

Hello World!  
{ css; } 세상에 오신 것을 환영합니다.



GuideBook (가이드북)

# { css; } - 목차



- **텍스트 서식 스타일**

- ✓ css 기본 문법 / css 링크 방법 / css 선택자 / 적용 우선순위
- ✓ css 텍스트 관련 스타일 / 목록 스타일 / 파비콘 / 구글 웹폰트 / 폰트어썸

- **CSS Position 레이아웃**

- ✓ 레이아웃 스타일 (display 속성 : inline 요소, block 요소)
- ✓ 박스모델
- ✓ float / clear / z-index / visibility / opacity / display

- **배경 색상, 배경 이미지**

- ✓ css 배경 색상 및 이미지 제어 / css 배경 그라디언트

# { css; } - 목차



- **변형 & 애니메이션**
  - ✓ transform: translate/scale/rotate/origin/skew
- **CSS 가상클래스**
  - ✓ 마우스 오버 가상 클래스 :hover
  - ✓ :first-child, :last-child, :nth-child, :nth-of-type
- **CSS 고급선택자**
  - ✓ 인접 선택자, 속성 선택자, 형제 선택자
  - ✓ 가상 클래스 :before, :after

# { css; } - 목차



- calc() 연산함수
- Inline-block 사이의 마진 처리
- img 태그 사이의 마진 처리
- Inline-block 요소끼리 수직중앙 처리
- body, div 높이 100% 처리
- 인라인 요소와 블럭요소 수형 중앙 정렬
- 부모요소 기준 HTML 요소 수평 수직 중앙 정렬

# { css; } - 기본문법



selector(선택자)

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

property(속성) value(값)

```
h1 { font-size : 14px; color : blue; }
```

선택자

속성

값

속성

값

# { css; } - 링크방법



```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>CSS Link</title>
    <link rel="stylesheet" href="main.css">          외부 스타일
    <style>
        body{
            font-size: 14px;
            background-color: beige;                  내부 스타일
        }
        .item{ margin: 0; }
    </style>
</head>
<body> <p class="item">내용</p> </body>
</html>
```

# { css; } - 선택자



## tag 선택자

```
p {  
    text-align: center;  
    color: red;  
}
```

## class 선택자

```
.item {  
    text-align: center;  
    color: red;  
}
```

## id 선택자

```
#item {  
    text-align: center;  
    color: red;  
}
```

## tag 와 함께 사용하는 선택자

```
p.item {  
    text-align: center;  
    color: red;  
}
```

- ① <p>내용</p>
- ② <p class="item">내용</p>
- ③ <p id="item">내용</p>
- ④ <div class="item">내용</p>

# { css; } - 선택자



## 하위 선택자

```
div span {  
    color: red;  
}
```

## 자식 선택자

```
div > span {  
    color: blue;  
}
```

## 그룹 선택자

```
h1, p span {  
    color: red;  
}
```

## 전체 선택자

```
* {  
    font-size: 10px; /* 모든 text에 적용 */  
}
```

<h1>css selector(선택자)</p>

<div>

    <span>css에서 HTML element 선택하는 방법입니다.</span>

선택을 통해 특정 element에 스타일을 적용할 수 있습니다.

    <p>

        선택자를 잘 사용하면 <span>스타일시트(Style Sheet)</span>를 관리하기 편해집니다.

    </p>

</div>

# { css; } - 적용 우선순위



1. !important 스타일
2. inline 스타일
3. id selector 스타일
4. class selector 스타일
5. tag selector 스타일

동일한 CSS 속성이 중복 사용될 경우 왼쪽의 우선 순위에 따라 적용됩니다.

1. 내부 스타일
2. 외부 파일 스타일

동일한 CSS 속성이 내부, 외부 둘 다 있으면 왼쪽과 같이 적용됩니다.

# { css; } - 폰트 사이즈



고정 (pixels, points) / 상대적 (em, percent)

pixels	em	percent	points
10px	0.625em	62.5%	8pt
11px	0.688em	68.8%	8pt
12px	0.75em	75%	9pt
13px	0.813em	81.3%	10pt
14px	0.875em	87.5%	11pt
15px	0.938em	93.8%	11pt
16px	1em	100%	12pt
17px	1.063em	106.3%	13pt
18px	1.125em	112.5%	14pt
19px	1.188em	118.8%	14pt
20px	1.25em	125%	15pt

- 일반적으로 pixels를 사용함
- 별다른 font-size 정의가 없으면 기본 폰트 사이즈는 16px로 적용됨

# { css; } - 텍스트 서식 1



- **font-size** : 텍스트의 사이즈를 지정하는 속성  
값 : px, em, %, pt, rem
- **font-weight** : 글꼴의 두께를 지정하는 속성  
값 : 100~900, bold, bolder, lighter, normal (400=normal, 700=bold)
- **font-family** : 글꼴을 지정하는 속성  
값 : 'font-name' ('sans-serif etc')
- **font-style** : 글꼴의 스타일(기울림체) 지정하는 속성  
값 : normal, italic, oblique
- **line-height** : 줄 간격을 지정하는 속성  
값 : px, em, %, pt, rem

# { css; } - 텍스트 서식 2



- **color** : 글꼴 색상을 지정하는 속성  
값 : color name, #000000, rgb(0,0,0), rgba(0,0,0,0.5)
- **text-decoration** : 텍스트 줄 표시 / 제거 여부 설정  
값 : none, underline, overline, line-through
- **text-transform** : 텍스트를 대문자 및 소문자로 변환  
값 : none, capitalize, uppercase, lowercase
- **text-shadow** : 텍스트 그림자 효과  
값1 : [x축거리] [y축거리] [흐림정보] [색상]  
값2 : none
- **text-align** : 문자 정렬 방법을 지정하는 속성  
값 : left(기본값). center, right, justify

# { css; } - 텍스트 서식 3



- direction : 쓰기 방향 지정하는 속성  
값 : ltr, rtl
- text-indent : 텍스트 들여쓰기  
값 : px, em, %, pt, rem
- white-space : 자동줄바꿈 속성  
값 : normal(기본값), nowrap(줄바꿈 안함)
- overflow : 자동줄바꿈 하지 않는 경우 넘친 텍스트 감추기 속성  
값 : hidden
- text-overflow : 넘친 텍스트 감춘 부분을 어떻게 처리할 지 결정하는 속성  
값 : clip(기본값), ellipsis(… 표기)

# { css; } - 목록 스타일



Value	example
disc	●
circle	○
square	■
none	

Value	example
decimal	1,2,3,... 10,11
decimal-leading-zero	01,02,03... 10,11
lower-roman	i,iii,iii,iv
upper-roman	I,II,III,IV
lower-alpha	a,b,c,d,e
upper-alpha	A,B,C,D,E

```
ul {  
    list-style : square; /* 왼쪽표의 value */  
}
```

```
ol {  
    list-style : lower-alpha;  
/*  
    ul의 value를 ol에 사용해도 무방합니다.  
    list-style : circle;  
*/  
}
```

# { css; } - 목록 스타일



## 2020년 국내 게임업체

- 넥슨
- 넷마블
- 엔씨소프트
- NHN
- 컴투스

## 2020년 세계 게임업체

- i. Tensest
- ii. NetEase
- iii. Activision Blizzard
- iv. Supercell
- v. BANDAI NAMCO

### html

```
<h2>2020년 국내 게임업체</h2>
<ul>
    <li>넥슨</li>
    <li>넷마블</li>
    <li>엔씨소프트</li>
    <li>NHN</li>
    <li>컴투스</li>
</ul>
<br>
<h2>2020년 세계 게임업체</h2>
<ol>
    <li>Tensest</li>
    <li>NetEase</li>
    <li>Activision Blizzard</li>
    <li>Supercell</li>
    <li>BANDAI NAMCO</li>
</ol>
```

### CSS

```
ul{
    list-style: square;
}
ol{
    list-style: lower-roman;
}
```

# { css; } - 목록 스타일 (중첩)



## html

### 모바일

1. 제2의 나라
2. ROBLOX Roblox
3. 트릭스터M
4. 쿠키런: 킹덤
5. 바람의나라 연

### 온라인

1. 리그 오브 레전드
2. 피파온라인4
3. 로스트아크
4. 메이플스토리
5. 아이온

### 스팀

1. 플레이어언노운스 배틀그라운드
2. 어몽 어스
3. GTA 5
4. 잇 테이크 투
5. 스타듀밸리

```
<ul class="list">
  <li><b>모바일</b>
    <ol>
      <li>제2의 나라</li>
      <li>ROBLOX Roblox</li>
      <li>트릭스터M</li>
      <li>쿠키런: 킹덤</li>
      <li>바람의나라 연</li>
    </ol>
  </li>
  <li><b>온라인</b>
    <ul>
      <li>리그 오브 레전드</li>
      <li>피파온라인4</li>
      <li>로스트아크</li>
      <li>메이플스토리</li>
      <li>아이온</li>
    </ul>
  </li>
  <li><b>스팀</b>
    <ul>
      <li>플레이어언노운스 배틀그라운드</li>
      <li>어몽 어스</li>
      <li>GTA 5</li>
      <li>잇 테이크 투</li>
      <li>스타듀밸리</li>
    </ul>
  </li>
</ul>
```

## CSS

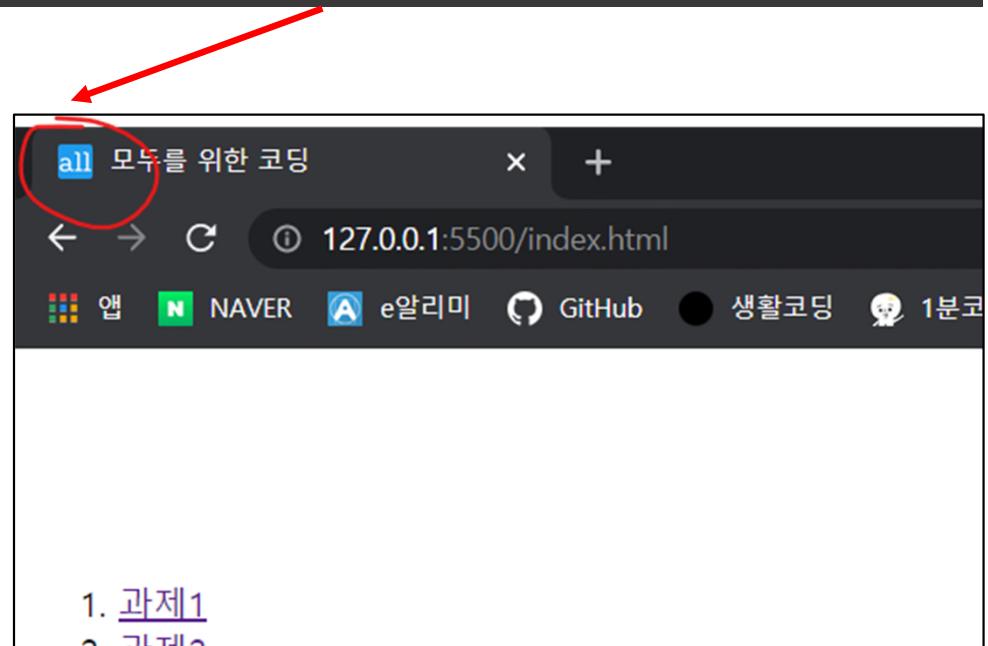
```
ul.list{
  padding: 0;
  line-height: 2em;
}
ul.list>li{
  list-style: none;
}
ul.list>li b{
  background-color: black;
  color: white;
  font-weight: normal;
}
ul.list>li>ul{
  list-style: decimal;
}
```

# { css; } - 파비콘



```
<head>
  <link rel="shortcut icon" type="image/x-icon" href="pavicon.ico">
  <link rel="icon" type="image/x-icon" href="pavicon.ico">
  <link rel="icon" type="image/png" sizes="32x32" href="favicon-32x32.png">
  <link rel="icon" type="image/png" sizes="16x16" href="favicon-16x16.png">
  <link rel="apple-touch-icon" sizes="180x180" href="apple-touch-icon.png">
</head>
```

- <https://favicon.io> 와 같은  
파비콘 무료 생성 페이지를 이용할  
수 있습니다.
- 다양한 디바이스에 맞춰서 위와  
같이 여러 케이스의 <link> 태그를  
사용하기도 합니다.



# { css; } - 구글 웹폰트



- 구글(<http://www.google.com>) 검색에서 “구글웹폰트” 검색  
or
- 구글 웹 폰트 공식사이트 바로가기 (<http://fonts.google.com>)

The screenshot shows the Google Fonts homepage. At the top, there's a search bar with 'Search' and a placeholder 'Type something'. Below it are filters for 'Categories' (with 'Korean' selected), 'Font properties', and a checkbox for 'Show only variable fonts'. A red circle highlights the 'Korean' filter button. The main area displays font families: Noto Sans KR (Google), Nanum Gothic (Sandoll), Nanum Fontrix, Gothic A1 (HanYang I&C Co), Noto Serif KR (Google), and Nanun Sandoll. Each family has a preview box showing Korean text. A red arrow points from the 'Nanum Gothic' preview box to the text '그들의 장비와 기구는 모두 살아 있다.'.

The screenshot shows the 'Nanum Gothic' font details page. It features the font's name, designer (Sandoll), and a 'Styles' section with 'Regular 400' and 'Bold 700'. Below each style is a preview of the text '그들의 장비와 기구는 모두 살아 있다.' in that specific weight. A red circle highlights the 'Regular 400' preview, and another red circle highlights the 'Select this style' link next to it. A red arrow points from the 'Regular 400' preview to the 'Select this style' link.

# { css; } - 구글 웹폰트



## 1. link 방식

The screenshot shows the Google Fonts interface for the Nanum Gothic font. It includes sections for 'Selected family', 'Review', 'Use on the web' (with options for <link> or @import), and 'Styles' (Regular 400, Bold 700, Extra-bold 800). The 'Use on the web' section highlights the <link> method, showing the code:

```
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Nanum+Gothic&display=swap" rel="stylesheet">
```

Below this, it shows the CSS rules to specify families:

```
font-family: 'Nanum Gothic', sans-serif;
```

## 2. import 방식

The screenshot shows the Google Fonts interface for the Nanum Gothic font. It includes sections for 'Selected family', 'Review', 'Use on the web' (with options for <link> or @import), and 'Styles' (Regular 400, Bold 700, Extra-bold 800). The 'Use on the web' section highlights the @import method, showing the code:

```
<style>
@import url('https://fonts.googleapis.com/css2?family=Nanum+Gothic&display=swap');
</style>
```

Below this, it shows the CSS rules to specify families:

```
font-family: 'Nanum Gothic', sans-serif;
```

# { css; } - 구글 웹폰트

## 1. link 방식

```
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Nanum+Gothic&display=swap" rel="stylesheet">

<style>
  body{
    font-family: 'Nanum Gothic', sans-serif;
  }
</style>
```

## 2. import 방식

```
<style>
  @import url('https://fonts.googleapis.com/css2?family=Nanum+Gothic&display=swap');

  body{
    font-family: 'Nanum Gothic', sans-serif;
  }
</style>
```

# { css; } - 폰트어썸



<http://fontawesome.com>

The screenshot shows the official Font Awesome website homepage. At the top is the Font Awesome logo (a flag icon) and the text "Font Awesome". Below it is a sub-headline: "Get vector icons and social logos on your website with Font Awesome, the web's most popular icon set and toolkit.". Two main calls-to-action are visible: a dark blue button with a star icon labeled "Start for Free" and a light blue button with a rocket icon labeled "Get More with Pro". At the bottom, there's a footer bar with links: "Version 5.15.3", "1,609 Free Icons", "7,865 Pro Icons", and "Download".

## 1. 다운로드 방법

<https://fontawesome.com/download>

에서 다운로드 후 <head> tag 사이에  
다운받은 css 파일을 link

```
<head>
<link rel="stylesheet" href="all.css">
</head>
```

## 2. CDN 링크 방법

<head> tag 사이에 폰트어썸 사이트의  
css 파일을 link

```
<head>
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.15.3/css/all.css" i
ntegrity="sha384-
SZxX4whJ79/gErwc0Yf+zWLeJdY/qpuqC4cAa9r0GUstPomtqpuNWT9wdPEn2fk" crossorigin="anonymous">
</head>
```

# { css; } - 폰트어썸

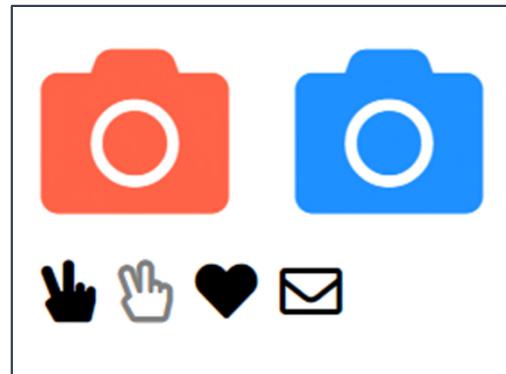
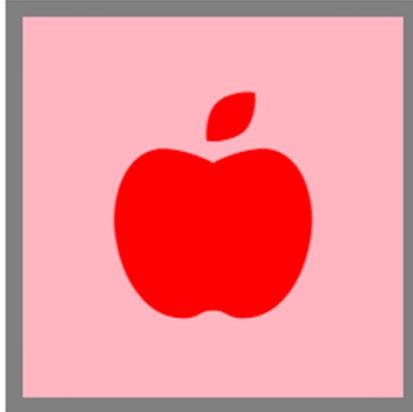


```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <link href="assets/fontawesome/css/all.min.css" rel="stylesheet">
  <title>폰트어썸</title>
</head>
<body>
  <i class="fas fa-user"></i> <!-- uses solid style -->
  <i class="far fa-user"></i> <!-- uses regular style -->
  <i class="fal fa-user"></i> <!-- uses light style -->
  <i class="fab fa-github-square"></i> <!-- uses brands style -->

  <!--this icon's 1) style prefix == fas and 2) icon name == camera -->
  <i class="fas fa-camera"></i>

  <!-- using a <span> element to reference the icon -->
  <span class="fas fa-camera"></span>
</body>
</html>
```

# { css; } - 폰트어썸



```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <link href="assets/fontawesome/css/all.min.css" rel="stylesheet">
    <title>폰트어썸</title>
    <style>
        .apple{
            font-size: 60px;
            color: red ;
            background-color: lightpink;
            width: 100px;
            height: 100px;
            text-align: center;
            line-height: 100px;
            border: 5px solid grey;
        }
    </style>
</head>
<body>
    <span class="apple fas fa-apple-alt"></span>
</body>
</html>
```

# { css; } - 배경 (색상)



## background-color : 배경색상

P{

background-color : green;

background-color : #00ff00;

background-color : rgb(0,255,0);

background-color : rgba(0,255,0, 0.5);

}

```
<style>
  div{
    background-color: royalblue;
  }
  p{
    background-color: yellowgreen;
  }
</style>
```

색상이름 : 기본 색상

16진수 색상 (미세조정)

rgb : windows rgb

rgba : 투명도 조정가능

## 배경색 예제

양식은 보통 사용자가 입력한 값을 활용하기 위해 사용합니다. 하지만, 사용자가 입력하지 않아도 필요한 값이 있을 수 있습니다.

이러한 값들은 type이 hidden인 input 태그로 만들어서 넘깁니다.

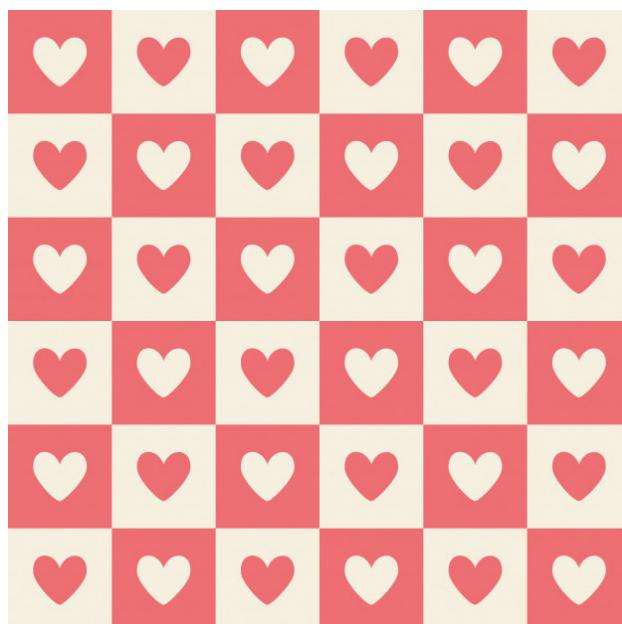
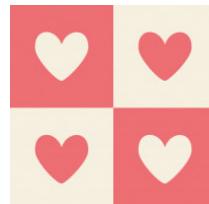


```
<div>
  <h1>배경색 예제</h1>
  <p>
    양식은 보통 사용자가 입력한 값을 활용하기 위해 사용합니다.
    하지만, 사용자가 입력하지 않아도 필요한 값이 있을 수 있습니다.
  </p>
  <span>이러한 값들은 type이 hidden인 input 태그로 만들어서 넘
  급니다.</span>
</div>
```

# { css; } - 배경 (이미지)



background-image.jpg



**background-image** : html 요소에 이미지 배경 넣기

**background-image** :  
`url('images/background-image.jpg');`

**background-repeat** : 배경 이미지 반복

- **repeat** : 가로세로 모두 반복(기본값)
- **repeat-x** : 가로만 반복
- **repeat-y** : 세로만 반복
- **no-repeat** : 반복 안함

# { css; } - 배경 (이미지)



contain



auto(기본값)



ex)

- background-size : cover;
- background-size : 50%;
- background-size : 200px 100px;

숫자를 하나만 넣으면 width 값을 의미  
**(높이는 이미지 비율로 자동 설정)**

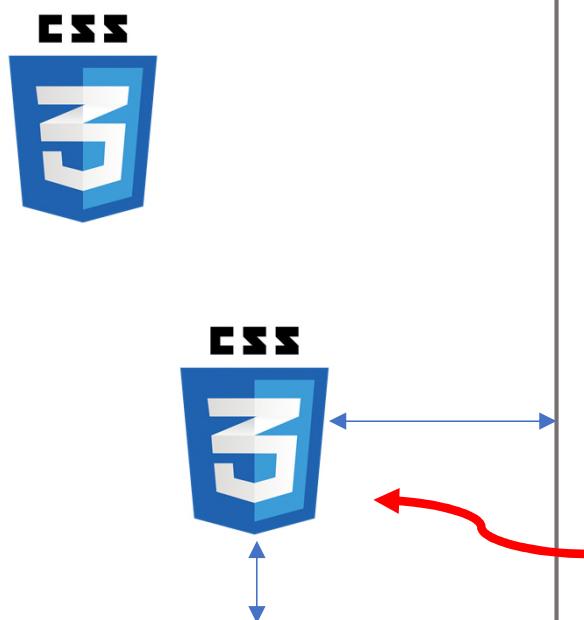
cover

# { css; } - 배경 (이미지)



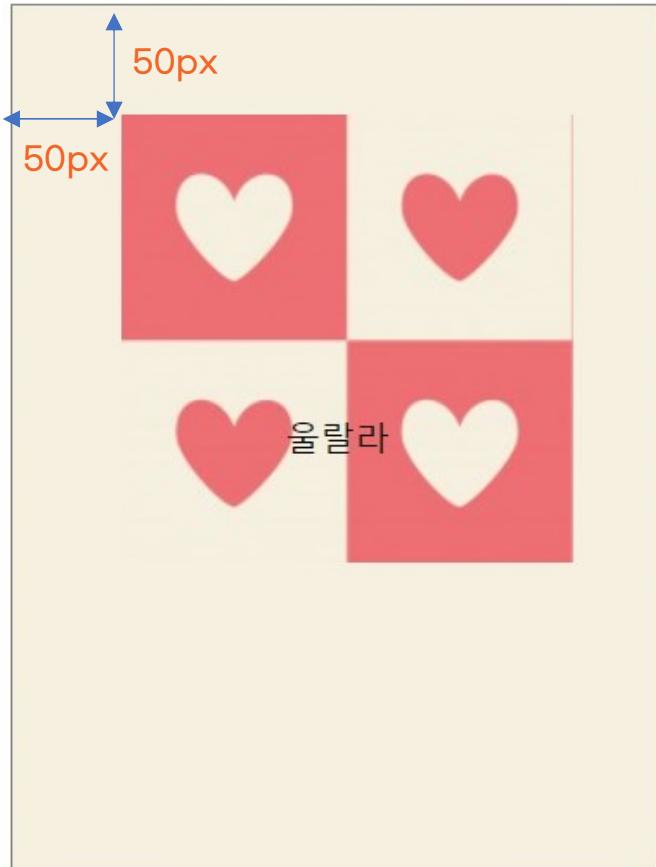
background-position : 이미지 위치

ex) background-position : <가로위치> <세로위치>



- 가로위치 : left center right or px %
  - 세로위치 : top center bottom or px %
- 
- background-position : center center;
  - background-position : left top;
  - **background-position : right 100px bottom 50px;**

# { css; } - 배경 표현 (한 줄로)



```
<style>
  div{
    width: 300px;
    height: 400px;
    text-align: center;
    line-height: 400px;
    border: 1px solid grey;

    background-color: #F6F0E0;
    background-image: url(images/heart_bg2.jpg);
    background-repeat: no-repeat;
    background-position: left 50px top 50px;
    background-size: auto;
  }
</style>
```

background: #F6F0E0 url(images/heart\_bg2.jpg) no-repeat left 50px top 50px;

# { css; } - Position & Layout



## 블록 요소 vs 인라인 요소 (block vs inline)

블록 요소(Block Elements)



인라인 요소(Inline Elements)



인라인 요소.inline elements)

span, a, small, big, em, u, i, s, del, br, q, b, mark, sub, sup, video, audio…

- 기본넓이 : 컨텐츠 넓이/한줄에 여러 개 배치/size, margin 값 없음

블록 요소.block elements)

div, h1, table, tr, td, caption, header, footer, section, article, p, ul, ol, li, hr…

- 기본넓이 : 100%/한줄에 하나만 배치/size, margin 값 있음

인라인 블록 요소.inline block elements)

Img, button, 폰트어썸, input 태그들…

- 기본넓이 : 컨텐츠 넓이 / 한줄에 여러 개 배치 /size, margin 값 있음

# { css; } - Position & Layout



## 블록 요소 vs 인라인 요소

예) 가로 배치된 메뉴 리스트

```
HTML CSS jQuery WordPress Photoshop
```

```
<ul class="navi1">
  <li><a href=""><i class="item fab fa-html5"></i>HTML</a></li>
  <li><a href=""><i class="item fab fa-css3-alt"></i>CSS</a></li>
  <li><a href=""><i class="item fab fa-js-square"></i>jQuery</a></li>
  <li><a href=""><i class="item fab fa-wordpress"></i>WordPress</a></li>
  <li><a href=""><i class="far fa-file-image"></i>Photoshop</a></li>
</ul>
```

```
<style type="text/css">
  .navi1{
    padding-left: 0px;
  }
  .navi1 li {
    list-style: none;
    display: inline-block;
  }
  .navi1 li a {
    text-decoration: none;
    color: #222;
    margin-right: 20px;
  }
  .item {
    margin-right: 5px;
    width: 15px;
  }
</style>
```

# { css; } - Position & Layout



## 블록 요소 vs 인라인 요소

예) 가로 배치된 메뉴 리스트

```
 HTML  CSS  jQuery  WordPress  Photoshop
```

```
<style type="text/css">
  .item {
    margin-right: 5px;
    width: 15px;
  }
  .navi2 a{
    display: inline-block;
    text-decoration: none;
    color: #222;
    margin-right: 20px;
  }
</style>
```

```
<div class="navi2">
  <a href=""><i class="item fab fa-html5"></i>HTML</a>
  <a href=""><i class="item fab fa-css3-alt"></i>CSS</a>
  <a href=""><i class="item fab fa-js-square"></i>jQuery</a>
  <a href=""><i class="item fab fa-wordpress"></i>WordPress</a>
  <a href=""><i class="item far fa-file-image"></i>Photoshop</a>
</div>
```

# { css; } - Position & Layout



레이아웃 스타일 (display 속성 : inline, inline-block, block, none)

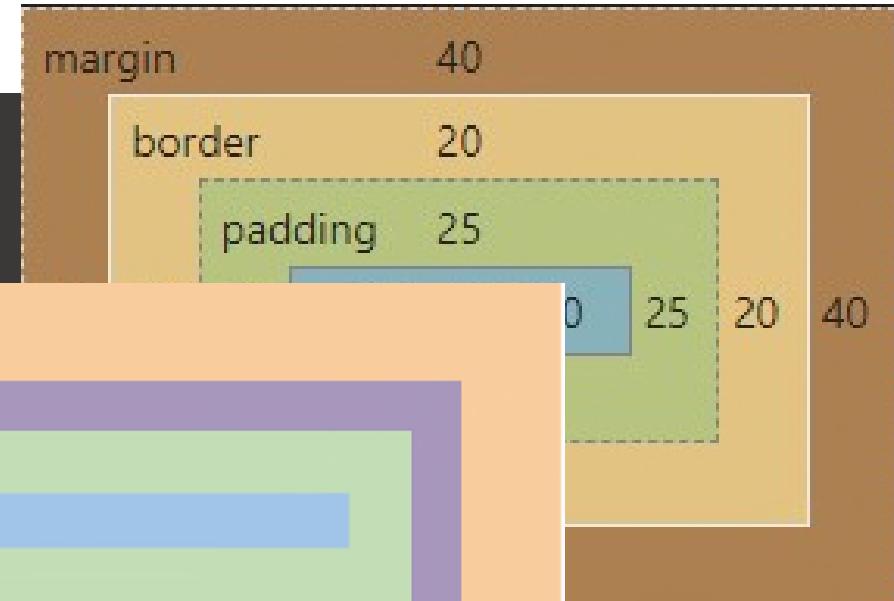


```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Display</title>
    <style>
        .image-box img { display: block; }
    </style>
</head>
<body>
    <div class="image-box">
        
        
        
    </div>
</body>
</html>
```

# { css; } - Box Model



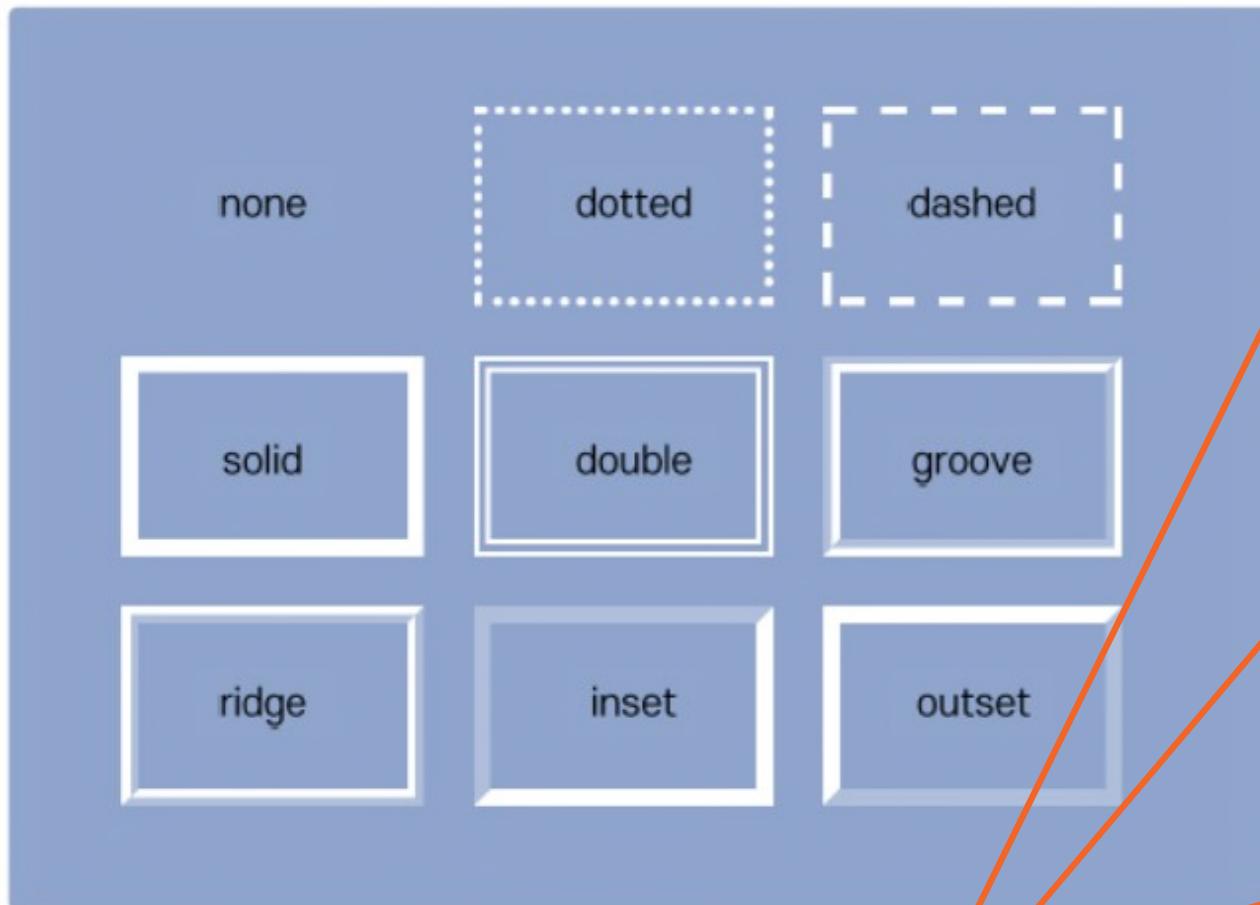
```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
    <div class="content" style="background-color: #f0f0ff; padding: 10px; border: 1px solid #ccc; margin: 20px auto; width: fit-content; height: 150px; font-family: inherit; font-size: inherit; color: inherit; position: relative; z-index: 1;>
        모두를 위한 코딩 (Coding for all)
    </div>
</body>
</html>
```



# { css; } - Box Model



border style



한 줄로 표현 →

`border: green solid 10px;`

**border-color** : 테두리 색상

**value** : color

border-color : green;

**border-style** : 테두리 스타일

**value** : none, solid, dashed, dotted, double, groove, ridge, inset, outset

border-style : solid;

**border-width** : 테두리 두께

**value** : px, %, thin, medium, thick

border-width : 10px;

# { css; } - Box Model



## 마진, 패딩 적용 방법 (margin & padding)

margin(padding) : 10px;



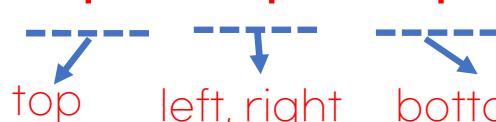
top, bottom, left, right

margin(padding) : 10px 20px;



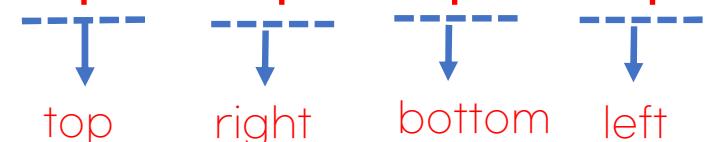
top, bottom      left, right

margin(padding) : 10px 20px 30px;



top      left, right      bottom

margin(padding) : 10px 20px 10px 20px;

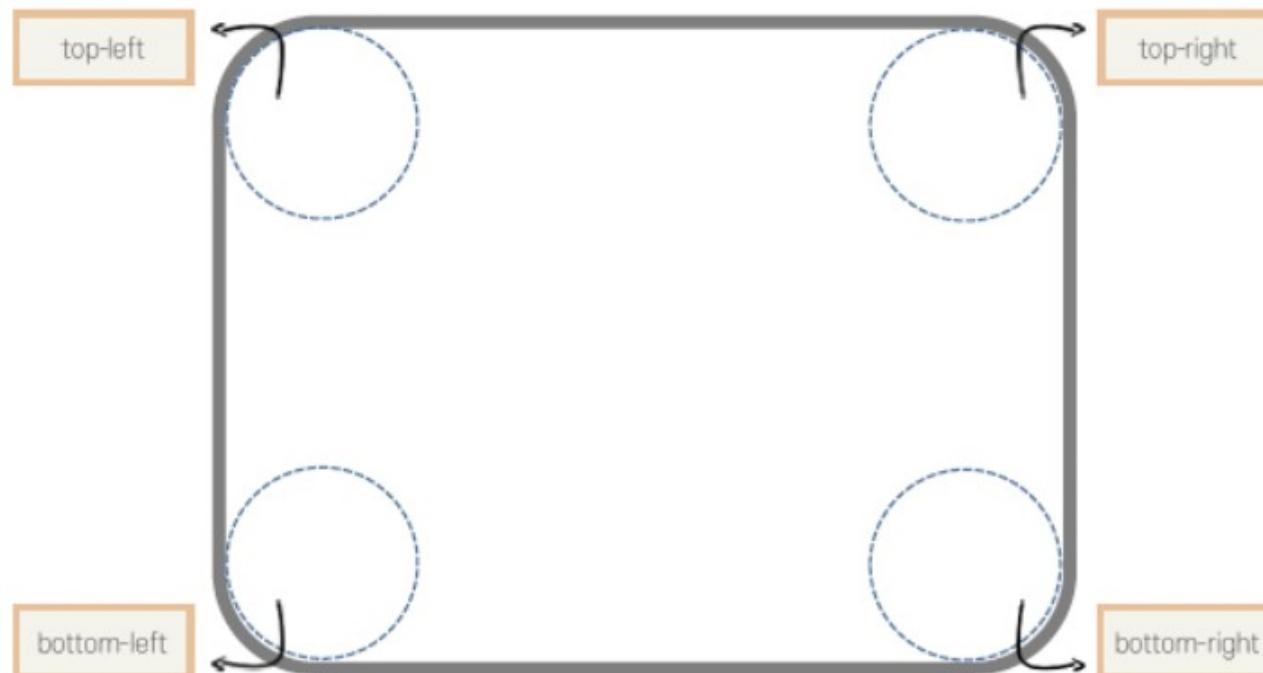


top      right      bottom      left

# { css; } - Box Model



border-radius : 요소 코너 둥글게



border-radius : 엘리먼트 코너 둥글게 하기  
value : px %

Ex) border-radius: 20px;

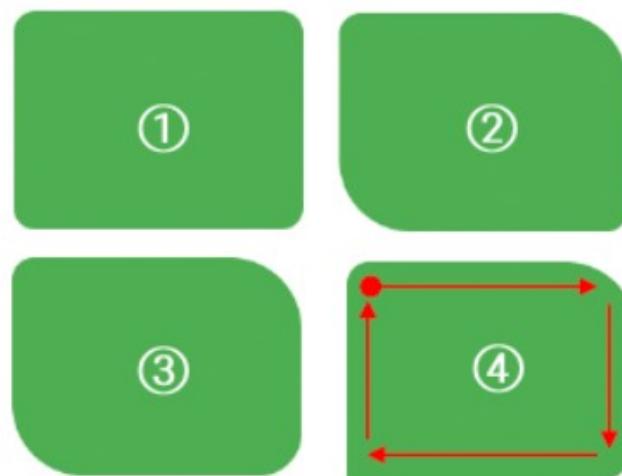
HTML 요소 코너 개별적으로 둥글게 하기

- border-top-left-radius: px %;
- border-top-right-radius: px %;
- border-bottom-right-radius: px %;
- border-bottom-left-radius: px %;

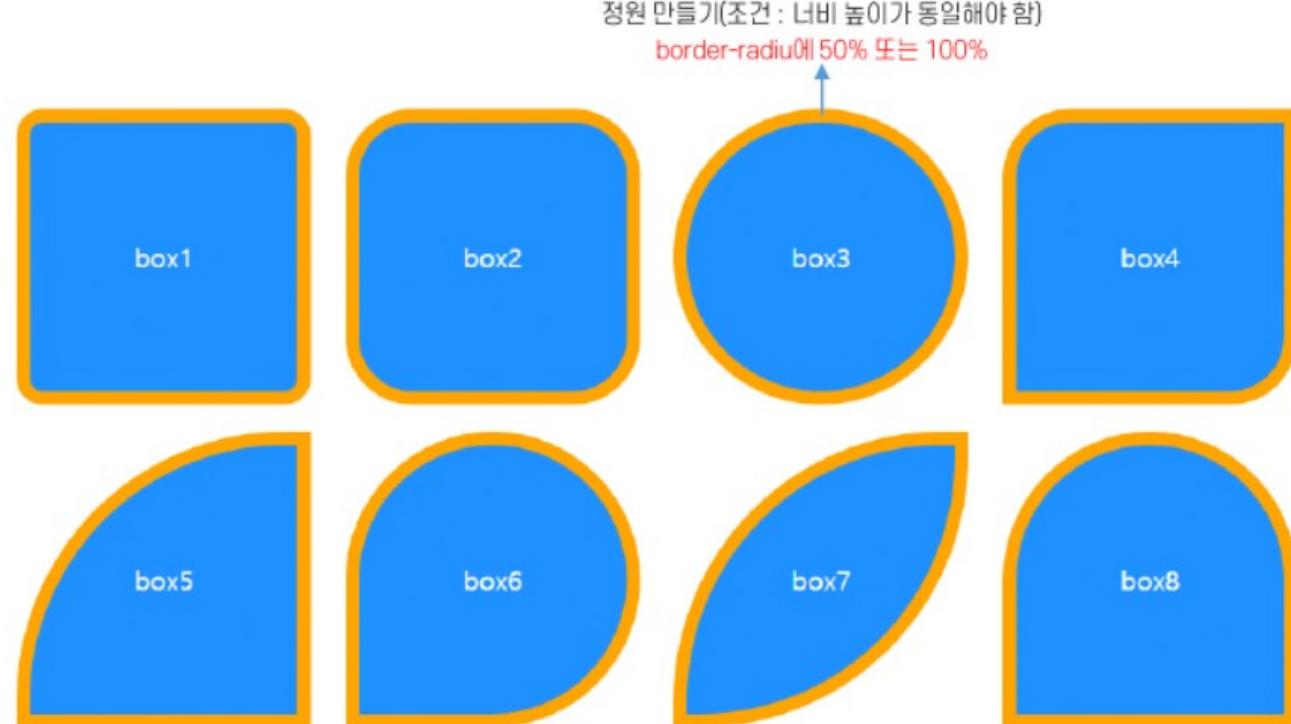
# { css; } - Box Model



border-radius : 요소 코너 둥글게



- ① 1가지 값 - border-radius : 15px;
- ② 2가지 값 - border-radius : 15px 50px;
- ③ 3가지 값 - border-radius : 15px 50px 30px;
- ④ 4가지 값 - border-radius : 15px 50px 30px 5px;

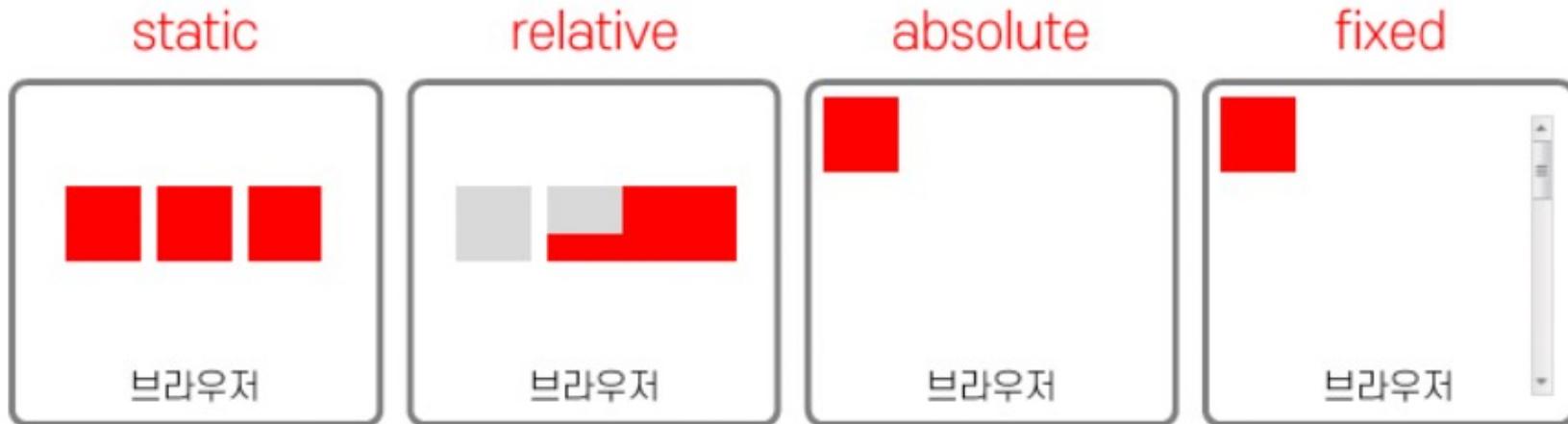


정원 만들기(조건 : 너비 높이가 동일해야 함)  
border-radius 50% 또는 100%

# { css; } - Position & Layout



position 속성 : **static & relative & absolute & fixed**



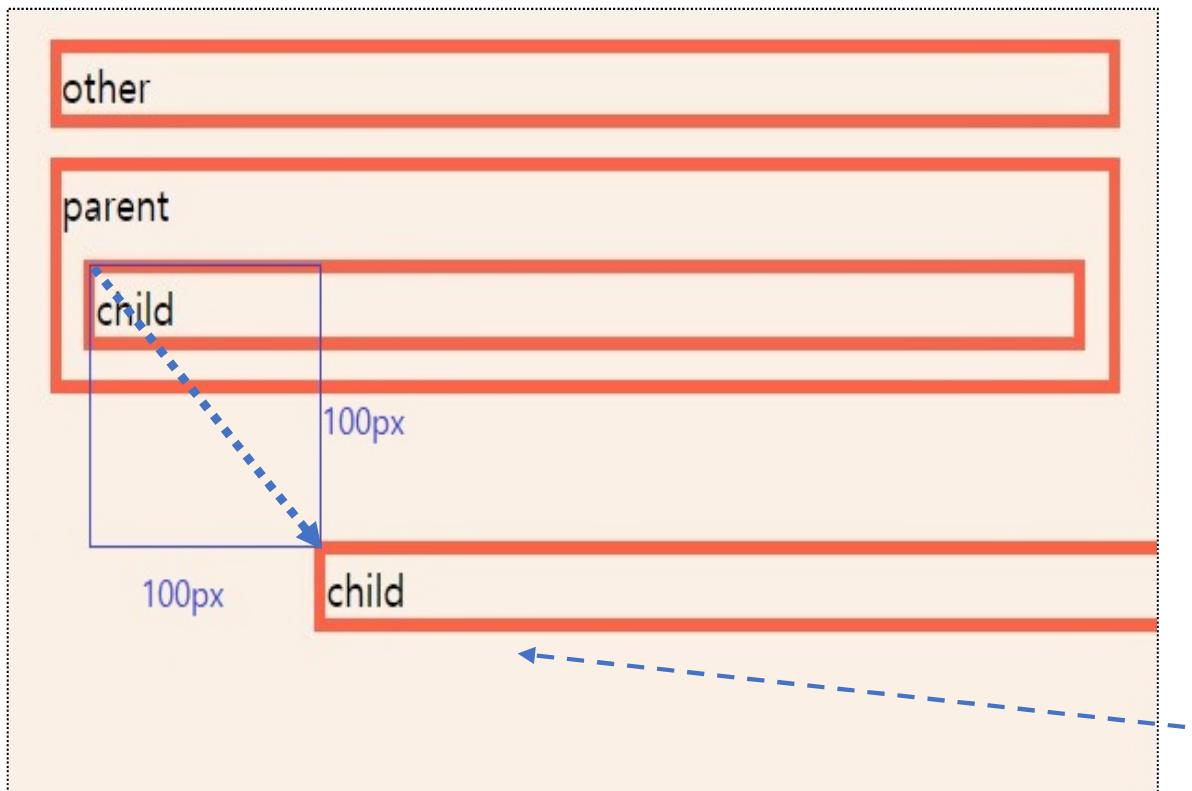
position: static | fixed | absolute | relative

- static(default) : position 속성을 없음
- fixed : 요소가 브라우저 창에 고정되어 배치
- absolute : 요소가 절대적인 위치에 배치
- relative : 요소가 상대적인 위치에 배치

# { css; } - Position & Layout



## 1. position : **relative**



```
<div id="other">other</div>
<div id="parent">parent
  <div id="child">child</div>
</div>
```

```
<style>
div{
  border:5px solid tomato;
  margin:10px;
}
</style>
```

```
#child{
  position: relative;
  left:100px;
  top:100px;
}
```

# { css; } - Position & Layout



## 2. position : absolute



부모 : relative, 자식 : absolute



```
<div class="parent">
  <div class="child inside1"></div>
  <div class="child inside2"></div>
  <div class="child inside3"></div>
  <div class="child inside4"></div>
</div>
```

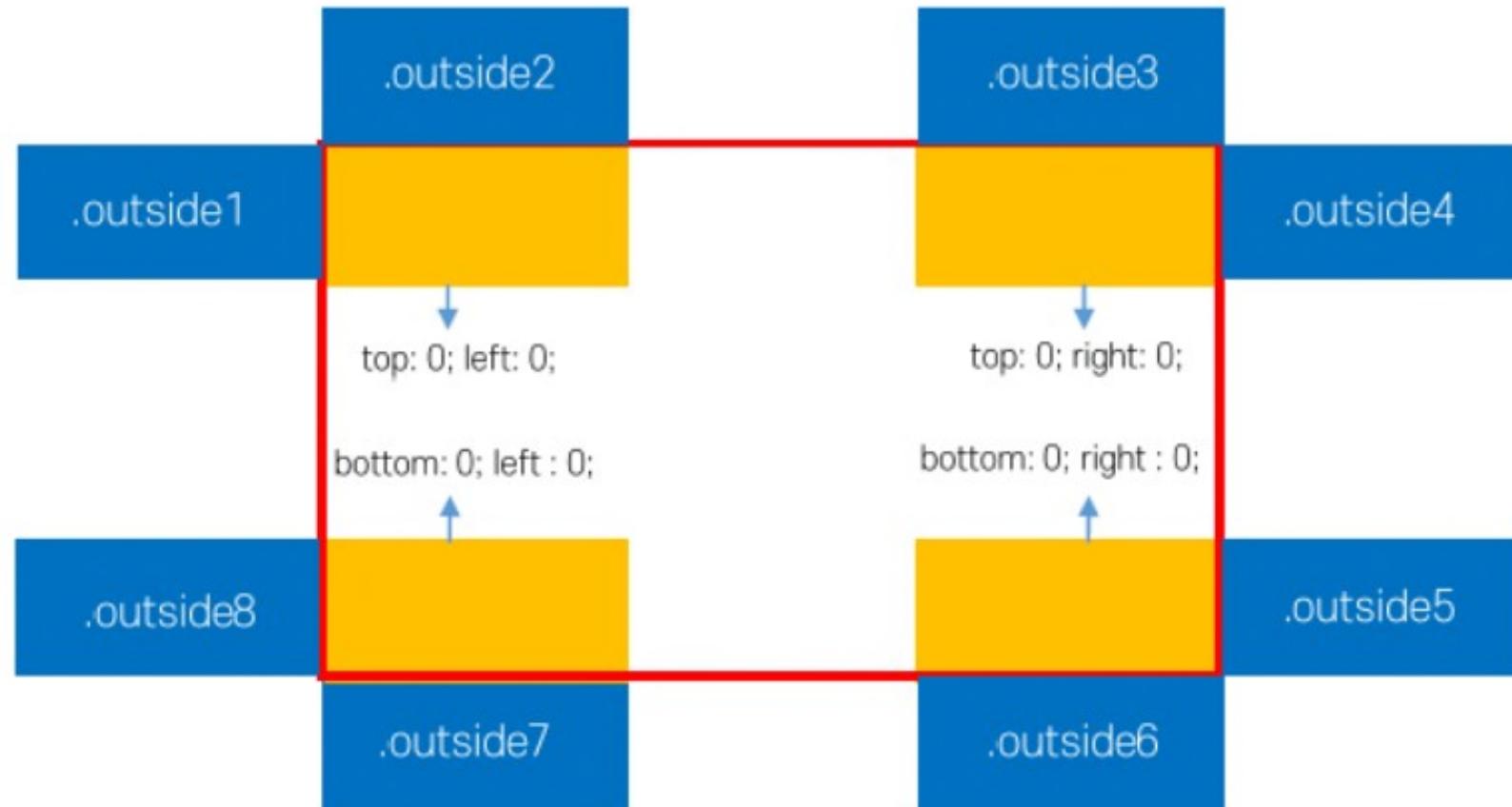
300px

```
<style>
  .inside1{
    top: 0;
    left: 0;
  }
</style>
```

# { css; } - Position & Layout



## 2. position : absolute



# { css; } - Position & Layout

### 3. position : fixed (정중앙에 위치)

**Position: fixed**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam harum volitantes eius, sequi, voluptate dolores eorum odit ipsum illum, facere dolores itaque odio inre nostrum ea dolores dicta. Vitae provident veniam quas tempora obsecuti nihil quisquam totius docimis laboriosam est tempore, iustus voluptate, quidem repudiandae consequitur repellendus sed? Adipisci cum quas laboriosam voluptatibus blanditiis, rerum id assit uniusquisque alii, ab illius reperiens distractio consumidi doloresque quae eorum non emper. Distinctio virtutis, conjectetur nesciit quo tempora, voluptates? Culpa saram ipsum niven voluptate vero odit solita silium pessentium dignissimum eius voluptatem ipsa non similique delecto: laborem potio eos, quam eligendi provident et qui molestiae? Consectetur non laboriosam ad deleniti, necessitatibus adipisci solita nimirum optio nisi sit error! Aut adipisci nam labore, impedit quaeque. Aliquid et vitae ex excepturi remanentes laboriosi vel enim magni reperiens voluptas consequitur ratione nesciunt ipsum nemo sed sit nonnuda, dolor id dociamus desenatur aut quidea beatas. Expedita fugiat blanditiis, aliquid impedit nonnuda quia, docimis quidem nesciit adipisci lalentium, cum. Perferendas dolorem similique molestias architecto ipsa quid culpi, temporis illo cupiditate sequi laboriosam, nonnuda plicest potio quisquam expedita temporis temporibus iusto dolor, officiis artus virtus tenet, doloremque exercitationes eorum non! Excepturi tempora alias rerum, nostrum velit nulla animo enim, molestiae qui nonnuda accessum exercitationem deserunt culpa silium eorum deleniti consequentes fiscos nihil aut voluptatis: odit expedita unde nimirum

```
.btn-top {  
    position: fixed;  
    left: 20px;  
    bottom: 20px;  
}
```



스크롤해도 상단 상단으로 버튼이 고정됩니다.

```
<a href="#" class="btn-top"><i class="fa fa-arrow-up"></i></a>
```

HTML

# { css; } - Position & Layout



## float : 수평 정렬하기



float: left



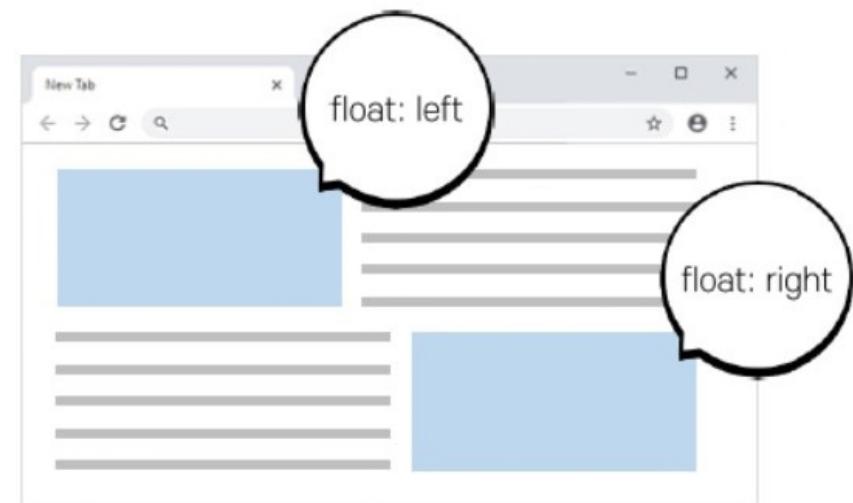
margin: auto



float: right

### float 속성

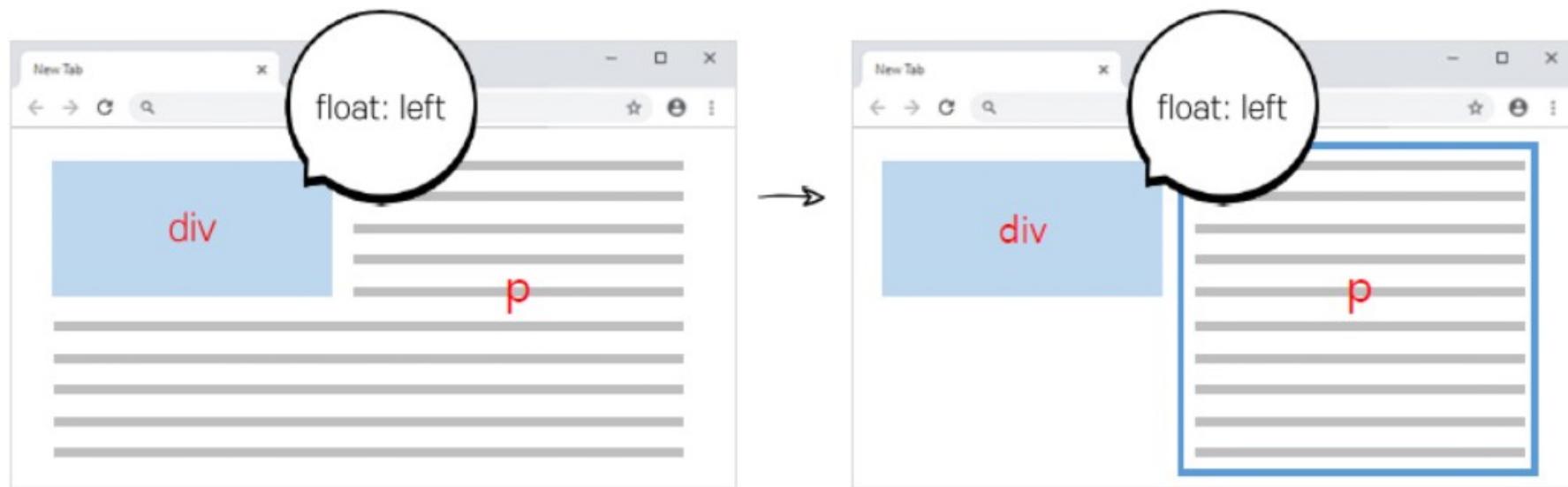
- none(default) : float 속성 적용안함.
- left : 요소가 컨테이너의 왼쪽 배치
- right : 요소가 컨테이너의 오른쪽 배치
- margin: auto 요소를 중앙에 배치



# { css; } - Position & Layout



## float : 수평 정렬하기



```
div { float: left; }
p { line-height: 1.6em; }
```

```
div { float: left; }
p { line-height: 1.6em; overflow: hidden; }
```

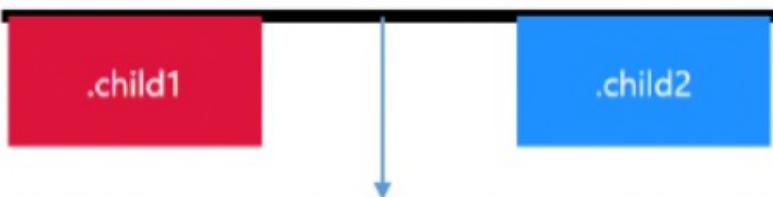
# { css; } - Position & Layout



## float : 수평 정렬하기



```
.parent { border: 5px solid black; width: 600px; }
.child1 { width: 200px; height: 100px; background-color: crimson; }
.child2 { width: 200px; height: 100px; background-color: dodgerblue; float: right; }
```



```
.parent { border: 5px solid black; width: 600px; }
.child1 { width: 200px; height: 100px; background-color: crimson; float: left; }
.child2 { width: 200px; height: 100px; background-color: dodgerblue; float: right; }
```

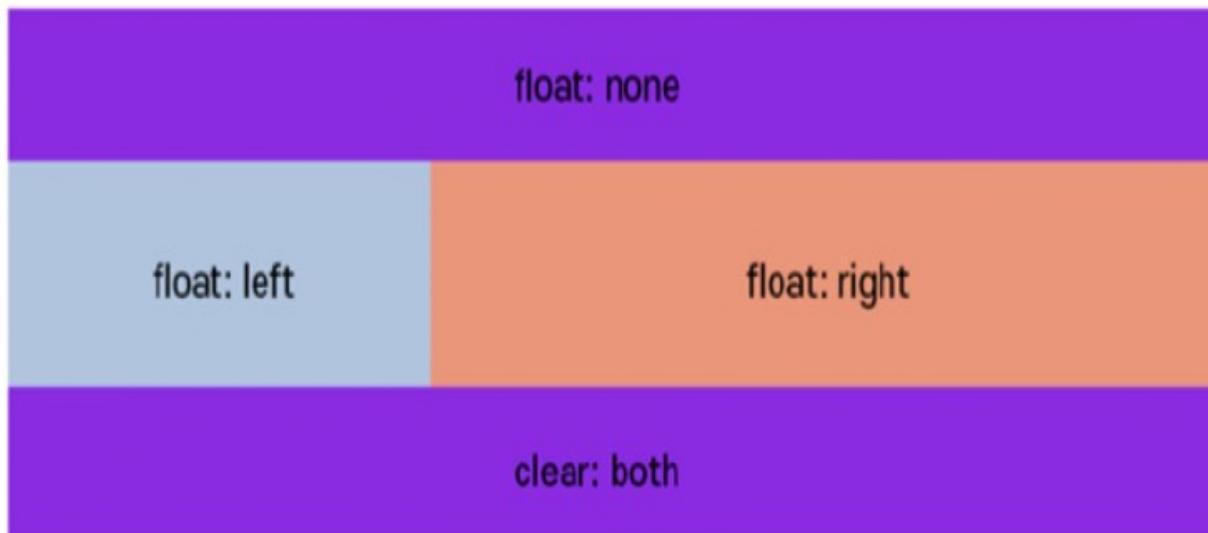
자식요소에 float 속성이 있으면 부모요소는 높이 값을 잃게 됩니다.  
다. 부모요소의 높이 값을 찾는 방법은 ① .parent에 높이를  
100px 주기 ② .parent에 overflow: hidden 주는 방법(추천)

# { css; } - Position & Layout



`clear : left | right | both`

Holy grail layout ↗



- `left` : `float:left` 상속 해지
- `right` : `float:right` 상속 해지
- `both` : 모든 `float` 상속 해지

# { css; } - Position & Layout

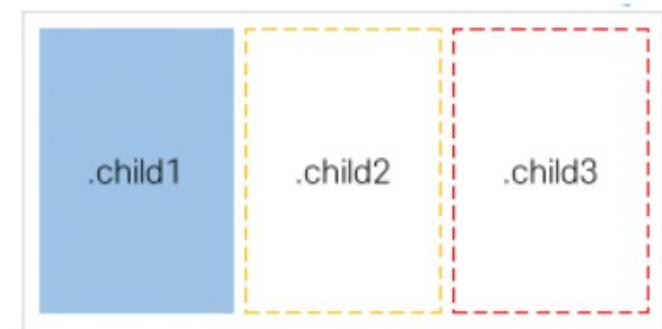


## visibility, opacity, display

```
<div class="parent">  
  <div class="child1"></div>  
  <div class="child2"></div>  
  <div class="child3"></div>  
</div>
```



```
.child1{  
  display:none;  
}
```



```
.child2{  
  opacity: 0;  
}  
.child3{  
  visibility: hidden;  
}
```

# { css; } - Position & Layout

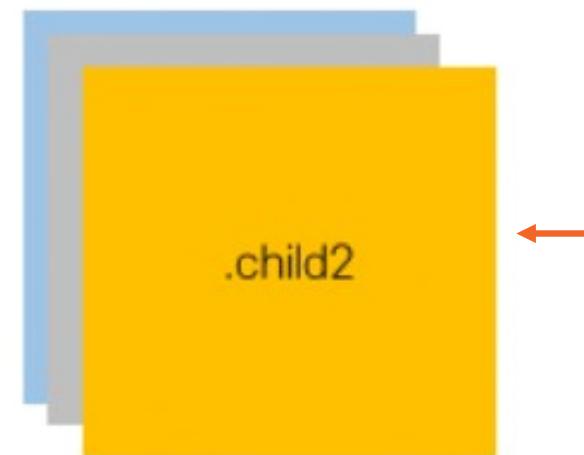
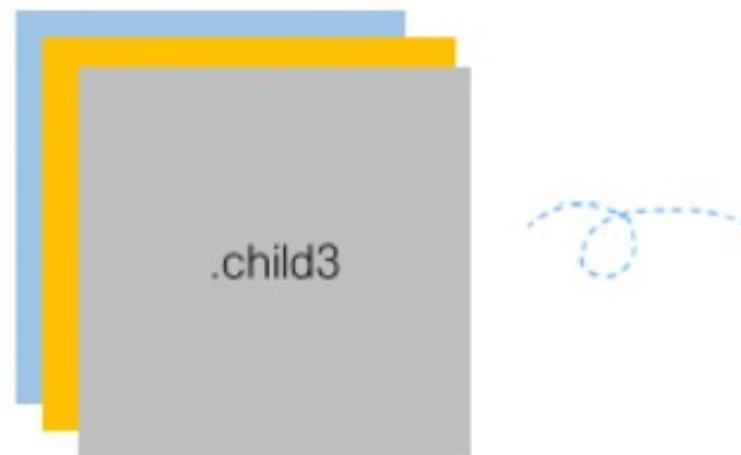


z-index : 겹쳐진 html 요소의 순서

z-index는 반드시  
position 속성과 함께

```
<div class="parent">  
  <div class="child1"></div>  
  <div class="child2"></div>  
  <div class="child3"></div>  
</div>
```

```
.parent{  
  width: 200px;  
  height: 200px;  
  position: absolute;  
}
```

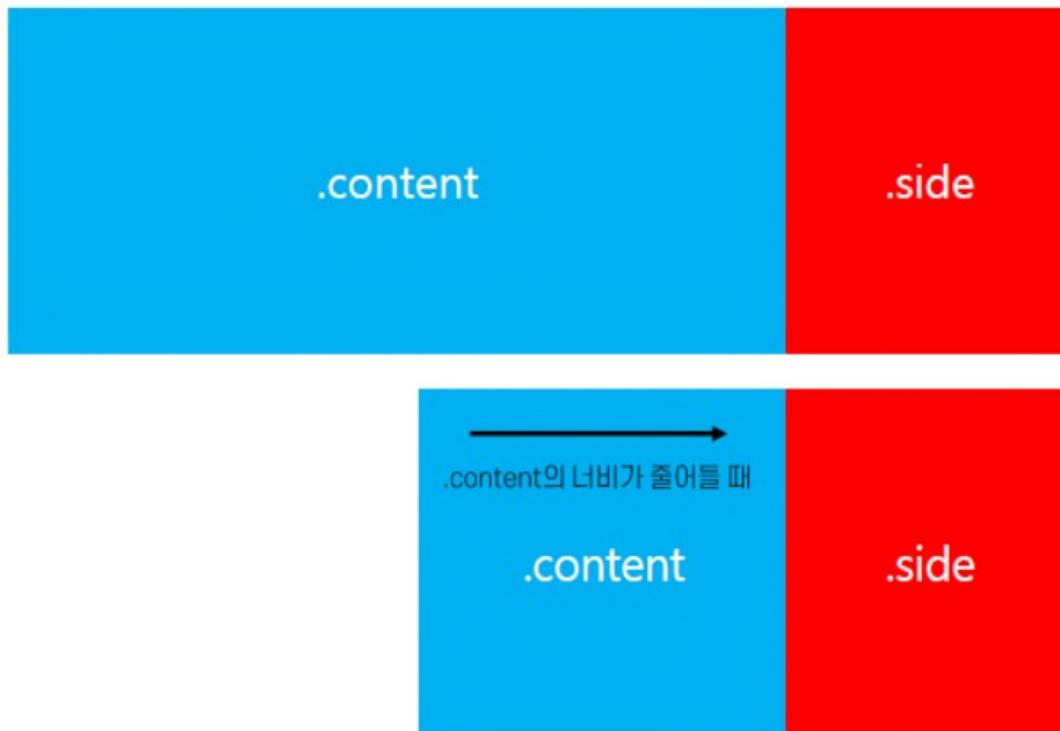


```
.child1{  
  z-index:0;  
}  
.child2{  
  z-index:2;  
}  
.child3{  
  z-index:1;  
}
```

# { css; } - 실천 이론 & Tip



## 가변영역, 고정영역 함께 사용하는 레이아웃



```
.container div { float: left; }
```

```
.side {
```

background-color: crimson; width: 300px; height: 100vh;

```
}
```

```
.content {
```

background-color: royalblue;

width: calc(100% - 300px);

height: 100vh;

```
}
```

높이를 강제로 100% 만드는 vh 단위

.side의 너비는 300px로 고정되어 있어서 변  
하지 않고 .content의 너비가 줄어들어도  
calc() 함수에 따라 유동적으로 변합니다.

연산자 + 또는 - 좌우에 띄어쓰기가 반드시 있  
어야 작동합니다.  
예) calc(100%-300px) --- (X)

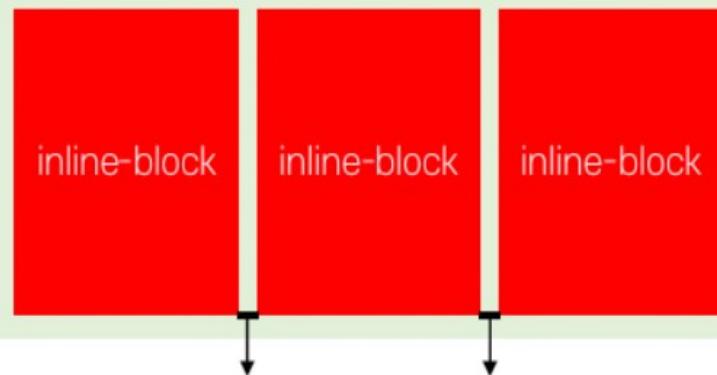
```
<div class="container">  
  <div class="content">Content</div>  
  <div class="side">Side</div>  
</div>
```

HTML

# { css; } - 실천 이론 & Tip



## Inline-block에 생기는 마진 없애기



인라인 블록을 가로 배치한 후 생기는 마진

```
<div class="content">  
  <div>inline-block</div>  
  <div>inline-block</div>  
  <div>inline-block</div>  
</div>
```

HTML

```
.content { text-align: center; }  
.content div {  
  background-color: crimson;  
  width: 200px;  
  height: 300px;  
  line-height: 300px;  
  color: #fff;  
  display: inline-block;  
  margin-right: -6px;  
}
```

네거티브 마진으로 오른쪽 요소를 강제  
로 왼쪽으로 당기는 방법

```
.content {  
  text-align: center;  
  font-size: 0; →  
}  
.content div {  
  background-color: crimson;  
  width: 200px; height: 300px;  
  line-height: 300px; color: #fff;  
  display: inline-block;  
  font-size: 16px; →  
}
```

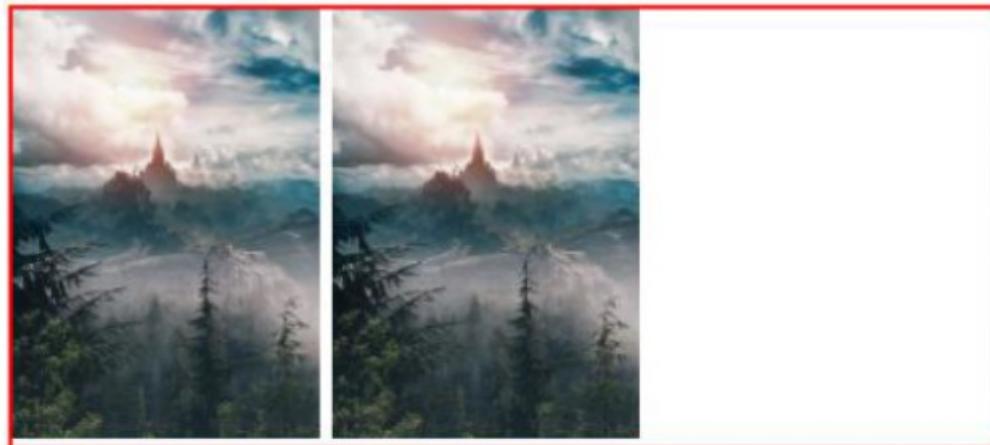
인라인 블록에 생긴  
마진은 원래 폰트에  
서 생긴 문제이므로  
폰트크기를 0으로 하  
면 마진이 없어짐

부모요소에서 폰트 크기 0을 상속받았으니 폰트  
크기를 다시 지정해줌

# { css; } - 실천 이론 & Tip



## img에 생기는 마진 없애기



img 태그는 img 사이에 우측 마진 뿐만 아니라 아래에도 마진이 발생합니다. div에 넣지 않으면 인식하기 힘든데 위 그림처럼 div 안에 넣으면 확연히 아래쪽 마진을 확인할 수 있습니다.

img에 생기는 마진도 폰트에 관련된 마진이므로 img를 감싸고 있는 부모요소에 `font-size: 0;`를 넣어주면 해결됩니다.

이미지 선택자에 `display: block;`으로 변경해도  
이미지 아래 생기는 마진을 없앨 수 있습니다.

<div>

```



</div>
```

HTML

```
div {
    border: 2px solid red;
    font-size: 0;
}
```

# { css; } - 실천 이론 & Tip



## Inline-block끼리 수직중앙 맞추기

모두를 위한 코딩 모집중

```
<div class="content">
  <b>모두를 위한 코딩</b> <span>모집중</span>
</div>
```

```
.content{
  background-color: lightskyblue;
  height: 150px;
  text-align: center;
  line-height: 150px;
}
.content b{
  font-size: 60px;
  vertical-align: middle; // This line is highlighted with a red box
  margin-right: 10px;
}
.content span{
  background-color: red;
  color: #fff;
  border-radius: 3px;
}
```

뒤에 오는 인라인  
요소를 자신의  
수직중앙에 배치

# { css; } - 실천 이론 & Tip



## Inline, inline-block, block 요소 수평중앙 정렬



```
div {  
    text-align: center; → 인라인 또는 인라인 블록 요소를 수평 중앙에 보  
}                                낼 때는 부모요소에 text-align: center를 줍니다.
```

```
<div>  
      
</div>
```

HTML

텍스트 뿐만 아니라 인라인 블록요소인 img도 해당됩니다.  
인라인 블록도 인라인 성질을 가지고 있기 때문입니다.



```
div {  
    border: 2px solid red; width: 400px;  
    text-align: center; → 이 중앙 정렬은 div h4를 div 안에서 중앙 정렬 시키는 것이 아  
}                                니라 h4안에 있는 '블록요소'라는 텍스트를 h4 안에서 중앙 정  
div h4 {                            력 시킨 것 입니다. h4 자신을 div 안에 수평 중앙 정렬시키려면  
    border: 2px solid blue;           h4 안에 margin: auto; 속성을 주어야 합니다.  
    width: 200px;                  }  
}
```

# { css; } - 실천 이론 & Tip



## body기준 수평 수직 중앙 맞추기

body      포지션 속성 사용하기 ▼

```
.content {
    width: 100px;
    height: 100px;
    background-color: #6495ed;
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%, -50%);
}
```

<div class="content">  
인라인 또는 인라인블록  
</div>

▼ 플렉스(Flex) 사용하기      body

```
body {
    height: 100vh; → body의 높이 값이 반드시 필요합니다.
    display: flex;
    justify-content: center;
    align-items: center;
}

.content {
    width: 100px;
    height: 100px;
    background-color: #6495ed;
}
```

<div class="content">  
인라인 또는 인라인블록  
</div>

```
body {
    height: 100vh; → body의 높이 값이 반드시 필요합니다.
    display: flex;
    justify-content: center;
    align-items: center;
}

.content {
    width: 100px;
    height: 100px;
    background-color: #6495ed;
}
```

<div class="content">  
인라인 또는 인라인블록  
</div>

# { css; } - 실천 이론 & Tip



## 부모요소 기준 수평 수직 중앙 맞추기

body      포지션 속성 사용하기▼

.parent {  
width: 400px; height: 400px;  
background-color: #fff;  
position: absolute;  
top: 50%; left: 50%;  
transform: translate(-50%, -50%);  
}  
.child {  
width: 100px; height: 100px;  
background-color: #6495ED;  
position: absolute;  
top: 50%; left: 50%;  
transform: translate(-50%, -50%);  
}

<div class="parent">  
  <div class="child">인라인 또는 인라인블록</div>  
</div>

HTML

▼ 플렉스(Flex) 사용하기      body

body {  
background-color: #eee;  
height: 100vh;  
display: flex;  
justify-content: center;  
align-items: center;  
}  
.parent {  
width: 400px; height: 400px;  
background-color: #fff;  
display: flex;  
justify-content: center;  
align-items: center;  
}

<div class="parent">  
  <div class="child">인라인 또는 인라인블록</div>  
</div>

HTML