

UX/UI를 활용한 모바일 웹 디자인 실무과정

# CSS

담당: 이은정

## css란?

- CSS는 웹 페이지에 “디자인”이라는 시각적 가치를 부여하기 위한 언어입니다.
- HTML 문서의 레이아웃과 스타일을 정의합니다.
- W3C에서 표준안을 만들어 내고 있으며 1996년 CSS Level 1 이후, 현재 CSS Level 2.1에서 CSS Level 3까지 보급되어 있음.
- W3C 표준 : CSS Level 2 Revision 1 (CSS2.1), 모바일 : CSS3 (CSS2.1 기술 포함됨)

## 서식

```
선택자 {속성      :      값;
         속성      :      값;
         속성      :      값;      }
```

- selector (선택자) : 스타일을 적용하는 대상
- property (속성) : 스타일의 종류
- value (값) : 속성이 가질 수 있는 값

```
body {color:gray;}
div {border:1px solid red;background-color:gray;}
```

## 선택자-1

### 기본선택자

선택자	기능설명
요소	특정 요소 선택
#id명	특정 id명을 가진 요소 선택
.class명	특정 class명을 가진 요소 선택
요소1 요소2	요소1 안의 있는 모든 요소자손 요소2
부모요소1>자식요소2	부모요소1 안에 있는 자식 요소2
형제요소1+형제요소2	형제요소1 옆에 있는 형제 요소2
형제요소1~형제요소2	요소1 의 모든 형제 요소2
선택자1, 선택자2	여러 선택자 한꺼번에 선택
*	모두 선택

### 타입선택자

요소명을 선택자로 사용

```
h1 {font-size:20px;}
ul {margin:0px;}
```

### class 선택자 (요소명.class명 : 요소명 생략가능)

- 클래스 이름으로 선택하기, 문서내에서 여러 요소를 한 번에 선택할 수 있다.

```
div.note { color : blue ;}
.clr {clear:both;}
```

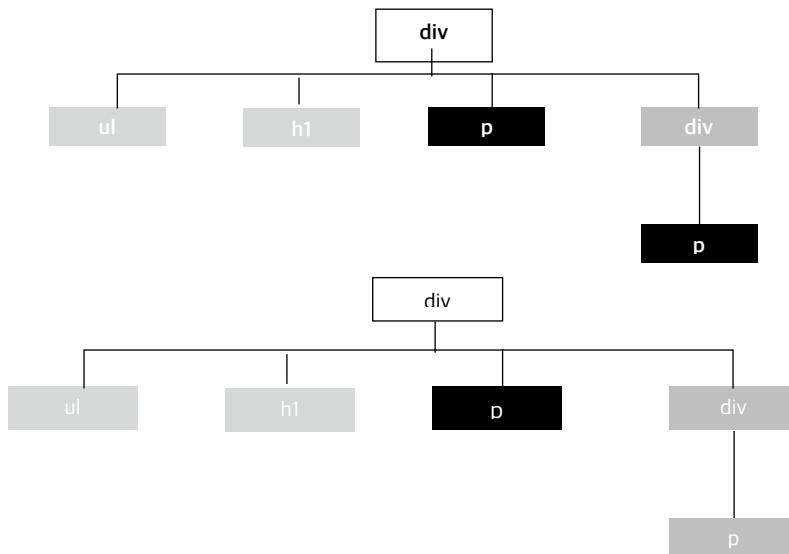
### id 선택자 (요소명#id명 : 요소명 생략가능)아이디 이름으로 선택하기

```
ul#gnb {list-style-type : none ;}
#bodyContent {width:100%; height:500px;}
```

### 자손 ,종속선택자

- 특정 엘리먼트 하위의 엘리먼트를 지정할 때 사용
- 공백(space) 으로 구분함
- ID, CLASS등 여러 선택자와도 함께 사용 가능

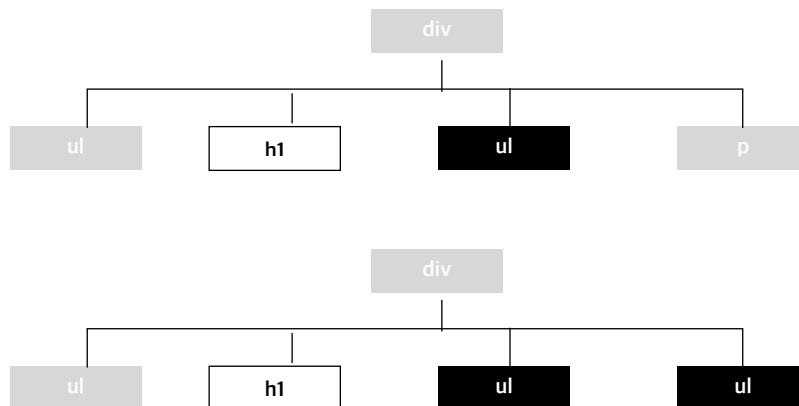
```
div p {height:30px;}
div>p {color:blue;}
```



### 형제 이웃선택자

기준이 되는 요소 와 형제 또는 다음에 오는 요소 선택

```
h1+ul{background-color:red;}
h1~ul{background-color:red;}
```



### 그룹 선택자

- 선택자와 선택자를 콤마(,)로 구분하여 그룹화 하고 복수의 요소에 같은 스타일을 적용합니다.
- class 선택자나 id 선택자, 속성 선택자, 하위 선택자, 자식 선택자, 인접 선택자 등을 지정할 때도 콤마(,)로 구분하여 그룹화 할 수 있다.

```
#wrap, p.note, blockquote p, h1[title] { color : red ; }
```

## 적용방식

### . Inline Style Sheet

- 요소에 직접 스타일 적용
- 태그와 함께 CSS가 코딩되어 있어 CSS분리원칙에 위배됨
- 단, 디버깅용(오류찾기)으로만 사용  
<태그 style="속성:값;">

### Internal Style Sheet

html문서의 <head></head> 사이에 CSS를 선언하는 방식 (meta 뒤에, script 앞에)마찬가지로, CSS분리원칙에 위배됨)

```
<head>
    <style type="text/css">
        selector { 속성 : 값; }
    </style>
</head>
```

### External Style Sheet:@link사용(가장권장)

```
<link>요소로 외부 파일 사용하기
<head>
    <link rel="stylesheet" type="text/css" href="css파일명" />
</head>
```

```
@charset "utf-8";
p{
    background: beige;
    color: gray;
}
p span{
    font-weight: bold;
    color: blue;
}
```

### External Style Sheet:@import사용하기

여러개의 CSS파일 연결해서 사용

@import url (경로/파일명.css); 또는 @import “파일명.css”; 로 작성

```
<style>
    @import url(경로)
</style>
```

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

## Basic

### CSS 주석

- 작성자의 관리와 편의를 위한 찾아보기나 메모
- “/\*”와 “\*/” 사용.
- 주석으로 작성은 내용은 스타일에 영향을 미치지 않는다.  
/\* comment \*/

### 단위

CSS에서 사용할 수 있는 단위(units)는 크게 '절대 단위'와 '상대 단위'가 있습니다.

### 절대단위

pt: 포인트 1pt는 1/72 inch

cm, mm: 센티미터, 미리미터

pc: 파이카, 1pc는 12pt

in: inch

### 상대단위

px : 디스플레이 해상도에 따라 화면에서 보이는 크기 변동

% : 부모 요소에 비해 얼마나 커야하는지 비율로 명시 기준의 크기를 100%로 잡는다.

em : 부모 요소에 비해 얼마나 커야하는지 척도로 명시 기준의 크기를 1로 잡는다.

### %와 em의 계산법

%: 자신의 크기/기준의 크기 \* 100

$320/450 * 100 = 71.11111111111\%$

em: 자신의 크기/기준의 크기

$320/450 = 0.711111em$

참고사이트: <http://pxtoem.com>

## Color

### Keyword (색상명, 17가지)

- black, silver, gray, white, maroon, red, purple, teal, fuchsia, green, lime, olive, yellow, navy, blue, aqua

color:lime;

background-color:maroon

샘플참조: [https://en.wikipedia.org/wiki/Web\\_colors](https://en.wikipedia.org/wiki/Web_colors)(구글에서 webcolor검색)

[http://www.w3schools.com/colors/colors\\_picker.asp](http://www.w3schools.com/colors/colors_picker.asp)(구글에서 webcolor검색)

### RGB값 (16진수, 6자리)

color : #ffff00 ;

### RGB값 (16진수, 3자리)

color : #ff0 ; (#ffff00 =#ff0)

### RGB값 (10진수)

color : rgb(255,0,0) ;

### RGB값 (%)

color : rgb(100%,0,0) ;

주의할 점은 값이 '0' 이더라고 '%'를 생략해서는 안된다.

white	#ffffff	rgb(255,255,255)	rgb(100%,100%,100%)
black	#000000	rgb(0,0,0)	rgb(0%,0%,0%)

## css3추가된 칼라의 속성

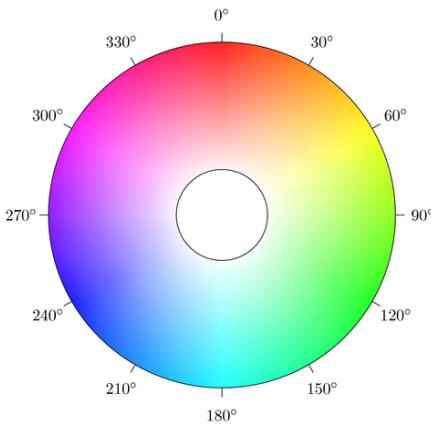
### rgba(red,green,blue,alpha)

- red green blue alpha로 칼라 지정
- alpha:0~1(0.5=50%)

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

### hsl(hue,saturation,lightness)

- hue: 색상환에서 0-360까지의 값으로 표현
  - 채도: %로 나타냄 (0: 무채색 ~ 100%: 순색)
  - 밝기: %로 나타냄 (0~100%)
- background-color:hsl(0, 0%, 78%)



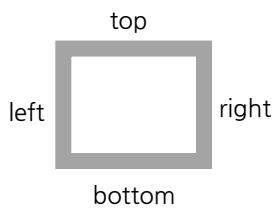
### hsla(hue,saturation,lightness,alpha)

background-color:hsla(0, 0%, 78%,0.5)

### opacity

- 요소에 투명도 지정
  - 0.0~1.0(0.15=15%)
- ```
p{background-color:rgb(0,0,0,0.5); opacity:0.15;}
```

### 방향



- 네 개 (top, right, bottom, left)  
margin:5pt 0pt 15pt 20pt;
- 세 개 (top, right과 left, bottom)  
margin:5pt 10pt 15pt;
- 두 개 (top과 bottom, right과 left)  
margin:5pt 10pt;
- 한 개 (모두 같음)  
margin:5pt;

## 스타일의 기본 성격

### 상속(Inherit)

- 하위 요소가 상위 요소의 사용값을 물려 받느냐 아니냐를 나타낸다.
- 값의 상속은 상속되는 것이 있고 안 되는 것이 있다. 자동으로 자식 요소에 상속되지 않는 경우 “inherit” 키워드를 이용하여 강제로 상속시킬 수 있음.

<p>부모 요소의 스타일이 자식 요소에 계속되는 것을 <a href="inherit.html">상속(inherit)</a>라고 한다. </p>

```
p { border : 1px solid red ;}
a { border : inherit ;}
```

### 겹침(Cascading)

스타일의 충돌 : 두 개 이상의 규칙이 동일한 한 개의 요소에 적용되는 현상

스타일의 우선순위 결정 규칙 : 가장 마지막에 지정된 스타일이 우선적으로 적용

스타일 선언이 위에서 아래로 순차적으로 적용이되기 때문에 항상 가장 마지막에 선언 된 스타일이 우선권을 갖는다.

```
<link rel="stylesheet" type="text/css" href="default.css" />
```

```
<style type="text/css">
  @import url(base.css);
  p { color : gray ; }
</style>
```

```
<p style="color : blue ;">스타일의 우선순위</p>
```

## 특정도(Specificity)

여러 선택자로 선택이 된 경우에는

in-line 스타일 > id 선택자 > class 선택자 > 가상 클래스 선택자 > 가상요소 선택자 > type 선택자

# Box Model

박스모델은 블록박스와 인라인 박스가 화면에 표시되는 영역과 다른 박스와의 배치등에 관한 내용으로 주요한 속성은 크기, 테두리, 여백에 관한 것들이 있다.

## width,height

Block 요소또는 블록 성격을 가진 인라인 요소의 가로, 세로 사이즈를 지정한다.

width:길이 | 퍼센트 | auto | inherit

height:길이 | 퍼센트 | auto | inherit

width:300px; height:500px;

## border

- 콘텐츠의 외곽선을 그려줌
- 선의 두께와 선의 스타일, 그리고 칼라 값이 모두 같다면 한 번에 모두 지정할 수 있고,
- 만약에 서로 다르다면 속성에 방향을 지정하여 정할 수 있다.
- 방향을 지정하는 속성:top,right,bottom,left;

### 1. border-width:외곽선의 굵기를 지정한다.

- 외곽선의 굵기를 지정한다.

border-width:두께;

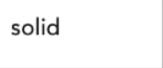
border-width:5px;

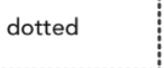
border-방향-width:두께;

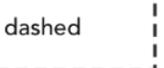
border-top-width:5px;

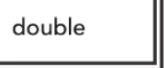
### 2. border-style:외곽선의 모양을 지정할 수 있다.

- 외곽선 종류:solid, dotted, dashed,double,groove,ridge,inset,outset,none,hidden

 solid

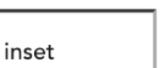
 dotted

 dashed

 double

 groove

 ridge

 inset

 outset

border-style:스타일 종류;

border-style:solid;

border-방향-style:스타일 종류;

border-left-style:solid;

### 3. border-color:외곽선의 색상을 설정한다.

border-color:칼라값;

border-color:#000;

border-방향-color:칼라값;

border-right-color:rgb(200 56 78);

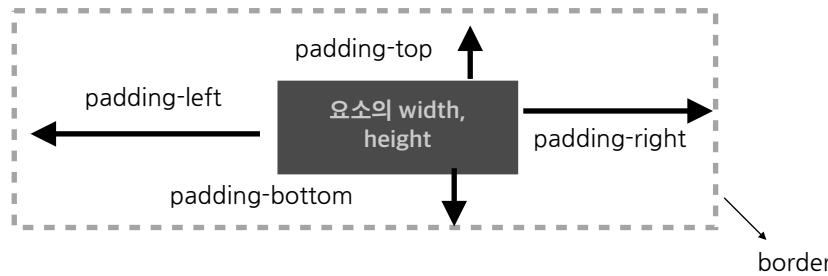
### 4. border:네방향의 외곽선의 두께, 스타일, 색상이 모두 같다면 border를 이용한다.

border:width style color

border:5px solid #000;

## padding

- 콘텐츠의 안쪽 여백으로 요소의 border를 기준으로 안쪽에 생긴다.
- width와 height값에 영향을 준다.'



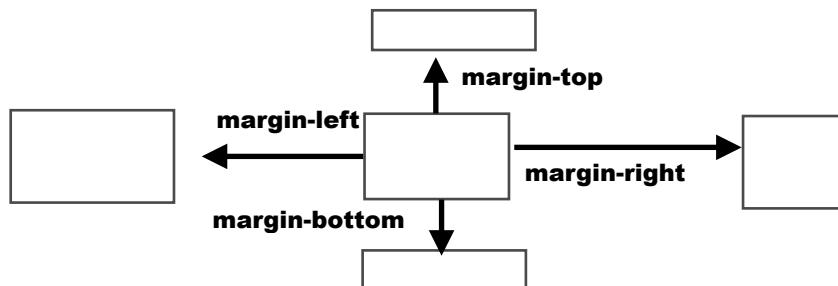
padding:길이 , % , auto , inherit

### padding값 지정하기

네 방향 모두 같을 때	top=bottom=left=right	padding:값;	padding:5px;
두 방향이 같을 때	top=bottom, left=right	padding:top right	padding:5px 10px;
한 방향이 같을 때	left=right	padding:top right bottom	padding:5px 10px 7px
네 방향 모두 다를 때		padding:top right bottom left	padding:5px 10px 8px 4px
한 방향만 지정할 때		padding-방향:길이	padding-top:5px;

### margin

- 블록 요소와 블록 요소 사이를 구분하는 빈 공간
- border를 기준으로 바깥쪽 여백이다.
- 투명으로 채워져 있으며, width와 height값에 영향을 주지 않는다.



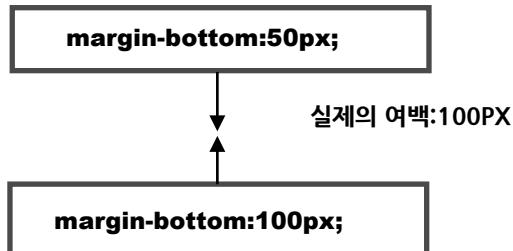
margin:길이 , % , auto , inherit

### margin값 지정하기

네 방향 모두 같을 때	top=bottom=left=right	margin:값;	margin:5px;
두 방향이 같을 때	top=bottom, left=right	margin:top right	margin:5px 10px;
한 방향이 같을 때	left=right	margin:top right bottom	margin:5px 10px 7px
네 방향 모두 다를 때		margin:top right bottom left	margin:5px 10px 8px 4px
한 방향만 지정할 때		margin-방향:길이	margin-top:5px;

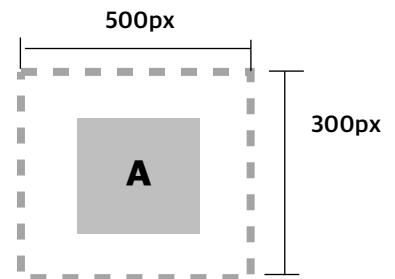
### margin 겹침 현상(margin collapsing)

두개 이상의 margin값이 세로 방향으로 만났을 때 하나의 margin으로 합쳐지는 현상.



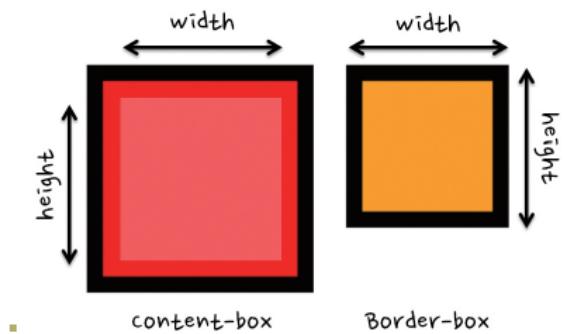
### width와 height값 정하기

padding-left	padding-right	border-left	border-right	width
20px	50px	5px	45px	
50px	10px	15px	25px	
70px	25px	8px	12px	
35px	45px	10px	6px	



## 크기 기준 정하기

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

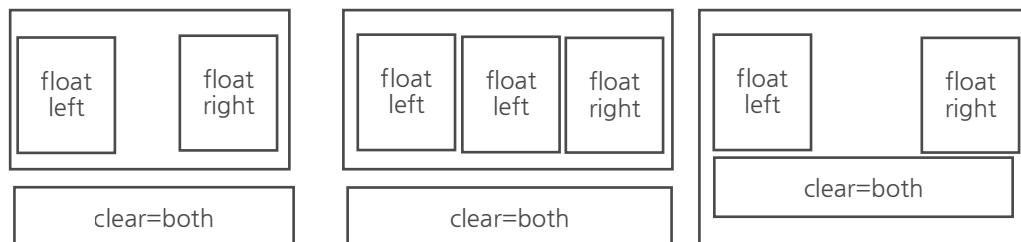


# 위치지정하기1

## float

- 요소의 박스를 부모 요소 안에서 왼쪽 혹은 오른쪽으로 이동 시키는 기능(인라인 요소인 경우 자동으로 block성격을 갖는다.)
- float된 요소는 다른 것에 방해 받지 않고 원하는 위치에 떠 있으면서 주변 박스의 위치를 변경 시킨다.
- float 된 요소는 다른 박스 위에 떠 있으므로 다음의 오는 박스의 인라인 적인 요소들은 float된 박스에 밀려서 보인다.
- 해결방법은 다음에 오는 요소에게는 clear속성을, float를 자식으로 가지는 요소에게는 overflow 속성을 사용한다.

```
float:left | right | none | inherit;
```



## clear

플로트 된 요소 다음에 오는 요소에게 플로트의 영향에서 벗어나서 자신이 가진 속성에 따라 구현 되도록 해준다  
블록 요소나 블록 요소 대체할 수 있는 요소에 지정해야 한다

```
clear:none | left | right | both | inherit
```

## overflow

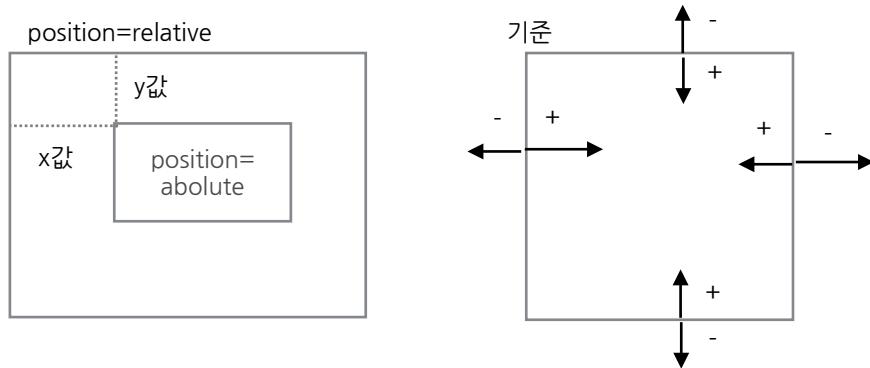
- 부모 box요소가 자식의 크기를 감지하고
- 만약 자식(콘텐츠)요소이 부모요소보다 크기보다 더 클 경우 상황을 처리 하고자 할 때 사용 합니다.

```
overflow:visible | hidden | scroll | auto | overflow-x | overflow-y | inherit
```

```
overflow : visible;  
overflow : hidden;
```

## position

- 박스요소를 배치하는 방법이다.
- float과는 다르게 왼쪽, 오른쪽 정렬하는 형태가 아닌, 위치하는 곳에 대한 값(오프셋)을 지정



### static

- 요소이 본래 성격으로 자리 배치

### relative

- 요소의 본래 위치으로 이동한다.
- relative가 된 요소는 자신의 offset 값으로 움직이고 자신의 원래 있어야 할 곳 위치에는 자신의 영역을 그대로 유지한다..
- 자신의 위치가 좌표 값으로 offset이 되면 다른 요소와 층(layer)을 가지게 된다.
- 자신의 자식요소 중 absolute요소가 있으면 자신의 기준 점으로 위치를 잡게 한다.

### absolute :

- 자신의 부모나 조상의 position속성이 있을 경우 그 요소의 위치를 이동한다.
- 자신을 포함한 컨테이닝 블록만 지정한 offset값으로 움직인다.
- 본문 위에 떠 있는 형태로 다른 요소에 방해하지 않고 층(layer)을 만든다.

### fixed :

- 스크린을 기준으로 위치를 이동한다.

position : static | relative | absolute | fixed | inherit

top, bottom:y축 값

right, left:y축 값

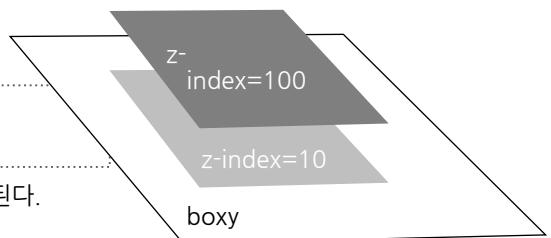
position:absolute; top:150px; left:200px;

## z-index

- layer를 가진 요소 박스가 겹쳐지는 순서를 지정할 때 사용 합니다.
- 수치가 작을수록 아래쪽에 있다는 것을 의미함.
- body는 기준이 0값이다.

z-index : auto | 정수(1, 2, 3 ...);

\*\*만약에 순서를 정하지 않는 경우 박스의 생성 순서에 따라 정수로 할당된다.



# 타이포그래피

## font-family

- 글자의 모양을 설정한다.
  - 하나 이상의 글꼴을 선언할 수 있으며 글꼴은 콤마(,)로 구분한다.
  - 주의사항
    1. 한글글꼴은 “ ” 반드시 필요하다. 한글 글꼴 뒤에는 꼭 글꼴의 영문이름을 추가한다.
    2. 영문글꼴이더라도 글꼴이름에 공백이 있는 경우 “ ” 필요하다.(ex:PT mono, Malgun Gothic)
  - 해당 글꼴이 없을 경우 제일 마지막에 대표 패밀리체가 렌더링되며, 만약에 폰트 페밀리를 지정하지 않으면, 사용자 컴퓨터에 설정된 브라우저 글꼴로 렌더링 된다.

형식:font-family:1순위글꼴패밀리명 . 2순위글꼴패밀리명 . 3순위글꼴패밀리명... 대표패밀리명

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

대표 패밀리란?

serif(인쇄물), sans-serif(온라인) cursive, fantasy, monospace

serif: 문자 끝부분에 장식용 가시와 고리모양고상하고 고전적인 느낌을 준다.

Times, Times New Roman, Georgia, 바탕, 궁서

sans-serif: 고리모양이 없다. 깔끔하고 읽기쉽다.

Verdana Arial black Arial 굴림 돌울

Cursive(흘림체): 손으로 쓴것같은 폰트 (필기체)

Caffish script. Sanvito

Fantasy 화려한 장식

Impact

**Monospace**      익정한 폰트의 문자 폰트로 이루어진 타자기로 타이핑한 것 같은 느낌

Courier Prestige

**font-size** 1 웹 문서의 글자 크기를 지정한다.

- 방식:  
    키워드:xx-small, x-small, small, normal, large, x-large, xx-large  
    절대 삼대크기:pt px % em

`font-size`: 키워드 또는 길이(상대, 절대, inherit)

font-size:12px;  
font-size:1.5em;  
font-size:150%

## font-weight

- 글꼴의 굵기를 표현
  - 방식:

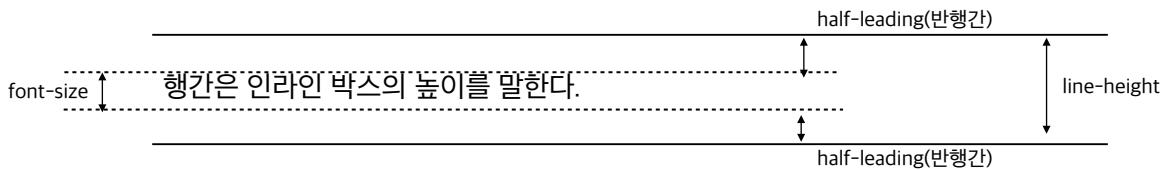
- 키워드: normal, bold, bolder, light, lighter
- 숫자: 100(lighter)-400(normal)-600(bold)-900(bolder)

```
font-weight:키워드 , 또는 100-900,inherit
```

```
font-weight:normal;  
font-weight:bold;  
font-weight:900;
```

## line-height

- 문단의 행간을 지정 할 때 사용한다.
- line-height는 font-size로 지정된 글자 크기를 포함하여 글자의 위쪽과 아래쪽 여백까지를 말한다.



- font-size보다 작게 지정할 경우에는 글자가 겹칠 수 있다.
- 보통은 120%로 신문등 글자가 많은 경우(신문)는 150%로 사용되는 경우가 많다.
- 수치는 짹수로 사용하는 것이 좋다.

```
line-height:normal,실수,길이,%,em,inherit
```

line-height:	normal;	기본값 = 1.2
line-height:	1.6;	normal의 1.6배
line-height:	16px;	
line-height:	200%	

## font-style

- 글꼴의 스타일을 지정할 때 사용합니다.

```
font-style:normal ,italic,oblique;
```

```
font-style : normal ;  
font-style : italic ;  
font-style : oblique ;
```

## font-variant

- 영문 글꼴의 소문자를 대문자로 변형할 때 사용합니다.
- 값 : normal | small-caps

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

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

if I had thought about it I might have been scared.

IF I HAD THOUGHT ABOUT IT I MIGHT HAVE BEEN SCARED.

## font

- font-family, font-size, line-height, font-weight, font-style, font-variant 의 6가지 속성을 한번에 선언할 때 사용하는 대표 속성이다.
- 선언순서를 지켜야 하고, **font-size와 font-family**는 꼭 선언해야 한다.

```
font:[ font-weight | font-style | font-variant ] | [ font-size ] | [ /line-height ] | [ font-family ] ;
```

```
font : bold italic small-caps 12px /1.6 “돋움”,Datum,sans-serif;
```

## text-indent

- 문단의 첫 줄 들여쓰기를 적용(값 적용 가능값만큼 첫줄만 앞당겨짐)
- 블록요소에만 적용 가능 (인라인요소인 a, span태그에는 안먹힘)
- 음수도 가능하다. 하지만 음수인 경우에는 같은 값인 안의 여백이 필요하다.

```
text-indent:길이 | 퍼센트 | inherit ;
```

```
text-indent:50px;  
text-indent:2em;  
p{text-indent:-50px; padding-left:50px;}
```

## text-align

- 단락의 텍스트를 가로 기준으로 정렬할 때 사용 합니다.

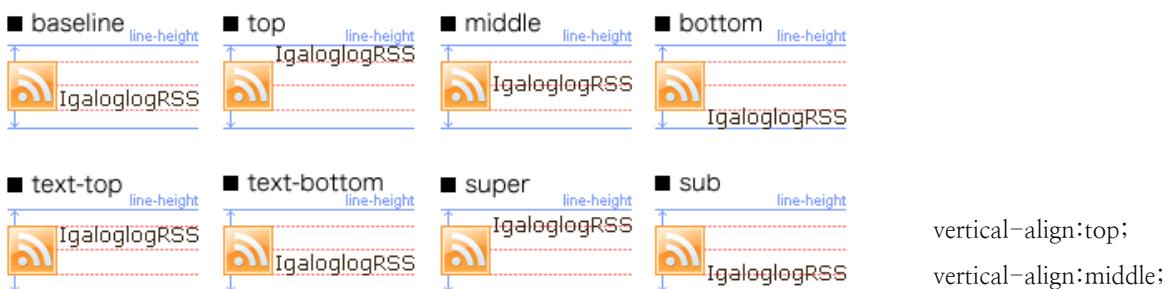
```
text-align:left | center | right | justify | inherit ;
```

```
text-align : left;  
text-align : center;  
text-align : justify;
```

## vertical-align

- 인라인 요소끼리의 세로 위치를 정렬할 때 사용된다.
- 인라인요소(이미지, 인풋, 스패n 등), 일부 블록요소(테이블의 th, td)에 사용가능
- 지정한 인라인요소 뒤에 오는 요소에 대한 세로정렬 가능

```
vertical-align: baseline | sub | super | top | text-top | middle | bottom | text-bottom | 길이 | 퍼센트 | inherit;
```



## text-decoration

- 텍스트에 밑줄을 긋거나 취소선 등의 효과를 추가할 때 사용한다.

```
text-decoration:none | underline | overline | line-through | blink | inherit
```

text-decoration : underline

text-decoration: underline;

text-decoration : line-through none;

~~text-decoration: line-through;~~

text-decoration :overline;

text-decoration: overline;

text-decoration :none;

text-decoration: none;

## letter-spacing

문자 간격을 조절할 때 사용된다. 음수 양수 모두 사용가능.

letter-spacing:길이 | normal | inherit

letter-spacing:1em; (글자크기에 비례하여 조절)

letter-spacing:-3px;

## word-spacing

단어와 단어 사이의 간격을 조절

word-spacing:길이 | normal | inherit

word-spacing:1em;

word-spacing:-3px;

## text-transform

영문 대소문자를 변환할 때 사용 합니다.

text-transform : none | uppercase | lowercase | capitalize | inherit

text-transform : none;

This is a paragraph(none)

text-transform : capitalize;

This Is A Paragraph(capitalize)

text-transform : uppercase;

THIS IS A PARAGRAPH(UPPERCASE)

text-transform : lowercase;

this is a paragraph(lowercase)

## white-space

공백 문자를 처리하는 방법

pre:입력한 형태 그대로 브라우저에 렌더링

nowrap:무조건 한 줄로 출력됨.

white-space : normal | nowrap | pre

# Color와 Background

## color

요소의 글자 색상을 지정할 때 사용 합니다.

color:색상명 | RGB 16진수 | RGB 10진수 | RGB 백분율 | inherit

```
color : red ;  
color : #ff0000 ;  
color : rgb(255,0,0) ;  
color : rgb(100%,0%,0%) ;
```

## background-color

요소의 배경 색상을 지정할 때 사용 합니다.

background-color:색상명 | RGB 16진수 | RGB 10진수 | RGB 백분율 | inherit

```
background-color : red ;  
background-color : #ff0000 ;  
background-color : rgb(255,0,0) ;  
background-color : rgb(100%,0%,0%) ;
```

## background-image

요소의 배경 이미지를 지정할 때 사용 합니다.

background-image:none | uri(경로명) | inherit

```
background-image: none ;  
background-image: url(images/bg.gif) ;
```

## background-repeat

요소에 지정한 배경 이미지의 반복 여부를 변경하고자 할 때 사용 합니다.

repeat:x,y 축 모두 반복  
repeat-x:x축만 반복  
repeat-y:y축만 반복  
no-repeat:반복 없어.

background-repeat:repeat | repeat-x | repeat-y | no-repeat | inherit

```
background-image: url(images/bg.gif) ;background-repeat:no-repeat;  
background-image: url(images/bg.gif) ;background-repeat:repeat-y;
```

## background-position

요소에 지정한 배경 이미지의 위치를 변경하고자 할 때 사용 합니다.

background-position:가로 세로

	가로	세로
키워드	left, right center	top, bottom center
길이	px	px
상대값	%, em	%, em

```
background-image: url(images/bg.gif) ;background-position:left top;
background-image: url(images/bg.gif) ;background-repeat:100px 200px;
```

## background-attachment

요소에 지정된 배경 이미지를 스크롤되어도 항상 같은 위치에 고정시키고자 할 때 사용합니다.

scroll:스크롤 적용

fixed: 지정한 스크린 위치에 그대로 남음.

background-attachment:scroll | fixed | inherit

```
background-image: url(images/bg.gif) ;
background-attachment:fixed;
```

## background

font 속성과 같이 배경과 관련된 다섯 가지 속성을 단축해서 한 번에 선언하고자 할 때 사용합니다.

background:color | image | repeat | position | attachment | inherit

```
background: yellow url(images/bg.gif) no-repeat left top fixed ;
```

## background Image중첩사용

### 1. background-image

- 중첩사용이 가능하다.

```
background:url(img/coforwardQr2.png);  
background:url(img/checkBox1.png) left top no-repeat,  
url(img/checkbox2.png) right top no-repeat,  
url(img/coforwardText.png);background-size
```

- IE8이하 사용 불가

## background-size : [x] [y]:

[x]% [y]%	적용되는 요소의 크기에 비례하여 배경 이미지 적용
[x]px [y]px	절대 크기로 배경 이미지 적용
cover	배경 이미지를 늘여 적용되는 요소 전체에 표시
contain	배경 이미지의 가로 세로 비율을 맞춰 요소에 표시할수 있는 최대 크기로 표시

```
#box_rel{background-size:40% 70%;}  
#box_abs{background-size:50px 50px;}  
#box_cover{background-size:cover;}
```

## 리스트 꾸미기

### list-style-type

목록을 생성할 때 기본적으로 제공되는 마커를 변경하고자 할 때 사용합니다.

```
list-style-type: none | disc | circle | square | decimal | decimal-leading-zero | upper-alpha | lower-alpha  
upper-roman | lower-roman | upper-latin | lower-latin | lower-greek | armenian | georgian | inherit
```

```
list-style-type: none;  
list-style-type: circle;
```

### list-style-image

목록 마커 대신 이미지로 변경할 때 사용합니다.

```
list-style-image: url() | none | inherit
```

```
list-style-image : url(images/icon.gif);
```

## Table

### border-collapse

테이블의 border와 cell의 보더를 표시하는 방법을 지정할 때는 사용 합니다.

table요소에 적용해야 한다.

```
border-collapse: separate | collapse | inherit
```

```
border-collapse :separate;  
border-collapse :collapse;
```

## 선택자2

### 기본선택자

선택자	기능설명
요소	특정 요소 선택
#id명	특정 id명을 가진 요소 선택
.class명	특정 class명을 가진 요소 선택
요소1 요소2	요소1 안의 있는 모든 요소자손 요소2
부모요소1>자식요소2	부모요소1 안에 있는 자식 요소2
형제요소1+형제요소2	형제요소1 옆에 있는 형제 요소2
형제요소1~형제요소2	요소1 의 모든 형제 요소2
선택자1, 선택자2	여러 선택자 한꺼번에 선택
*	모두 선택

### 속성선택자

선택자[속성~=값]	속성 안의 값이 특정 값을 단어로 포함하는 태그를 선택
선택자[속성!=값]	속성 안의 값이 특정 값을 단어로 포함하는 태그를 선택
선택자[속성 ^=값]	속성 안의 값이 특정 값으로 시작하는 태그를 선택
선택자[속성\$=값]	속성 안의 값이 특정 값으로 끝나는 태그를 선택
선택자[속성*=값]	속성 안의 값이 특정 값을 포함하는 태그를 선택
[속성 선택자][속성선택자]	

### 가상 선택자

:first-child	첫 번째에 위치하는 자손을 선택
:last-child	마지막에 위치하는 자손을 선택
:first-letter	첫 번째 글자를 선택
:first-line	첫 번째 줄을 선택
:before	선택자의 앞에 가상 공간을 만든다.
:after	마지막에 위치하는 자손을 선택

content: 원하는 값을 넣어 준다.

이미지: url()

텍스트: “ 내용 ”

스크립트: “ “ 없이 넣기

### 하이퍼 링크태그 관련 선택자

a	하이퍼링크요소
a:link	하이퍼링크에 연결 경로가 있는 경우
a:visited	하이퍼링크 주소에 전에 방문한적 있는 경우 (쿠키에 등록 되어 있는 경우)
a:hover	하이퍼링크에 마우스가 를 오버인 상태
a:active	하이퍼링크의 활성 상태
:focus	사용하기 위해 포커스가 맞추어져 있는 상태

## css3

편집자초안(Editor's Draft)

초안 Working Draft

최종 초안 Last Call Working Draft

권고 후보

권고안

권고(Recommendation)

css3는 현재 완벽하게 완성되어 있지는 않다.

css3 자체가 브라우저 개발사들이 하나 둘 출시 중인 모듈을 모아놓은 것들로 아직까지는 완벽하게 실험에 통과 하지 못한 스펙도 있고 실험중인 스펙을 부라우저사에서 먼저 사용하는 경우도 있다. 이러한 문제점이 있기 때문에 우리는 이러한 속성을 사용 할 때는 브라우저의 접두어를 사용하여야 한다.

### 접두어 사용하기(vendor Prefix)

벤더 프리픽스는 웹 브라우저 공급 업체(마이크로소프트, 구글, 모질라, 애플, 오페라)에서 제공하는 실험적인 기능을 사용할 때 사용한다.

예를 들어 변환 속성은 CSS3 표준에 있지만 아직 완벽하게 제정된 상태가 아니다. 따라서 웹 브라우저 업체가 무턱대고 속성을 추가할 수 없다.

하지만 다른 웹 브라우저를 이기려면 새로운 기능을 모두 제공해야 하므로 벤더 프리픽스를 사용해 지원을 하는 것이다.. 각 웹 브라우저마다 다음과 같은 벤더 프리픽스를 갖는다.



사용 방법 및 순서:

-webkit-속성

-moz-속성

속성

-webkit-boxshadow  
-moz-boxshadow  
boxshadow

## FONT(@font-face)

- 브라우저 사용자의 컴퓨터에 폰트가 설치돼 있지 않아도 사용자의 컴퓨터등에서 내려 받을 수 있는 경로를 지정하여 해당 폰트를 사용할 수 있게 하는 방법

### 사용 가능 글꼴 파일 형식

truetype(ttf)  
opentype(otf),  
embedded-opentype(eot)  
scalable-vector-graphic(svg, svgz)

### 사용법

- 지정: @font-face{ font-family:'폰트이름';  
src : url('폰트경로') format('포맷이름');  
src : url('폰트경로') format('포맷이름'),  
url('폰트경로') format('포맷이름'); }
- 사용: 선택자{fontfamily: '폰트이름', 대표폐밀리체명;}

```
@font-face {
  font-family: 'ChunkFiveRegular';
  src: url('fonts/chunkfive.eot');
  src: url('fonts/chunkfive.eot') format('embedded-opentype'),
       url('fonts/chunkfive.woff') format('woff'),
       url('fonts/chunkfive.ttf') format('truetype'),
       url('fonts/chunkfive.svg') format('svg');
}
```

브라우저	포맷			
	eot	woff	ttf / otf	svg
크롬(모두)				●
크롬 6+		●	●	●
파이어폭스 3.5			●	
파이어폭스 3.6+		●	●	
IE 5 ~ 8	●			
IE 9+	●	●	●	
오페라 10+			●	●
사파리 3.1+			●	●
iOS 4.2 미만				●
iOS 4.2+			●	●

국내 무료서체:

서울서체, 제주서체, skt 외비우스테, 아모레퍼스픽 아리따체

구글 폰트 사용하기:

<https://www.google.com/fonts/earlyaccess>

폰트 컨버트 사이트:

<http://www.fontsquirrel.com>

# Shadow

## box-shadow

box-shadow:[dx] [dy] [blur radius] [spread] [칼라] [set];

- 지정된 요소의 그림자를 넣어준다.
- 블록 요소 뿐 아니라 이미지등에도 사용할 수 있다.

dx	그림자의 가로방향 위치
dy	그림자의 세로방향 위치
blur	흐려짐 정도
spread	변침 정도
set	그림자의 형태 outset : 요소 바깥쪽으로 그림자가 떨어짐 (기본값) inset : 요소 안쪽으로 그림자가 떨어짐

box-shadow:10px 20px 10px 5px #666666 inset;

## text-shadow

text-shadow:[dx] [dy] [blur] [color];

- 텍스트에 그림자를 지정함
- 속성

dx	그림자의 가로방향 위치
dy	그림자의 세로방향 위치
blur	흐려짐 정도

text-shadow:3px 3px 5px #000;

\*\*여러개의 그림자 사용하기:text-shadow:3px 3px 5px #000,3px 43px 3px #FC0;

## background2

### background-clip

배경 속성이 적용되는 영역을 지정

border-box	백그라운드 적용 영역을 보더에서부터 한다.
padding-box	백그라운드 적용 영역을 안의 여백에서부터 한다.
content-box	백그라운드 적용 영역을 콘텐츠에서부터 한다.

### background-origin

배경 이미지의 시작점을 지정

border-box	백그라운드 시작을 보더에서부터 한다.
padding-box	백그라운드 시작을 안의 여백에서부터 한다.
content-box	백그라운드 시작을 콘텐츠에서부터 한다.

### gradient

- 아직 실험단계에 있는 대표적인 속성이다. 각 브라우저마다 형식이 달리 표현되며, 요즘에 와서 조금 형식이 통일되기 시작했다.
- 절대로 중요한 부분에서는 사용하지 말아야 한다. 브라우저마다 사양이 많이 차이난다.

#### linear형식

background: linear-gradient(방향, 칼러 위치, 칼러 위치, 칼러 위치);

```
background: -webkit-linear-gradient(45deg, red 10%, yellow 65%, #666 85%, blue 95%); /* For Safari 5.1 to 6.0 */
background: -moz-linear-gradient(45deg, red 10%, yellow 65%, #666 85%, blue 95%); /* For Firefox 3.6 to 15 */
background: linear-gradient(45deg, red 10%, yellow 65%, #666 85%, blue 95%); /* Standard syntax (must be last) */
```

#### radial형식

background: radial-gradient(칼러 위치, 칼러 위치, 칼러 위치);

```
background: -webkit-radial-gradient(red 5%, green 15%, blue 60%); /* Safari 5.1 to 6.0 */
background: -moz-radial-gradient(red 5%, green 15%, blue 60%); /* For Firefox 3.6 to 15 */
background: radial-gradient(red 5%, green 15%, blue 60%); /* Standard syntax (must be last) */
```

참고 사이트 <https://chrome.google.com/webstore/search/colorzilla>

<http://www.colorzilla.com/>

## border2

### border-radius

- Box Model 외각선을 부드럽게 해 준다.

1. 기본형: border-radius:x1 x2 x3 x4/y1 y2 y3 y4;

border-radius:20px 30px 5px 15px/45px 23px 34px 5px;

2. 각 방향 다르게 설정하기

border-top-left-radius

border-top-right-radius

border-bottom-left

border-bottom-right

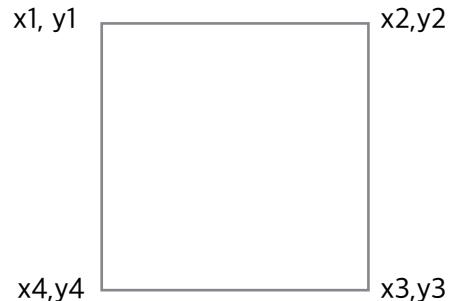
border-top-left-radius:20px;

border-top-right-radius:20px;

border-bottom-left-radius:20px;

border-bottom-right-radius:20px;

border-radius: 50px 100px 200px 5px;



### border-image

- 테두리에 이미지 넣기

#### 1. 속성:

- source : 외각선에 지정될 이미지 경로
- 이미지영역 : 외각선으로 사용하기 위한 가로, 세로 방향의 이미지 영역
- 표시방법 : 이미지를 표시하는 방법

속성	설명
border-image-slice	테두리에 지정할 이미지가 요소 박스 상하좌우 테두리 영역에 보여질 크기를 지정
border-image-repeat	요소 박스에 지정된 이미지가 반복되는 스타일을 지정 repeat, round, stretch
border-image-width	각 방향별로 border 이미지 표시 영역의 길이를 지정
border-image-outset	테두리에 지정한 이미지가 border 영역에서 벗어나 요소 박스 외부에 표시되는 이미지 영역을 지정
border-image	요소 박스의 테두리 이미지 지정과 관련된 속성을 일괄 적용하는 대표 속성

## 변형과 움직임

2차원 변환	9	4.0	3.5	3.2	12.1.10.5
3차원 변환	10	12.0	16	4.0	23.0.15.0
애니메이션	10	4.0	5.0	4.0	12.1.12.0

## 변형

### Transform

- 요소의 형태를 변경함

```
#sample_01{transform:속성명(값);}
```

translate([가로거리],[세로거리])	지정한 크기 만큼 이동 시kim translateX, translateY
scale([가로비율, 세로비율])	지정한 배율 만큼 확대함 scaleX, scaleY
rotate([각도])	지정한 각도만큼 회전 시kim (degree:deg)
skew([x각도],[y각도])	지정한 경사로 기울임 skewX, skewY
perspective(length);	

```
#sample_01{transform:translate(50px, 30px);}
```

```
#sample_02{transform:scale(2);}
```

```
#sample_03{transform:rotate(45deg);}
```

```
#sample_04{transform:skew(30deg, 30deg);}
```

- 다중입력일 때는 공백으로 구분

```
#sample_04{transform:skew(30deg, 30deg) rotate(45deg);}
```

### • transform-origin

transform의 기준점을 정해준다.

transform-origin: 가로 세로

# transition

## Transition

- 위치 속성 top, left, bottom, right
- 크기 속성 width, height
- 박스 속성 margin, padding
- 테두리 속성 border-width, border-radius, border-color
- 색상 속성 color, background-color
- 투명도 속성 opacity
- 변환 속성 transform

transition-delay	움직이기 이전의 기다리는 시간
transition-property	트랜지션할 속성(none, all)
transition-duration	트랜시션 시간
transition-timing-function	이펙트의 가속조절 (ease-in, ease-out, cubic-bezier, linear)

transition: 속성 시간 옵션 delay

transition:background 3s ease;

- transition 효과는 처음에는 적용안되고 마우스를 올리거나 다른이벤트에 다른 변형(transform)을 줄때만 발생된다. 그리고 이벤트가 제거되면 처음 변형(transform)값으로 다시 transition 효과가 적용된다.
- 처음부터 transition 효과를 내려면 animation (애니메이션 @keyframes) 효과를 적용하여야 한다.

## Column

블록 요소에 단을 설정해 준다.

column-width	다단 구성 시 단의 너비를 지정
column-count	다단 구성 시 단의 개수를 지정
column-gap	단과 단 사이의 간격을 지정
column-rule	단과 단 사이의 구분선을 표현하기 위한 속성

```
-webkit-column-width:300px;
-webkit-column-count: 2;
-webkit-column-rule: #999 solid 2px;
-webkit-column-gap:20px;
```

- columns : 단의 너비와 개수를 일괄 지정하기 위한 대표 속성

-webkit-columns : 300px 2 ;

## display:table

display:table, display:table-cell, display:table-caption, display:table-row

table	부모요소
table-cell	자식요소
table-caption	테이블의 제목
table-row	테이블의 가로(tr기능)

display:inline-table (적용할 요소)

장점: 가운데 정렬 가능, 내용: 가로, 세로 정렬 가능 (정중앙)

border-spacing:값 css2 td,th 간격 (부모요소)

### • animation (에니메이션) @keyframes

- transition 효과와 비슷하지만 transition은 객체에 직접 적용하지만, animation은 별도의 이름을 가진 keyframe을 생성시켜 프레임 구간별로 다른 속성을 적용한 다음 적용 할 요소에 미리정의된 animation 이름과 여러가지 애니메이션속성을 적용한다.

```
@keyframes 애니메이션이름 {
```

```
  from {초기상태속성값}
  키프레임구간위치 % {해당구간 속성값}
  to { 종료상태 속성값}
```

## animation

animation: name time function counter direction

animation-name	적용할애니메이션(keyframes) 이름
animation-delay	실행전 지연시간
animation-duration	1회실행시간
animation-iteration-count	반복횟수 ( infinite 무한반복)
animation-timing-function	가속도형태 ( linear일정, ease중간빠르고시작끝은느리게, ease-in끝을빠르게, ease-out끝을느리게, ease-in-out시작끝을느리게 )
animation-direction	반복형태 normal 다시처음부터, alternate 역방향재생

- 주의) 애니메이션을 사용할때는 움직이기 전에 변화시켜줄 속성의 초기값이 설정되어있어야 한다.
- 처음부터 움직임을 진행할때는 animation 기능을 사용하는게 좋고 , 마우스반응으로 움직일때는 transform과 transition 효과를 사용하는 것 이좋다.

## css3선택자

### 일반구조 선택자

:first-child	첫번째 자식요소 선택
:last-child	마지막 자식요소 선택
:nth-child(수열)	자식관계에서 앞에서 수열 번째 요소 선택
:nth-last-child((수열))	자식관계에서 뒤에서 수열 번째 요소 선택

### 수열

:nth-child(odd)	
:nth-child(even)	
:nth-child(숫자)	:nth-child(3)
:nth-child(@n)	:nth-child(3n)
:nth-child(@n+숫자)	:nth-child(3n+2)



### 형태구조 선택자

:first-of-type	형제 관계 중에서 앞에서 첫 번째로 등장하는 요소 선택
:last-of-type	형제 관계 중에서 앞에서 마지막으로 등장하는 요소 선택
:nth-of-type(수열)	형제 관계 중에서 앞에서 수열 번째로 등장하는 요소 선택
:nth-last-of-type(수열)	형제 관계 중에서 뒤에서 수열 번째로 등장하는 요소 선택

### 반응문자선택자

::selection	사용자가 드래그한 글자를 선택
-------------	------------------

### 반응 선택자

:hover	마우스가 오버인 상태
:active	마우스가 클릭 한 상태

### 링크선택자

:link	하이퍼링크에 연결 경로가 있는 경우
:visited	하이퍼링크 주소에 전에 방문한적 있는 경우 (쿠키에 등록 되어 있는 경우)

### 상태선택자

:checked	체크 상태인 input
:focus	초점이 맞추어진 input

# 위치지정하기3: display

## display

요소가 가지고 있는 박스의 성격 변경할 때 사용 합니다.

### 고전 방식 기법

inline요소를 block요소로, block요소를 inline요소로 변경 가능

none : 요소의 존재를 없애줌. 스크린 리더 등에서 접근이 안 될 수 있다.

block: 인라인 요소를 블록 요소로 만들어 줌

inline : 블록요소를 인라인 요소로 만들어 줌

inline-block: inline요소처럼 배치되면서 block요소의 속성(width, height, margin, padding)들을 사용할 수 있는 속성

display : none | block | inline | inline-block | inherit

p{display:inline;}

## Flexbox

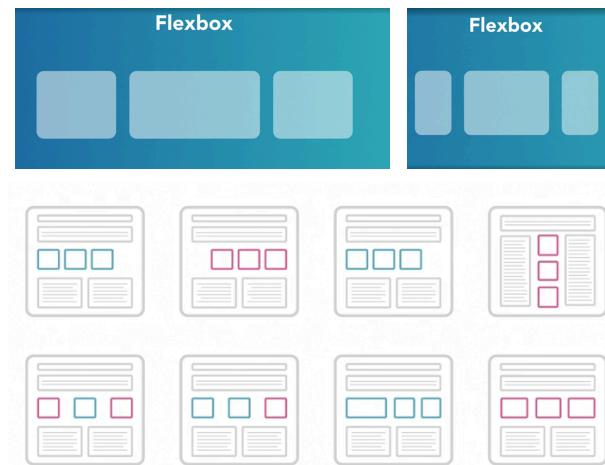
### Flexbox와 Grid 비교

- Flexbox와 Grid는 모두 이전 기술보다 더 진보 된 레이아웃을 만드는 데 사용할 수 있는 새로운 레이아웃 모듈입니다. 유사한 구문 과 일부 동일한 속성을 공유하지만 각각 고유 한 특성이 있습니다



### Flexbox

Flexbox는 요소를 정렬하고 크기를 조정하고보다 유연한 레이아웃을 만든다.



## 용어와 구조

flex-container: 기준이 되는 부모 요소

display, flex-flow,

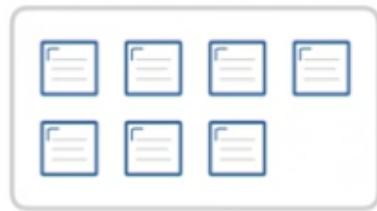
justify-content, flex-direction, flex-wrap, align-content, align-items

flex-items:flex-container 내의 직접적인 자식 요소

order, flex, align-self, flex-grow, flex-shrink, flex-basis



flex-container



flex-items

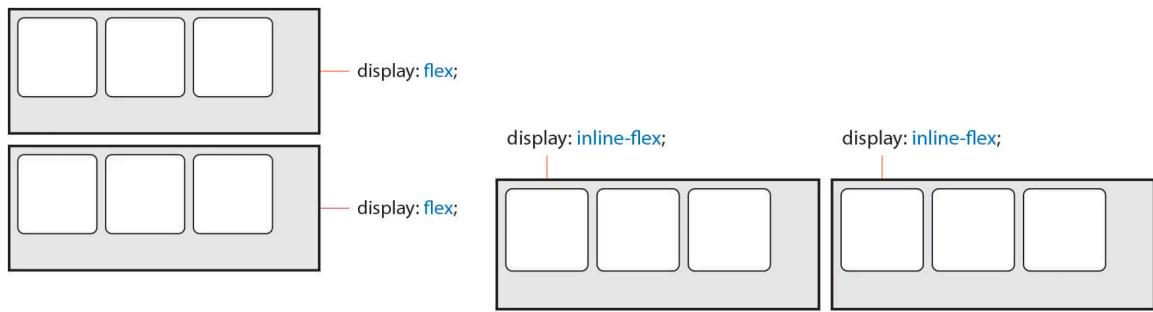
## flex-container 속성

display	Flex Container를 정의
flex-flow	flex-direction와 flex-wrap의 단축 속성
flex-direction	Flex Items의 주 축(main-axis)을 설정
flex-wrap	Flex Items의 여러 줄 묶음(줄 바꿈) 설정
justify-content	주 축(main-axis)의 정렬 방법을 설정
align-content	교차 축(cross-axis)의 정렬 방법을 설정(2줄 이상)
align-items	교차 축(cross-axis)에서 Items의 정렬 방법을 설정(1줄)

## display:flex, inline-flex

display: flex;로 지정된 Flex Container는 Block 요소와 같은 성향(수직 쌓임)을 가지며,

display: inline-flex로 지정된 Flex Container는 Inline(Inline Block) 요소와 같은 성향(수평 쌓임)을 가집니다.



## flex-flow: direction wrap

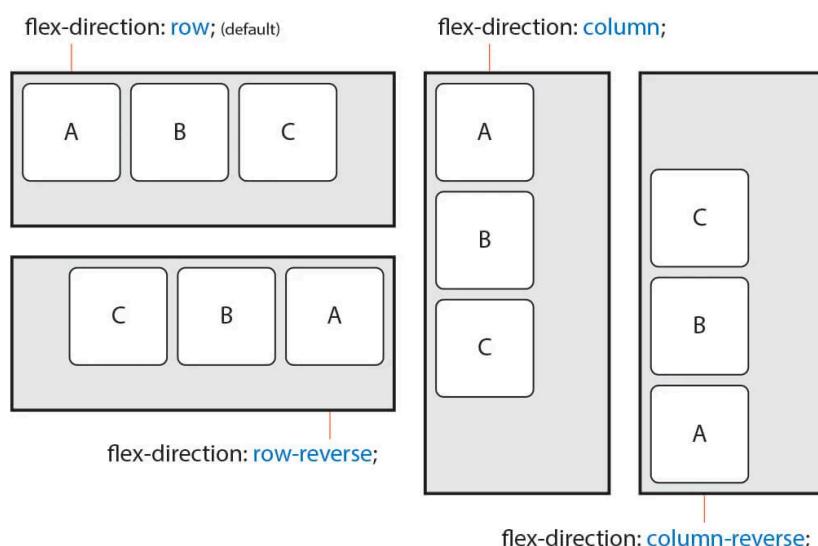
단축 속성으로 Flex Items의 주 축(main-axis)을 설정하고 Items의 여러 줄 묶음(줄 바꿈)도 설정합니다.

값	의미	기본값
<code>flex-direction</code>	Items의 주 축(main-axis)을 설정	<code>row</code>
<code>flex-wrap</code>	Items의 여러 줄 묶음(줄 바꿈) 설정	<code>nowrap</code>

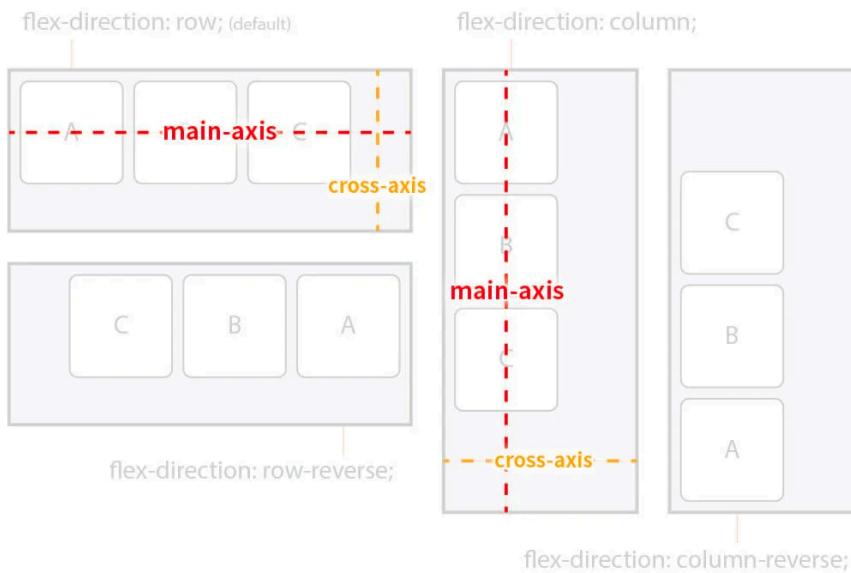
## flex-direction

Items의 주 축(main-axis)을 설정합니다.

값	의미	기본값
<code>row</code>	Items를 수평축(왼쪽에서 오른쪽으로)으로 표시	<code>row</code>
<code>row-reverse</code>	Items를 <code>row</code> 의 반대 축으로 표시	
<code>column</code>	Items를 수직축(위에서 아래로)으로 표시	
<code>column-reverse</code>	Items를 <code>column</code> 의 반대 축으로 표시	



### \*\*\*축(main-axis)과 교차 축(cross-axis)

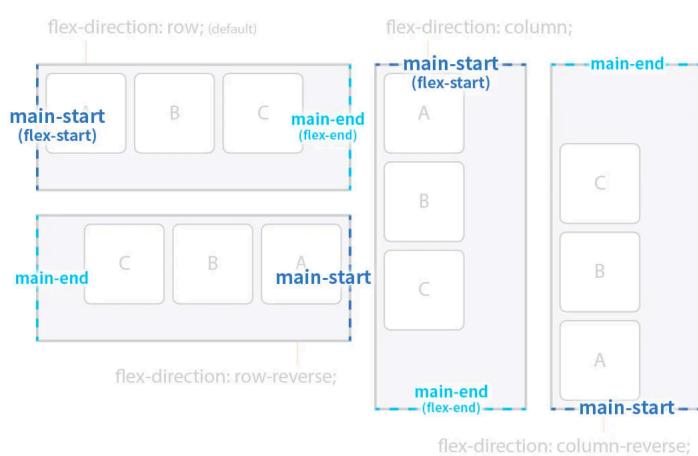


값 **row**는 Items를 수평축으로 표시하므로 이때는 주 축이 수평이며 교차 축은 수직이 됩니다.

반대로 값 **column**은 Items를 수직축으로 표시하므로 주 축은 수직이며 교차 축은 수평이 됩니다.

즉, 방향(수평, 수직)에 따라 주 축과 교차 축이 달라집니다.

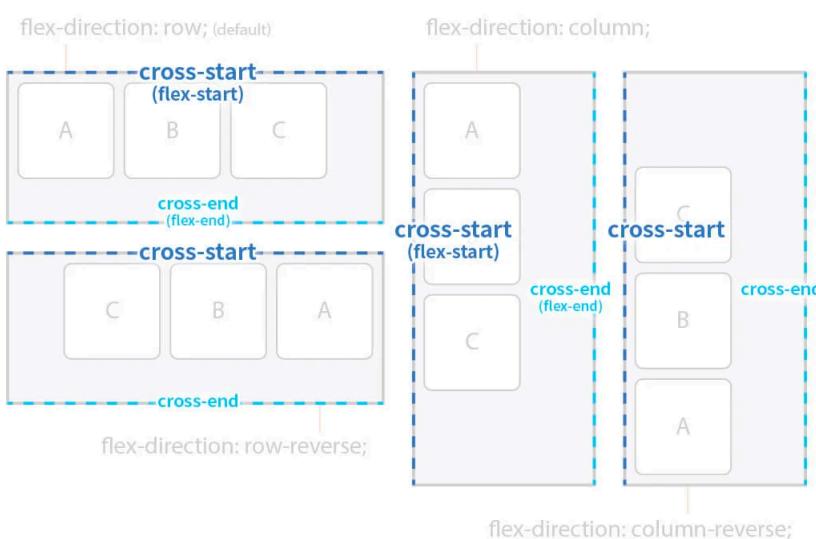
### 시작점(flex-start)과 끝점(flex-end)



시작점(flex-start)과 끝점(flex-end)이라는 개념도 있습니다.

이는 주 축이나 교차 축의 시작하는 지점과 끝나는 지점을 지칭합니다.

flex-start와 flex-end를 사용하는데 이는 방향에 맞는 그 시작점과 끝점을 의미합니다.



## Flex-wrap

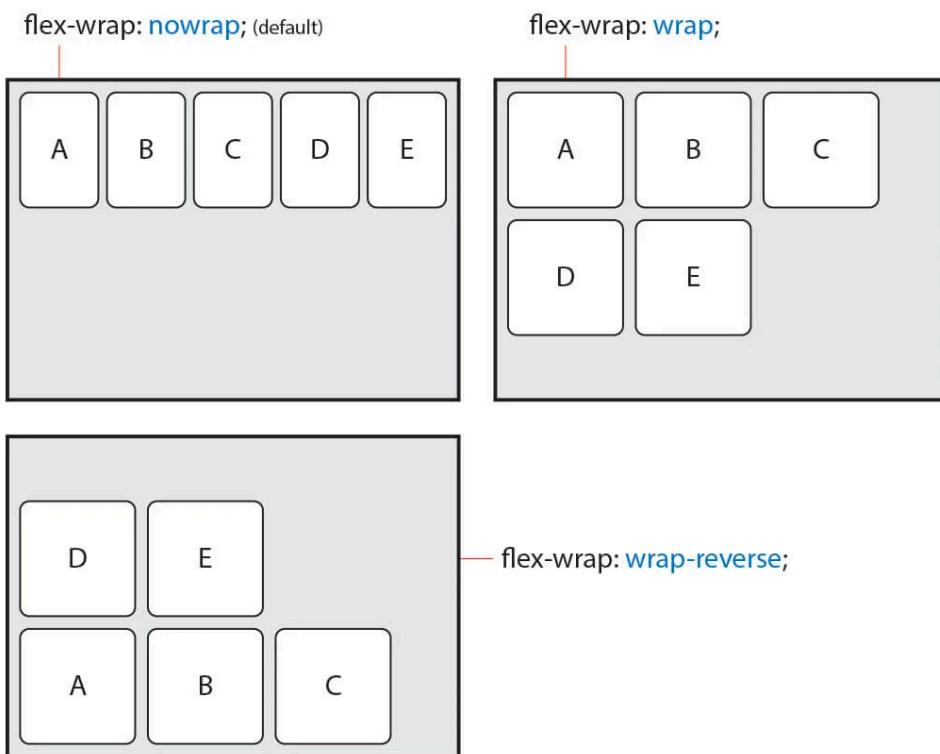
:Items의 여러 줄 묶음(줄 바꿈)을 설정합니다.

기본적으로 Items는 한 줄에서만 표시되고 줄 바꿈 되지 않습니다.

이는 지정된 크기(주 축에 따라 width나 height)를 무시하고 한 줄 안에서만 가변합니다.

Items를 줄 바꿈 하려면 값으로 wrap을 사용해야 합니다.

의미	설명	기본값
nowrap	모든 Items를 여러 줄로 묶지 않음(한 줄에 표시)	nowrap
wrap	Items를 여러 줄로 묶음	
wrap-reverse	wrap-reverse	



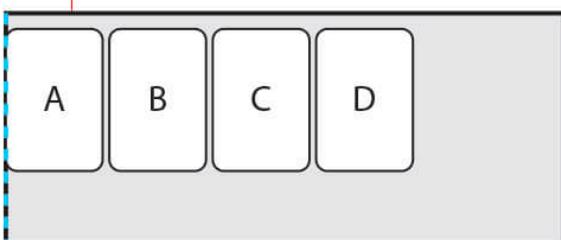
## justify-content

주 축(main-axis)의 정렬 방법을 설정합니다.

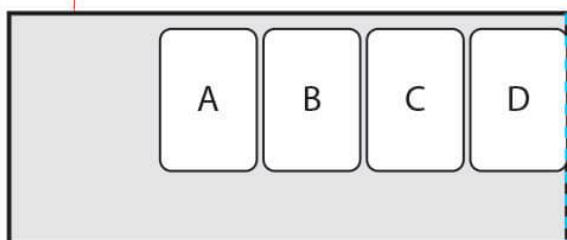
값	의미	기본값
flex-start	Items를 시작점(flex-start)으로 정렬	flex-start
flex-end	Items를 끝점(flex-end)으로 정렬	
center	Items를 가운데 정렬	

space-between	시작 Item은 시작점에, 마지막 Item은 끝점에 정렬되고 나머지 Items는 사이에 고르게 정렬됨	
space-around	Items를 균등한 여백을 포함하여 정렬	

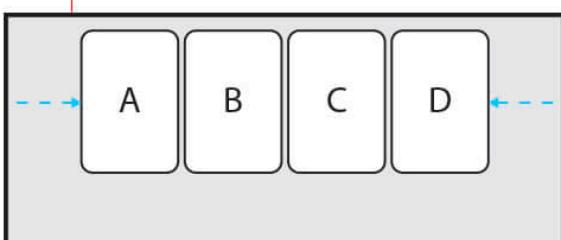
justify-content: **flex-start**; (default)



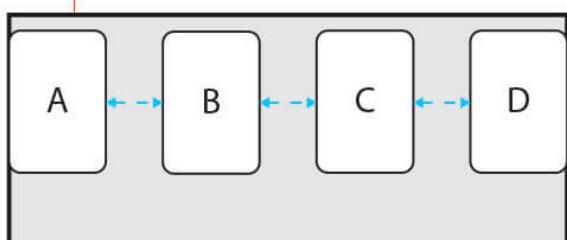
justify-content: **flex-end**;



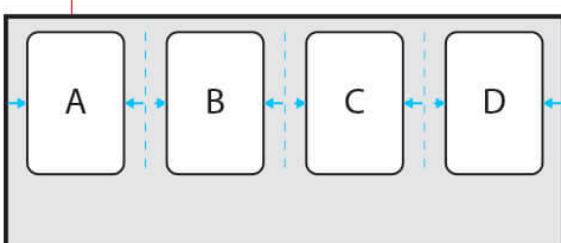
justify-content: **center**;



justify-content: **space-between**;



justify-content: **space-around**;



## align-content

교차 축(cross-axis)의 정렬 방법을 설정합니다.

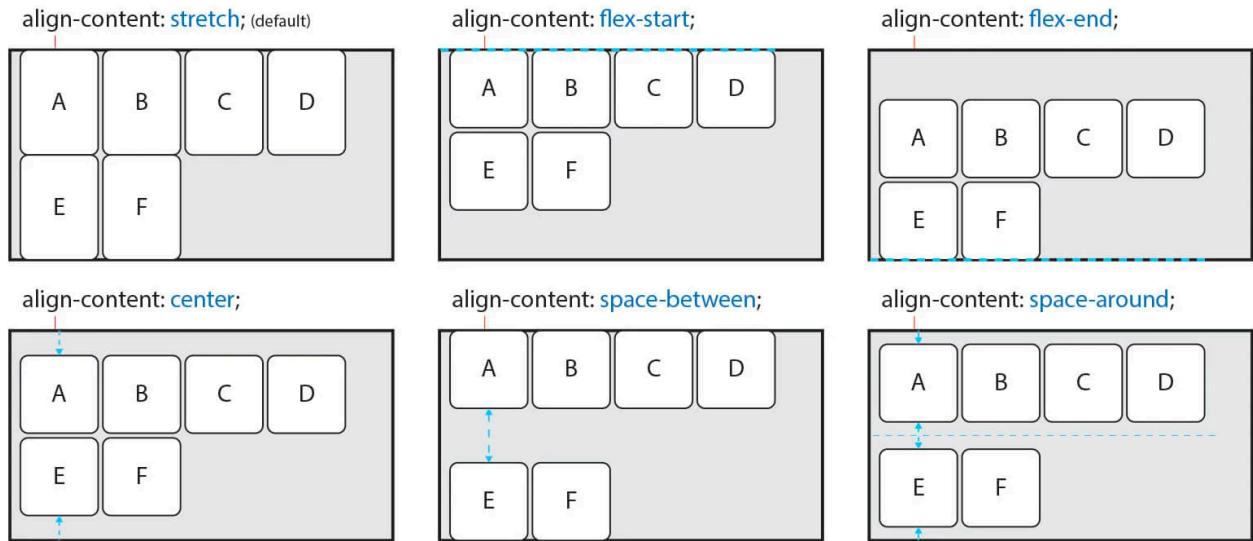
주의할 점은 flex-wrap 속성을 통해 Items가 여러 줄(2줄 이상)이고 여백이 있을 경우만 사용할 수 있습니다.

\*\*\*\*Items가 한 줄일 경우 align-items 속성을 사용하세요.

값	설명	기본값
stretch	Container의 교차 축을 채우기 위해 Items를 늘림	stretch
flex-start	Items를 시작점(flex-start)으로 정렬	
flex-end	Items를 끝점(flex-end)으로 정렬	
center	Items를 가운데 정렬	

space-between	시작 Item은 시작점에, 마지막 Item은 끝점에 정렬되고 나머지	
space-around	Items를 균등한 여백을 포함하여 정렬	

값 stretch는 교차 축에 해당하는 너비(속성 width 혹은 height)가 값이 auto(기본값)일 경우 교차 축을 채우기 위해 자동으로 늘어납니다.



## align-items

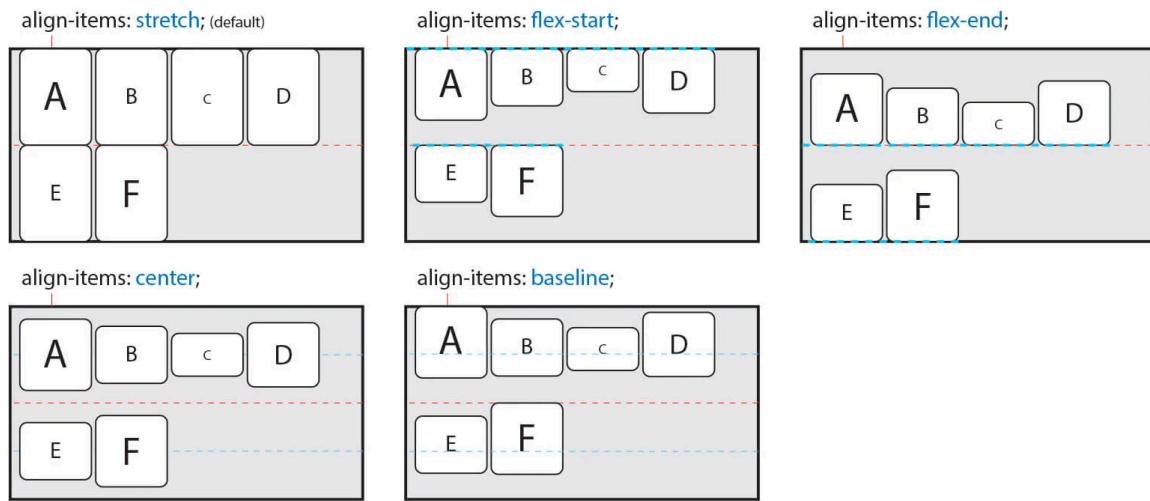
교차 축(cross-axis)에서 Items의 정렬 방법을 설정합니다.

Items가 한 줄일 경우 많이 사용합니다.

주의할 점은 Items가 flex-wrap을 통해 여러 줄(2줄 이상)일 경우에는 align-content 속성이 우선합니다.

따라서 align-items를 사용하려면 align-content 속성을 기본값(stretch)으로 설정해야 합니다.

값	의미	기본값
stretch	Container의 교차 축을 채우기 위해 Items를 늘림	stretch
flex-start	Items를 각 줄의 시작점(flex-start)으로 정렬	
flex-end	Items를 각 줄의 끝점(flex-end)으로 정렬	
center	Items를 가운데 정렬	
baseline	Items를 문자 기준선에 정렬	



## Flex Items

속성	의미
order	Flex Item의 순서를 설정
flex	flex-grow, flex-shrink, flex-basis의 단축 속성
flex-grow	Flex Item의 증가 너비 비율을 설정
flex-shrink	Flex Item의 감소 너비 비율을 설정
flex-basis	Flex Item의 (공간 배분 전) 기본 너비 설정
align-self	교차 축(cross-axis)에서 Item의 정렬 방법을 설정

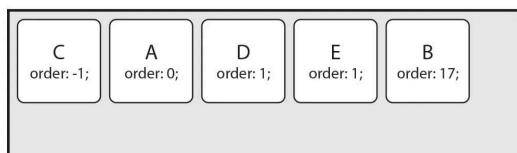
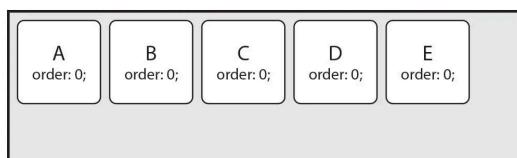
### order

Item의 순서를 설정합니다.

Item에 숫자를 지정하고 숫자가 클수록 순서가 밀립니다.

음수가 허용됩니다.(HTML 구조와 상관없이 순서를 변경할 수 있기 때문에 유용합니다.)

값	의미	기본값
숫자	Item의 순서를 설정	0



## flex-grow

Item의 증가 너비 비율을 설정합니다.

숫자가 크면 더 많은 너비를 가집니다.

Item이 가변 너비가 아니거나, 값이 0일 경우 효과가 없습니다.

의미	기본값	
숫자	Item의 증가 너비 비율을 설정	0

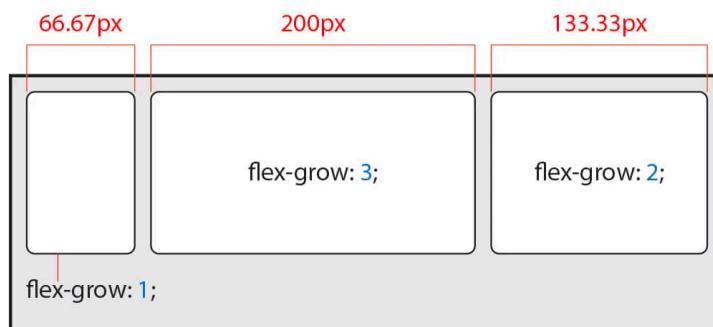
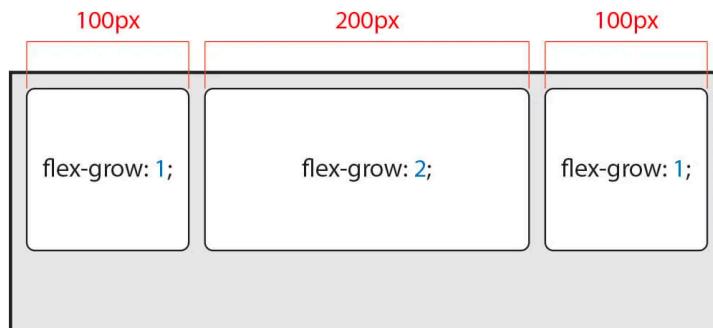
모든 Items의 총 증가 너비(flex-grow)에서 각 Item의 증가 너비의 비율 만큼 너비를 가질 수 있습니다.

예를 들어 Item이 3개이고 증가 너비가 각각 1, 2, 1이라면,

첫 번째 Item은 총 너비의 25%(1/4)을,

두 번째 Item은 총 너비의 50%(2/4)를,

세 번째 Item은 총 너비의 25%(1/4)을 가지게 됩니다.



## flex-shrink

Item이 감소하는 너비의 비율을 설정합니다.

숫자가 크면 더 많은 너비가 감소합니다.

Item이 가변 너비가 아니거나, 값이 0일 경우 효과가 없습니다.

값	의미	기본값
숫자	Item의 감소 너비 비율을 설정	1

감소 너비(flex-shrink)는 요소의 너비에 영향을 받기 때문에 계산하기 까다롭습니다.

영향을 받는다는 요소의 너비는 width, height, flex-basis 등으로 너비가 지정된 경우를 의미합니다.

Container의 너비가 줄어 Items의 너비에 영향을 미칠 경우, 영향을 미치기 시작한 지점부터 줄어든 거리 만큼 감소 너비 비율에 맞게 Item의 너비가 줄어듭니다.

## flex-basis

Item의 (공간 배분 전) 기본 너비를 설정합니다.

값이 auto일 경우 width, height 등의 속성으로 Item의 너비를 설정할 수 있습니다.

하지만 단위 값이 주어질 경우 설정할 수 없습니다.

값	의미	기본값
auto	가변 Item과 같은 너비	auto
단위 px, em, cm 등 단위로 지정		

## flex

Item의 너비(증가, 감소, 기본)를 설정하는 단축 속성입니다.

속성	설명	기본값
flex-grow	Item의 증가 너비 비율을 설정	0
flex-shrink	Item의 감소 너비 비율을 설정	1
flex-basis	Item의 (공간 배분 전) 기본 너비 설정	aut

flex: 증가너비 감소너비 기본너비;

```
.item {  
  flex: 1 1 20px; /* 증가너비 감소너비 기본너비 */  
  flex: 1 1; /* 증가너비 감소너비 */  
  flex: 1 20px; /* 증가너비 기본너비 (단위를 사용하면 flex-basis가 적용됩니다) */  
}
```

## align-self

교차 축(cross-axis)에서 개별 Item의 정렬 방법을 설정합니다.

align-items는 Container 내 모든 Items의 정렬 방법을 설정합니다.

필요에 의해 일부 Item만 정렬 방법을 변경하려고 할 경우 align-self를 사용할 수 있습니다.

이 속성은 align-items 속성보다 우선합니다.

값	의미	기본값
auto	Container의 align-items 속성을 상속받음	auto
stretch	Container의 교차 축을 채우기 위해 Item을 늘림	
flex-start	Item을 각 줄의 시작점(flex-start)으로 정렬	
flex-end	Item을 각 줄의 끝점(flex-end)으로 정렬	
center	Item을 가운데 정렬	
baseline	Item을 문자 기준선에 정렬	

