

CSS

1. CSS 기본 문법
2. 다양한 선택자
3. 선택자 우선순위
4. 서체, 글 스타일
5. 목록 스타일
6. display, visibility
7. Background 스타일

1. CSS 기본 문법

CSS

CASCADING STYLE SHEET

- 스타일 선언이 위에서 아래로 순차적으로 적용이 되고, 마지막에 선언된 스타일이 우선순위 있음
- 마크업 언어(HTML/XHTML) 가 실제 화면에 표시되는 방법을 기술하는 언어
- HTML문서의 레이아웃과 스타일을 정의한다.
- W3C에서 표준안을 만들어 내고 있으며 1996년 CSS Level 1 이후, 현재 CSS Level 2.1에서 CSS Level 3까지 보급되고 있음

skip navigation
[skip to content](#)

LOGO

top navigation
[Login](#)
[Member join](#)
[Sitemap](#)

main navigation

- [About](#)
- [Services](#)
- [Portfolio](#)
- [Contact us](#)

About

- [Company Info](#)
- [History](#)
- [Location](#)

[Home](#) > [About](#) > [Company Info](#)

Company Info



Lorem ipsum dolor sit amet, consectetur adipisicing elit. Iusto, ea, impedit ut perspiciatis assumenda ipsam m
facere vitae cumque sapiente sequi qui asperiores eos ipsum perferendis tenetur veritatis nesciunt dolores co
incidunt beatae. Quod, amet recusandae earum iste debitis! Amet, expedita, aperiam, sint omnis fuga magnar
ipsum!

bottom navigation
[About](#)
[Services](#)
[Portfolio](#)
[Contact us](#)
[Location](#)

Copyright © COMPANY Design
Powered by Webazit, e-mail : webmaster@company.com

LOGO

[Login](#) | [Member Join](#) | [Sitemap](#)

[ABOUT](#) [SERVICES](#) [PORTFOLIO](#) [CONTACT US](#)



About

Company Info

[Home](#) > [About](#) > [Company Info](#)

Company Info

History

Location



Lorem ipsum dolor sit amet, consectetur adipisicing elit. Iusto, ea, impedit ut perspiciatis assumenda ipsam minus veritatis tenetur
nisi commodi laboriosam quae sint nulla? Obcaecati, corrupti,
placeat, quasi repudiandae deleniti quod aliquam facere vitae
cumque sapiente sequi qui asperiores eos ipsum perferendis
tenetur veritatis nesciunt dolores corporis sint nisi numquam
excepturi dolor aspernatur natus provident et itaque quaerat
saepe unde quo quidem odit sunt soluta animi incidunt beatae,
Quod, amet recusandae earum iste debitis! Amet, expedita,
aperiam, sint omnis fuga magnam eos quidem alias soluta nobis
sunt unde incidunt! Ratione, autem eligendi molestiae saepe et
numquam facilis pariatur voluptates ipsum!

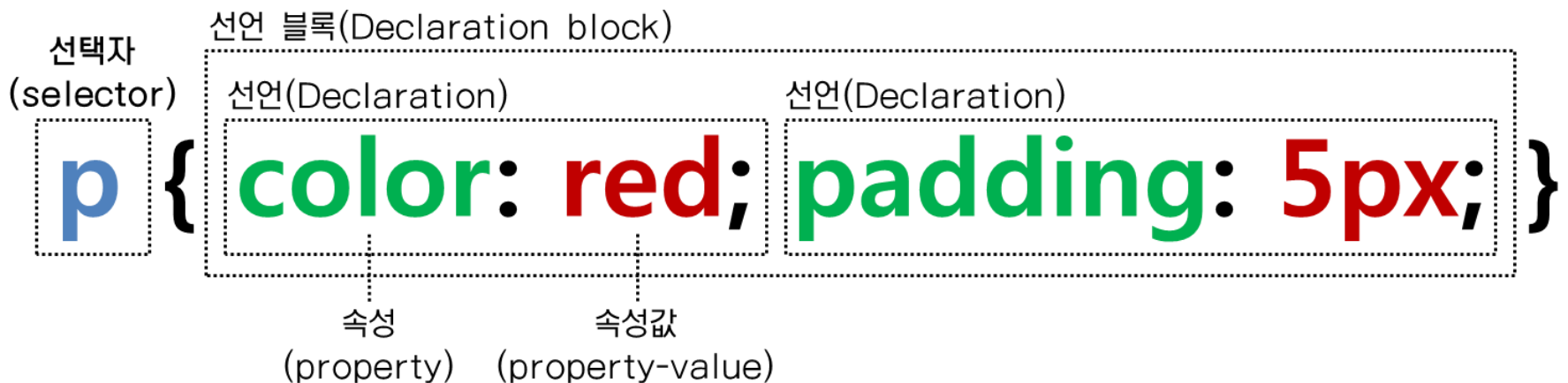
Copyright © COMPANY Design

Powered by Webazit, e-mail : webmaster@company.com

[About](#) | [Services](#) | [Portfolio](#) | [Contact us](#) | [Location](#)

1. CSS 적용 방법과 기본 문법

1 CSS 적용방법



스타일 적용 대상인 선택자(**selector**)를 지정하고,
중괄호 { } 로 감싼 영역에 속성(Attribute)과 값(Value)를 작성.

속성들은 '**세미콜론(;)**' 으로 구분,
선택자(selector) 에는 적용하고자 하는 요소의 **지정태그명, 클래스명, 아이디명**

1. CSS 적용 방법과 기본 문법

<1> 각 요소에 직접 적용하는 CSS

웹표준과 어긋나는 방법으로,

구조를 작성하는 HTML과 표현을 담당하는 CSS를 함께 쓰는 방법.

이방법은 일부 문서나 게시판의 내용을 만들 때 **제한적으로 사용.**

```
<p style="color: gray; background: beige;">  
    <span style="font-weight: bold;">Lorem ipsum dolor sit amet,</span> consectetur  
adipisicing elit. Qui, excepturi quis velit veniam quas impedit ipsam sit perferendis  
voluptatem ex.  
</p>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Qui, excepturi quis velit veniam quas impedit ipsam sit preferendis voluptatem ex.

<2> style 요소를 활용

<head> </head> 사이에 <style> 요소를 사용하여 css 스타일링을 모아둔다.

```
<html lang="ko">
<head>
<meta charset="utf-8">
<title>css 적용하기 <style>요소 안에 작성하기</title>
<style type="text/css">
p{
    color: gray;
    background: beige;
}
p span{
    font-weight: bold;
}
</style>
</head>
<body>

<p>
    <span>Lorem ipsum dolor sit amet,</span> consectetur adipisicing elit. Qui,
    excepturi quis velit veniam quas impedit ipsam sit preferendis voluptatem ex.
</p>

</body>
</html>
```


<3> CSS 내용을 외부 파일로 만들어 연결하는 방법

파일 확장자는 '.css' . 연결은 <link> 요소 활용.

```
<!DOCTYPE html>
<html lang="ko">
<head>
<meta charset="utf-8">
<title>css 적용하기, link요소를 연결하기</title>
<link rel="stylesheet" type="text/css" href="css/style.css">
</head>
<body>

<p>
    <span>Lorem ipsum dolor sit amet,</span> consectetur adipisicing elit
    obcaecati explicabo aliquam dolore. Beatae, temporibus itaque nihil incidunt
</p>

</body>
</html>
```

style.css

```
@charset "utf-8";

p{
    background: beige;
    color: gray;
}
p span{
    font-weight: bold;
    color: blue;
}
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Alias dicta obcaecati explicabo aliquam dolore. Beatae, temporibus itaque nihil incidunt voluptatum.

'rel' 은 현재 문서와의 관계를 나타내는 속성 / 'href' css 파일의 경로 지정
'type' 문서의 타입 지정. **HTML5 문서라면 'type' 속성 생략 가능**

<3> CSS 내용을 외부 파일로 만들어 연결하는 방법

```
@charset "utf-8";

p{
    background: beige;
    color: gray;
}
p span{
    font-weight: bold;
    color: blue;
}
```

style.css 파일에서, **시작부분에 문자 코드셋은 필수.**

문자코드셋이 지정되지 않았을 경우 일부 브라우저에서 제대로 적용되지 않을 수 있다.

웹표준에서는 많은 문자 코드를 표현할 수 있는 '**utf-8**'을 권장.

<3> CSS 내용을 외부 파일로 만들어 연결하는 방법

<style>요소 내에서 외부 css 파일 연결 방법

```
<style type="text/css">  
@import url(sample.css);  
</style>
```

@import url (경로/파일명.css); 또는 **@import "파일명.css";** 로 작성

<3> CSS 내용을 외부 파일로 만들어 연결하는 방법

CSS 파일 내부에서 다른 CSS 파일을 불러올 경우

```
@charset "utf-8";  
@import url(sample.css);
```

연결된 sample.css 파일은 utf-8 문자셋 지정 이후에 불러온 것으로 sample.css 파일 내부에서 문자셋을 다시 지정해야 한다.

2 CSS의 값을 표현하는 규칙

<1> CSS로 표현하는 색상값

16진수 hexa 표기법 / 색상의 키워드 (이름) / RGB 범위값
HTML5에서는 '색조/채도/명도'와 '투명도추가' 방법이 도입됨.

① '**16진수 표기법**' '#'이후 색상값 16진수
#rrggbb 형식, 각 값들이 반복되는 경우 #rbg 형식으로 한자리 생략 가능

#ff0000 -> #f00

```
<style type="text/css">
p{
    background-color: #c0c0c0;
    color: #00f;
}
</style>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Alias dicta obcaecati explicabo aliquam dolore. Beatae, temporibus itaque nihil incidunt voluptatum.

② 색상 키워드

http://www.w3schools.com/tags/ref_colorpicker.asp

기본 색상 키워드는 aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, yellow.....etc

```
p{  
    background-color: silver;  
    color: blue;  
}
```

③ 'RGB 범위값' 10진수 형식으로 적용. 각 색상은 0~255 범위, 쉼표(,)로 구분

```
p{  
    background-color: rgb(192,192,192);  
    color: rgb(0,0,255);  
}
```

http://www.w3schools.com/tags/ref_colorpicker.asp

④ 'RGB 비율 적용법'

각각의 값을 %로 대입

주의할 점은 값이 '0' 이더라도 '%'를 생략해서는 안된다.

```
p{  
    background-color: rgb(80%,80%,80%);  
    color: rgb(0%,0%,100%);  
}
```


⑤ HTML5 '색조/채도/ 명도 입력법'

HSL(Hue, Saturation, Lightness)

색조(Hue)는 색상환의 각도(0~ 360도)로 표현.
360=red, 120=green, 240 = blue

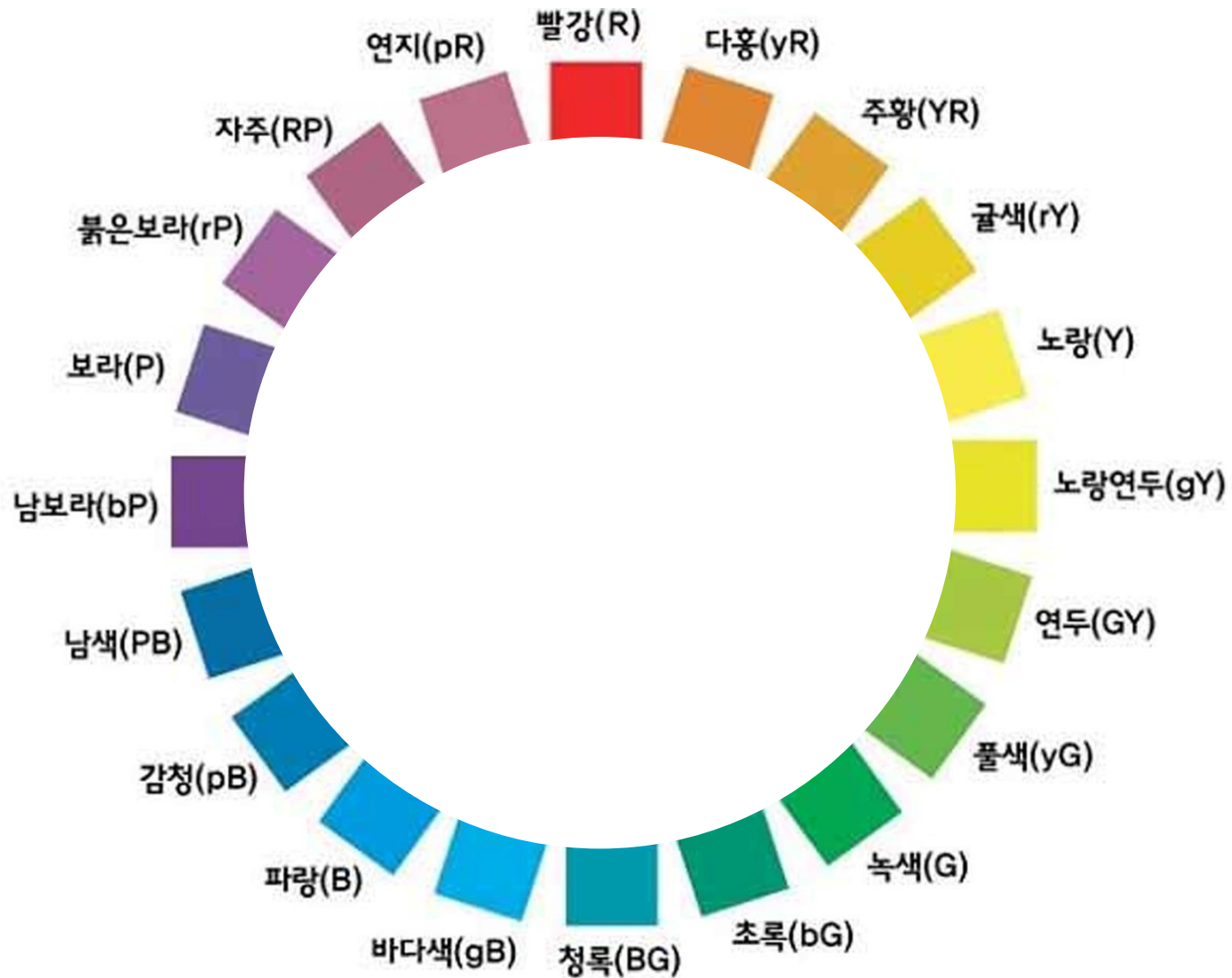
채도(Saturation)는 백분율로 표현.

명도(Lightness)는 백분율로 표현.

0%= 완전한 회색, 100% = 가장높은 채도

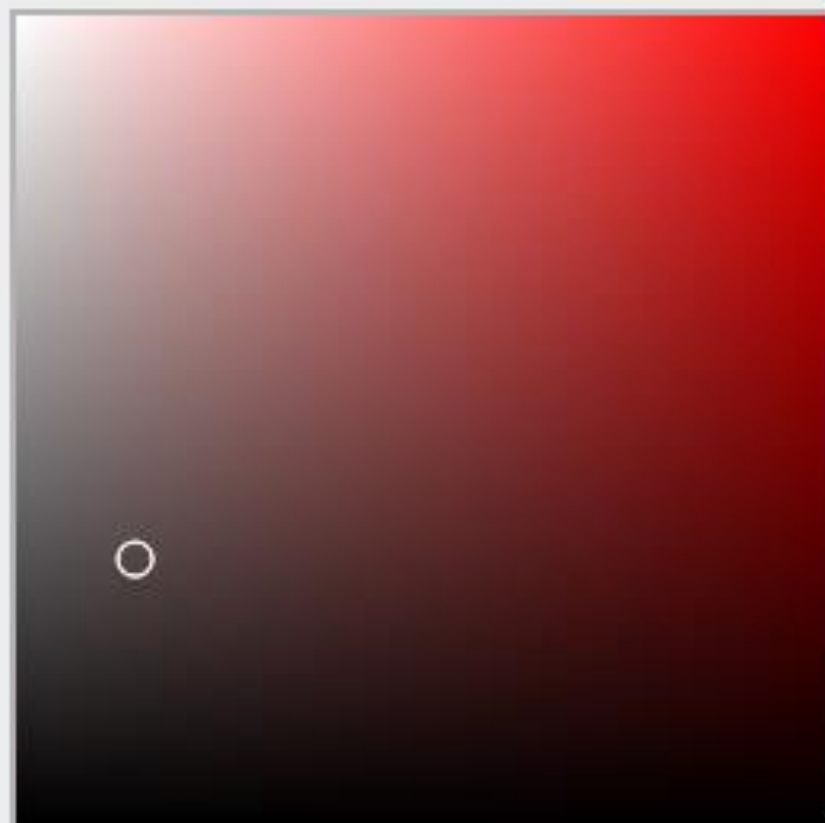
명도0% = 검은색, 100% = 흰색 50% = 기본값

```
p{  
    background-color: hsl(0,0%,80%);  
    color: hsl(240,100%,50%);  
}
```

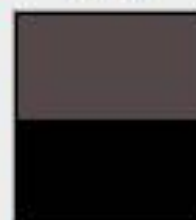


색조(Hue)는 색상환의 각도(0~ 360도)로 표현.
360=red, 120=green, 240 = blue

Color Picker (Foreground Color)



new



current

OK

Cancel

Add to Swatches

Color Libraries

☒ H: 0 °
☐ S: 15 %
☐ B: 33 %
☐ R: 84
☐ G: 72
☐ B: 72

☐ L: 32
☐ a: 5
☐ b: 2

C: 70 %
M: 70 %
Y: 65 %
K: 26 %

☐ Only Web Colors

544848

⑥ HTML5 투명도 (alpha)

RGBA(red, green, blue, **alpha**), HSLA(hue, saturation, lightness, **alpha**)로 표현
alpha 값은 0.0 ~ 1.0 사이에 지정,
0 = 완전한 투명 / 1 = 완전한 불투명

```
body{  
    background: url(img/tree.png);  
}  
p{  
    width: 415px;  
    background-color: rgba(192,192,192,0.5);  
}  
p span{  
    font-weight: bold;  
    font-size: 2em;  
    font-family: verdana,sans-serif;  
    color: hsla(0,100%,50%,0.3);  
}
```



<2> CSS로 지정하는 길이의 단위

CSS에서 지정하는 길이 값의 단위는 문서 매체의 종류(인쇄물/모니터)에 따라 달라짐.

인쇄용 : 절대값과 관련된 단위 'in(인치), cm, mm, pt(포인트), pc(파이크)

모니터,스크린용: '**em, px, %**' 등의 상대값
웹작업에서는 상대값이 효과적.

HTML5에서는 'rem 과 ch'가 추가되었고, 뷰포트(viewport)의 백분율 길이인 'vw, vh, vmin, vmax' 가 추가됨.

<2> CSS로 지정하는 길이의 단위

① 절대값 in, cm, mm, pt

| | | |
|---------------|--------------|--------------|
| 1in = 96px | 1cm = 37.8px | 1mm = 3.78px |
| 1pt = 1/72 in | 1pc = 12pt | |

② 'px' 해상도가 높으면 상대적으로 작아 보이고, 해상도가 낮으면 상대적으로 커 보임.

```
<style>
p{
    width: 400px;
    background-color: lime;
}
</style>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Alias dicta obcaecati explicabo aliquam dolore. Beatae, temporibus itaque nihil incidunt voluptatum.



400px

- ③ '백분율 (%)로 값을 지정하면, 부모요소의 길이를 기준으로 비율을 표현

```
<style>
p{
    width: 50%;
    background-color: lime;
}
</style>
```

Lorem ipsum dolor sit amet,
consectetur adipisicing elit. Alias
dicta obcaecati explicabo aliquam
dolore. Beatae, temporibus itaque
nihil incidunt voluptatum.

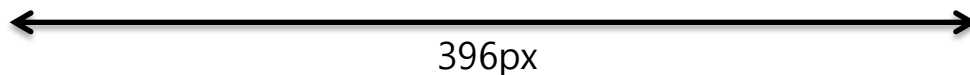


④ 'em' 상대적인 단위,
em(엠) 단위가 쓰여진 곳의 원래 폰트 사이즈의 배수
부모요소 값이 없을 경우 $1em = 16px$
부모요소 글자 크기가 $12px$ 이면 $1.1em = 13.2px$

```
<style>
body{
    font-size: 12px;
}
p{
    width: 30em;
    background-color: lime;
    font-size: 1.1em;
}
</style>
```

<body>요소의 지정된 'font-size' 값은 12px
 <p> 요소의 'font-size' 1.1em = '12 x 1.1' = 13.2px
 <p> 요소의 width 30em = 12 x 1.1 x 30 = 396px
 (p 요소내에서 font-size를 기준으로 계산)

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Alias
 dicta obcaecati explicabo aliquam dolore. Beatae, temporibus
 itaque nihil incidunt voluptatum.



⑤ 'HTML5 의 'rem' 은 부모요소가 아니라 루트(문서의 기본값, 1em=16px)을 기준으로 표현

```
body{  
    font-size: 12px;  
}  
p{  
    font-size: 1.1em  
}  
span{  
    font-size: 2rem;  
}
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Alias dicta obcaeca
aliquam dolore. Beatae, temporibus itaque nihil
incididunt voluptatum.

 font-size = 16 x 2 = 32px

⑤ 'HTML5 의 '**rem**' 은 부모요소가 아니라

루트(**문서의 기본값, 1em=16px**)를 기준으로 표현

root(기본값)을 변경하고자 할때는 '**: root**' 선택자를 활용

다음코드에서 root의 'font-size'를 12px 지정한 이후에는 문서안의 rem 값은 모두 12px을 기준으로 함.

```
<style>
:root{
    font-size: 12px;
}
body{
    font-size: 12px;
}
p{
    font-size: 1.1em;
}
span{
    font-size: 2rem;
}
</style>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Alias dicta obcaecati exp
aliquam dolore. Beatae, temporibus itaque nihil incidunt
voluptatum.

⑥ viewport에 대응해서 크기를 지정하는 방법 'vm, vh, vmin, vmax'

| | |
|-------|------------------------|
| 1vw | 뷰포트 폭의 1% |
| 1vh | 뷰포트 높이의 1% |
| 1vmin | 1vw 또는 1vh 중 작은값을 기준으로 |
| 1vmax | 1vw 또는 1vh 중 큰값을 기준으로 |

```
<style>
p{
  width: 50vw;
  background-color: lime;
}
```

width: 50vw

브라우저 창의 가로크기를 변경해 보면 <p> 요소의 가로크기가 50% 와 같은 결과

Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Alias dicta obcaecati explicabo
aliquam dolore. Beatae, temporibus itaque
nihil incidunt voluptatum.

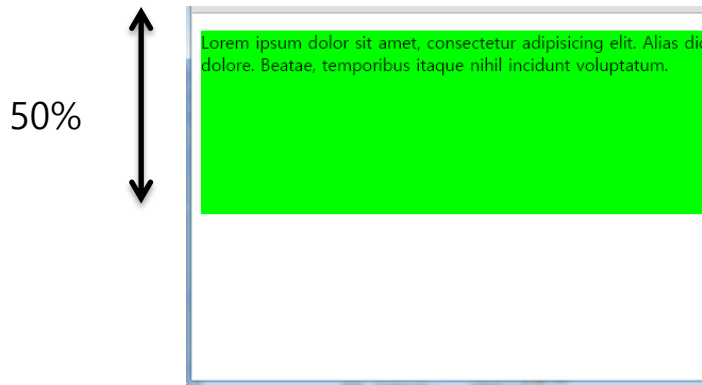


⑥ viewport에 대응해서 크기를 지정하는 방법 'vm, vh, vmin, vmax'

| | |
|-------|------------------------|
| 1vw | 뷰포트 폭의 1% |
| 1vh | 뷰포트 높이의 1% |
| 1vmin | 1vw 또는 1vh 중 작은값을 기준으로 |
| 1vmax | 1vw 또는 1vh 중 큰값을 기준으로 |

height: 50vh

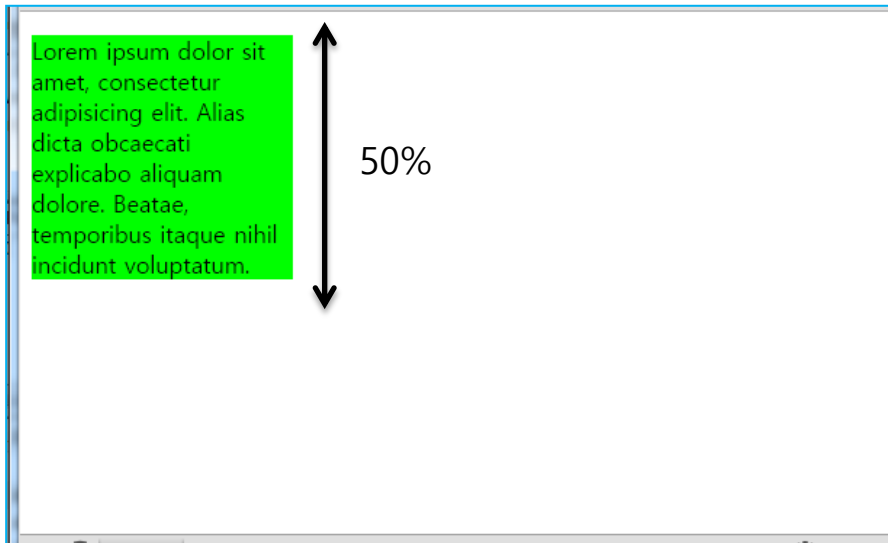
브라우저 창의 가로크기를 변경해 보면 <p> 요소의 세로크기가 50% 와 같은 결과



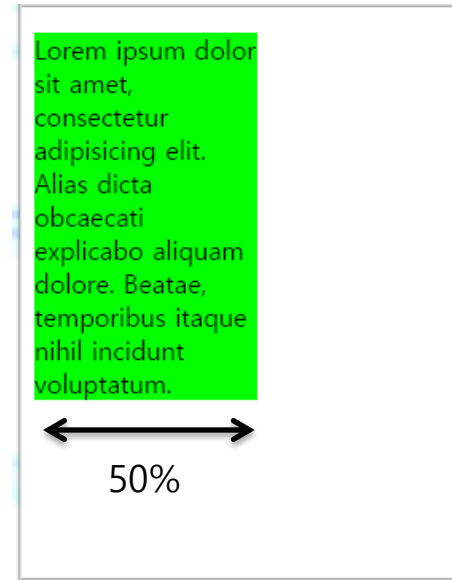
⑥ viewport에 대응해서 크기를 지정하는 방법 'vm, vh, vmin, vmax'
'vmin' 와 'vmax'의 경우 브라우저 크기를 가로와 세로 모두 비교해서 큰쪽, 또는 작은쪽을 기준으로 함.

width: 50vmin

가로나 세로중 짧은 쪽을 기준으로 50% 계산



세로가 짧을 때 기준



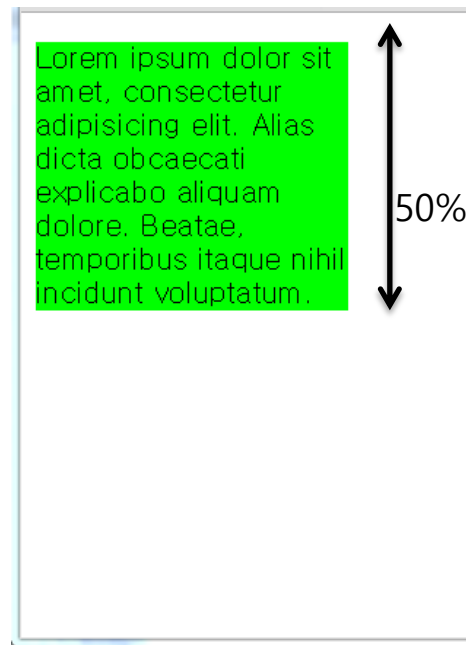
가로가 짧을 때 기준

⑥ viewport에 대응해서 크기를 지정하는 방법 'vm, vh, vmin, vmax'
'vmin' 와 'vmax'의 경우 브라우저 크기를 가로와 세로 모두 비교해서 큰쪽, 또는 작은 쪽을 기준으로 함.

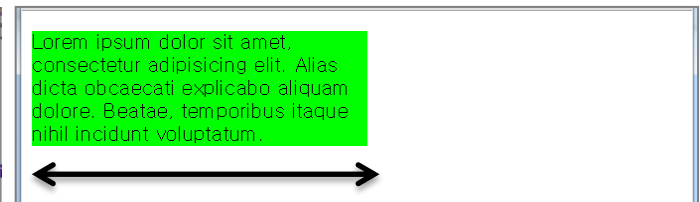
width: 50vmax

가로나 세로중 긴 쪽을 기준으로 50% 계산

```
<style>
p{
    width: 50vmax;
    background-color: lime;
}
</style>
```



세로가 길 때 기준



가로는 길 때 기준

Review TEST

1. 본문에 이미지 배치
2. 문단(p)을 그림 위에 배치 / 문단의 텍스트 크기 12pt/ 문단가로크기 500px / 배경색상: 192,192,192 / 투명도 50%
3. 'CSS TEST 1' 굵게, 글씨크기는 문서 본래의 기본값의 2배, 색상은 HSL 값으로 0, 100%, 50% / 투명도는 0.3 폰트는 verdana, sans-serif



2. 다양한 선택자

1 기본 선택자

<1> 태그 선택자 (타입 선택자)

문서의 **html** 태그를 선택자로 지정하는 방법.

다음 코드는 문서의 **<p>** 요소의 글자 색상을 'lime'색으로 지정한다는 의미.

```
<style type="text/css">
p{
    color: lime;
}
</style>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Alias dicta obcaecati explicabo aliquam dolore. Beatae, temporibus itaque nihil incidunt voluptatum.

```
h1 {font-size:20px;}
```

```
ul {margin:0px;}
```

```
a {color:blue;}
```

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

```
h1, h2 {color:blue;}
```

<2> 전체 선택자 (공통 선택자)

① ' * ' 선택자는 모든 선택자를 선택.

아래 문서에서는 문서내 요소인 body, h1, p, span, address, a 요소를 각각 선택하는 것과 같다.

```
<!DOCTYPE html>
<html lang="ko">
<head>
<meta charset="utf-8">
<title>css, selector</title>
<style type="text/css">
*{
    color: green;
}
</style>
</head>
<body>

<h1>Selector</h1>
<p>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit
    dicta obcaecati explicabo aliquam dolore. Beatae, <span>temporib
    nihil incidunt voluptatum.</span>
</p>
<address>
    mail : <a href="mailto:sample@sample.com">sample.com</a>
</address>
```

Selector

Lorem ipsum dolor sit amet, consectetur
obcaecati explicabo aliquam dolore. &
incididunt voluptatum.

mail : [sample.com](#)

- ② ' * ' 선택자는 '**selector ***' 와 같이 활용가능.
해당 선택자(요소)안에 모든 요소를 선택

```
<!DOCTYPE html>
<html lang="ko">
<head>
<meta charset="utf-8">
<title>css, selector</title>
<style>
```

```
div *{
    color: green;
}
```

```
</style>
</head>
<body>
```

```
<div>
    <h1>Selector</h1>
    <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit.
        Alias dicta obcaecati explicabo aliquam dolore. Beatae
        nihil incidunt voluptatum.
    </p>
</div>
```

```
<p>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    dicta obcaecati explicabo aliquam dolore. Beatae
    incidunt voluptatum.
</p>
```

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

Selector

Lorem ipsum dolor sit amet, consectetur adipiscing elit. obcaecati explicabo aliquam dolore. Beatae nihil incidunt voluptatum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. obcaecati explicabo aliquam dolore. Beatae nihil incidunt voluptatum.

<3> 클래스 선택자

① 각 요소에 고유의 '이름' 지정. 'id'와 'class' 속성으로 지정.

class ' '. '으로 연결. 같은 이름 중복 가능.

여러 가지의 클래스 명을 한 요소에 지정하는 방법도 있음.

아래 예제는 클래스명 '**lead**'가 지정된 요소를 라임색으로 적용

```
<style>
.lead{
    background-color: lime;
}
</style>
</head>
<body>

<p>
    <span class="lead"> Lorem ipsum dolor sit a
    adipiscing elit.</span> Alias dicta obcaecati exp
    Beatae, temporibus itaque nihil incidunt voluptatu
    <span class="lead"> Lorem ipsum dolor sit a
    adipiscing elit.</span> Alias dicta obcaecati exp
    Beatae, temporibus itaque nihil incidunt voluptatu
</p>
```

Lorem ipsum dolor sit amet, consectetur ad
obcaecati explicabo aliquam dolore. Beatae
incidunt voluptatum.
Lorem ipsum dolor sit amet, consectetur ad
obcaecati explicabo aliquam dolore. Beatae
incidunt voluptatum.

- ② 한 요소에 여러 개의 클래스 명을 동시에 적용할 때는 '띄어쓰기'로 표현

``

```
<style>
.lead{
    background-color: lime;
}
.text{
    color: green;
    font-weight: bold;
}
.size{
    font-size: 11px;
}
</style>
</head>
<body>

<p>
    <span class="lead text size">Lorem ipsum do
adipisicing elit.</span> Alias dicta obcaecati expl
Beatae, temporibus itaque nihil incidunt voluptatum
</p>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Alias explicabo aliquam dolore. Beatae, temporibus voluptatum.

<3> 아이디 선택자

요소의 이름을 'id'로 지정할 경우 ' #' 으로 아이디 명을 연결

중복사용 불가, 하나의 요소에 여러 개의 아이디명 동시 적용 불가.

* '아이디 명'이나 '클래스 명' 작성시 **숫자로 시작 불가**, 띄어쓰기, 특수문자는 피한다.

```
<style>
#spot{
    background-color: lime;
}
</style>
</head>
<body>

<p>
    <span id="spot"> Lorem ipsum dolor sit
elit.</span> Alias dicta obcaecati explicabo
temporibus itaque nihil incidunt voluptatum.
</p>
```

Lorem ipsum dolor sit amet, consectetur
obcaecati explicabo aliquam dolore. Beata
incidunt voluptatum.

2 구체적으로 찾아가는 방법

<1> 자손(하위) 선택자 (Descendant selectors)

자손(하위) 선택자는 'E F'의 형식(띄어쓰기)으로 작성,
'E' 요소의 하위에 있는 'F' 요소를 선택
'area' 하위의 <p>만을 선택

```
<style>
.area p{
    color: lime;
}
</style>
</head>
<body>

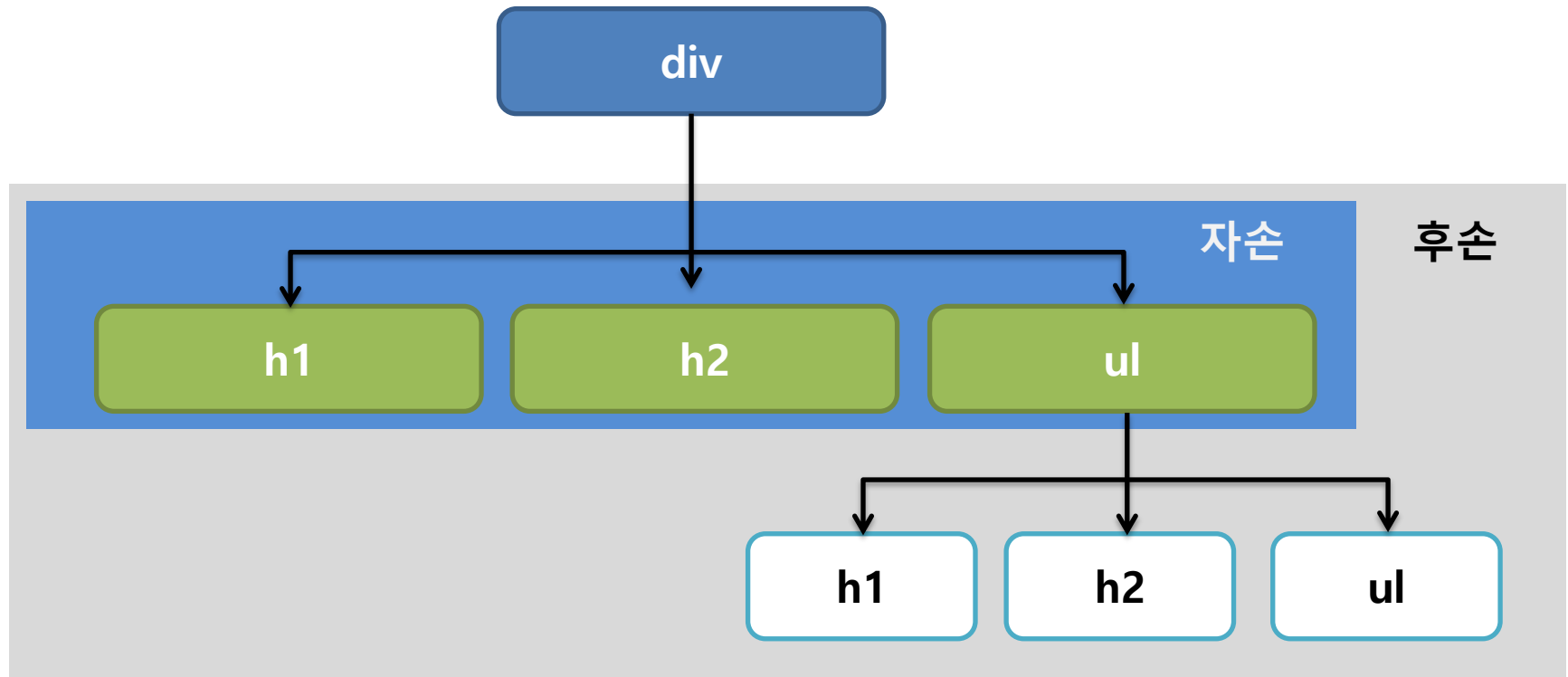
<div class="area">
    <p>Lorem ipsum dolor sit amet,
dicta obcaecati explicabo aliquam dolo
incidunt voluptatum.</p>
    <blockquote>
        <p>Lorem ipsum dolor s
lpsum, eaque.</p>
    </blockquote>
</div>
<p>Lorem ipsum dolor sit amet, consecte
officiis.</p>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit
obcaecati explicabo aliquam dolore. Beatae, temporibus
incidunt voluptatum.

Lorem ipsum dolor sit amet, consectetur adipisicing
 Ipsum, eaque.

Lorem ipsum dolor sit amet, consectetur adipisicing elit
officiis.

자손과 후손



<2> 자식 선택자 (child selectors)

'E > F' 형식으로 작성. 'E'의 자식요소인 'F'를 선택

자식(child) 선택자는 **바로 다음**에 놓인 **하나의 자식만** 선택
자손(하위) 선택자는 하위에 있는 **모든** 요소

```
<style>
.area > p{
    color: lime;
    font-size: 12px;
}
</style>
</head>
<body>

<div class="area">
    <p>Lorem ipsum dolor sit amet, consetetur
dicta obcaecati explicabo aliquam dolore. Beatae,
incididunt voluptatum.</p>
    <blockquote>
        <p>Lorem ipsum dolor sit amet
Ipsum, eaque.</p>
    </blockquote>
</div>
<p>Lorem ipsum dolor sit amet, consetetur ad
officiis.</p>
```

Lorem ipsum dolor sit amet, consetetur adipiscing elit, Alias dicta obcaecati explicabo aliquam dolore, Beatae, temporibus itaque nihil incididunt voluptatum.

Lorem ipsum dolor sit amet, consetetur adipiscing elit, Alias dicta obcaecati explicabo aliquam dolore, Beatae, temporibus itaque nihil incididunt voluptatum.
Ipsum, eaque.

Lorem ipsum dolor sit amet, consetetur adipiscing elit, Alias dicta obcaecati explicabo aliquam dolore, Beatae, temporibus itaque nihil incididunt voluptatum.
officiis.

'E > F' 형식으로 작성. 'E' 의 자식요소인 'F'를 선택

ALIKEROCK

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
div > p {
```

```
    background-color: yellow;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>
```

```
<p>Paragraph 1 in the div.</p>
```

```
<p>Paragraph 2 in the div.</p>
```

```
<span>
```

```
<p>Paragraph 3 in the div.</p>
```

```
</span>
```

```
</div>
```

```
<p>Paragraph 4. Not in a div.</p>
```

```
<p>Paragraph 5. Not in a div.</p>
```

```
</body>
```

```
</html>
```

div 요소의 첫번째 자식들 중 P 모두

<http://goo.gl/UKuQH4>

'E > F' 형식으로 작성. 'E' 의 자식요소인 'F'를 선택

```
<!DOCTYPE html>
<html>
<head>
<style>
div > p {
    background-color: yellow;
}
</style>
</head>
<body>

<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
  <span>
    <p>Paragraph 3 in the div.</p>
    <!-- not Child but Descendant 자식은 아니고 후손 -->
  </span>
</div>

<p>Paragraph 4. Not in a div.</p>
<p>Paragraph 5. Not in a div.</p>

</body>
</html>
```

Paragraph 1 in the div.

Paragraph 2 in the div.

Paragraph 3 in the div.

Paragraph 4. Not in a div.

Paragraph 5. Not in a div.

<3> 인접 형제 선택자 (Adjacent sibling selectors)

'E + F'로 작성, 'E' 요소 다음에 오는 'F' 요소를 선택하는 방법.

아래 예제에서 '.area' 요소 하위의 요소 중에서 첫번째 <p> 요소 다음의 <p>요소를 지정 즉, 첫번째 <p>는 건너뛰고, 다음에 오는 <p>요소에 스타일 적용

```
<style>
.area p + p{
    color: lime;
    font-size: 12px;
}
</style>
</head>
<body>

<div class="area">
    <h1>sample text</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipis
ex inventore voluptate. Assumenda, expedita, aliquid.</p>
    <p>Lorem ipsum dolor sit amet, consectetur adipis
Possimus, perspiciatis.</p>
</div>
```

sample text

Lorem ipsum dolor sit amet, consectetur a
inventore voluptate. Assumenda, expedita,

Lorem ipsum dolor sit amet, consectetur adipisicing elit.

E + F'로 작성, 'E' 요소 바로 다음에 오는 'F' 요소를 선택하는 방법.

ALIKEROCK

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
div + p {
```

```
    background-color: yellow;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>
```

```
<p>Paragraph 1 in the div.</p>
```

```
<p>Paragraph 2 in the div.</p>
```

```
</div>
```

```
<p>Paragraph 3. Not in a div.</p>
```

```
<p>Paragraph 4. Not in a div.</p>
```

```
</body>
```

```
</html>
```

div 요소 바로 다음에 오는 P

<http://goo.gl/BGRg1z>

```
<!DOCTYPE html>
<html>
<head>
<style>
div + p {
  background-color: yellow;
}
</style>
</head>
<body>

<div>
<p>Paragraph 1 in the div.</p>
<p>Paragraph 2 in the div.</p>
</div>

<p>Paragraph 3. Not in a div.</p>
<p>Paragraph 4. Not in a div.</p>

</body>
</html>
```

div 요소 바로 다음에 오는 P

Paragraph 1 in the div.

Paragraph 2 in the div.

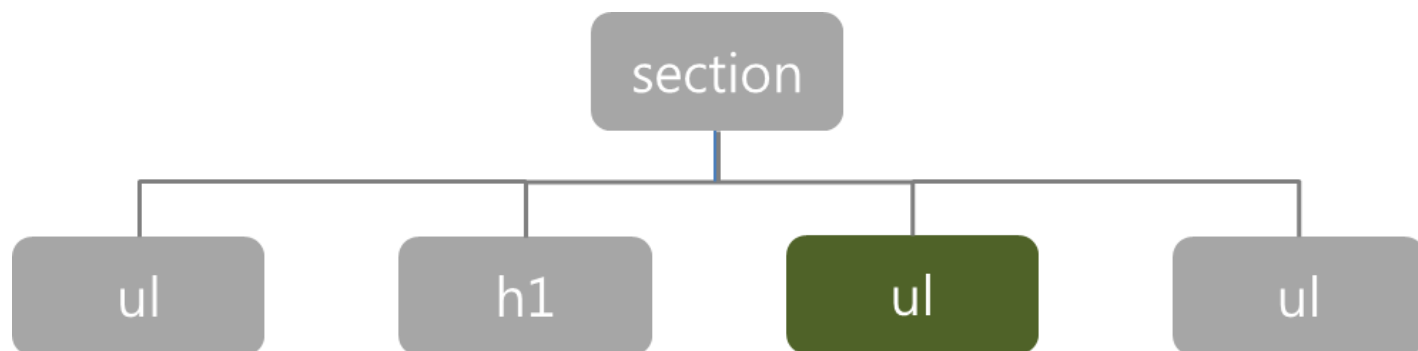
Paragraph 3. Not in a div.

Paragraph 4. Not in a div.

<http://goo.gl/BGRg1z>

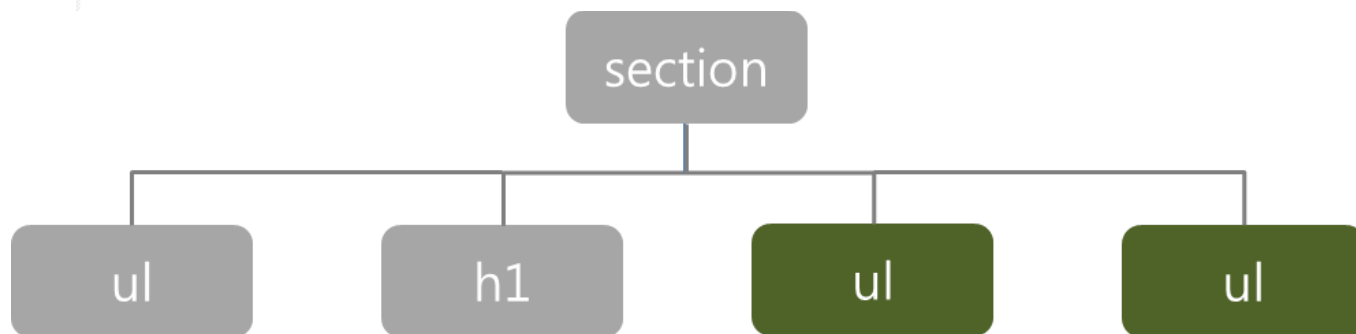
```
/* 인접 형제 선택자 */
```

```
h1+ul { background: yellowgreen; color: darkgreen; }
```



```
/* 일반 형제 선택자 */
```

```
h1~ul { background: darkgreen; color: yellowgreen; }
```



<4> 형제 선택자 (Sibling selectors)

'E ~ F'로 작성, 'E' 요소 다음에 오는 'F' 요소를 모두 선택하는 방법.
(인접 형제 선택자의 경우, 바로 다음에 오는 'F'요소만을 지정)

.area h1 ~ p : <h1> 요소 다음에 오는 <p>요소 모두를 선택

```
<style>
.area h1 ~ p{
    color: lime;
    font-size: 12px;
}
</style>
</head>
<body>

<div class="area">
    <h1>sample text</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    ex inventore voluptate. Assumenda, expedita, a
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    Possimus, perspiciatis.</p>
    <h2>Lorem ipsum</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    Distinctio, voluptatum.</p>
</div>
```

sample text

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Assumenda, expedita, aliquid.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Lorem ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

'E ~ F'로 작성, 'E' 요소 다음에 오는 'F' 요소를 모두 선택

ALIKEROCK

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
div ~ p {
```

```
    background-color: yellow;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>
```

```
<p>Paragraph 1 in the div.</p>
```

```
<p>Paragraph 2 in the div.</p>
```

```
</div>
```

```
<p>Paragraph 3. Not in a div.</p>
```

```
<p>Paragraph 4. Not in a div.</p>
```

```
</body>
```

```
</html>
```

div 요소 다음에 오는 P 모두

<http://goo.gl/fSrjWT>

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
div ~ p {
```

```
    background-color: yellow;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>
```

```
<p>Paragraph 1 in the div.</p>
```

```
<p>Paragraph 2 in the div.</p>
```

```
</div>
```

```
<p>Paragraph 3. Not in a div.</p>
```

```
<p>Paragraph 4. Not in a div.</p>
```

```
</body>
```

```
</html>
```

div 요소 다음에 오는 P 모두

Paragraph 1 in the div.

Paragraph 2 in the div.

Paragraph 3. Not in a div.

Paragraph 4. Not in a div.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
/* div 태그의 후손중 p들만 선택해서 글씨 색상을 'red'로 변경 */
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>
```

```
<p>This is a paragraph inside a div element.</p>
```

```
<p>This is another paragraph inside a div element.</p>
```

```
<span><p>This a paragraph inside a span element, inside a div element.</p></span>
```

```
</div>
```

```
<p>This is a paragraph, not inside a div element.</p>
```

```
<p>This is another paragraph, not inside a div element.</p>
```

```
</body>
```

```
</html>
```

This is a paragraph inside a div element.

This is another paragraph inside a div element.

This a paragraph inside a span element, inside a div element.

This is a paragraph, not inside a div element.

This is another paragraph, not inside a div element.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
/* div 태그의 자식들중(후손아님) 모든 p를 선택하여 글씨 색상을 'red'로 */
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>
```

```
<p>This is a paragraph inside a div element.</p>
```

```
<p>This is another paragraph inside a div element.</p>
```

```
<span><p>This a paragraph inside a span element, inside a div element.</p></span>
```

```
</div>
```

```
<p>This is a paragraph, not inside a div element.</p>
```

```
<p>This is another paragraph, not inside a div element.</p>
```

```
</body>
```

```
</html>
```

This is a paragraph inside a div element.

This is another paragraph inside a div element.

This a paragraph inside a span element, inside a div element.

This is a paragraph, not inside a div element.

This is another paragraph, not inside a div element.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
/* div 태그 이후의 오는 첫번째 p 를 선택, 칼라 red로 */
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>
```

```
<p>This is a paragraph inside a div element.
```

```
<p>This is another paragraph inside a div element.
```

```
<span><p>This a paragraph inside a span element, inside a div element.
```

```
</div>
```

```
<p>This is a paragraph, not inside a div element.</p>
```

```
<p>This is another paragraph, not inside a div element.</p>
```

```
</body>
```

```
</html>
```

This is a paragraph inside a div element.

This is another paragraph inside a div element.

This a paragraph inside a span element, inside a div element.

This is a paragraph, not inside a div element.

This is another paragraph, not inside a div element.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
/* div 태그 이후의 P 요소들 모두, 'red' 컬러로 */
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>
```

```
<p>This is a paragraph inside a div element.</p>
```

```
<p>This is another paragraph inside a div element.</p>
```

```
<span><p>This a paragraph inside a span element, inside a div element.</p></span>
```

```
</div>
```

```
<p>This is a paragraph, not inside a div element.</p>
```

```
<p>This is another paragraph, not inside a div element.</p>
```

```
</body>
```

```
</html>
```

This is a paragraph inside a div element.

This is another paragraph inside a div element.

This a paragraph inside a span element, inside a div element.

This is a paragraph, not inside a div element.

This is another paragraph, not inside a div element.

<5> 그룹화(Grouping)

여러 선택자(요소)들에 같은 CSS 내용을 적용할 때, 쉼표 (,)를 사용하여 그룹화

```
<style>
p, address{
    color: lime;
    font-size: 12px;
}
</style>
</head>
<body>

<div class="area">
    <h1>sample text</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipis
ex inventore voluptate. Assumenda, expedita, aliquid.</p>
    <p>Lorem ipsum dolor sit amet, consectetur adipis
Possimus, perspiciatis.</p>
</div>
<address>
    Seoul, Korea
</address>
```

sample text

Lorem ipsum dolor sit amet, consectetur adipisicing elit
Assumenda, expedita, aliquid.

Lorem ipsum dolor sit amet, consectetur adipisicing elit

Seoul, Korea

3 속성 선택자(Attribute selector)

- ① **E[foo]**는 'foo'라는 속성을 가진 <E> 요소를 선택하는 방법.
다음은 'title' 속성을 가진 <h1> 요소를 선택해서 글자 색상을 변경.

```
h1[title] { color: green }
```

- ② **E[foo="warning"]**는 'foo="warning"'이라는 속성과 값을 모두 만족시키는 <E> 요소를 선택
< a > 요소 중에서 target="_blank" 라는 속성과 값을 가진 것만 선택. 적용.

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

- ③ **E[foo~="warning"]**는 'foo'의 속성에 포함된 값이 'warning' 을 포함한다는 의미.
여러 단어로 지정된 값이라면 띄어쓰기를 통해 구분된 단어를 지정.
다음은 <p> 요소 중에서 'title'의 속성을 가지고, 그 값으로 'lead'를 포함하는 요소를 찾아서 적용

```
p[title~="lead"]{background-color:lime;}
```

- ④ **E[foo] ^=“warning”**는 'foo'의 속성의 값 중에서 'warning' 으로 시작하는 값을 선택. 다음은 <a> 요소 중에서 'href' 속성의 값이 'mailto'로 시작하는 요소를 선택

```
a[href^="mailto"]{background-color:lime;}
```

- ⑤ **E[foo] |=“warning”**는 'foo' 속성의 값으로 'warning'을 가지거나 'warning' 으로 시작하는 값을 선택. 이 방법은 주로 요소에 지정된 특정 언어를 값으로 찾아낼 때 사용. 다음은 문서 내의 모든 요소 중에서 lang="en" 을 지정한 요소만 모두(*) 찾아서 적용.

```
*[lang |=“en”]{background-color:silver;}
```

다음 방법으로도 가능 (IE6, IE7에서는 적용 안됨)

```
*:lang(en){background-color:silver;}
```

```
<style>
body{
    font: 12px/1.6 Verdana,sans-serif;
}
```

```
span[data-title="lead"]{
    font-weight: bold;
    text-decoration: underline;
}
```

```
a[href^="mailto"]{
    background-color: lime;
}
```

```
</style>
</head>
<body>
```

```
<p><span data-title="lead">Lorem ipsum dolor
adipisicing elit.</span> Dolorem, dolorum, de
modi provident cum iste blanditiis porro.</p>
<p>Lorem ipsum dolor sit amet, consectetur ad
molestias.</p>
```

```
<hr>
```

```
<p><a href="#top" title="top">top</a></p>
```

```
<address>
```

```
<a href="mailto:sample@sample.com">sa
```

```
</address>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit.

delectus tenetur minima esse modi provident cum iste blanditiis

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Volupta

[top](#)

sample@sample.com

[attribute^=value] value로 시작하는

```
<!DOCTYPE html>
<html>
<head>
<style>
div[class^="test"] {
    background: #ffff00;
}
</style>
</head>
<body>

<div class="first_test">The first div element.</div>
<div class="second">The second div element.</div>
<div class="test">The third div element.</div>
<p class="test">This is some text in a paragraph.</p>

</body>
</html>
```

The first div element.

The second div element.

The third div element.

This is some text in a paragraph.

[attribute^=value] value로 시작하는

```
<!DOCTYPE html>
<html>
<head>
<style>
[class^="test"] {
    background: #ffff00;
}
</style>
</head>
<body>

<div class="first_test">The first div element.</div>
<div class="second">The second div element.</div>
<div class="test">The third div element.</div>
<p class="test">This is some text in a paragraph.</p>

</body>
</html>
```

The first div element.
The second div element.
The third div element.

This is some text in a paragraph.

<http://goo.gl/zMPfqO>

[attribute\$=value] value로 끝나는

```
<!DOCTYPE html>
<html>
<head>
<style>
[class$="test"] {
    background: yellow;
}
</style>
</head>
<body>

<div class="first_test">The first div element.</div>
<div class="second">The second div element.</div>
<div class="my-test">The third div element.</div>
<p class="mytest">This is some text in a paragraph.
</p>

</body>
</html>
```

The first div element.

The second div element.

The third div element.

This is some text in a paragraph.

[attribute*=value] value를 포함하는

```
<!DOCTYPE html>
<html>
<head>
<style>
[class*="te"] {
    background: yellow;
}
</style>
</head>
<body>

<div class="first_test">The first div element.</div>
<div class="second">The second div element.</div>
<div class="my-test">The third div element.</div>
<p class="mytest">This is some text in a paragraph.
</p>

</body>
</html>
```

The first div element.

The second div element.

The third div element.

This is some text in a paragraph.

<http://goo.gl/uQJlJe>

[attribute]="value"] value라는 클래스명이거나,
단어로 구분되는 'value-' 처럼 '-'가 오는 요소

```
<!DOCTYPE html>
<html>
<head>
<style>
[class|=top] {
    background: yellow;
}
</style>
</head>
<body>

<h1 class="top-header">Welcome</h1>
<p class="top-text">Hello world!</p>
<p class="topcontent">Are you learning CSS?</p>

<p><b>Note:</b> For [<i>attribute</i>|=<i>value</i>]
to work in IE8 and earlier, a DOCTYPE must be
declared.</p>

</body>
</html>
```

Welcome

Hello world!

Are you learning CSS?

Note: For [*attribute*=*value*] to work in IE8 and earlier, a DOCTYPE must be declared.

<http://goo.gl/6CbbMA>

`[attribute]="value"` value라는 클래스명이거나,
'value-' 처럼 '-'가 오는 요소

```
<!DOCTYPE html>
<html>
<head>
<style>
[class|=top] {
    background: yellow;
}
</style>
</head>
<body>

<h1 class="top-header">Welcome</h1>
<p class="top-text">Hello world!</p>
<p class="topcontent">Are you learning CSS?</p>

<p><b>Note:</b> For [<i>attribute</i>|=<i>value</i>]
to work in IE8 and earlier, a DOCTYPE must be
declared.</p>

</body>
</html>
```

Welcome

Hello world!

Are you learning CSS?

Note: For `[attribute]=value` to work in IE8 and earlier, a DOCTYPE must be declared.

[title~="value"]

value라는 클래스명이 있거나,
공백으로 구분되어 value가 있는 요소

```
<!DOCTYPE html>
<html>
<head>
<style>
[title~=flower] {
    border: 5px solid yellow;
}
</style>
</head>
<body>

<p>All images with the title attribute containing the
word "flower" get a yellow border.</p>





<p><b>Note:</b> For [<i>attribute</i>~=<i>value</i>]
to work in IE8 and earlier, a DOCTYPE must be
declared.</p>

</body>
</html>
```

All images with the title attribute containing the word "flower" get a yellow border.



4 가상 선택자(Pseudo selector)

선택하려고 하는 요소를 좀더 구체적으로 또는 그 요소의 상태를 같이 지정하는 방법

※ ie 6 버전에서는 :link, :visited, :hover, :active 만을 지원.

<1> 링크 가상 클래스와 동적 가상 클래스 선택자

:link	링크가 지정되어 있을 때
:visited	해당 링크를 방문했었을 때
:hover	마우스를 링크 위에 올릴 경우
:active	클릭하는 경우
:focus	키보드의 탭키(TAB)로 접근하는 경우

```
<style>
body{
    font: 12px/1.6 Verdana,sans-serif;
}
a:link{
    color: blue;
    text-decoration: none;
}
a:visited{
    color: silver;
    text-decoration: none;
}
a:hover{
    color: brown;
    text-decoration: underline;
}
a:active{
    color: white;
    text-decoration: underline;
    background-color: red;
}
</style>
```

```
<style type="text/css">
a {
    color: blue;
    text-decoration: none;
}
a:hover, a:active, a:focus{
    color: brown;
    text-decoration: underline;
}
</style>
```

<2> 의사 요소 선택자(Pseudo-elements)

어떤 요소를 선택하는 경우 그 요소의
시작지점(:before),
첫번째 자식요소(:first-child),

끝지점 (:after),
마지막 자식요소 (:last-child)

① 요소의 시작지점을 선택 **:before** , 끝 지점을 선택 **:after**

```
<style>
body{
    font: 12px/1.6 Verdana,sans-serif;
}
h1:before{
    content: "before ";
    color: green;
}
h1:after{
    content: " after";
    color: red;
}
</style>
</head>
<body>

<h1>heading 1</h1>
```

'content' 속성은 CSS를 이용하여
특정 내용을 입력하고자 할 때 사용

<2> 의사 요소 선택자(Pseudo-elements)

HTML 코드 기준으로 앞 과 뒤

<h1> ○ text content ○ </h1>

h1:before

h1:after

```
<style>
body{
    font: 12px/1.6 Verdana,sans-serif;
}
h1:before{
    content: "before ";
    color: green;
}
h1:after{
    content: " after";
    color: red;
}
</style>
</head>
```

before heading 1 after

의사 요소 선택자는 ie6, ie7 지원되지 않는다.

구조 선택자 (일반 구조 선택자)

특정한 위치에 있는 태그를 선택

선택자 형태	설명
:first-child	형제 관계 중에서 첫번째 위치하는 태그
:last-child	형제 관계 중에서 마지막에 위치하는 태그
:nth-child(수열)	형제 관계 중에서 앞에서 수열번째에 태그
:nth-last-child(수열)	형제 관계 중에서 뒤에서 수열번째 태그

수열 $2n + 1$

숫자 0 부터 적용 : “1, 3, 5, 7, 9...”

구조 선택자 (일반 구조 선택자)

[코드 3-26]

```
<style>
  ul { overflow: hidden; }
  li {
    list-style: none;
    float:left; padding: 15px;
  }

  li:first-child { border-radius: 10px 0 0 10px; }
  li:last-child { border-radius: 0 10px 10px 0; }

  li:nth-child(2n) { background-color: #FF0003; }
  li:nth-child(2n+1) { background-color:#800000; }
</style>
```

```
<ul>
  <li> First</li>
  <li> Second</li>
  <li> Third</li>
  <li> Fourth</li>
  <li> Fifth</li>
  <li> Sixth</li>
  <li> Seventh</li>
</ul>
```

First Second Third Fourth Fifth Sixth Seventh


```
<style>
body{
    font: 12px/1.6 Verdana,sans-serif;
}
p:first-letter{
    font-size: 2em;
    font-weight: bold;
    color: red;
}
p:first-line{
    background-color: lime;
}
li:first-child, li:last-child{
    text-decoration: underline;
}
li:nth-child(3){
    background-color: silver;
}
</style>
</head>
<body>
```

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing
maxime quisquam eum sunt voluptatibus voluptate et
optio doloreque provident iure cumque quasi nam
```

```
<ul>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
</ul>
```

p:first-letter

<p>요소의 첫번째 글자

p:first-line

<p>요소의 첫번째 라인

li:first-child, li:last-child

요소의 첫번째, 마지막

li:nth-child(3)

 요소의 3번째

()안에 숫자로 지정,

odd(홀수), even(짝수) 지정 가능

```
<style>
body{
    font: 12px/1.6 Verdana,sans-serif;
}
p:first-letter{
    font-size: 2em;
    font-weight: bold;
    color: red;
}
p:first-line{
    background-color: lime;
}
li:first-child, li:last-child{
    text-decoration: underline;
}
li:nth-child(3){
    background-color: silver;
}
</style>
</head>
<body>
```

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
maxime quisquam eum sunt voluptatibus voluptate eum.
optio doloremque provident iure cumque quasi nam.
```

```
<ul>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
    <li>Lorem ipsum dolor sit amet.</li>
</ul>
```

Lorem ipsum dolor sit amet, consectetur

eum sunt voluptatibus voluptate earum iure cumque quasi nam.

- Lorem ipsum dolor sit amet.
- Lorem ipsum dolor sit amet.
- Lorem ipsum dolor sit amet.
- Lorem ipsum dolor sit amet.
- Lorem ipsum dolor sit amet.
- Lorem ipsum dolor sit amet.

구조 선택자 (형태 구조 선택자)

특정한 위치에 있는 특정한 태그 형태를 구분

선택자 형태	설명
:first-of-type	형제 관계 중에서 첫번째로 나오는 특정 태그
:last-of-child	형제 관계 중에서 마지막에 나오는 특정 태그
:nth-of-type(수열)	형제 관계 중에서 앞에서 수열번째에 나오는 특정 태그
:nth-last-of-type(수열)	형제 관계 중에서 뒤에서 수열번째 나오는 특정태그

구조 선택자 (형태 구조 선택자)

[코드 2-9]

```
<style>
  h1:first-of-type { color: red; }
  h2:first-of-type { color: red; }
  h3:first-of-type { color: red; }
</style>
</head>
<body>
  <h1>Header - 1</h1>
  <h2>Header - 2</h2>
  <h3>Header - 3</h3>
  <h3>Header - 3</h3>
  <h2>Header - 2</h2>
  <h1>Header - 1</h1>
</body>
```

Header - 1

Header - 2

Header - 3

Header - 3

Header - 2

Header - 1

부정 선택자

선택자를 반대로 적용할 수 있게 만드는 선택자

선택자 형태	설명
:not(선택자)	선택자를 반대로 적용

부정 선택자

```
<head>
  <title>CSS3 Selector Basic</title>
  <style>
    input:not([type=password]) {
      background: red;
    }
  </style>
</head>
<body>
  <input type="password" />
  <input type="text" />
  <input type="password" />
  <input type="text" />
</body>
```

<input type="password"/>	<input type="text"/>	<input type="password"/>	<input type="text"/>
--------------------------	----------------------	--------------------------	----------------------

Review TEST

1. 제목 뒤에 테스트 글자가 나오도록, 색상은 블루
2. 자식 선택자를 이용하여 첫번째 문단에만 글자색상 라임, 폰트크기는 12px
3. 구분선 아래 문단 문단들과, 주소부분 모두 라임색상, 크기는 12px
4. 리스트 첫번째와 마지막에 밑줄

선택자 테스트

Lorem ipsum dolor sit amet, consectetur adipisicing
dolore, Beatae, temporibus itaque nihil incidunt volu

Lorem ipsum dolor sit amet, conse
Ipsum, eaque.

Lorem ipsum dolor sit amet, consectetur

sample text 테스트

Lorem ipsum dolor sit amet, consectetur adipisicing
Assumenda, expedita, aliquid.

Lorem ipsum dolor sit amet, consectetur adipisicing

Seoul, Korea

Lorem ipsum dolor sit amet.
Lorem ipsum dolor sit amet.
Lorem ipsum dolor sit amet.
Lorem ipsum dolor sit amet.
Lorem ipsum dolor sit amet.
Lorem ipsum dolor sit amet.

3. 선택자의 우선순위

1 선택자의 우선순위란?

공통선택자 '0' < 태그 선택자 '1' < 클래스 선택자(.) '10' < 아이디 선택자 (#) '100'

① 아래에서 '.area p' 선택자는 그 값이 클래스와 타입을 더해서 '11'

.area #text'는 클래스와 아이디를 더해서 '110'

즉, '.area #text'로 선택한 방법이 적용된다.

```
<style>
body{
    font: 12px/1.6 Verdana,sans-serif;
}
.area p{
    color: green;
}
.area #text{
    color: red;
}

</style>
</head>
<body>

<div class="area">
    <p id="text">
        Lorem ipsum dolor sit amet
        blanditiis maxime quisquam eum sunt volupt
        voluptates optio doloremque provident iure
    </p>
</div>
</body>
</html>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit.
eum sunt voluptatibus voluptate earum nihil omnis quis
iure cumque quasi nam.

② 스타일을 인라인 형식으로 직접 지정하면 그 값이 '**1000**' 정도로 높아집니다.

```
<div class="area">  
<p id="text" style="color:blue">  
    Lorem ipsum dolor sit amet, conseq  
blanditiis maxime quisquam eum sunt voluptatibus v  
voluptates optio doloremque provident iure cumque  
</p>  
</div>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Nulla, aperiam, voluptate earum nihil omnis quis rerum voluptates optio doloremque pr

2 우선순위에 관계없이 적용되는 css, !important

선택자의 우선순위와 같이 없이 무조건 적용하고자 할 때는 해당 스타일의 속성값 마지막에 **!important** 를 입력한다.

'area p'의 우선순위가 '.area #text'보다 낮음에도 불구하고 '.area p'의 선택자가 우선 적용됨.

```
<style>
body{
    font: 12px/1.6 Verdana,sans-serif;
}
.area p{
    color: green !important;
}
.area #text{
    color: red;
}

</style>
</head>
<body>

<div class="area">
    <p id="text"style="color:blue">
        Lorem ipsum dolor sit amet,
        blanditiis maxime quisquam eum sunt voluptat
        voluptates optio doloremque provident iure c
    </p>
</div>
```

Lorem ipsum dolor sit amet, c
adipisicing elit. Nulla, aperiam
maxime quisquam eum sunt

4. 서체(font)와 글(text)스타일

1 서체 지정

<1> font-family 속성

서체 지정 시에는 쉼표(,)를 통해 여러 개의 후보 서체를 제시할 수 있다.

값의 마지막에는 서체의 타입을 지정하는데,

'san-serif (고딕체), serif (명조체), cursive (필기체), monospace (동일공간 글꼴) 등이 있다.

```
body {  
    font-family:'돋움', dotum, helvetica, sans-serif  
}
```

서체 값이 두 단어 이상
이거나 영어가 아닌 경
우 따옴표

영문윈도우 대비

맥 OS기반의 컴퓨터 대비

돋움체가 고딕체 계열이므로 'sans-serif' 지정

<2> @font-face

원하는 서체를 같이 올려 두고 연결하는 방법으로 @font-face를 지원
서체가 로딩되는 방법으로 용량이 큰 **서체는 화면에 늦게 적용될** 수 있다.
한글은 특히 용량이 큼.

```
@font-face {  
    font-family: 폰트명; src: url(폰트경로);  
}
```

<2> @font-face

각 브라우저 마다 지원하는 폰트의 확장자가 다르다.

IE 6, 7, 8은 **'.eot'** 지원, IE 9 은 **'.woff'** 지원,

크롬, 파이어폭스, 사파리, 오페라 등은 **.ttf, otf, woff** 등을 지원.

익스플로러는 버전에 따라 지원 확장자가 다름.

```
<style type="text/css">
@font-face{
    font-family: 'Nanum Gothic';
    src:url(fonts/NanumGothic.eot);
    src:url(fonts/NanumGothic.eot?#iefix) format('embedded-opentype'),
        url(fonts/NanumGothic.woff) format('woff'),
        url(fonts/NanumGothic.ttf) format('truetype')
}
p{
    font-family: 'Nanum Gothic', sans-serif;
}
</style>
```


<3> 구글 웹 폰트

비용을 지불하고 웹폰트 사용 하고자 할때, 모빌리스 웹폰트
(<http://api.mobilis.co.kr/webfonts/>)

무료 구글 웹폰트 <http://www.google.com/fonts>

※ ie9 이상에서 적용 가능

The screenshot shows the Google Fonts website interface. At the top, there's a navigation bar with 'Google Fonts' on the left and links for 'DIRECTORY', 'FEATURED', 'ARTICLES', and 'ABOUT' in the center. On the right of the navigation bar are icons for a download and a search. Below the navigation bar, the main content area displays 'Viewing 853 of 853 font families'. Two font families are highlighted: 'Roboto' by Christian Robertson (12 styles) and 'Nanum Gothic' by Sandoll (3 styles). Each font family has a red plus icon next to it. Below the font family names, there are two columns of text. The left column shows the text 'All their equipment and instruments are alive.' in the Roboto font. The right column shows the text '그들의 장비와 기구는 모두 살아 있다.' in the Nanum Gothic font. On the right side of the page, there is a sidebar with a search bar and a 'Categories' section. The categories listed are 'Serif', 'Sans Serif', 'Display', 'Handwriting', and 'Monospace', each with a red checkmark. Below the categories is a 'Sorting' section with a dropdown menu set to 'Trending'.

Google Fonts

DIRECTORY FEATURED ARTICLES ABOUT

Viewing 853 of 853 font families

Roboto
Christian Robertson (12 styles)

Nanum Gothic
Sandoll (3 styles)

All their equipment
and instruments are
alive.

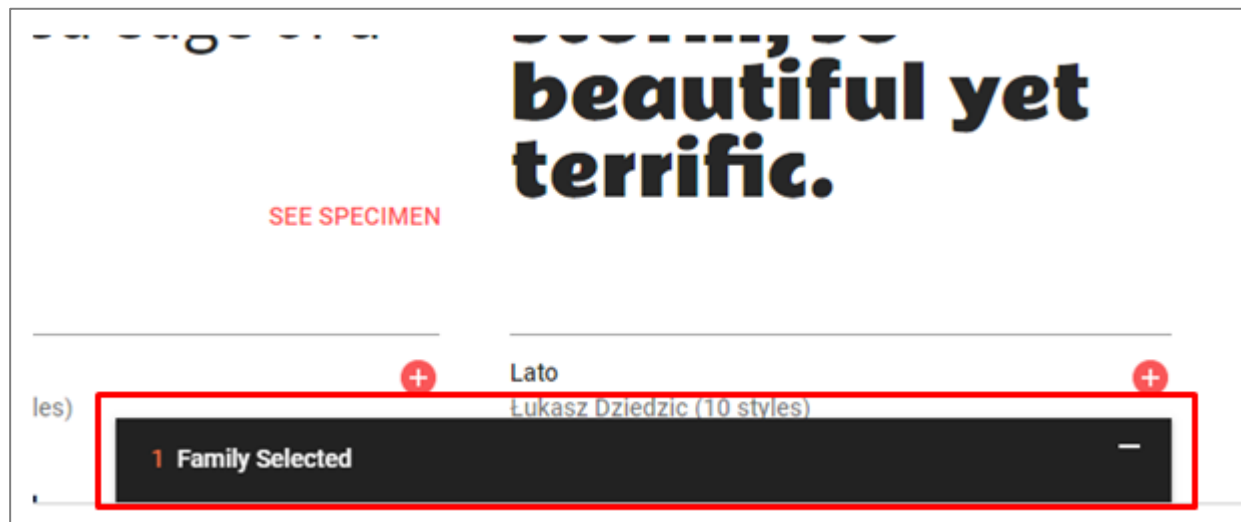
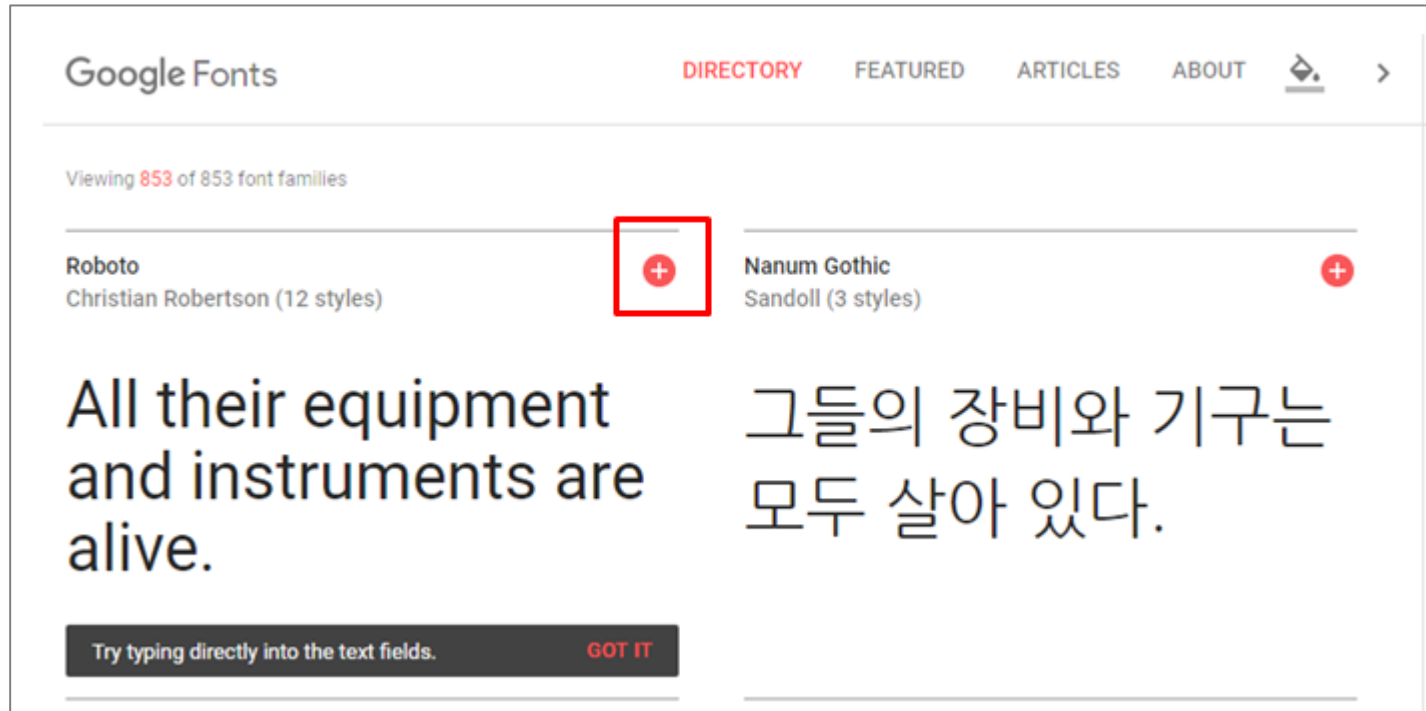
그들의 장비와 기구는
모두 살아 있다.

Categories

- ☒ Serif
- ☒ Sans Serif
- ☒ Display
- ☒ Handwriting
- ☒ Monospace

Sorting
Trending

4. 서체(font)와 글(text)스타일




4. 서체(font)와 글(text)스타일

ALIKEROCK

1 Family Selected

Your Selection **Clear All**

Roboto 

EMBED CUSTOMIZE

Load Time: Fast

Embed Font

To embed your selected fonts into a webpage, copy this code into the <head> of your HTML document.

STANDARD @IMPORT

```
<link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">
```

Specify in CSS

Use the following CSS rules to specify these families:


```
font-family: 'Roboto', sans-serif;
```

For examples of how fonts can be added to webpages, see the [getting started guide](#).

standart, import 방식 제공
두께 등 customize 가능

주의사항
웹폰트개수를 늘리면 로딩
시간이 걸리므로 주의

Google Fonts

[DIRECTORY](#) [FEATURED](#) [ARTICLES](#) [ABOUT](#)  [>](#)

Viewing 853 of 853 font families

Roboto
Christian Robertson (12 styles)

⊖

Nanum Gothic
Sandoll (3 styles)

⊕

All their equipment
and instruments are
alive.

그들의 장비와 기구는
모두 살아 있다.

Categories

☒ Serif

☒ Sans Serif

☒ Display

☒ Handwritten

☒ Monospace

Sorting

Trending ▾

your own highly customized collections by filtering families, weights, and scripts, plus test color themes, and review sample copy. Collections can be shared, making it easy to collaborate on projects and ensure typography is optimized and streamlined throughout the design and engineering process.

Collaborate with Open Source

All the fonts in our catalog are free and open source, making beautiful type accessible to anyone for any project. This means you can share favorites and collaborate easily with friends and colleagues. Google Fonts takes care of all the licensing and hosting, ensuring that the latest and greatest version of any font is available to everyone.

Support

[Frequently Asked Questions](#)[API documentation](#)[GitHub](#)[Analytics](#)[Early Access](#)[Privacy Policy](#)[Terms of Use](#)[Contact us on Twitter](#)

↻

안전함 | <https://fonts.google.com/earlyaccess>

Noto Sans Kannada (Kannada)

When text is rendered by a computer, sometimes there will be characters in the text that can not be displayed, because no font that supports them is available to the computer. When this occurs, small boxes are shown to represent the characters. We call those small boxes “tofu,” and we want to remove tofu from the Web. This is how the Noto font families got their name.

Noto helps to make the web more beautiful across platforms for all languages. Currently, Noto covers over 30 scripts, and will cover all of Unicode in the future. This is the Sans Kannada family. It has Regular and Bold styles and is unhinted.

Noto fonts are intended to be visually harmonious across multiple languages, with compatible heights and stroke thicknesses. For the currently released Noto fonts see code.google.com/p/noto/wiki/FontList

Link

```
@import url(//fonts.googleapis.com/earlyaccess/notosanskannada.css);
```

Example

```
font-family: 'Noto Sans Kannada', sans-serif;
```

Arabic

Amiri

Lateef

Scheherazade

Korean

Nanum Brush Script

[Nanum Gothic](#)

Nanum Gothic Coding

Nanum Myeongjo

Nanum Pen Script

Myanmar

Padauk

Telugu

Dhurjati

Gidugu

Gurajada

Lakki Reddy

Mallanna

4. 서체(font)와 글(text)스타일 – 한글 웹폰트

The screenshot displays the Google Fonts website interface for Korean fonts. The browser address bar shows the URL <https://fonts.google.com/?subset=korean>. The page header includes the Google Fonts logo and navigation links: DIRECTORY, FEATURED, ARTICLES, and ABOUT. A search bar is located on the right side of the header.

The main content area displays four font families, each with a preview of Korean text and a red plus icon indicating it can be added to the collection:

- Nanum Gothic** (Sandoll (3 styles))
Preview text:

빠죽빠죽한 날개 윤곽
이 붉게 빛났다.
- Nanum Myeongjo** (Sandoll (3 styles))
Preview text:

세 시간 전 항구에서
출발한 배를 안개가
감쌌다.
- Nanum Gothic Coding** (Sandoll (2 styles))
Preview text:

파도가 푸르른 저녁
에 내던져져 부서졌
다.
- Nanum Pen Script** (Sandoll (1 style))
Preview text:

내 안의 두 가지 본성은 공통된 기억
을 갖고 있었다.

A dark button with the text "Try typing directly into the text fields." and a "GOT IT" link is positioned below the first two font previews.

The right sidebar contains filtering options:

- Categories:** Serif, Sans Serif, Display, Handwriting, Monospace (all checked).
- Sorting:** Trending (dropdown arrow).
- Languages:** Korean (dropdown arrow).
- Number of styles:** A slider control.
- Thickness:** A slider control.
- Slant:** A slider control.
- Width:** A slider control.

영문 사용법과 동일함

구글 웹 폰트 적용 예제

```
<!DOCTYPE html>
<html lang="ko">
<head>
<meta charset="utf-8">
<title>font, text</title>
<style type="text/css">
@import url(http://fonts.googleapis.com/css?family=Londrina+Shadow|Autour+One);
@import url(http://fonts.googleapis.com/earlyaccess/nanumgothic.css);
p.ko{
    font-family: 'Nanum Gothic', sans-serif;
}
p.en{
    font-family: 'Londrina Shadow', cursive;
    font-size: 1.2em;
}
</style>
</head>
<body>

<p class="ko">
    웹 표준은 월드 와이드 웹의 측면을 서술하고 정의하는 공식 표준이나 다른
    가리키는 일반적인 용어이다. 최근에 이 용어는 웹 사이트를 작성하는 데 중요도가 높
    며 웹 디자인, 개발과 관계가 있다.
</p>
<p class="en">
    Lorem ipsum dolor sit amet, consectetur adipisicing elit. Veniam, bland
</p>

</body>
</html>
```

구글 웹 폰트 적용 예제

웹 표준은 월드 와이드 웹의 측면을 서술하고 정의하는 공식 표준이나 다른 기술 규격을 가리키는 일반적인 용어이다. 최근에 이 용어는 웹 사이트를 작성하는 데 중요도가 높아지고 있으며 웹 디자인, 개발과 관계가 있다.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Veniam, blandit.

2 글자 크기 지정

글자 크기는 'font-size' 속성으로 지정. 절대값, 상대값, px, em, % 사용 가능
※ 아무런 지정도 없으면, 기본값 16px로 설정된다.

① 글자 크기는 7단계의 절대값을 가지고 있다. **medium** 이 기본값

```
font-size: xx-small;  
font-size: x-small;  
font-size: small;  
font-size: medium;  
font-size: large;  
font-size: x-large;  
font-size: xx-large;
```

- ② 'font-size' 속성은 상속을 받게 되는데, 부모요소로부터 더 크게는 'larger', 더 작게는 'smaller'

```
font-size: larger;  
font-size: smaller;
```

- ③ 길이값은 주로 'px'와 'em'을 활용

상속되는 특징을 살려서 상위선택자(주로 body) 에 'px'로 폰트의 크기를 지정하고

그 자식 선택자들은 'em'을 값으로 지정하면

이후 전체적인 폰트 크기를 수정할 경우 상위 선택자 하나만 수정하면 되는 편리함이 있다.

```
<style type="text/css">  
body{  
    font-family: '돋움',dotum,Helvetica,sans-serif;  
    font-size: 12px;  
}  
.area{  
    font-size: 1.1em;  
}  
.area p{  
    font-size: 2em;  
}  
</style>
```

Lorem ipsum dolor sit amet, consectetur adipiscing
repellat enim vitae nam nulla sequi facere?

Lorem ipsum dolor sit am
adipiscing elit. Sequi, ob

.area = 13.2px , .area p = 26.4px

④ %로 지정할 때, 100%는 16px, 75%는 12px...

Pixels	EMs	Percent	Points
6px	0.375em	37.5%	5pt
7px	0.438em	43.8%	5pt
8px	0.500em	50.0%	6pt
9px	0.563em	56.3%	7pt
10px	0.625em	62.5%	8pt
11px	0.688em	68.8%	8pt
12px	0.750em	75.0%	9pt
13px	0.813em	81.3%	10pt
14px	0.875em	87.5%	11pt
15px	0.938em	93.8%	11pt
16px	1.000em	100.0%	12pt
17px	1.063em	106.3%	13pt
18px	1.125em	112.5%	14pt
19px	1.188em	118.8%	14pt
20px	1.250em	125.0%	15pt
21px	1.313em	131.3%	16pt
22px	1.375em	137.5%	17pt
23px	1.438em	143.8%	17pt
24px	1.500em	150.0%	18pt

<http://pxtoem.com/>

3 글자 두께 지정

'font-weight' 속성으로 지정, 브라우저에서는 기본값(**normal**), 볼드체 (**bold**), 2가지로 표현

① normal 두께 표현

```
font-weight: normal;  
font-weight: lighter;  
font-weight: 100;  
font-weight: 200;  
font-weight: 300;  
font-weight: 400;  
font-weight: 500;
```

② bold체 두께 표현

```
font-weight: bold;  
font-weight: bolder;  
font-weight: 600;  
font-weight: 700;  
font-weight: 800;  
font-weight: 900;
```

4 글자 장식

'text-decoration' 속성으로 선택한 요소의 위, 아래, 가운데 선을 긋거나 깜박거리게 할 수 있다.

none	장식 없음	Line-through	가운데 선
underline	밑줄	blink	깜박거림(지원 안함)
overline	윗줄		

① <a> 요소는 기본값으로 밑줄을 가지고 있는데, 마우스 올리는 경우(hover)에 밑줄이 나타나도록 많이 활용.

```
a {  
    color:#666666;  
    text-decoration: none;  
}  
a:hover {  
    color:blue;  
    text-decoration: underline;  
}
```

5 글자 스타일 지정

글자를 기울이는 방법. **italic**(기울임), **oblique**(비스듬하게)
바로 세울 때에는 '**normal**'

① italic 과 oblique 결과는 동일

```
<style type="text/css">
body{
    font-family: '돋움',dotum,Helvetica,sans-serif;
    font-size: 12px;
}
address,em{
    font-style: italic;
}
</style>
```

*Lorem ipsum dolor sit amet, consectetur adipiscing
dolores voluptate ab recusandae odio nesciunt adipis
provident!*

Seoul, Korea

- ② <address>, 요소 등은 원래 기울어진 형태. normal로 바로 세움.
font-style: normal;

```
address,em{
    font-style: normal;
}
</style>
</head>
<body>

<p><em>Lorem ipsum dolor sit amet,</em> con
dolores voluptate ab recusandae odio nesci
<address>Seoul, Korea</address>

</body>
</html>
```

Lorem ipsum dolor sit amet, consectetur adipisicing
odio nesciunt adipisci natus voluptatibus provident!

Seoul, Korea

6 글자 색상

'color' 속성으로 글자 색상 지정.

```
p {
    color:#555;
}
```

4. 서체(font)와 글(text)스타일

7 글의 정렬

'text-align' left, center, right, justify

① 'left', 'center', 'right', 'justify(양쪽 정렬)'.

```
<style type="text/css">
body{
    font-family: '돋움',dotum,Helvetica,sans-serif;
    font-size: 12px;
}
.area{width: 400px;}
.area p.left{text-align: left;}
.area p.center{text-align: center;}
.area p.right{text-align: right;}
.area p.justify{text-align: justify;}
</style>
</head>
<body>

<div class="area">
    <p class="left">Lorem ipsum dolor sit amet.</p>
    <p class="center">Lorem ipsum dolor sit amet.</p>
    <p class="right">Lorem ipsum dolor sit amet.</p>
    <p class="justify">
        Lorem ipsum dolor sit amet, consectetur ad
        possimus soluta corporis dicta dolores quidem inventore vo
        voluptatem iusto dolore deserunt excepturi officiis expedi
        temporibus nihil error rem architecto. Minima maiores erro
        rem consectetur doloremque porro molestiae!
    </p>
</div>
```

olor sit amet,

Lorem ipsum dolor sit amet,

Lorem ipsum dolor sit amet,

olor sit amet, consectetur adipisicing elit. Quod, nisi
soluta corporis dicta dolores quidem inventore
s facere vero est voluptatem iusto dolore deserunt
s expedita culpa provident libero temporibus nihil error
Minima maiores error tempore enim dignissimos sed
doloremque porro molestiae!

4. 서체(font)와 글(text)스타일

8 글의 간격

'letter-spacing' 글자 간격, 'word-spacing' 단어 간격, 'line-height' 줄 간격

※ 본문에 한글, 영문, 숫자 등이 섞여 있는 경우, 각 종류별 간격이 달라 보일 수 있다.

```
p { letter-spacing: 0.1em; }  
p { word-spacing: 0.3em; }  
p { line-height: 1.6; }
```

'line-height' 는 텍스트의 줄 간격 뿐 아니라, 블록 요소의 높이를 지정하면서 동시에 내부의 텍스트를 박스의 가운데(위와 아래의)에 배치할 수 있는 방법이 된다.

```
<style type="text/css">
body{
    font-family: '돋움', dotum, sans-serif;
    font-size: 12px;
}
.area{width: 400px;}
.area h1{
    letter-spacing: 1em;
}
.area p{
    word-spacing: 0.3em;
    letter-spacing: 0.1em;
    line-height: 1.6;
}
</style>
</head>
<body>

<div class="area">
    <h1>글자의 정렬</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing
    iste, adipisci a quo illo aliquid r
    aut deleniti saepe odio sunt.</p>
</div>
```

글 자 의 정 렬

Lorem ipsum dolor sit amet, consectetur adipiscing
elit. Odit, laboriosam, ut, iste, adipisci a quo illo
aliquid necessitatibus fuga tempora est animi quas ab
quibusdam aut deleniti saepe odio sunt.

4. 서체(font)와 글(text)스타일

ALIKEROCK

[코드 4-49]

```
<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;
  line-height: 70px;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="button">
```

```
<a href="#" class="font_big font_italic font_bold font_center">Click</a>
```

```
</div>
```

```
</body>
```

height와 line-height를 같게 설정하면
세로의 중앙에 텍스트를 배치할수 있다.



4. 서체(font)와 글(text)스타일

9 글자 들여쓰기

'text-indent' 한 글자 정도 들여쓰기

간격을 마이너스로 지정하면 화면에서 글자를 보이지 않도록 만들 수 있다.

CSS에서 메뉴를 만들거나, 본문의 텍스트를 이미지로 변환하는 경우 활용

```
<style type="text/css">
body{
    font-family: '돋움',dotum,Helvetica,sans-serif;
    font-size: 12px;
}
.area{width: 400px;}
.area h1{text-indent: -1em; border: 1px solid;}
.area p{text-indent: 1em; border: 1px solid;}
</style>
</head>
<body>

<div class="area">
    <h1>글자의 정렬</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adip
    iste, adipisci a quo illo aliquid necessitatibus fuga t
    aut deleniti saepe odio sunt.</p>
</div>
```

글자의 정렬

Lorem ipsum dolor sit amet, consectetur adipisicing laboriosam, ut, iste, adipisci a quo illo aliquid necessitatibus fuga t
tempora est animi quas ab quibusdam aut deleniti saepe

4. 서체(font)와 글(text)스타일

10 영문의 대/소문자

① text-transform', font-variant

※ HTML 문서 작성할 때 대문자로만 작성하게 되면, IE6에서 CSS를 통한 대/소문자 변경이 안 될 수 있다.

text-transform: uppercase; -> 대문자로 / text-transform: lowercase; 소문자로
text-transform: capitalize; -> 첫 글자만 대문자로

```
<style type="text/css">
body{
    font-family: '돋움',dotum,Helvetica,sans-serif;
    font-size: 12px;
}
.area{width: 400px; font-family: verdana,sans-serif;}
h1.uppercase{text-transform: uppercase;}
h1.lowercase{text-transform: lowercase;}
h1.capitalize{text-transform: capitalize;}
h1.smallcaps{font-variant: small-caps;}
</style>
</head>
<body>

<div class="area">
    <h1 class="uppercase">Lorem Ipsum Dolor.</h1>
    <h1 class="lowercase">Lorem Ipsum Dolor.</h1>
    <h1 class="capitalize">Lorem Ipsum Dolor.</h1>
    <h1 class="smallcaps">Lorem Ipsum Dolor.</h1>
</div>
```

LOREM IPSUM DOLOR.

lorem ipsum dolor.

Lorem Ipsum Dolor.

LOREM IPSUM DOLOR.

4. 서체(font)와 글(text)스타일

- ② **font-variant** 속성의 값 '**small-caps**' 대문자와 소문자가 섞여 있을 경우 소문자를 대문자로 변경하지만, 그 크기는 **소문자의 크기를 유지**하는 특징이 있다. 변경을 막기 위한 값은 '**normal**'

```
font-variant : small-caps;
```

```
<style type="text/css">
body{
    font-family: '돋움',dotum,Helvetica,sans-serif;
    font-size: 12px;
}
.area{width: 400px; font-family: verdana,sans-serif;}
h1.uppercase{text-transform: uppercase;}
h1.lowercase{text-transform: lowercase;}
h1.capitalize{text-transform: capitalize;}
h1.smallcaps{font-variant: small-caps;}
</style>
</head>
<body>

<div class="area">
    <h1 class="uppercase">Lorem Ipsum Dolor.</h1>
    <h1 class="lowercase">Lorem Ipsum Dolor.</h1>
    <h1 class="capitalize">Lorem Ipsum Dolor.</h1>
    <h1 class="smallcaps">Lorem Ipsum Dolor.</h1>
</div>
```

LOREM IPSUM DOLOR.

lorem ipsum dolor.

Lorem Ipsum Dolor.

LOREM IPSUM DOLOR.

4. 서체(font)와 글(text)스타일

11 수직 정렬(Vertical Align)

‘vertical align’ 속성은 인라인 요소끼리 위, 아래, 간격을 맞춤.

baseline, sub, super, top, text-top, middle, bottom, text-bottom

백분율(%)을 지정할 수 있고, 길이값을 적용 가능

vertical-align: baseline;	-> 글자의 baseline 위치에 맞춤
vertical-align: sub;	-> 부모요소의 아래 첨자 위치에
vertical-align: super;	-> 부모요소의 위 첨자 위치에 맞춤
vertical-align: top;	-> 부모요소의 상단에 맞춤
vertical-align: text-top;	-> 부모요소의 글꼴 요소의 상단 위치에 맞춤
vertical-align: middle;	-> 부모요소의 소문자를 기준으로 중간에 맞춤
vertical-align: bottom;	-> 부모요소의 아래쪽에 맞춤
vertical-align: text-bottom;	-> 부모요소의 글꼴의 아래쪽에 맞춤

4. 서체(font)와 글(text)스타일

11 수직 정렬(Vertical Align)

‘vertical align’ 속성은 인라인 요소끼리 위, 아래, 간격을 맞춤.

baseline, sub, super, top, text-top, middle, bottom, text-bottom

백분율(%)을 지정할 수 있고, 길이값을 적용 가능

```
<style type="text/css">
body{
    font-family: arial, sans-serif;
    font-size: 20px;
}
.base{
    background-color: lime;
    font-size: 1.2em;
}
.vertical{
    background-color: orange;
    font-size: 0.6em;
    vertical-align: baseline;
}
</style>
</head>
<body>

<h1>
    <span class="base">Lorem ipsum dolor
    <span class="vertical">Lorem.</span>
</h1>
```

sum dolor.

Lorem.

baseline

sum dolor.

Lorem.

sub

sum dolor.

Lorem.

super

sum dolor.

Lorem.

top

4. 서체(font)와 글(text)스타일

```
<style type="text/css">
body{
    font-family: arial, sans-serif;
    font-size: 20px;
}
.base{
    background-color: lime;
    font-size: 1.2em;
}
.vertical{
    background-color: orange;
    font-size: 0.6em;
    vertical-align: baseline;
}
</style>
</head>
<body>

<h1>
    <span class="base">Lorem ipsum dolor
    <span class="vertical">Lorem.</span>
</h1>
```

psum dolor.

Lorem.

text-top

psum dolor.

Lorem.

middle

psum dolor.

Lorem.

bottom

psum dolor.

Lorem.

text-bottom

4. 서체(font)와 글(text)스타일

정밀한 위치를 지정할 때는 px 값을 지정.

Lorem ipsum dolor.

Lorem.

vertical-align:0;

Lorem ipsum dolor.

Lorem.

vertical-align:20px;

4. 서체(font)와 글(text)스타일

 요소는 블록 요소 안에 있을 경우 약간의 **공백**이 발생하는 경우가 있다.
이때 '**vertical-align**' 속성을 활용해서 공백을 없앨 수 있다.

```
h1{
    border: 1px solid;
}

</style>
</head>
<body>

<h1>
    
</h1>
```



```
h1{
    border: 1px solid;
}

img{
    vertical-align: top;
}

</style>
</head>
<body>

<h1>
    
</h1>
```



12 글의 줄바꿈 처리

인라인 요소인 '글(text)'은 자신을 감싸고 있는 상위 블록 요소의 'width (너비)'를 넘어서게 되면 줄이 바뀐다.

① 단어를 깨뜨려 줄바꿈을 지정하는 '**word-break**' 속성

기본 줄바꿈 기준은 단어 단위이므로, 긴 단어가 있으면 부자연스럽다.

'**word-break**' 와 '**text-align:justify**'와 같이 사용하면 좋은 정렬 상태 유지 가능.

```
<style>
body{
    width: 400px;
    font-family: '돋움',dotum,helvet;
    font-size: 12px;
    text-align: justify;
}
.area{
    background-color: #ddd;
    word-break: break-all;
}
</style>
</head>
<body>
<p class="area">Lorem ipsum dolor sit an
adipisci saepe nemo excepturi magnam ut
<p>Lorem ipsum dolor sit amet, consectet
nemo excepturi magnam ut atque harum san
</body>
</html>
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Nam, quia voluptas adipisci saepe nemo excepturi magnam ut atque harum sampletextsampletext.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Nam, quia voluptas adipisci saepe nemo excepturi magnam ut atque harum sampletextsampletextsampletext.

4. 서체(font)와 글(text)스타일

- ② 줄바꿈을 금지하거나 <pre>요소의 특성을 부여하는 '**white-space**' 속성
작성한 **그대로** 줄바꿈을 보이게 만들어 준다.
'no-wrap', pre-wrap, pre-line 등을 지정할 수 있다.

속성값	효과
nowrap	줄 바꿈 금지
pre	<pre> 요소를 지정한 것처럼 띄어쓰기, 줄바꿈이 '작성한 그대로' 표현
pre-wrap	앞의 pre와 비슷하지만, 지정한 영역을 넘어가지 않음.
pre-line	'pre-wrap'과 비슷하지만, 띄어쓰기 한 공백은 한 칸만 표현

4. 서체(font)와 글(text)스타일

```
.nowrap{
    white-space: nowrap;
}
.pre{
    white-space: pre;
}
.pre-wrap{
    white-space: pre-wrap;
}
.pre-line{
    white-space: pre-line;
}
</style>
</head>
<body>
```

```
<p class="area nowrap">Lorem ipsum
quisquam!</p>
<p class="area pre">
    Lorem ipsum dolor sit ame
    Totam,
    doloremque,
    provident tenetur reprehe
</p>
<p class="area pre-wrap">
    Lorem ipsum dolor sit ame
    Totam,
    doloremque,
    provident tenetur reprehe
</p>
<p class="area pre-line">
    Lorem ipsum dolor sit ame
    Totam,
    doloremque,
    provident tenetur reprehe
</p>
```

1 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Totam, doloremque, provident te

nowrap

2 Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Totam,
doloremque,
provident tenetur reprehenderit blanditiis quaerat consequuntur dolores quas officiis

pre

3 Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Totam,
doloremque,
provident tenetur reprehenderit blanditiis quaerat consequuntur
dolores quas officiis quisquam!

pre-wrap

4 Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Totam,
doloremque,
provident tenetur reprehenderit blanditiis quaerat consequuntur dolores
quas officiis quisquam!

pre-line

- ① 줄바꿈 금지
- ② 작성한 그대로 (영역 넘어감)
- ③ 작성한 그대로 (영역 내)
- ④ 띄어쓰기 한 공백은 한칸만

Review TEST

본문 폰트: 돋움체, 크기 12px
로고부분 이미지로 교체, 텍스트는 화면에 나타나지 않도록 한다.
문단 크기 400px, 문단 들여쓰기 1em, 1px 테두리

구분선 이하 **문단**
글자단위로 자연스럽게 모두 채워지는 문단으로 설정.

036_test.html

LOGO

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Odit, laboriosam, ut, iste, adipisci a quo illo aliquid necessitatibus fuga tempora est animi quas ab quibusdam aut deleniti saepe odio sunt.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Nam, quia voluptas adipisci saepe nemo excepturi magnam ut atque harum sampletextsampletext.

5. 목록(List) 스타일

5. 목록(List) 스타일

1 list-style-type

목록 ,

<1> Ul(unordered list) 지정

```
ul li:nth-child(1){
    list-style-type: disc;
}
ul li:nth-child(2){
    list-style-type: circle;
}
ul li:nth-child(3){
    list-style-type: square;
}
</style>
</head>
<body>

<ul class="nav">
    <li>disc</li>
    <li>circle</li>
    <li>square</li>
</ul>
```

- disc
- circle
- square

블릿(bullet) 지정 대상은 개별 항목인 보다는 ,요소에 지정하는 것이 좋다. 이와 같은 블릿 타입들은 각 항목을 꾸미기 위한 것이기보다 항목을 그룹화하려는 목적이 더 크기 때문.

5. 목록(List) 스타일

메인메뉴와 서브메뉴의 블릿을 다르게 지정한 예.

```
ul.nav{
    list-style-type: disc;
}
ul.nav ul{
    list-style-type: square;
}
</style>
</head>
<body>

<ul class="nav">
    <li><a href="#">About</a>
        <ul>
            <li><a href="#">Message</a></li>
            <li><a href="#">Location</a></li>
        </ul>
    </li>
    <li><a href="#">Services</a></li>
    <li><a href="#">Portfolio</a></li>
    <li><a href="#">Contact</a></li>
</ul>
```

- [About](#)
 - [Message](#)
 - [Location](#)
- [Services](#)
- [Portfolio](#)
- [Contact](#)

5. 목록(List) 스타일

<2> ol(ordered list) 타입 지정

 요소의 기본값은 10진수로 적용.

CSS로는 , 관계없이 적용가능. 소수, 로마숫자, 그리스, 알파벳, 아르메니아, 그르지아 넘버링

```
list-style-type: decimal;
list-style-type: decimal-leading-zero;
list-style-type: lower-roman;
list-style-type: upper-roman;
list-style-type: lower-greek;
list-style-type: lower-alpha;
list-style-type: upper-alpha;
list-style-type: armenian;
list-style-type: georgian;
```

예제에서는 요소에 각각 적용했지만,
 요소에 지정해서 하나의 목록 그룹에서
같은 형식으로 지정하는 것이 바람직.

1. decimal
02. decimal-leading-zero
- iii. lower-roman
- IV. upper-roman
- ε. lower-greek
- f. lower-alpha | lower-latin
- G. upper-alpha | upper-latin
- Լ. armenian
- თ. georgian

css2 에서 추가된 값

- א. hebrew
- 二. cjk-ideographic
- う. hiragana
- 工. katakana
- ほ. hiragana-iroha
- へ. katakana-iroha

Mozilla의 확장

- 가. hangul
- ㄴ. hangul-consonant
- 丙. cjk-heavenly-stem

5. 목록(List) 스타일

2 list-style-position으로 목록 위치 지정

목록이 지정된 블록 영역의 '안쪽, 바깥쪽'에 블릿 표현

```
.inside{
    list-style-position: inside;
}
.outside{
    list-style-position: outside;
}
</style>
</head>
<body>

<p>inside</p>
<ul class="inside">
    <li>list item 1</li>
    <li>list item 2</li>
    <li>list item 3</li>
</ul>

<p>outside</p>
<ul class="outside">
    <li>list item 1</li>
    <li>list item 2</li>
    <li>list item 3</li>
</ul>
```

inside

- list item 1
- list item 2
- list item 3

outside

- list item 1
- list item 2
- list item 3

5. 목록(List) 스타일

3 list-style-image로 이미지 블릿 지정

list-style-image 속성은 블릿을 이미지로 표현할 수 있는데, 이미지의 위치를 섬세하게 지정할 수 없어 실무에서는 'background' 속성을 활용하는 편.

```
.list{
    list-style-image: url(img/bull.png);
}
</style>
</head>
<body>

<p>list-style-image</p>
<ul class="list">
    <li>Lorem ipsum dolor sit amet, consectetur.</li>
    <li>Lorem ipsum dolor sit amet, consectetur.</li>
    <li>Lorem ipsum dolor sit amet, consectetur.</li>
</ul>
```

list-style-image

- ★ Lorem ipsum dolor sit amet, consectetur.
- ★ Lorem ipsum dolor sit amet, consectetur.
- ★ Lorem ipsum dolor sit amet, consectetur.

6. Display 속성과 Visibility 속성

1 Display 속성

display 속성은 CSS의 많은 속성 중에서도 **매우 중요**

각 HTML 요소는 블록 혹은 인라인을 나타나게 되는데, display 속성으로 **'block'**, 혹은 **'inline'**을 의도적으로 지정 가능. 또한 원래 존재하지 않는 것처럼 **'none'**을 지정가능.

(※ ie의 경우 ie9이상에서 안정적으로 지원)

<1> display 속성으로 block, inline 값 지정하기

HTML에서 **<div>**는 블록요소, ****은 인라인 요소,

다음예제에서는 **<div>**요소가 인라인으로, **** 요소가 블록으로 표현

width 속성은 블록 요소에만 적용되므로 **'display:block'**을 지정한 요소에 **크기**가 지정되고 있다.

```
<style type="text/css">
.block{
    display: inline;
    width: 300px;
    background-color: lime;
}
.inline{
    display: block;
    width: 300px;
    background-color: silver;
}
</style>
</head>
<body>

<div class="block">div : block elements</div>
<span class="inline">span : inline elements</span>
```

div : block elements
span : inline elements

6. Display 속성과 Visibility 속성

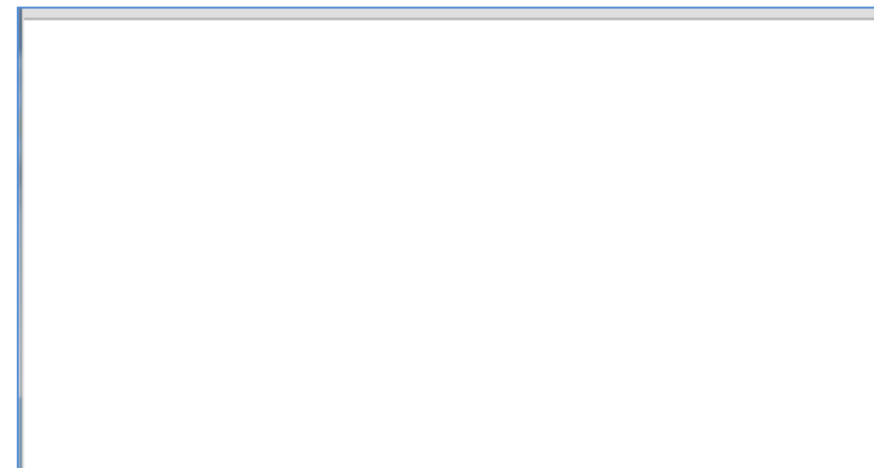
<2> display 속성 none

none으로 지정하면 브라우저에서는 원래 없었던 것으로 간주.
웹접근성 차원에서는 주의가 필요.

```
<style type="text/css">
.none{
    display: none;
}
</style>
</head>
<body>

<p class="none">Lorem ipsum dolor s

</body>
```



6. Display 속성과 Visibility 속성

주요 태그별 display 속성

- 기본값은 직접 CSS코드에서 'display:inline' 또는 'display:block'으로 변경 가능

태그	속성	태그	속성
<a>	inline	<div>	block
	inline	<form>	block
	inline	<h1> ~ <h6>	block
<input>	inline	, , 	block
		<p>, <pre> ,	block
		<table>	block

<3> 'inline-block' 인라인과 블록 속성 모두 가짐

CSS2 버전에서 추가된 'inline-block'은 인라인과 블록 특징 모두 가짐.

인라인 요소는 크기를 가질 수 없다.

블록 요소는 텍스트 정렬이 안 된다.

하지만, 'inline-block'이 지정되면 크기지정 가능하고, 상위에서 'text-align'속성으로 게시판 하단의 페이지징(1,2,3,4,..) 부분을 꾸미는데 유용하다.

```
.paging{
    text-align: center;
}
.paging a{
    display: inline-block;
    padding: 5px 8px;
    border: 1px solid #ccc;
    background-color: beige;
    font-weight: bold;
}
</style>
</head>
<body>

<p class="paging">
    <a href="#">1</a>
    <a href="#">2</a>
    <a href="#">3</a>
    <a href="#">4</a>
    <a href="#">5</a>
</p>
```



<4> 목록을 표현하는 'list-item'

'list-item'을 값으로 지정하면, 해당요소가 목록의 형태로 나타남.
지정하는 요소가 인라인 요소라도 **블록형태**로 나타남.

```
.area{
    width: 200px;
    margin: 0 auto;
    border: 1px solid;
}
.list {
    display: list-item;
}
</style>
</head>
<body>

<div class="area">
    <span class="list">list item</span>
    <span class="list">list item</span>
    <span class="list">list item</span>
    <span class="list">list item</span>
</div>
```

- 
- list item
 - list item
 - list item
 - list item

<5> 테이블의 형태를 표현하는 다양한 값 지원

테이블 형태를 표현하기 위한

'table, table-caption, table-cell, table-column, table-column-group, table-footer-group, table-header-group, table-row, table-row-group

<table>요소를 쓰지 않고 CSS로 테이블 표현

```
<style type="text/css">
.table{
    display: table;
    border: 1px solid;
}
.table-row{
    width: 400px;
    border-bottom: 1px solid gray;
}
.cell_no{
    display: table-cell;
    width: 40px;
    font-weight: bold;
    text-align: center;
}
.cell_tle{
    display: table-cell;
    width: 260px;
}
.cell_date{
    display: table-cell;
    width: 100px;
    text-align: center;
}
</style>
```

```
<div class="table">
    <div class="table-row-group">
        <div class="table-row">
            <div class="cell_no">01</div>
            <div class="cell_tle">Lorem ipsum dolor
            <div class="cell_date">2013-08-10</div>
        </div>
        <div class="table-row">
            <div class="cell_no">02</div>
            <div class="cell_tle">Lorem ipsum dolor
            <div class="cell_date">2013-08-10</div>
        </div>
    </div>
</div>
```

01	Lorem ipsum dolor sit amet.	2013-08-10
02	Lorem ipsum dolor sit amet.	2013-08-10

2 visibility 속성

기본값은 'visible', 'hidden' 지정할 경우 보이지 않게함
'display:none' = 원래 없었던 것으로 인식
'visibility: hidden'의 경우에는 보이지만 않고, 공간은 유지한다.

```
<style type="text/css">
*{
    margin: 0;
    padding: 0;
}
.visible{
    visibility: hidden;
}
</style>
</head>
<body>

<p>Paragraph Line 1</p>
<p class="visible">Paragraph Line 2</p>
<p>Paragraph Line 3</p>

</body>
</html>
```

Paragraph Line 1

Paragraph Line 3

3 overflow

- block요소의 자식 콘텐츠가 부모요소의 사이즈를 변화시킬 때 숨김 / 스크롤바 표시여부를 지정
- 부모의 사이즈가 지정 되어 있을 경우에 사용가능

```
overflow : visible ; (기본)  
overflow : hidden ;  
overflow : scroll ;  
overflow : auto ;
```

overflow : scroll 가로, 세로에 대한 속성을 따로 지정 할 수 있다.
overflow-x
overflow-y

7. 배경(background) 스타일

background-color (색상)
background-repeat (반복지정)
background-position(위치)

background-image (경로)
background-attachment(고정)

올바른 표현 순서. ※ 유사한 대부분의 CSS 속성에도 적용되는 순서.

```
background: url(img/bg.gif) no-repeat 0 0;  
background-color: red;
```

1 배경 색상 지정

```
.bg{ background-color: lime; }  
.bg{ background-color: rgba(0,255,0,0.4); }  
.bg{ opacity: 0.4; }
```

alpha값은 0~ 1.0까지 지정 가능하고, opacity로도 투명도를 지정하는데 마지막의 경우는 배경뿐 아니라 요소 전체에 투명도가 지정됨.

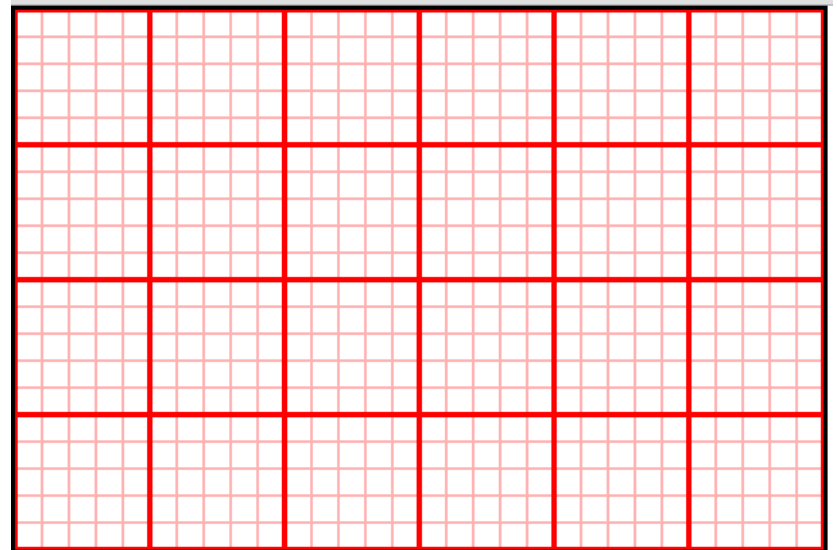
2 배경 이미지 지정과 관련 스타일들

<1> 괄호안에 이미지의 경로 지정

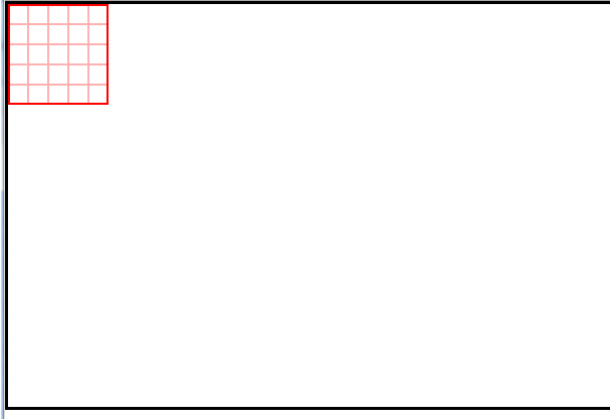
```
background-image: url(img/bg.png);
```

<2> 지정된 이미지의 반복 여부 설정. 기본값은 반복

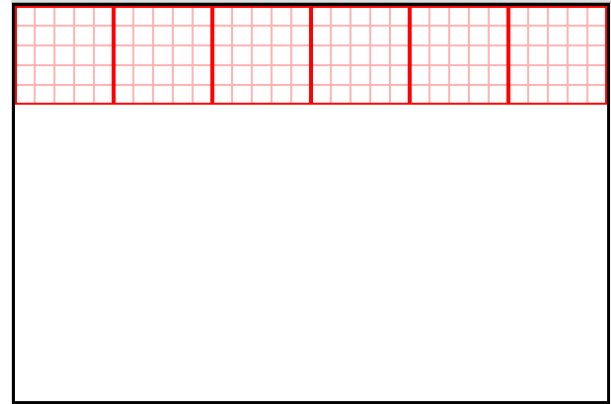
② background-repeat: repeat;



② background-repeat: no-repeat;



③ background-repeat: repeat-x;



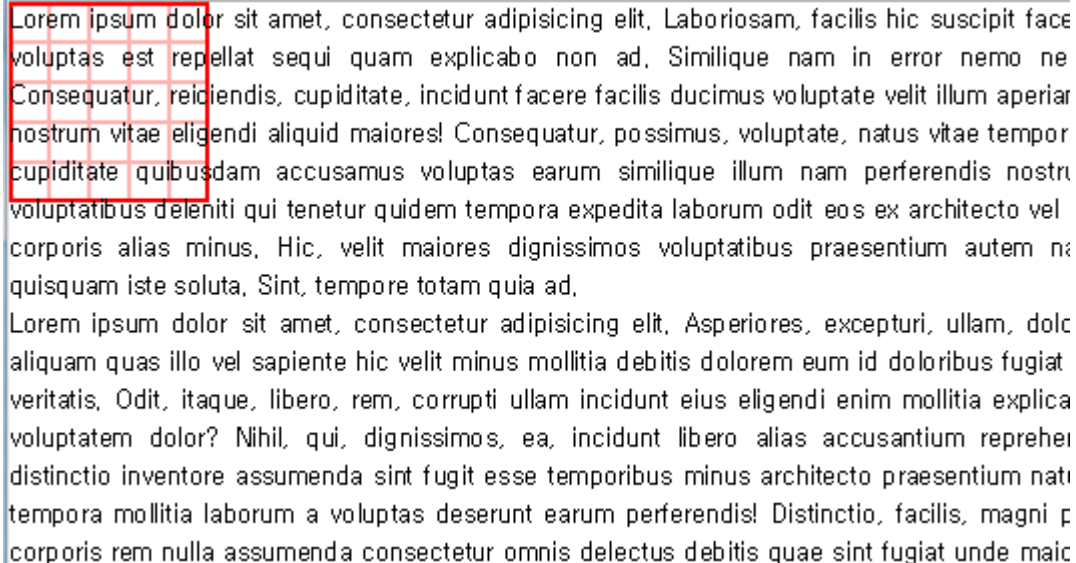
background-repeat: repeat-y;



<3> 지정된 배경 이미지가 뷰포트 내에서 고정되는지의 여부 지정. 고정은 fixed.

background-repeat: no-repeat;
background-attachment: fixed;

background-attachment: scroll;
(내용이 스크롤 될 때 같이 움직임)



Lorem ipsum dolor sit amet, consectetur adipiscing elit, Laboriosam, facilis hic suscipit face voluptas est repellat sequi quam explicabo non ad, Similique nam in error nemo ne Consequatur, reiciendis, cupiditate, incidunt facere facilis ducimus voluptate velit illum aperiar nostrum vitae eligendi aliquid maiores! Consequatur, possimus, voluptate, natus vitae tempor cupiditate quibusdam accusamus voluptas earum similique illum nam preferendis nostri voluptatibus deleniti qui tenetur quidem tempora expedita laborum odit eos ex architecto vel corporis alias minus, Hic, velit maiores dignissimos voluptatibus praesentium autem n quisquam iste soluta, Sint, tempore totam quia ad, Lorem ipsum dolor sit amet, consectetur adipiscing elit, Asperiores, excepturi, ullam, dolc aliquam quas illo vel sapiente hic velit minus mollitia debitis dolorem eum id doloribus fugiat veritatis, Odit, itaque, libero, rem, corrupti ullam incidunt eius eligendi enim mollitia explic voluptatem dolor? Nihil, qui, dignissimos, ea, incidunt libero alias accusantium repreh distinctio inventore assumenda sint fugit esse temporibus minus architecto praesentium nati tempora mollitia laborum a voluptas deserunt earum preferendis! Distinctio, facilis, magni p corporis rem nulla assumenda consectetur omnis delectus debitis quae sint fugiat unde maic

<4> 배경의 위치 지정

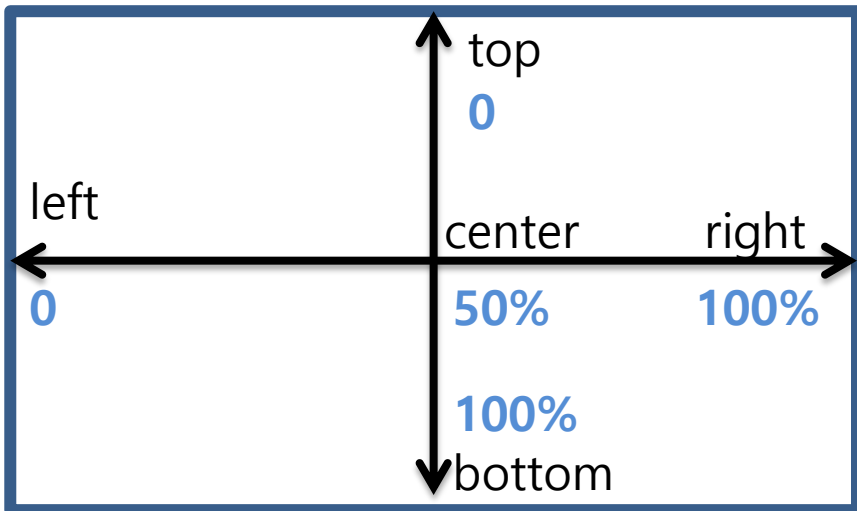
background-position: left top;

앞부분이 '가로' 기준, 뒷부분이 세로기준.

background-position: top = background-position: center top

background-position: 20px = background-position: 20px center

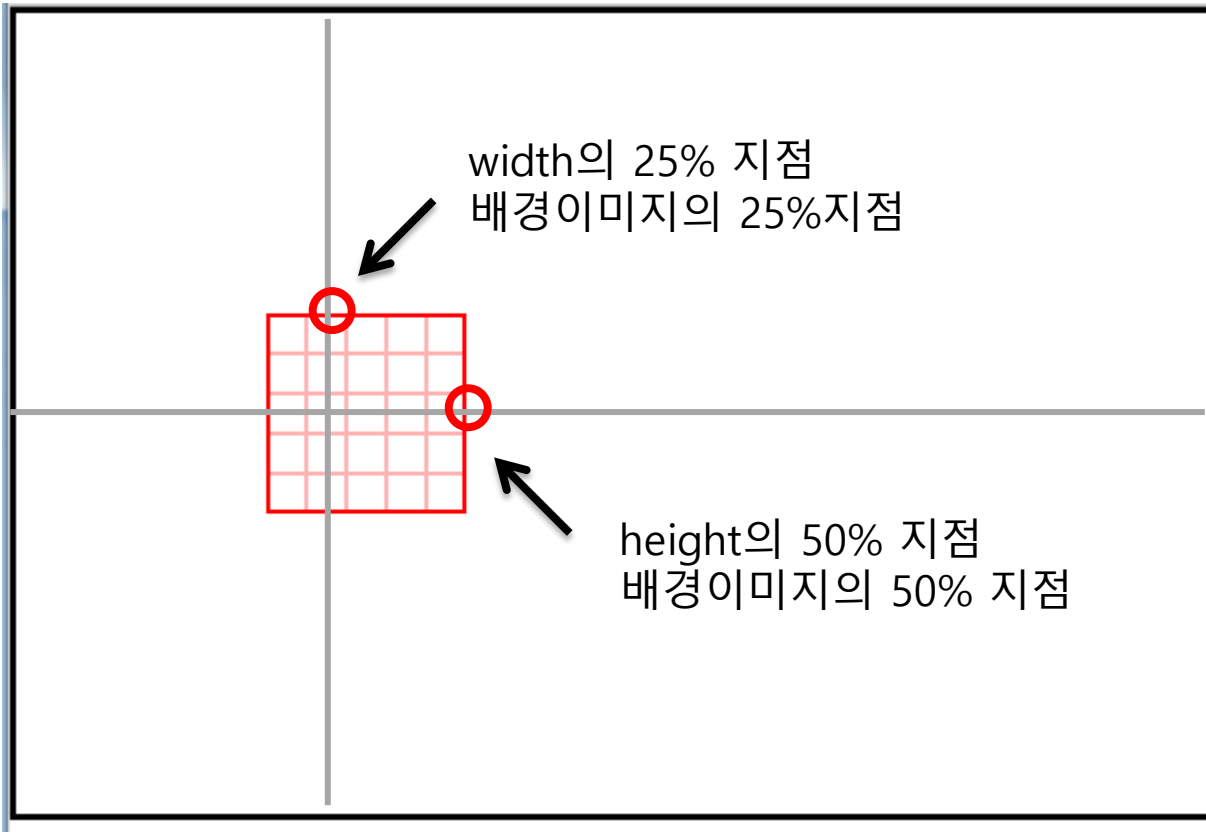
width 기준의 값으로는 'left, center, right' / 세로 기준의 값으로는 top, center, bottom.
%로 지정하면 0, 50%, 100%



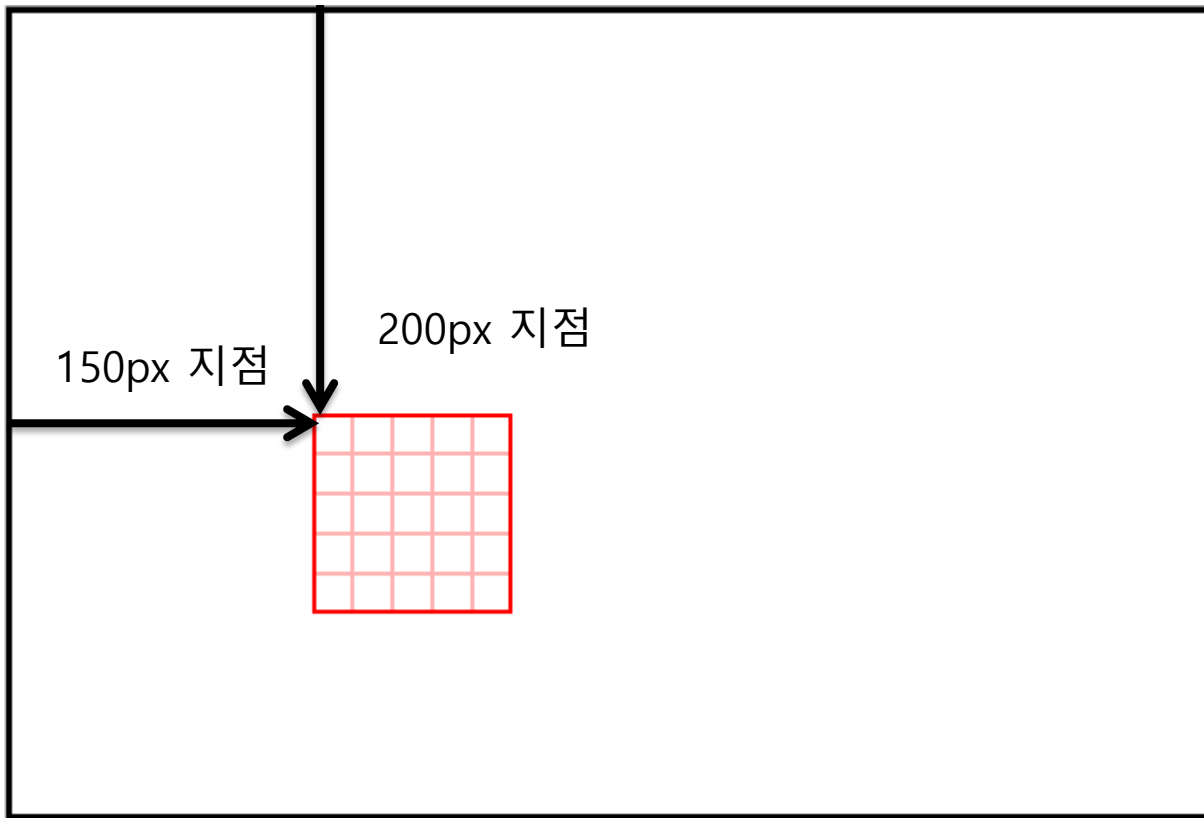
%로 값을 지정하면 해당영역의 %지점에
배경이미지의 **%지점**이 위치함.

px로 값을 지정하면 해당영역의 px 값 위치가
배경이미지의 **시작지점**

background-position: 25% 50%;



```
background-position: 150px 200px;
```

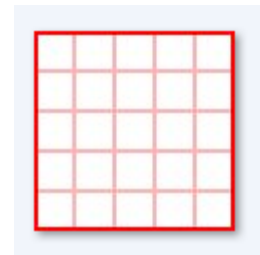
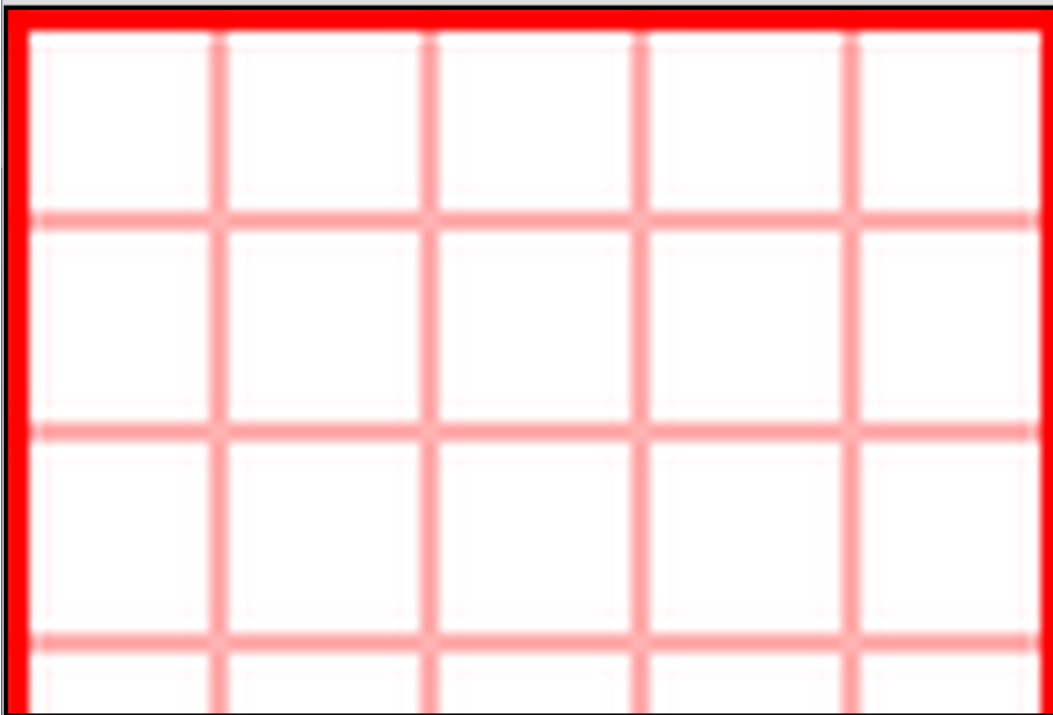


3 새로운 배경 스타일들

<1> 배경 이미지의 크기조절

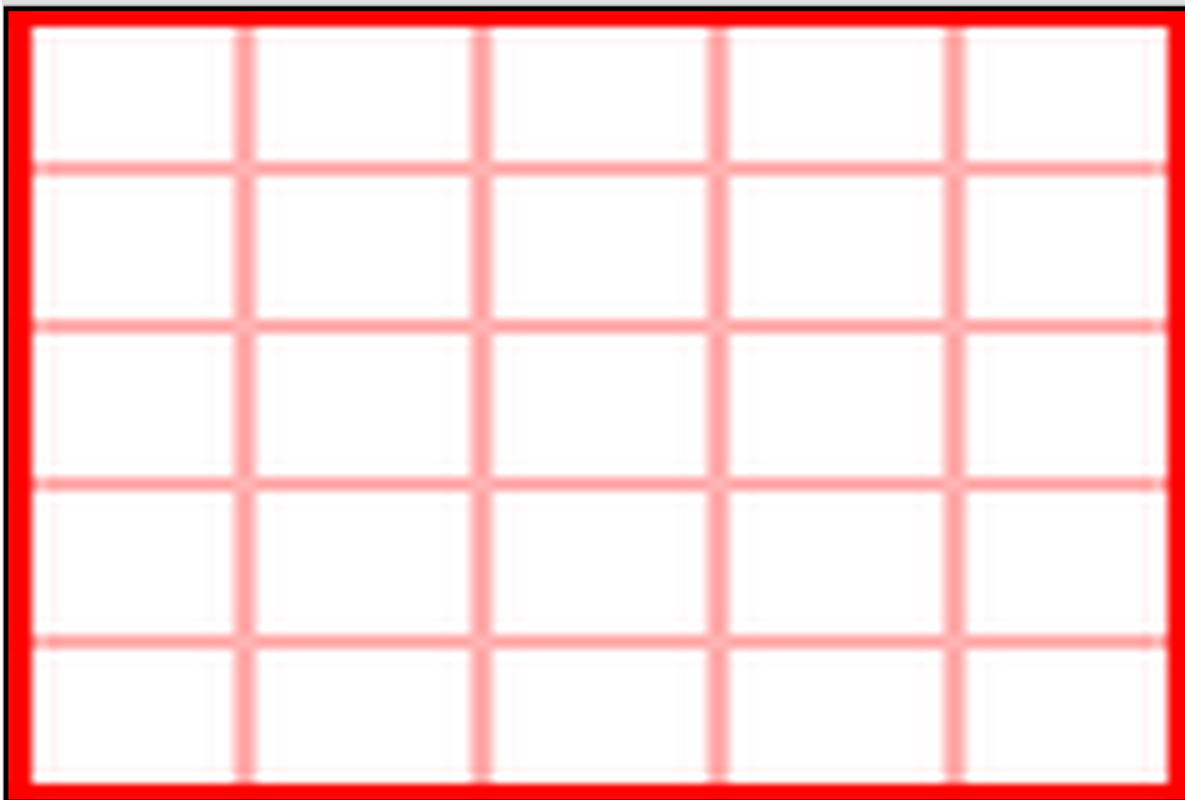
값을 하나만 입력하면 width 값이 되고, height는 이미지의 비율에 따라 나타난다.
둘을 입력해서 width와 height를 각각 지정 가능.

```
background-size: 100%;
```



100x100px

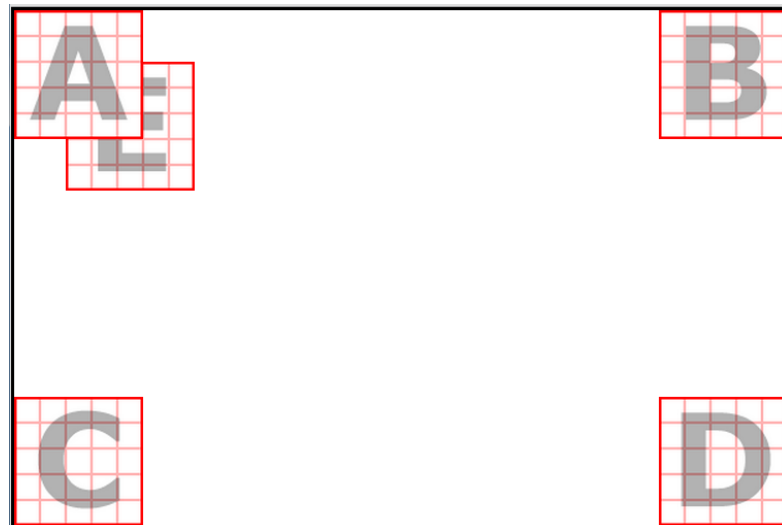

```
background-size: 100% 100%;
```



<2> 한 요소에 여러 개의 배경 이미지 적용

CSS3에서는 여러 개의 배경을 적용가능, 'background' 속성을 사용하며 쉼표(,)로 추가

```
<style type="text/css">
*{
    margin: 0;
    padding: 0;
}
.bg{
    width: 600px;
    height: 400px;
    border: 3px solid;
    background:
        url(img/bg_a.png) no-repeat 0 0,
        url(img/bg_b.png) no-repeat 100% 0,
        url(img/bg_c.png) no-repeat 0 100%,
        url(img/bg_d.png) no-repeat 100% 100%,
        url(img/bg_e.png) no-repeat 40px 40px;
}
</style>
```

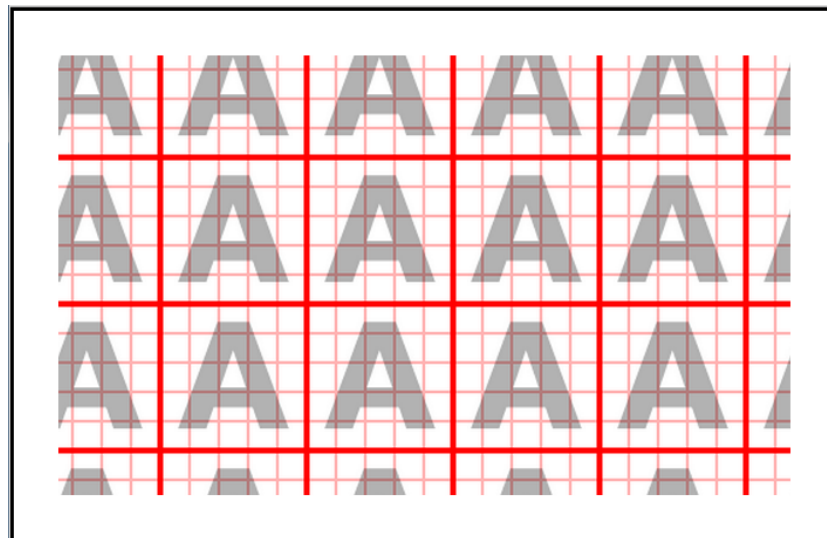


‘먼저 작성한 이미지가 위에’ 표시된다.

<3> 배경 이미지 클리핑하기

배경을 content 영역에만 보이게 하는 속성 '**background-clip**'
원래 padding으로 만들어진 여백 부분까지 배경이 나타나게 되지만, 이 속성으로 padding 영역의 배경을 보이지 않게 지정 가능.

```
<style type="text/css">
*{
    margin: 0;
    padding: 0;
}
.bg{
    width: 500px;
    height: 300px;
    border: 3px solid;
    padding: 30px;
    background: url(img/bg_a.png) repeat;
    background-clip: content-box;
}
</style>
```

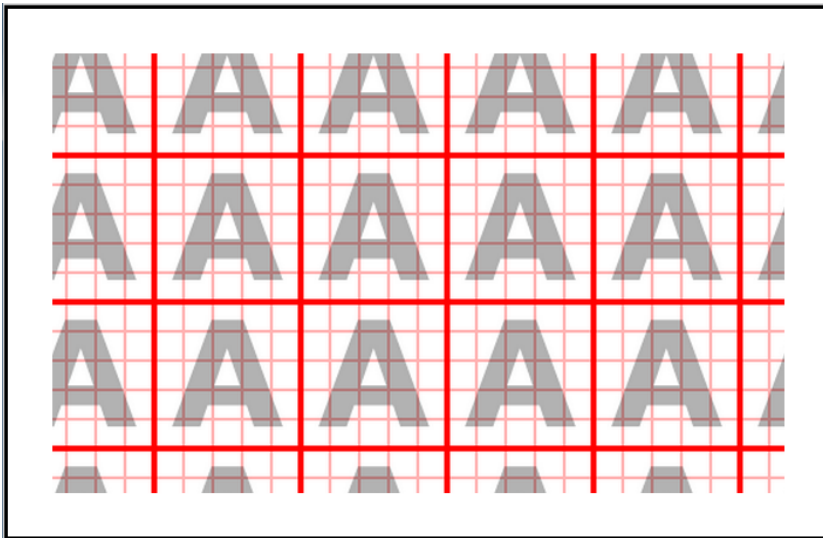


<4> 배경 영역 보정

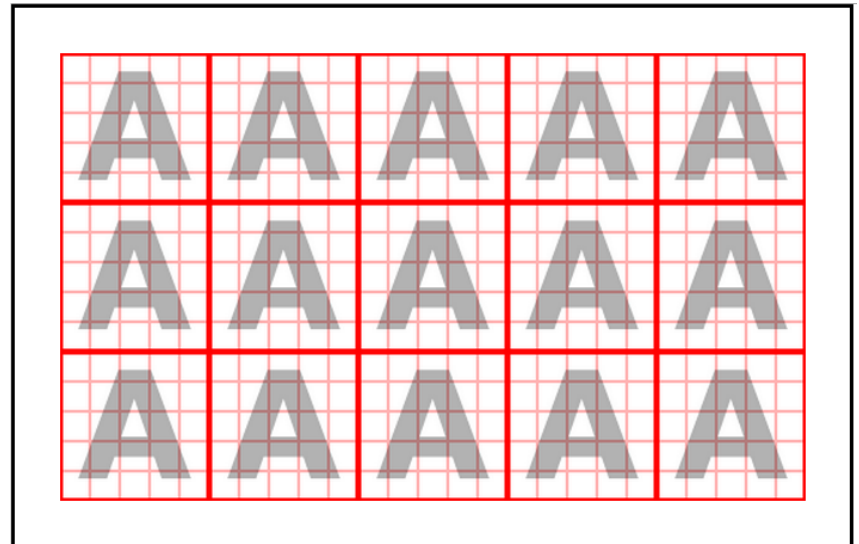
'background-origin'

background-clip: content-box 로 지정할 경우, padding 영역에 있는 배경을 잘라 보이게 되는데, 'background-origin'을 지정하면 배경의 시작되는 위치를 content-box에 맞출수 있다.

```
padding: 30px;  
background-clip: content-box;
```



```
padding: 30px;  
background-clip: content-box;  
background-origin: content-box;
```



Review TEST

00_REVIEW_02_HTML&CSS

CSS LAYOUT (background-position, position)

01_background-position

Background_position_01 <http://codepen.io/alikerock/pen/GZYRbP>

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s



Background_position_03 (image replacement)

<http://codepen.io/alikerock/pen/EgxjVa>



웹 표준은 월드 와이드 웹의 측면을 서술하고 정의하는 공식 표준이나 다른 기술 규격을 가리키는 일반적인 용어이다. 최근에 이 용어는 웹 사이트를 작성하는 데 중요도가 높아지고 있으며 웹 디자인, 개발과 관계가 있다.

Background_position_02 <http://codepen.io/alikerock/pen/ZWqYzB>



Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s

Sprite BG Base

스타	프로그램	이슈
스타	프로그램	이슈