Donec nec scelerisque magna. Integer sit amet ligula quam.

- text-align 속성: 글자의 정렬과 관련된 속성

## 4.7 폰트 속성 / 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>
```



- text-decoration 속성: a 태그에 href 속성을 사용하면 글자에 밑줄이 생기고 글자의 색상이 파란색으로 변경됨
- 하지만 일반적인 웹 페이지에서 링크에 밑줄이 없는 이유는 text-decoration 속성으로 밑줄을 제거했기 때문
- 해당 속성으로는 밑줄만 제거되며 색상은 color 속성을 사용해 별도로 적용해야 함

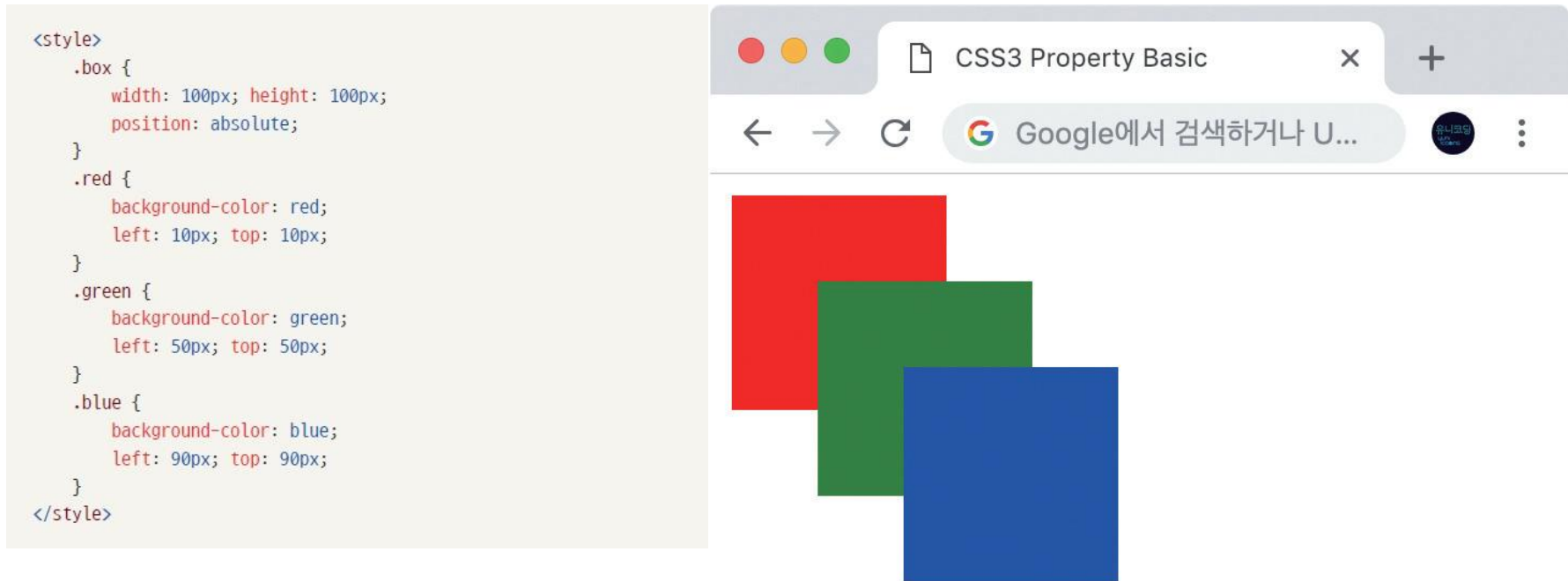
## 4.8 위치 속성 / position 속성

- 절대 위치 좌표: 요소의 X 좌표와 Y 좌표를 설정해 절대 위치를 지정
- 상대 위치 좌표: 요소를 입력한 순서를 통해 상대적으로 위치를 지정
- 상대 위치 좌표를 사용할 때는 position 속성에 static 키워드 또는 relative 키워드를 적용
- static 키워드 적용: 태그가 "위에서 아래로"와 "왼쪽에서 오른쪽으로" 순서에 맞게 배치
- relative 키워드 적용: static 키워드로 초기 위치가 지정된 상태에서 상하좌우로 이동 가능

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



## 4.8 위치 속성 / position 속성



- 빨간색 상자가 가장 아래에 있고 파란색 상자가 가장 위에 있게 됨
- HTML 페이지의 뒤에 입력한 태그가 상위에 올라가게 됨

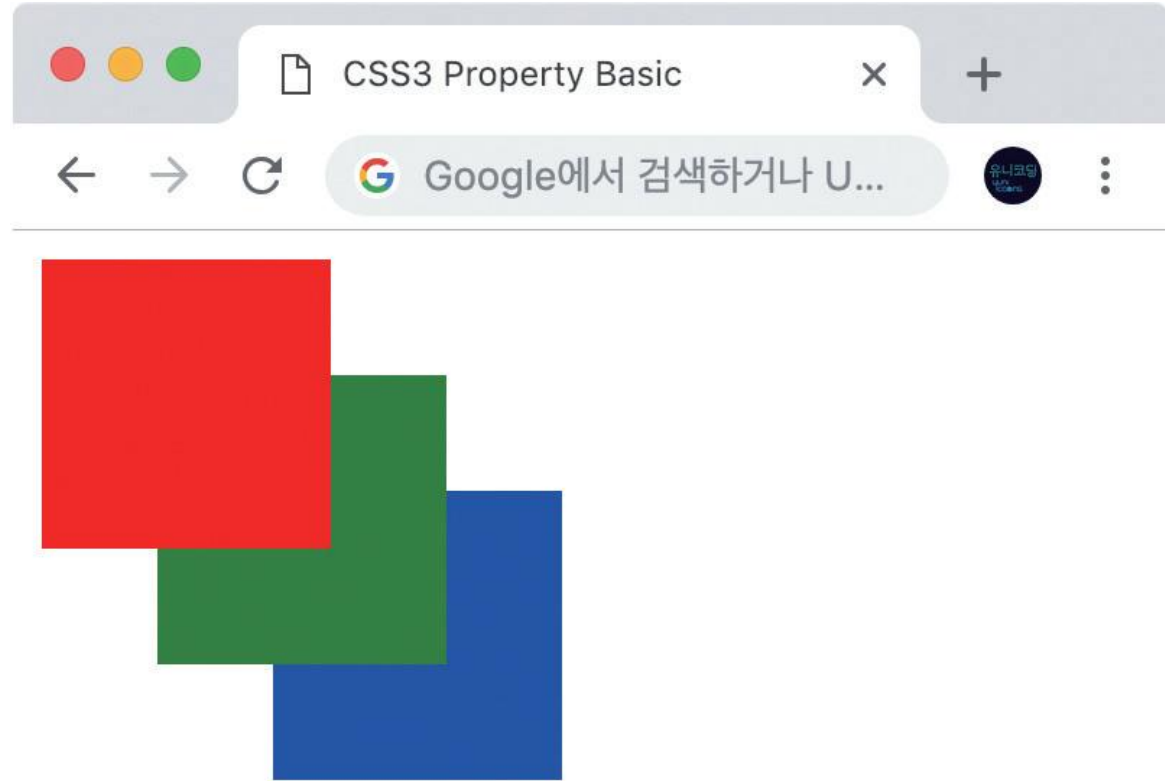
## 4.8 위치 속성 / z-index 속성

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

    z-index: 1;
  }
</style>
```



- 객체 순서를 변경하고 싶을 때 z-index 속성을 사용함
- z-index 속성에는 숫자를 적용하며 숫자가 클수록 앞에 위치하게 됨

## 4.8 위치 속성 / 위치 속성과 관련된 공식



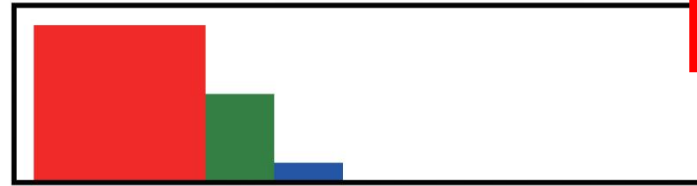
- 좌측 코드 실행 시의 문제 1) h1 태그 2개가 붙어 있음(div 태그가 영역을 차지하지 않음)
- 좌측 코드 실행 시의 문제 2) 색상이 적용된 상자가 자신의 부모를 기준으로 위치를 잡지 않음
- 1)번 문제 해결: 자손의 **position** 속성에 **absolute** 키워드를 적용하면 부모는 **height** 속성을 사용해야 함
- 2)번 문제 해결: 자손의 **position** 속성에 **absolute** 키워드를 적용하면 부모의 **position** 속성에 **relative** 키워드 적용

## 4.8 위치 속성 / overflow 속성

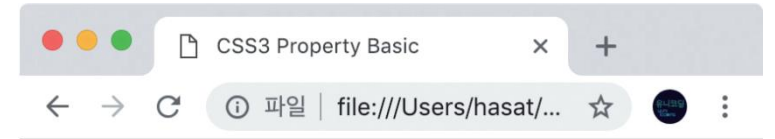
```
body > div {  
  width: 400px; height: 100px;  
  border: 3px solid black;  
  
  position: relative;  
  overflow: hidden;  
}
```



Lorem ipsum dolor amet



Lorem ipsum dolor amet



Lorem ipsum dolor amet



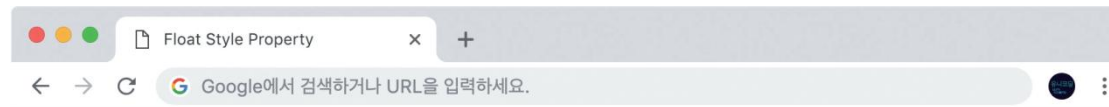
Lorem ipsum dolor amet

- hidden: 영역을 벗어나는 부분을 보이지 않게 만듦
- scroll: 영역을 벗어나는 부분을 스크롤로 만듦
- 만약 특정한 방향으로만 스크롤을 생성할 때는 overflow-x 속성과 overflow-y 속성을 사용

## 4.9 float 속성 / float 속성 개요

```
<!DOCTYPE html>
<html>
<head>
  <title>Float Style Property</title>
  <style>

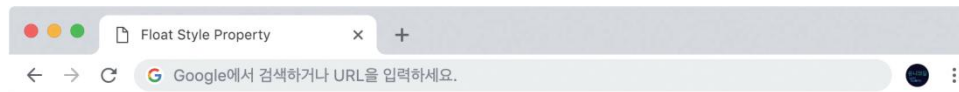
  </style>
</head>
<body>
  
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  <p>In hac habitasse platea dictumst. Donec lobortis augue a metus.</p>
</body>
</html>
```



img 태그는 inline 형식의 태그이고  
p 태그는 block 형식의 태그이므로  
그림과 글자를 분리하여 출력

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque justo ligula, convallis eu dignissim id, aliquet nec orci. Pellentesque ultricies erat sit amet felis hendrerit, ullamcorper rhoncus elit convallis. Maecenas ac dapibus est. Aenean tristique euismod libero, vitae vehicula odio. Proin ut arcu lacinia, fermentum nulla vel, placerat turpis. Phasellus vel dui eu dui sollicitudin hendrerit. Aenean eleifend magna lectus, accumsan tristique enim ultrices non. Vivamus quis consectetur metus, in tincidunt lorem. Morbi eget massa massa. Phasellus non sapien volutpat, vehicula nibh ac, efficitur mi. Fusce tempus metus non metus lacinia molestie. Donec porta felis fermentum semper malesuada. Praesent luctus rhoncus mi. Vivamus volutpat iaculis molestie. Aliquam rhoncus urna ut ullamcorper ultrices.

```
<style>
  img {
    float: left;
  }
</style>
```



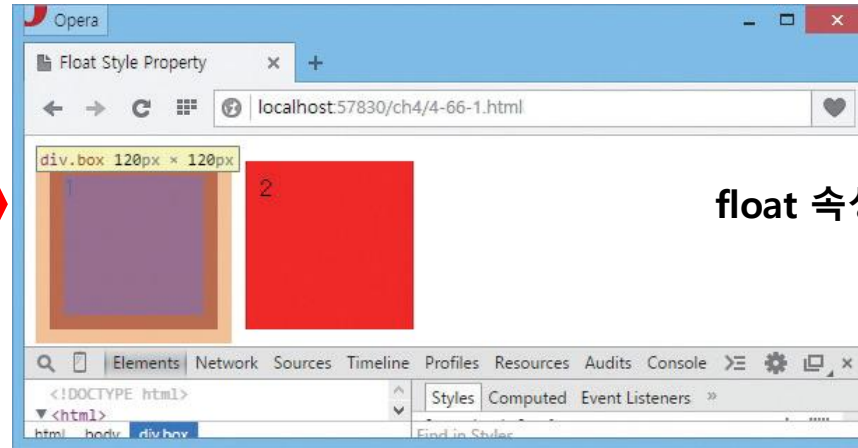
이미지가 글자 위에  
부유하게 됨

Curabitur odio ipsum, finibus non nulla in, efficitur aliquet elit. Vestibulum rutrum, sem nec dignissim molestie, enim sapien interdum justo, at dapibus elit nunc quis eros. In faucibus nibh ligula, placerat tincidunt erat dignissim vel. Vestibulum eu ornare nisi. Phasellus porta elit at imperdiet suscipit. Aliquam non malesuada massa, quis dictum quam. Nulla ultrices viverra tortor, eu porttitor ligula bibendum in. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque pretium nulla sit amet lacus vehicula luctus.

## 4.9 float 속성 / float 속성을 사용한 수평 정렬

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

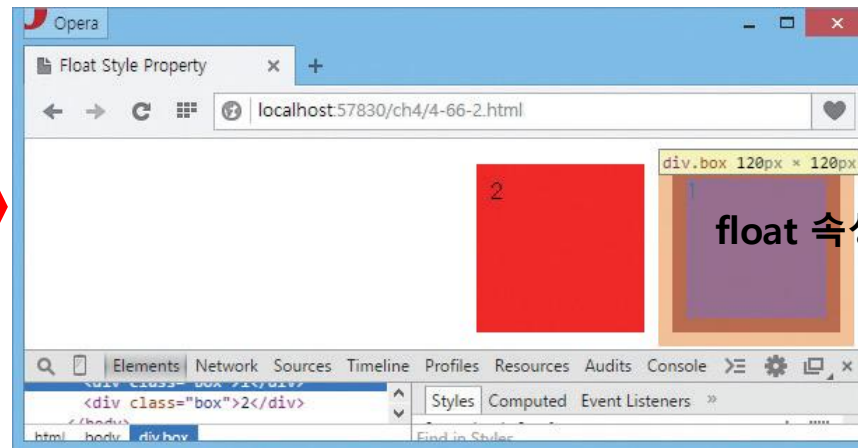
    /* 태그를 왼쪽으로 붙입니다. */
    float: left;
  }
</style>
```



float 속성에 left 키워드 적용한 결과

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

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



float 속성에 right 키워드 적용한 결과



## 4.9 float 속성 / float 속성을 사용한 레이아웃 구성

(1)

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Property Basic</title>
  <style>

</style>
</head>
<body>
  <div id="header"></div>
  <div id="navigation"></div>
  <div id="wrap">
    <div id="aside"></div>
    <div id="section"></div>
  </div>
  <div id="footer"></div>
</body>
</html>
```

(3)

```
<style>
/* body 태그를 중앙 정렬합니다. */
body {
  width: 960px;
  margin: 0 auto;
}
</style>
```

(2)

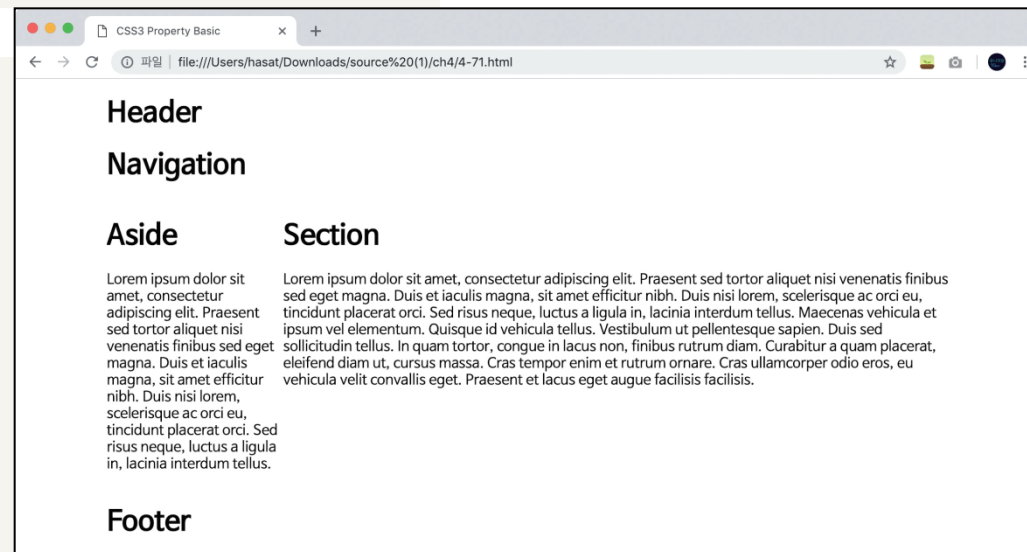
```
<body>
  <div id="header"><h1>Header</h1></div>
  <div id="navigation"><h1>Navigation</h1></div>
  <div id="wrap">
    <div id="aside">
      <h1>Aside</h1>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
    </div>
    <div id="section">
      <h1>Section</h1>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
    </div>
  </div>
  <div id="footer"><h1>Footer</h1></div>
</body>
```

(4)

```
<style>
/* body 태그를 중앙 정렬합니다. */
body {
  width: 960px;
  margin: 0 auto;
}

#aside {
  width: 200px;
  float: left;
}

#section {
  width: 760px;
  float: left;
}
</style>
```



## 4.10 clear:both를 사용한 레이아웃 / float 속성을 사용한 레이아웃 구성

(1)

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Property Basic</title>
  <style>

  </style>
</head>

<body>
  <div id="header"></div>
  <div id="navigation"></div>
  <div id="aside"></div>
  <div id="section"></div>
  <div id="footer"></div>
</body>
</html>

<body>
  <div id="header"></div>
  <div id="navigation"></div>
  <div class="clear"></div>

  <div id="aside"></div>
  <div id="section"></div>
  <div class="clear"></div>

  <div id="footer"></div>
  <div class="clear"></div>
</body>
```

사용하지 않아도 문제 없습니다.

사용하지 않아도 문제 없습니다.

(2)

```
<body>
  <div id="header"><h1>Header</h1></div>
  <div id="navigation"><h1>Navigation</h1></div>
  <div class="clear"></div>

  <div id="aside">

    <h1>Aside</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  </div>
  <div id="section">
    <h1>Section</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  </div>
  <div class="clear"></div>

  <div id="footer"><h1>Footer</h1></div>
  <div class="clear"></div>
</body>

<style>
  body {
    width: 960px;
    margin: 0 auto;
  }

  .clear {
    clear: both;
  }

  #aside {
    float: left;
    width: 260px;
  }

  #section {
    float: right;
    width: 700px;
  }
</style>
```

clear: both를 적용합니다.

Header

Navigation

Aside

Section

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed quis arcu vel enim tempus egestas. Mauris purus lorem, hendrerit vel cursus quis, viverra id purus. Sed quis arcu vel enim tempus Phasellus luctus pellentesque lectus, in rhoncus diam convallis sed. Duis ac volutpat magna. Praesent eget tempus quam. Etiam laoreet at ex sit amet condimentum. In id aliquam sapien. Ut ornare, orci id lacinia iaculis, est tortor elementum leo, et ornare est dolor sit amet leo. Fusce nunc lectus, ullamcorper consequat metus ut, facilisis aliquam turpis. Duis dictum commodo tortor, ut viverra eros. Aliquam mattis iaculis arcu et efficitur. Ut augue mauris, vestibulum eu mollis id, sodales id mauris. Curabitur mollis augue in nisi blandit tincidunt. Nam dui enim, facilisis condimentum scelerisque quis, gravida gravida sem. Aenean ut arcu fermentum, rutrum nunc a, rhoncus leo. Duis posuere, nisi vel facilisis aliquam, felis sem interdum enim, at tempus velit mauris vel dui.

Footer



## 4.11 그림자 속성 / text-shadow 속성

text-shadow: 5px 5px 5px black

오른쪽      아래      흐림도      색상

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Property Basic</title>
  <style>
    h1 {
      text-shadow: 5px 5px 5px black;
    }
  </style>
</head>
<body>
  <h1>Lorem ipsum dolor amet</h1>
</body>
</html>
```



태그에 그림자를 부여해주는 속성

## 4.11 그림자 속성 / box-shadow 속성

box-shadow: 5px 5px 5px black

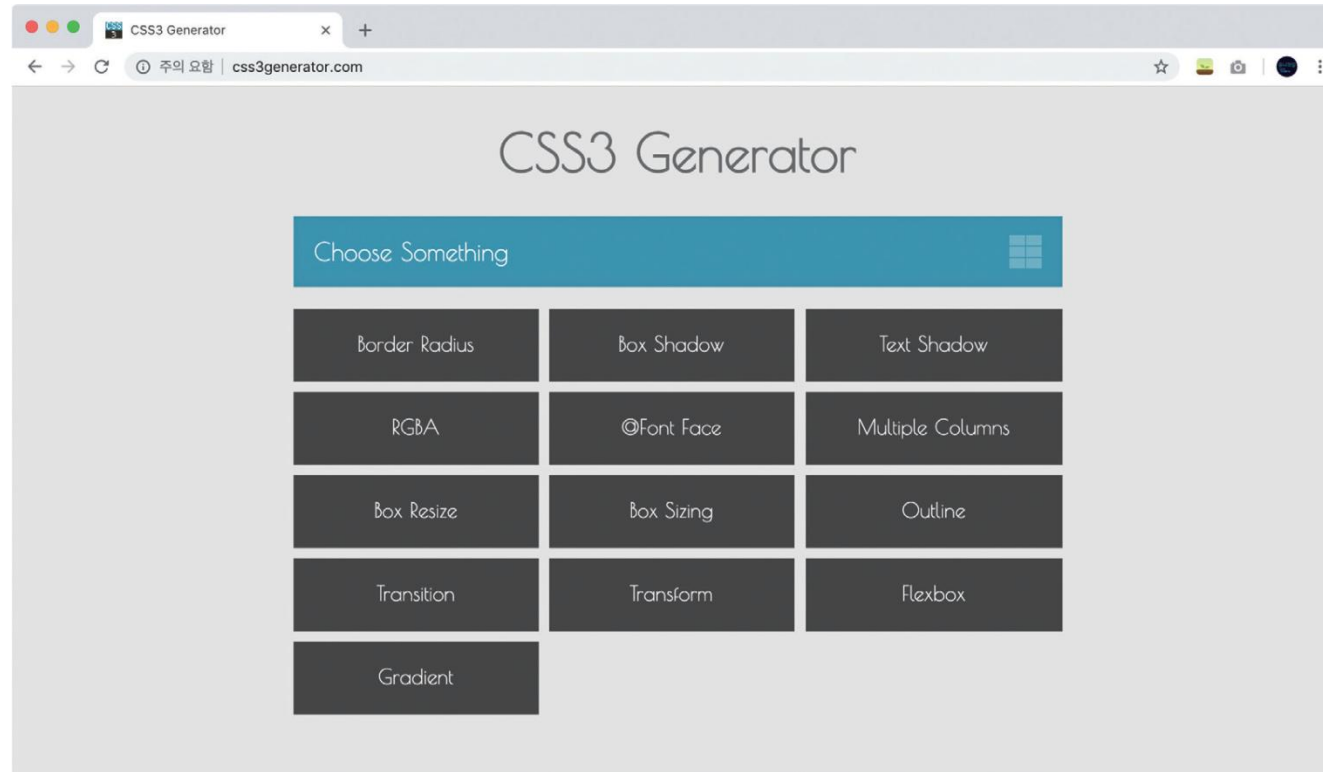
오른쪽      아래      흐림도      색상

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Property Basic</title>
  <style>
    div {
      border: 3px solid black;
      box-shadow: 10px 10px 30px black;
      text-shadow: 5px 5px 5px black;
    }
  </style>
</head>
<body>
  <div>
    <h1>Lorem ipsum dolor amet</h1>
  </div>
</body>
</html>
```



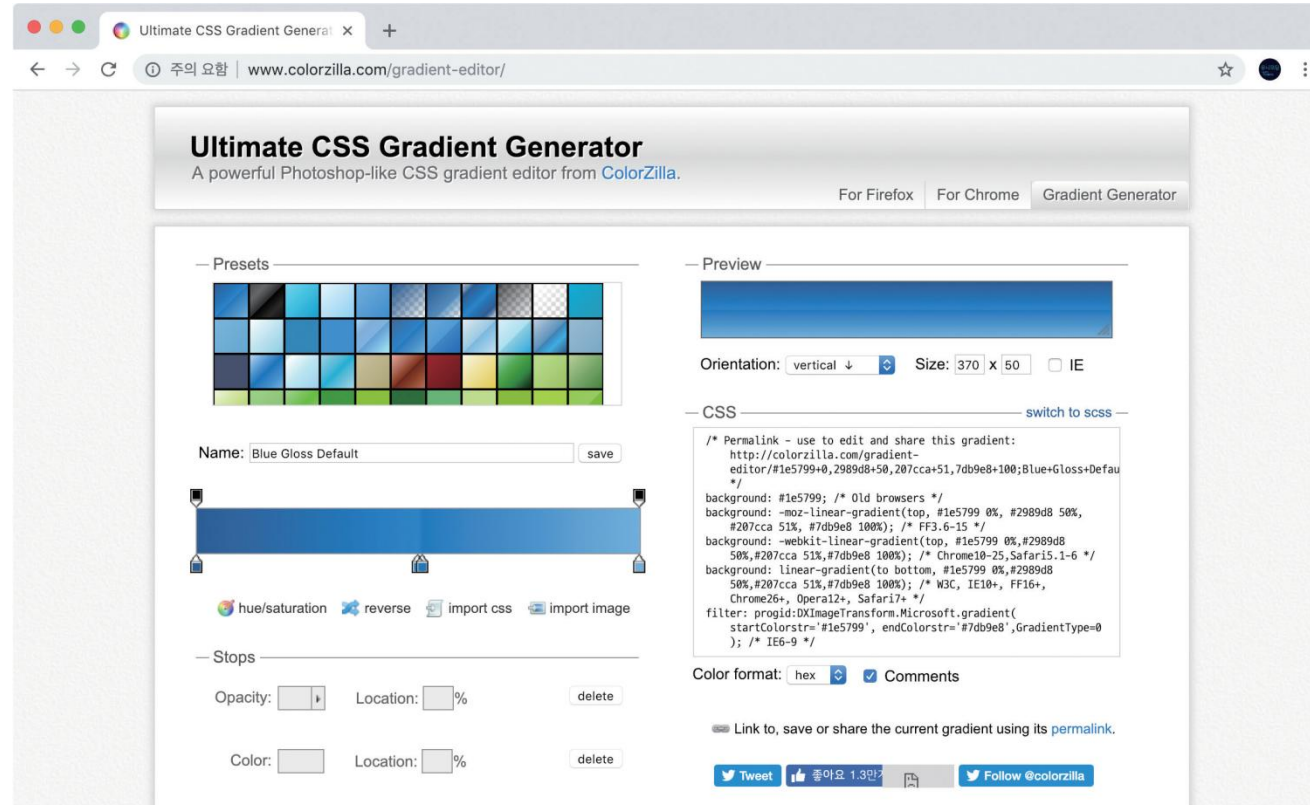
박스에 그림자를 부여해주는 속성

## 4.11 그림자 속성 / box-shadow 속성



CSS3 Generator 툴 사용: <http://css3generator.com/> 접속  
사용하면 box-shadow와 text-shadow 속성을 쉽게 생성  
원하는 형태의 스타일을 정하고 생성된 코드를 복사해 스타일시트에 붙여 넣으면 됨

## 4.12 그레이디언트



- <http://www.colorzilla.com/gradient-editor/> 접속: Ultimate CSS Gradient Generator
- 왼쪽에서 원하는 형태의 그레이디언트를 생성하고 오른쪽에 생성된 CSS 코드를 복사하는 식으로 사용
- 선형 그레이디언트 이외에 원형 그레이디언트도 쉽게 생성할 수 있음

## 4.12 그레이디언트

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Style Property Basic</title>
  <style>
    div {
      font-family: 'Times New Roman';
      height: 60px;
      line-height: 60px;
      text-align: center;

      /* 복사해서 붙여 넣으세요. */
      /* Old browsers */
      background: #f85032;
      /* FF3.6+ */
      background: -moz-linear-gradient(top, #f85032 1%, #f16f5c 50%,
        #f6290c 51%, #e73827 100%);
      /* Chrome, Safari4+ */
      background: -webkit-gradient(linear, left top, left bottom,
```

```
        color-stop(1%, #f85032), color-stop(50%, #f16f5c),
        color-stop(51%, #f6290c),color-stop(100%, #e73827));
      /* Opera 11.10+ */
      background: -o-linear-gradient(top, #f85032 1%, #f16f5c 50%,
        #f6290c 51%, #e73827 100%);
      /* IE10+ */
      background: -ms-linear-gradient(top, #f85032 1%, #f16f5c 50%,
        #f6290c 51%, #e73827 100%);
      /* W3C */
      background: linear-gradient(top bottom, #f85032 1%, #f16f5c 50%,
        #f6290c 51%, #e73827 100%);
      /* IE6-9 */
      filter: progid:DXImageTransform.Microsoft.gradient(
        startColorstr='#f85032', endColorstr='#e73827', GradientType=0
      );
    }
  </style>
</head>
<body>
  <div>
    <h1>CSS3 Gradient</h1>
  </div>
</body>
</html>
```



**CSS3 Gradient**

## 4.13 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;
    transition-duration: 1s;
  }
</style>
```



- 벤더 프리픽스는 웹 브라우저 공급 업체(마이크로소프트, 모질라, 구글, 애플, 오페라)에서 제공
- 실험적인 기능이 필요할 때 주로 사용함