

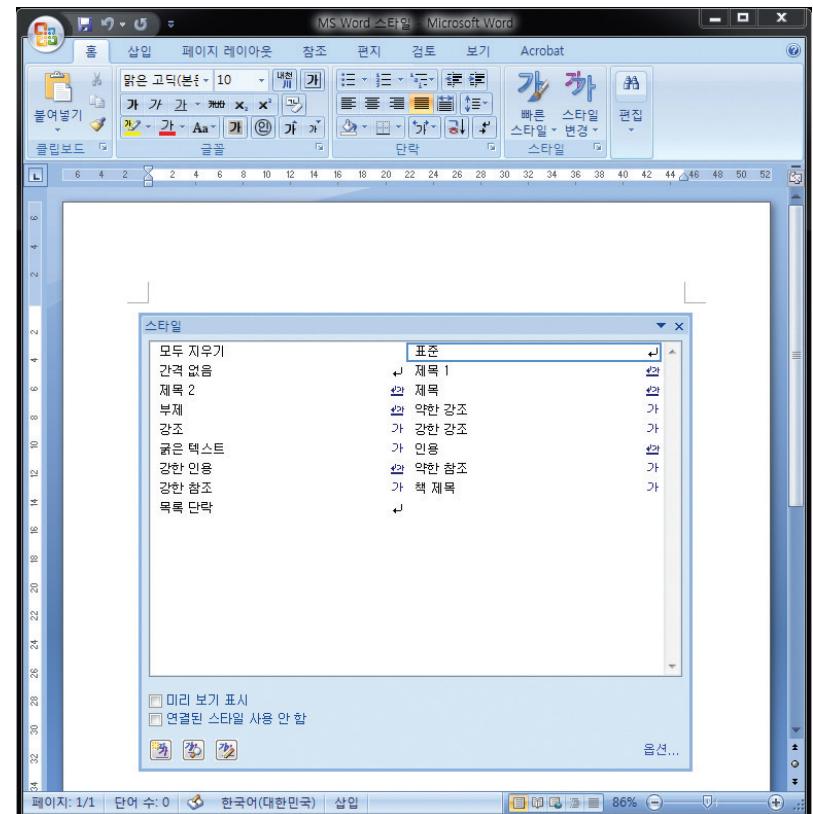
# CSS3 Design

김데레사

# **CSS에 대하여**

# CSS란?

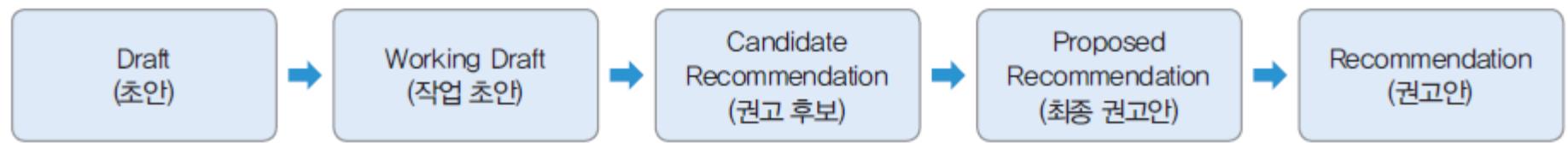
- CSS 또는 캐스케이딩 스타일 시트 (Cascading Style Sheet)는 마크업 언어가 실제로 표시되는 방법을 기술하는 언어로, HTML과 XHTML에 주로 쓰이며, XML에서도 사용할 수 있음.
  - CSS는 W3C의 표준이며, 레이아웃과 스타일을 정의할 때의 자유도가 높은 언어임.
  - CSS는 웹 사이트에서 사용되는 스타일을 지정하기 위한 언어로, 스타일은 웹의 독창적인 개념이 아니라 웹 이전부터 워드프로세스 등에 사용했던 개념임.



# CSS의 과거와 현재

- 1996년 W3C의 주도하에 첫 번째 CSS 버전인 CSS Level 1이 발표.
- 1998년에 CSS Level 2가 등장하면서 대부분의 웹 브라우저들이 CSS Level 2를 지원하기 시작
- 그 후 CSS Level 2의 버그를 수정한 CSS Level 2.1이 2006년에 발표되면서 현재까지 표준으로 사용되고 있음.
- CSS Level 3는 CSS Level 2.1과 달리 모든 명세가 포함된 버전이 아닌 모듈 단위로 개발되고 있으며, 표준화가 모듈 단위로 진행되고 있음. 이 중 몇몇 모듈은 현재 Recomendation(권고안) 단계에 있으며, Working Draft(초안) 단계에 머물러 있는 모듈도 있음.

# 웹표준 개발 단계



# CSS3 모듈

- Selectors Level 3 : CSS 속성을 적용하기 위해 HTML 요소를 선택하는 방법에 대한 규정으로, CSS3에서는 이전 버전에서 제공하던 선택자 이외에 HTML의 정보 구조 관계를 탐색하여 요소를 선택하는 Structural Pseudo Selector가 다수 추가 되었음.
- CSS Color Module Level 3 : CSS에서 사용할 수 있는 색상에 관한 규정으로, CSS3에서는 색상의 투명도를 지정할 수 있는 RGBA 형식을 사용할 수 있게 되었음.
- CSS Namespaces Module : CSS가 XML 형식에서 사용될 경우의 "네임 스페이스"를 지정하기 위한 규정.
- CSS Backgrounds and Borders Module Level 3 : CSS를 이용한 웹 디자인에서 사용 빈도가 높은 배경과 테두리에 대한 규정으로, 많은 기능이 추가되었음.

# CSS3 모듈

- CSS Multi–Column Layout Module : 인쇄물의 다단 구성과 같은 형태의 레이아웃을 구현하기 위한 규정.
- Media Queries : Media Queries는 CSS의 media 속성의 확장 기능. 기존의 media 속성에서는 웹을 표시하는 매체의 형식만을 구분하였지만, CSS3 추가된 기능으로 인해 매체의 특성에 따라서도 CSS 속성을 구분하여 적용할 수 있게 되었음.
- CSS Mobile Profile 2.0 : 휴대 전화에 적용하기 위한 CSS의 하위 집합으로, CSS의 전체 스펙을 축소한 형태.
- CSS Marquee Module Level 3 : 움직이는 텍스트와 같이 사용자의 인터렉션에 독립적으로 흐르는 효과에 대한 규정.

# CSS3 모듈

- CSS3 Paged Media : 인쇄된 웹과 같이 페이지로 구분되는 미디어를 위한 규정.
- CSS Print Profile : 성능이 낮은 저가형 프린터에 적용하기 위한 CSS의 하위 집합으로, CSS의 전체 스펙을 축소한 형태.
- CSS Values and Units : CSS에서 사용되는 값과 단위에 대한 규정
- CSS Cascading and inheritance : CSS의 주요한 특징인 단계적 상속에 관한 규정.
- CSS Text Level 3 : 문자열에 관한 규정.
- CSS Writing Modes Module Level 3 : 문자에 따른 글쓰기 방향에 관한 규정.

# CSS3 모듈

- CSS3 Line Grid : 영문자의 베이스라인과 같은 문자열의 가이드 라인에 대한 규정.
- CSS3 Ruby Module : 한자나 일본어 등에서 독음 등을 표시하기 위한 작은 글자에 대한 규정.
- CSS Generated Content for Paged Media Module : 출력물과 같은 페이지로 나누어지는 매체에서 쪽 번호나 장 번호를 생성하기 위한 규정.
- CSS Fonts Module Level 3 : 글자체에 관한 규정. CSS3에서는 사용자 측에 미리 설치되어 있는 폰트뿐만 아니라 서버 측의 폰트도 지정할 수 있음.
- CSS basic box model : CSS가 HTML 요소를 표시하는 기본 규칙인 Box Model에 관한 규정.

# CSS3 모듈

- CSS Template Layout Module : CSS 속성으로 레이아웃 템플릿을 작성하기 위한 규정.
- CSS Speech Module : 음성 합성 장치를 위한 볼륨, 음성, 속도, 피치 등을 제어할 수 있는 속성에 대한 규정.
- CSS3 Basic User Interface Module : <form>의 서식 요소와 같이 기본적으로 제공되는 사용자 환경에 대한 설정 방법의 규정.
- CSS Scoping : CSS 속성의 적용 범위에 대한 규정.
- CSS Grid Positioning Module Level 3 : Grid Layout에서의 요소 배치에 관한 사항에 대한 규정.

# CSS3 모듈

- CSS Grid Template Layout : 그리드 기반의 레이아웃을 구현하기 위한 속성에 대한 규정. 기존 CSS에서 규정하던 position 속성 등과는 별도의 것임.
- Flexible Box Layout Module : CSS3에 등장한 새로운 방식의 레이아웃을 구성하는 방법에 대한 규정으로, 박스의 배치 순서나 방향을 지정하는 방법에 대한 규정.
- CSS Image Values and Replaced Content Module Level 3 : 이미지나 대치되는 요소를 위한 속성들에 대한 규정. 이 속성들은 CSS에서 사용되는 이미지 또는 대치되는 요소 자체에 대한 제어를 목적으로 함.

# CSS3 모듈

- CSS 2D Transforms Module Level 3 : 2차원 형태 변환을 위한 규정. 이 속성은 CSS 요소 박스의 확대, 축소, 회전 등의 변환을 적용.
- CSS 3D Transforms Module Level 3 : 3차원 형태 변환을 위한 규정.
- CSS Transitions Module Level 3 : CSS 속성 효과의 전환을 일정 시간 동안 부드럽게 진행하기 위한 규정. Transition이 수행되기 위해서는 요소에 지정된 CSS의 속성 값이 변경되어야 하므로 특정한 트리거가 필요.
- CSS Animations Module Level 3 : CSS로 애니메이션을 구현하기 위한 속성에 관한 규정. Transition도 애니메이션의 형태로 구현되지만, 이 속성은 애니메이션을 수행하기 위한 트리거가 특별히 필요 없음.

# CSS3 모듈

- CSS Style Attributes : HTML 등의 마크업 언어에 inline으로 CSS를 적용하기 위한 규정.
- CSSOM View Module : 스크롤된 거리나 화면 폭과 같은 화면이 보여지고 있는 형태에 대한 정보에 기계적으로 접근할 수 있는 CSSOM의 부분적인 규정.
- CSS Extended Box Model : CSS의 기본 Box Model을 추가적으로 제어하기 위한 규정.
- CSSOM : CSS의 Object Model에 대한 규정으로, 스크립트나 API를 위한 정보를 제공.
- CSS3 Syntax : CSS를 작성하는 문법에 대한 규정.

# CSS3 모듈

- CSS Lists and Counters Module Level 3 : 리스트 요소를 위한 속성과 번호 지정을 위한 규정.
- CSS Tables : 테이블 요소를 위한 속성에 관련된 규정.
- The CSS ‘Reader’ Media Type : 화면 판독기와 같은 리더 기기를 위한 규정.
- CSS Positioning : 요소의 위치를 지정하는 속성에 관련된 규정.
- CSS3 Generated and Replaced Content Module : CSS로 내용 요소 또는 대치되는 요소를 생성하는 것에 대한 규정.
- CSS3 Line Layout : Inline Box에 대한 규정.

# CSS3 모듈

- CSS3 Hyperlink Presentation Module : 하이퍼링크가 표시되는 형태에 대한 내용을 다룸.
- CSS Math : 수식을 작성하는 MathML의 스타일을 위한 규정.
- CSS Aural Style Sheets : 음성 장치를 위한 규정.
- CSS TV Profile 1.0 : TV에 적용하기 위한 CSS의 하위 집합으로, CSS의 전체 스펙을 축소한 형태.
- Behavioral Extensions to CSS : CSS를 확장하기 위한 동적 제어에 대한 규정.
- Cascading Style Sheets (CSS) : CSS에 대한 전반적 설명과 CSS3의 각 모듈에 대한 간략한 소개를 담은 문서.

# 웹 브라우저별 CSS3 지원 Test

<http://css3test.com>

The screenshot shows a browser window displaying 'The CSS3 Test' website at [css3test.com](http://css3test.com). The main content area features a large 'Your browser scores' heading followed by a massive '60%' in black. Below this, a note states 'Determined by passing 589 tests out of 1040 total for 243 features'. To the right, a 'Caution' box explains that the test checks which CSS3 features the browser recognizes, not whether they are implemented correctly. It also notes a 'Time taken: 51ms'. A sidebar on the left lists 'THE CSS3 TEST' and a pink circular icon. The bottom section, titled 'Backgrounds and Borders', shows a 100% score with 'TR' and 'DEV' status indicators. A green 'Properties' list includes 'background-repeat', 'background-attachment', 'background-position', and 'background-clip', all marked ':D'. On the right, a 'Specs tested' sidebar lists 'Backgrounds and Borders', 'Image Values and Replaced Content', 'Selectors', and 'Media Queries', each with a status indicator like ':D' or ':o'.

The CSS3 TEST

Your browser scores

# 60%

Determined by passing 589 tests out of 1040 total for 243 features

Backgrounds and Borders TR DEV 100%

Properties

- ▶ background-repeat :D
- ▶ background-attachment :D
- ▶ background-position :D
- ▶ background-clip :D

Caution: This test checks which CSS3 features the browser recognizes, *not* whether they are implemented correctly.

Time taken: 51ms

Campaign Monitor  
Send something beautiful

via Ad Packs

Specs tested:

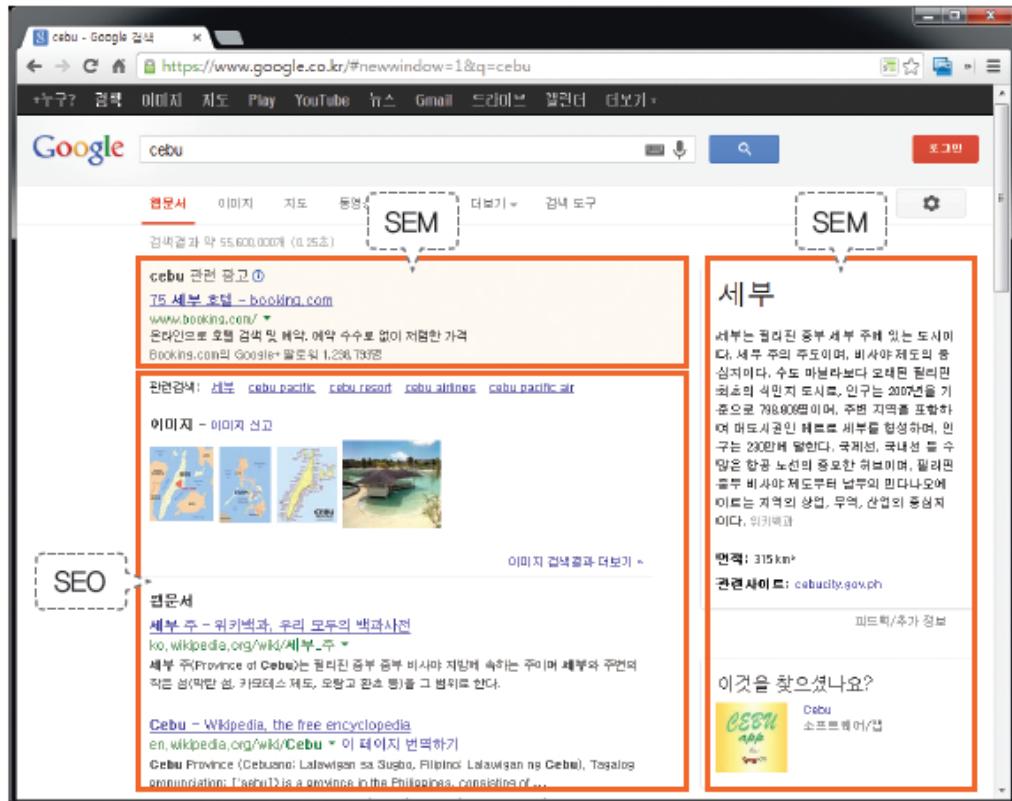
- Backgrounds and Borders :D
- Image Values and Replaced Content :o
- Selectors :D
- Media Queries :D

# CSS 사용의 의의

- 스타일 사용의 중요한 의의  
는 문서의 구조와 표현을  
분리할 수 있다는 점을 들  
수 있으며 이는 구조와 표  
현을 분리함으로써 문서 구  
조의 수정 없이 스타일의  
변경만으로 다양한 표현을  
할 수 있다는 것을 의미함.



# CSS 사용의 의의



- 웹 문서에 있어 구조와 표현의 분리는 워드프로세서 문서보다 중요할 수 있음. 그 이유 중 하나는 웹 문서가 기계적으로 해석될 수 있는 가능성이 높아지기 때문임. 기계적으로 해석되는 것은 CSS가 아니라 HTML이지만, 구조와 무관한 표현적 요소를 CSS로 분리함으로써 HTML은 간결해지고 더욱 구조화될 수 있기 때문임.

# **CSS 기초**

# CSS 기본문법

- 하나의 규칙은 크게 선택자(selector)와 선언부(declaration block)로 이루어지며 선언부는 다시 속성(property)과 속성 값(value)으로 구성되어 있음. 이때 선언부는 세미콜론(;)으로 속성과 속성 값을 구분하여 여러 개의 선언(declaration)을 지정할 수 있음.

세미콜론으로 구분하여 여러 개의 선언을 지정할 수 있음.

```
selector { property : value ; property : value }
```

속성

속성 값

속성

속성 값

선택자(selector)

선언부(declaration block)

# CSS3 웹 브라우저별 접두사(vendor prefix)

- CSS3는 표준안이 아직 확정되지 않은 상태이기 때문에 최신 웹 브라우저들은 CSS3 속성을 실험적으로 제공하고 있음. 이를 위해 속성이나 속성 값 앞에 웹 브라우저별로 접두사(vendor prefix)를 제공하고 있으며 이 접두사의 경우 웹 브라우저별로 다르기 때문에 하나의 속성을 선언하기 위해서는 여러 번의 동일한 선언을 지정해야 함.

웹 브라우저	접두사(prefix)
파이어폭스(Firefox)	- moz -
크롬, 사파리(Chrome, Safari)	- webkit -
오페라(Opera)	- o -
인터넷 익스플로러(Internet Explorer)	- ms -

# 주석 및 단위

/\* CSS에서 사용할 수 있는 주석입니다. \*/

- 문자열 타입 : "inherit"와 같이 CSS에서 미리 정의된 키워드나 제작자가 정의한 저작자 키워드 등 텍스트로 입력하는 속성 값이 이에 속함.
- 숫자 타입 : 개수 비율들을 나타낼 수 있는 정수, 실수 값, 퍼센트 값이 이에 속함.
- 길이 단위 타입 : 길이 단위 타입은 상대 길이 단위 값과 절대 길이 단위 값이 있으며 상대 길이 단위 값은 특정 길이에 비율로 정해지는 값임.

# 상대 단위와 절대 단위

## 상대 단위

단위	의미
em	폰트의 기본 크기 값에 비례한 단위
ex	폰트의 기본 X 높이에 비례한 단위
ch	"0"의 기본 폭에 비례한 단위
rem	최상위 요소의 폰트 크기에 비례한 단위
vw	뷰포트 너비에 비례한 단위
vh	뷰포트 높이에 비례한 단위
vmin	뷰포트의 최소 너비, 높이에 비례한 단위

## 절대 단위

단위	의미
cm	센티미터 단위
mm	밀리미터 단위
in	인치(inch) 단위로, 2.54cm과 같음.
px	픽셀(pixels) 단위로, 1/96inch와 같음.
pt	포인트(point) 단위로, 1/72inch와 같음.
pc	피카(piccas) 단위로, 12pt와 같음.

# 기타 단위 타입

- 이에 속한 단위들로는 각도를 지정하는 "deg", "grad", "rad", "turn", 시간을 나타내는 "s", "ms", 빈도를 지정하는 "Hz", "kHz" 그리고 해상도를 지정하는 "dpi", "dpcm", "dppx"를 들 수 있음. 이 단위들은 기존 CSS2에서 음성 브라우저 등의 특수 경우를 상정하여 정의되었던 것이지만, CSS3에서 는 표현 속성이 다양해짐에 따라 각도나 시간은 일반적으로 사용되는 값의 형식이 될 것임.

# 색상

- CSS3에서의 색상 단위는 좀 더 확대되었음. 특히, 투명 값(alpha)이나 색상, 채도, 명도 등을 지정 할 수 있는 HSL 방식 그리고 currentcolor 키워드 등이 추가되었음.  
(색상 이름, transparent, 16진수 RGB 색상 값, 256단계 RGB 색상 값, 백분율(%) RGB 색상 값, RGBA 색상 값, HSL 색상 값, HSLA 색상 값, currentcolor)

# RGBA 형식 및 HSLA 형식의 사용 예

## ※ RGBA 형식

```
color : rgba( 255 , 127 , 45 , 0.5 )
```

Red      Green      Blue      투명도  
↓      ↓      ↓      ↓

## ※ HSLA 형식

```
color : hsla( 0 , 0% , 100% , 0.5 )
```

색상      채도      명도      투명도  
↓      ↓      ↓      ↓

# CSS 적용하기 (External)

- CSS 파일을 외부에 생성하여 HTML 문서에 연결하는 방식으로 <link> 요소를 사용하는 방법과 @import 명령을 사용하는 두 가지 방식이 있음.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>External Style Sheet</title>
    <link rel="stylesheet" type="text/css" href="css/style.css">
    <style>
      @import url(style.css) ;
    </style>
  </head>
  <body>
    <h1>External 방식</h1>
    <p>&lt;link&gt; 요소를 사용하여 외부에 생성한 CSS 파일을 연결합니다.</p>
  </body>
</html>
```

# CSS 적용하기 (Embedded)

- HTML 파일 내에 CSS 코드를 직접 포함하여 스타일이 적용되도록 하는 방법으로, CSS 코드는 <style> 요소내에 선언합니다.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Embedded Style Sheet</title>
    <style>
      h1{
        color:#f00;
      }
      p{
        color:#00f;
      }
    </style>
  </head>
  <body>
    <h1>Embedded 방식</h1>
    <p>&lt;style&gt; 요소 내에 직접 CSS 코드를 포함합니다.</p>
  </body>
</html>
```

# CSS 적용하기 (Inline)

- 특정 HTML 요소에 style 속성을 사용하여 CSS 코드를 선언하는 방법임. 그러나 이 방법은 구조와 표현의 분리라는 관점에서 바람직하지 않은 면을 가지고 있음. 특히 Inline 방식의 스타일은 선택자의 우선순위가 가장 높기 때문에 스타일의 재정의가 어렵거나 불가능한 경우가 발생할 수 있으므로 사용에 주의가 필요함.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Inline Style Sheet</title>
  </head>
  <body>
    <h1 style="color:#f00">Inline 방식</h1>
    <p style="color:#00f">&lt;style&gt; 요소 내에 직접 CSS 코드를 포함합니다.</p>
  </body>
</html>
```



# **Section 3.**

# **CSS 선택자**

# 전체 선택자(Universal Selector)

- 전체 선택자(Universal Selector)는 모든 요소를 선택하는 방법으로, "\*"를 선택자로 선언.

선택자(Selector)	의미
*	모든 요소를 선택함

사용 예

```
* {  
    margin : 0 ;  
    padding : 0 ;  
}
```

# 요소 선택자(Type Selector)

- 요소 선택자(Type Selector)는 HTML 요소를 선택하는 방법으로, "h1", "p", "div" 등의 요소를 선택자로 지정할 수 있음.

선택자(Selector)	의미
E	"E" 요소를 선택함

사용 예    `div {  
    color : rgba( 255 , 0 , 0 , 0.5 ) ;  
}`

# 클래스 선택자(Class Selector)

- 클래스 선택자(Class Selector)는 HTML 요소의 class 속성 값을 참조하여 선택하는 방법. 이때 class 속성 값은 하나의 HTML 요소에 여러 개를 지정할 수 있기 때문에 다중 class를 선택자로 지정할 수도 있음.

선택자(Selector)	의미
E.warning	Class 속성 값이 "warning"인 "E" 요소를 선택함
E.warning.notice	Class 속성 값을 "warning"과 "notice"를 모두 가진 "E" 요소를 선택함

사용 예

```
.note {  
    font-size : 1.2em ;  
}
```

# 아이디 선택자(ID Selector)

- 아이디 선택자(ID Selector)는 HTML 요소의 id 속성 값을 참조하여 선택하는 방법. 이때 id 속성 값은 하나의 HTML 문서에 한 번만 사용할 수 있기 때문에 아이디 선택자를 사용하면 유일한 요소를 선택할 수 있음.

선택자(Selector)	의미
E#useful	id 속성 값이 "useful"인 "E" 요소를 선택함

사용 예    `#main {  
    font-weight : bold ;  
}`

# 속성 선택자(Attribute Selector)

- 속성 선택자(Attribute Selector)는 HTML 요소의 속성을 참조하여 선택하는 방법을 의미하며, 이때 속성의 지정 여부나 속성 값의 일치 여부로 선택할 수 있음.

선택자(Selector)	의미
E[attr]	"attr" 속성을 가진 "E" 요소를 선택함
E[attr="val"]	"attr" 속성 값이 "val"인 "E" 요소를 선택함
E[attr~="val"]	"attr" 속성 값 중에 공란으로 분리된 "val" 단어가 있는 "E" 요소를 선택함
E[attr^="val"]	"attr" 속성 값이 "val"로 시작되는 "E" 요소를 선택함
E[attr\$="val"]	"attr" 속성 값이 "val"로 끝나는 "E" 요소를 선택함
E[attr*="val"]	"attr" 속성 값이 "val"을 포함하는 "E" 요소를 선택함
E[attr = "en"]	"attr" 속성 값이 "en"이거나 "en-"로 시작하는 "E" 요소를 선택함

```
사용 예  a[title] {  
            text-decoration : underline ;  
        }
```

# 가상 클래스 선택자(Pseudo-classes Selector)

- 가상 클래스 선택자(Pseudo-classes Selector)는 요소의 상태나 상황에 따라 선택하는 방법으로, 링크의 경우 방문하기 전, 방문한 후, 링크 위에 마우스를 올려놓거나 포커스 시 등의 상황을 선택하여 스타일을 지정할 수 있음. 또한 언어에 따른 구분이나 마크업 구조에 따라 특정 요소를 선택할 수도 있음.
- 동적 가상 클래스(Dynamic pseudo-classes) – :link, :visited, :active, :hover, :focus
- 타깃 가상 클래스(Target pseudo-classes) – :target
- 언어 가상 클래스(Language pseudo-classes) – :lang
- UI 요소 상태 가상 클래스(UI element states pseudo-classes) – :enabled, :disabled, :checked
- 구조 가상 클래스(Structural pseudo-classes) – :root, :nth-child( ), :nth-last-child( ), :nth-of-type( ), :nth-last-of-type( ), :first-child, :last-child, :first-of-type( ), :last-of-type( ), :only-child, :only-of-type( ), :empty
- 부정 가상 클래스(Negation pseudo-classes) – :not( )

# 가상 클래스 선택자(Pseudo-classes Selector)

선택자(Selector)	의미
E:link	아직 방문하지 않는 하이퍼링크를 가진 "E" 요소를 선택함
E:visited	이미 방문한 하이퍼링크를 가진 "E" 요소를 선택함
E:active	현재 사용자 액션을 받고 있는 "E" 요소를 선택함
E:hover	마우스 포인터가 올라간 "E" 요소를 선택함
E:focus	키보드의 포커스를 받은 "E" 요소를 선택함
E:target	"E" 요소가 하이퍼링크 타깃이 되는 경우에 선택함
E:lang(fr)	Lang 속성이 "fr(프랑스어를 나타내는 속성 값)"인 "E" 요소를 선택함
E:enabled	사용 가능 상태의 "E" 요소를 선택함
E:disabled	사용 불가 상태의 "E" 요소를 선택함
E:checked	체크된 "E" 요소를 선택함
E:root	문서 최상위의 요소를 선택함
E:nth-child(n)	상위 요소의 n 번째 자식 요소가 "E"이면 선택함

# 가상 클래스 선택자(Pseudo-classes Selector)

E:nth-last-child(n)	상위 요소의 역순으로 n 번째 자식 요소가 "E"이면 선택함
E:nth-of-type(n)	동일한 "E" 타입의 형제 요소중 n 번째 "E" 요소를 선택함
E:nth-last-of-type(n)	동일한 "E" 타입의 형제 요소 중 역순으로 n 번째 "E" 요소를 선택함
E:first-child	첫 번째 자식 요소의 타입이 "E"이면 선택함
E:last-child	마지막 자식 요소가 타입이 "E"이면 선택함
E:first-of-type()	상위 요소에 대하여 첫 번째 자식 요소의 타입이 "E"이면 선택함
E:last-of-type()	상위 요소에 대하여 마지막 자식 요소의 타입이 "E"이면 선택함
E:only-child	상위 요소에 대하여 유일한 자식 요소가 타입이 "E"이면 선택함
E:only-of-type	상위 요소에 대하여 자식 요소중 다른 "E" 요소가 없이 유일한 "E"일 경우 선택함
E:empty	텍스트 노드를 포함하여 아무런 자식 요소를 갖고 있지 않는 "E" 요소를 선택함
E:not(S)	"S"로 지정된 선택자와 일치하지 않는 "E" 요소를 선택함

# 가상 클래스 선택자(Pseudo-classes Selector)

사용 예

```
a:link {  
    color : rgba( 255 , 100 , 100 , 0.8 ) ;  
}
```

# 가상 요소 선택자(Pseudo-element Selector)

- 가상 요소 선택자(Pseudo-element Selector)는 요소의 첫글자나 첫줄 또는 요소 앞이나 뒤 등 가상의 영역을 선택하고자 할 때 사용.

선택자(Selector)	의미
E::first-line	'E' 요소의 문자열 중 첫 번째 줄을 선택함
E::first-letter	'E' 요소의 문자열 중 첫 번째 문자를 선택함
E::before	'E' 콘텐츠 앞으로 생성된 가상 요소를 선택함
E::after	'E' 콘텐츠 뒤로 생성된 가상 요소를 선택함
E::selection	사용자가 선택한 'E' 요소의 범위를 선택함

사용 예    p::first-letter {  
              color : #ff0000 ;  
              font-size : 5em ;  
}

# 하위 선택자(Descendant Combinator)

- 하위 선택자(Descendant Combinator) 방식은 선택자와 선택자를 공란으로 선언하며, 선행 선택자의 하위 요소 중 후행 선택자에 해당하는 요소를 선택하는 방법.

선택자(Selector)	의미
[선택자] [선택자]	후행 선택자가 반드시 선행 선택자가 안에 포함되어 있을 경우 선택함

사용 예    `#main div {  
    border : 3px solid currentcolor ;  
}`

# 자식 선택자(Child Combinator)

- 자식 선택자(Child Combinators) 방식은 선행 선택자인 부모 요소 하위에 포함된 후행 선택자인 자식 요소를 선택하는 방법. 이때 부모 선택자와 자식 선택자는 ">"로 구분하여 선언.

선택자(Selector)	의미
[선택자] > [선택자]	후행 선택자가 부모 요소로 선행 선택자를 가지고 있는 경우 선택함

사용 예    `#main > div {  
    border : 3px solid red ;  
}`

# 형제 선택자(Sibling Combinators)

- 형제 선택자(Sibling Combinators)는 기본 형제 선택자(General Sibling Combinators)와 인접 형제 선택자(Adjacent Sibling Combinators)로 구분할 수 있음. 이때 기본 형제 선택자는 선행 선택자와 후행 선택자를 "+"로 구분하여 선언하고, 인접 형제 선택자는 "~"로 구분하여 선언.

선택자(Selector)	의미
[선택자] + [선택자]	선행 선택자 뒤에 후행 선택자를 선택함
[선택자] ~ [선택자]	선행 선택자 뒤에 인접하여 등장하는 모든 후행 선택자를 선택함

사용 예

```
h1 + p {  
    color : red ;  
}  
.note ~ p {  
    color : blue ;  
}
```

# 선택자 그룹화

- 앞에서 살펴본 모든 선택자는 콤마(,)를 사용하여 그룹으로 한 번에 선언할 수 있음. 선택자를 그룹으로 선언할 경우, 선언된 모든 선택자에는 동일한 선언이 적용됨.

선택자(Selector)	의미
[선택자] , [선택자]	콤마(,)로 구분된 모든 선택자에 동일한 선언이 적용됨

사용 예    `h1 , h1 + p , .note, #main, p[title] {  
    font-size : 1.2em ;  
}`

# 선택자 우선순위

- CSS에서는 하나의 요소에 여러 스타일이 중복 지정될 경우, 선택자의 우선순위에 따라 구체성(specificity)이 높은 선택자의 스타일이 적용됨.

선택자	A	B	C	구체성	우선순위
li	0	0	1	1	5
ul li	0	0	2	2	4
div.note	0	1	2	12	3
#list li	1	0	1	101	2
ul#list li	1	0	2	102	1



# **Section 4.**

## **CSS 속성**

# font-family

- 텍스트의 폰트를 지정

속성 값	<family-name>   <generic-family>
기본 값	사용자 환경에 따름
상속 여부	상속됨
지원 웹 브라우저	    

# font-weight

- 폰트가 표시되는 굵기를 지정

속성 값	normal   bold   bolder   lighter   100   200   300   400   500   600   700   800   900
기본 값	normal
상속 여부	상속됨
지원 웹 브라우저	    

# font-stretch

- 폰트의 장평을 지정

속성 값	normal   ultra-condensed   extra-condensed   condensed   semi-condensed   semi-expanded   expanded   extra-expanded   ultra-expanded
기본 값	normal
상속 여부	상속됨
지원 웹 브라우저	 



# font-style

- 폰트의 표시 형태를 지정

속성 값	normal   italic   oblique
기본 값	normal
상속 여부	상속됨
지원 웹 브라우저	    

# font-size

- 폰트의 크기를 지정

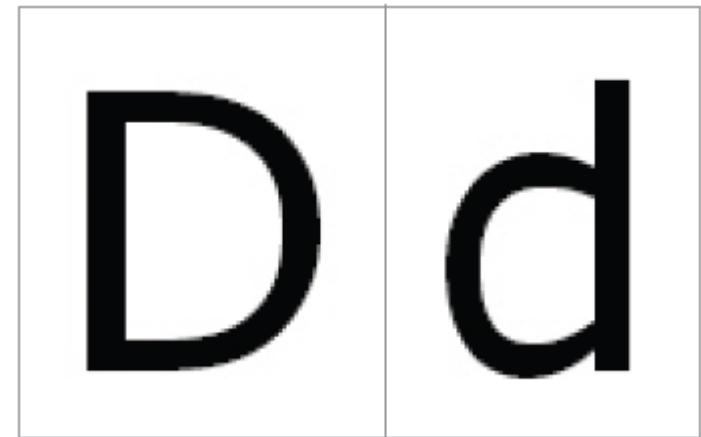
속성 값	<code>&lt;absolute-size&gt;</code>   <code>&lt;relative-size&gt;</code>   <code>&lt;length&gt;</code>   <code>&lt;percentage&gt;</code>
기본 값	medium
상속 여부	상속됨
지원 웹 브라우저	    

구분	<code>xx-small</code>	<code>x-small</code>	<code>small</code>	<code>medium</code>	<code>large</code>	<code>x-large</code>	<code>xx-large</code>
비율	3/5	3/4	8/9	1	6/5	3/2	2/1
Heading	h6		h5	h4	h3	h2	h1

# font-size-adjust

- 소문자의 높이를 기준으로 글자 크기를 조절할 때 사용

속성 값	none   auto   <number>
기본 값	medium
상속 여부	상속됨
지원 웹 브라우저	  ※ IE의 경우 none 값만 지원함



▲ font-size-adjust 속성 값이 0.7로 적용된 경우의 소문자 크기

# font

- 폰트 관련 속성을 일괄적으로 지정하기 위한 대표 속성

속성 값	[ [ <font-style>    <font-variant-css2>    <font-weight>    <font-stretch> ]? <font-size> [ / <line-height> ]? <font-family> ]   caption   icon   menu   message-box   small-caption   status-bar
기본 값	각 하위 속성 값 참조
상속 여부	상속됨
지원 웹 브라우저	    

# font-synthesis

- 지정된 글꼴에 볼드체와 기울임꼴체가 포함되어 있지 않을 경우, 웹 브라우저가 굵은 글꼴이나 기울어진 글꼴을 합성하여 표현할 것인지의 여부를 지정

속성 값	none   [ weight    style ]
기본 값	weight style
상속 여부	상속됨
지원 웹 브라우저	없음

# @font-face

- @font-face는 일반적인 속성과 달리 CSS에서 사용할 font-family의 이름과 자원을 정의할 수 있는 규칙으로, 사용자의 환경에 설치된 폰트 또는 제작자가 서버에서 제공한 폰트를 다운로드하여 사용할 수 있도록 하는 역할

지원 웹 브라우저



※ IE6~8의 경우 eot 형식만 지원

```
@font-face {  
    font-family : [ 사용할 폰트이름 ] ;  
    src : [폰트 자원의 위치] - local (폰트이름) , url (자원경로) ] ;  
}
```

# @font-face

폰트 유형	폰트 Format	폰트 확장자
"woff"	WOFF(Web Open Font Format)	.woff
"truetype"	TrueType	.ttf
"opentype"	OpenType	.ttf, .otf
"embedded-opentype"	Embedded OpenType	.eot
"svg"	SVG Font	.svg, .svgz

# font-variant

- 영문 폰트의 소문자 표시 형태를 변경

속성 값	normal   none   [ <common-lig-values>    <discretionary-lig-values>    <historical-lig-values>    <contextual-alt-values>    stylistic(<feature-value-name>)    historical-forms    styleset(<feature-value-name> #)    character-variant(<feature-value-name> #)    swash(<feature-value-name>)    ornaments(<feature-value-name>)    annotation(<feature-value-name>)    [ small-caps   all-small-caps   petite-caps   all-petite-caps   unicase   titling-caps ]    <numeric-figure-values>    <numeric-spacing-values>    <numeric-fraction-values>    ordinal    slashed-zero    <east-asian-variant-values>    <east-asian-width-values>    ruby ]
기본 값	normal
상속 여부	상속됨
지원 웹 브라우저	normal, none, small-caps 값은 모든 웹 브라우저에서 지원 그 밖의 속성 값은 지원 웹 브라우저 없음

없음

# font-feature-settings

- 오픈 타입 글꼴의 세부 설정을 조정하는 속성으로, 해당 오픈 타입 글꼴을 포함하는 글꼴의 문자 모양과 대체 문자, 글자 위치와 레이아웃 등을 조정할 수 있음.

속성 값	normal   <feature-tag-value>
기본 값	auto
상속 여부	상속됨
지원 웹 브라우저	    

# font-kerning

- 커닝 기능을 사용할 것인지의 여부를 지정

속성 값	auto   normal   none
기본 값	auto
상속 여부	상속됨
지원 웹 브라우저	  

# font-language-override

- lang 속성으로 정의된 언어를 다른 언어로 대체하는 속성

속성 값	normal   <string>
기본 값	normal
상속 여부	상속됨
지원 웹 브라우저	

# text-transform

- 텍스트의 대소 문자를 변환하기 위한 속성

속성 값	none   capitalize   uppercase   lowercase   full-width
기본 값	none
상속 여부	상속됨
지원 웹 브라우저	   

※ full-width 값의 경우 파이어폭스에서만 지원함

# white-space

- 텍스트의 공백과 줄바꿈의 처리 방법을 지정

속성 값	normal   pre   nowrap   pre-wrap   pre-line
기본 값	CSS 2에서의 기본 값은 normal이지만 CSS 3에서는 기본 값이 정의되지 않음
상속 여부	상속됨
지원 웹 브라우저	    

# tab-size

- 탭(tab) 크기를 지정

속성 값	<code>&lt;integer&gt;   &lt;length&gt;</code>
기본 값	8
상속 여부	상속됨
지원 웹 브라우저	없음

# line-break

- 요소 내 줄바꿈 규칙을 지정

속성 값	auto   loose   normal   strict
기본 값	auto
상속 여부	상속됨
지원 웹 브라우저	    ※ IE는 normal과 strict 값만 지원하고, 사파리는 strict 값만 지원함

# word-break

- 줄바꿈을 위한 단어 규칙을 지정

속성 값	normal   keep-all   break-all
기본 값	normal
상속 여부	상속됨
지원 웹 브라우저	     ※ 크롬, 사파리, 오페라에서 속성 값 keep-all은 지원하지 않음

# hyphens

- 줄바꿈으로 인해 단어가 나누어질 때 하이픈을 자동으로 삽입하는 방식을 지정

속성 값	none   manual   auto
기본 값	manual
상속 여부	상속됨
지원 웹 브라우저	    

# overflow-wrap/word-wrap

- overflow-wrap 속성과 word-wrap 속성은 동일한 속성으로 이 두 속성은 단어가 요소 박스의 너비보다 길어질 경우 자동 줄바꿈이 발생하는데 이때 단어를 나눌지 여부를 지정

속성 값	normal   break-word
기본 값	normal
상속 여부	상속됨
지원 웹 브라우저	  ※ 크롬과 오페라만 overflow-wrap 속성을 지원함      ※ word-wrap 속성은 모든 웹 브라우저에서 지원함

# text-align

- 단락 내 텍스트의 가로 방향 정렬 방법을 지정

속성 값	[ [ start   end   left   right   center ]    <string> ]   justify   match-parent   start end
기본 값	start
상속 여부	상속됨
지원 웹 브라우저	    

# text-align-last

- 단락의 마지막 줄 텍스트 정렬 방식을 지정

속성 값	auto   start   end   left   right   center   justify
기본 값	start
상속 여부	상속됨
지원 웹 브라우저	  ※ IE의 경우 start와 end 값은 지원하지 않음

# letter-spacing

- 글자와 글자 사이의 간격을 지정

속성 값	⟨spacing-limits⟩
기본 값	normal
상속 여부	상속됨
지원 웹 브라우저	    

# word-spacing

- 단어와 단어 사이의 간격을 지정

속성 값	⟨spacing-limits⟩
기본 값	normal
상속 여부	상속됨
지원 웹 브라우저	    

# text-indent

- 단락의 첫줄 들여쓰기를 지정

속성 값	[ <length>   <percentage> ] && [ hanging    each-line ]?
기본 값	0
상속 여부	상속됨
지원 웹 브라우저	    

# text-decoration

- 텍스트를 장식하는 속성으로, CSS3에서 세분화된 하위 속성이 추가되었음.

속성 값	<code>&lt;text-decoration-line&gt;    &lt;text-decoration-style&gt;    &lt;text-decoration-color&gt;</code>
기본 값	없음
상속 여부	상속되지 않음
지원 웹 브라우저	 ※ 파이어폭스에서는 세부 속성인 <code>text-decoration-line</code> , <code>text-decoration-style</code> , <code>text-decoration-color</code> 속성은 지원하지만 대표 속성인 <code>text-decoration</code> 속성은 지원하지 않음

# text-shadow

- 텍스트에 그림자를 지정

속성 값	none   [〈offset-x〉 〈offset-y〉 〈blur-radius〉 〈color〉]
기본 값	없음
상속 여부	상속됨
지원 웹 브라우저	    

# display

- HTML 요소의 표현 방식을 지정

속성 값	inline   block   inline-block   list-item   run-in   compact   table   inline-table   table-row-group   table-header-group   table-footer-group   table-row   table-column-group   table-column   table-cell   table-caption   ruby   ruby-base   ruby-text   ruby-base-group   ruby-text-group   <template>   none
기본 값	inline
상속 여부	상속되지 않음
지원 웹 브라우저	    

# padding

- content 영역과 border 사이의 안쪽 여백을 지정

속성 값	[ <length>   <percentage> ][1,4]
기본 값	세부 속성 참조
상속 여부	상속되지 않음
지원 웹 브라우저	    

# border

- 요소 박스의 테두리를 지정

속성 값	<code>&lt;border-width&gt;    &lt;border-style&gt;    &lt;color&gt;</code>
기본 값	세부 속성 참조
상속 여부	상속되지 않음
지원 웹 브라우저	    

# margin

- border를 기준으로 다른 요소와의 바깥쪽 여백을 지정.
- 속성 값에 음수 값을 사용할 수 있으며 상하로 인접한 박스의 display 속성 값이 "block"인 경우, 마진 겹침(Margin Collapsing) 현상이 발생함.

속성 값	[ <length>   <percentage>   auto ]{1,4}
기본 값	세부 속성 참조
상속 여부	상속되지 않음
지원 웹 브라우저	    

# width, height

- 요소의 너비와 높이를 지정

속성 값	<code>&lt;length&gt;   &lt;percentage&gt;   auto</code>
기본 값	<code>auto</code>
상속 여부	상속되지 않음
지원 웹 브라우저	    

## min-width, min-height, max-width, max-height

- 요소의 최소 및 최대 너비와 높이를 지정

속성 값	<code>&lt;length&gt;   &lt;percentage&gt;   inherit</code>
기본 값	0
상속 여부	상속되지 않음
지원 웹 브라우저	    

# visibility

- HTML 요소에 의해 만들어진 박스의 화면상 표시 여부를 지정

속성 값	visible   hidden   collapse
기본 값	visible
상속 여부	상속됨
지원 웹 브라우저	    

# overflow

- 콘텐츠가 요소의 영역을 벗어나는 경우의 처리 방법을 지정

속성 값	visible   hidden   scroll   auto   no-display   no-content
기본 값	visible
상속 여부	상속되지 않음
지원 웹 브라우저	    

# overflow-style

- 사용자의 선택에 따라 스크롤 방법을 결정하는 overflow 속성의 실행 방식을 지정하는 속성

속성 값	auto   [scrollbar   panner   move   marquee] [, [scrollbar   panner   move   marquee]]*
기본 값	auto
상속 여부	상속됨
지원 웹 브라우저	없음

# marquee-style, marquee-loop, marquee-direction, marquee-speed

- 기존에 인터넷 익스플로러에서 지원하던 `<marquee>` 요소와 동일하게 움직이는 효과를 지정하는 속성

## ※ marquee-style

속성 값	scroll   slide   alternate
기본 값	scroll
상속 여부	상속되지 않음
지원 웹 브라우저	없음

## ※ marquee-loop

속성 값	<code>&lt;non-negative-integer&gt;</code>   infinite
기본 값	1
상속 여부	상속되지 않음
지원 웹 브라우저	없음

## ※ marquee-direction

속성 값	forward   reverse
기본 값	reverse
상속 여부	상속됨
지원 웹 브라우저	없음

## ※ marquee-speed

속성 값	slow   normal   fast
기본 값	normal
상속 여부	상속되지 않음
지원 웹 브라우저	없음

# rotation, rotation-point

- 요소 박스의 회전 각도를 지정

※ rotation 속성

속성 값	$\langle\text{angle}\rangle$
기본 값	0
상속 여부	상속되지 않음
지원 웹 브라우저	없음

※ rotation-point 속성

속성 값	$\langle\text{bg-position}\rangle$
기본 값	50% 50%
상속 여부	상속되지 않음
지원 웹 브라우저	없음

# color

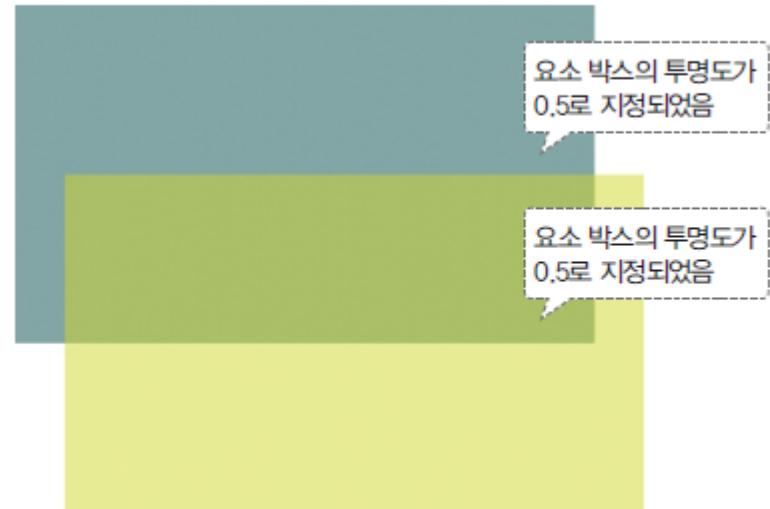
- 요소의 전경색을 지정

속성 값	<color>   inherit
기본 값	사용자 에이전트에 따름
상속 여부	상속됨
지원 웹 브라우저	    

# opacity

- 요소 박스의 투명도를 지정

속성 값	<code>&lt;alphavalue&gt;</code>   inherit
기본 값	1
상속 여부	상속됨
지원 웹 브라우저	    



# background-color

- 요소의 배경색을 지정

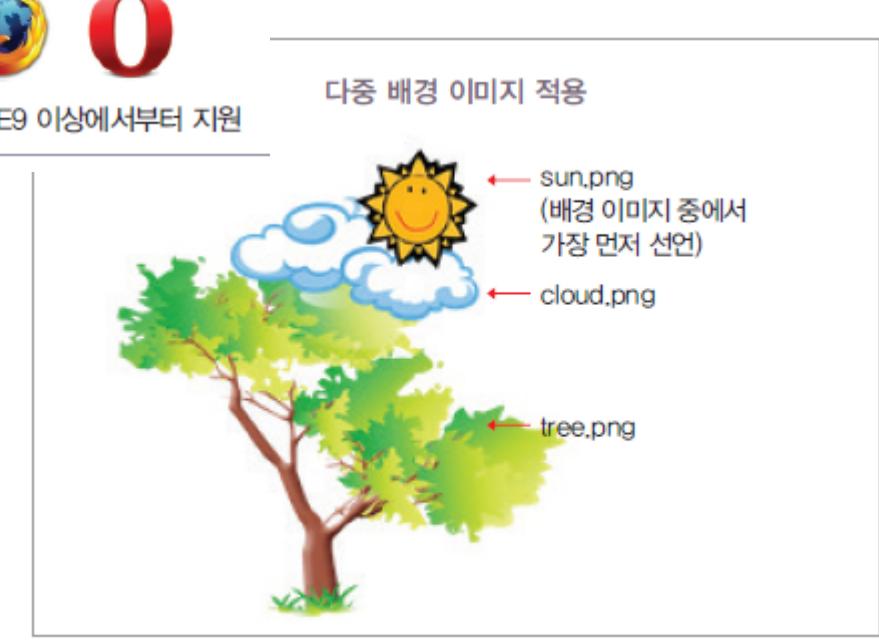
속성 값	<color>   inherit
기본 값	Transparent
상속 여부	상속되지 않음
지원 웹 브라우저	    

# background-image

- 요소 박스에 배경 이미지를 지정. CSS3에서는 하나의 요소 박스의 여러 개의 배경 이미지를 지정할 수 있음.

속성 값	<code>&lt;bg-image&gt; [ , &lt;bg-image&gt; ]*</code>
기본 값	없음
상속 여부	상속되지 않음
지원 웹 브라우저	     ※ 여러개의 배경 이미지를 지정할 경우 IE9 이상에서부터 지원

- CSS3에서 background-image 속성의 값으로 그레이디언트 효과를 지정할 수 있음.



# linear-gradient( )

- 선형 그레이디언트 효과를 지정

지원 웹 브라우저



```
형식    linear-gradient(  
           [angle]  
           , [color-stop]  
           )
```

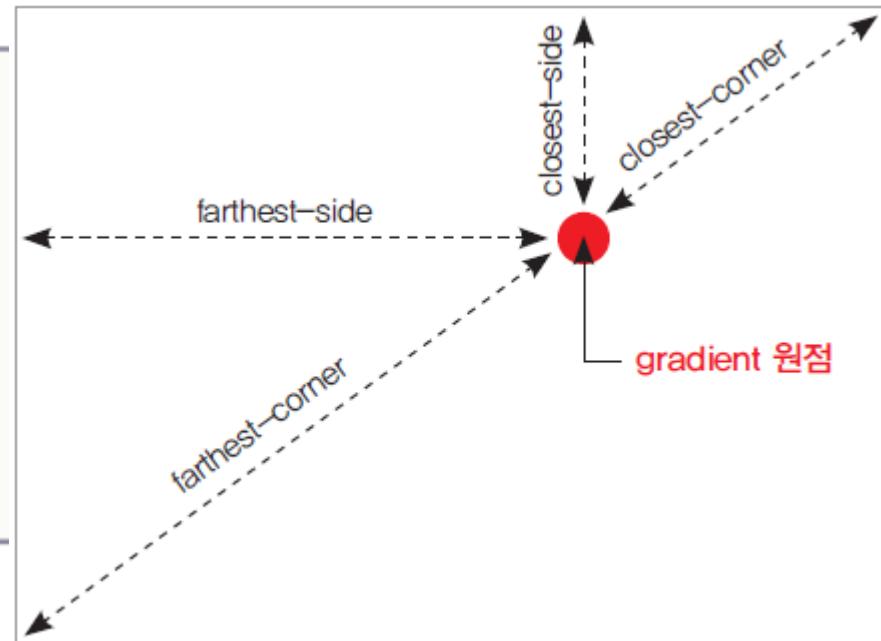
# radial-gradient( )

- 원형 그레이디언트 효과를 지정

지원 웹 브라우저



형식      `radial-gradient (`  
              [shape]  
              [size]  
              [position]  
              , [color-stop]  
              )



# repeating-linear-gradient( )

- 색상 변환이 반복되는 선형 그레이디언트 효과를 지정

지원 웹 브라우저



형식      repeating-linear-gradient (  
              [angle]  
              , [color-stop]  
              )

# repeating-radial-gradient( )

- 색상 변환이 반복되는 원형 그레이디언트 효과를 지정

지원 웹 브라우저



형식      repeating-radial-gradient (  
              [shape]  
              [size]  
              [position]  
              , [color-stop]  
              )

# background-repeat

- 배경 이미지의 반복 여부를 지정

속성 값	repeat, repeat-x, repeat-y, no-repeat, space, round, inherit
기본 값	repeat
상속 여부	상속되지 않음
지원 웹 브라우저	    

# background-attachment

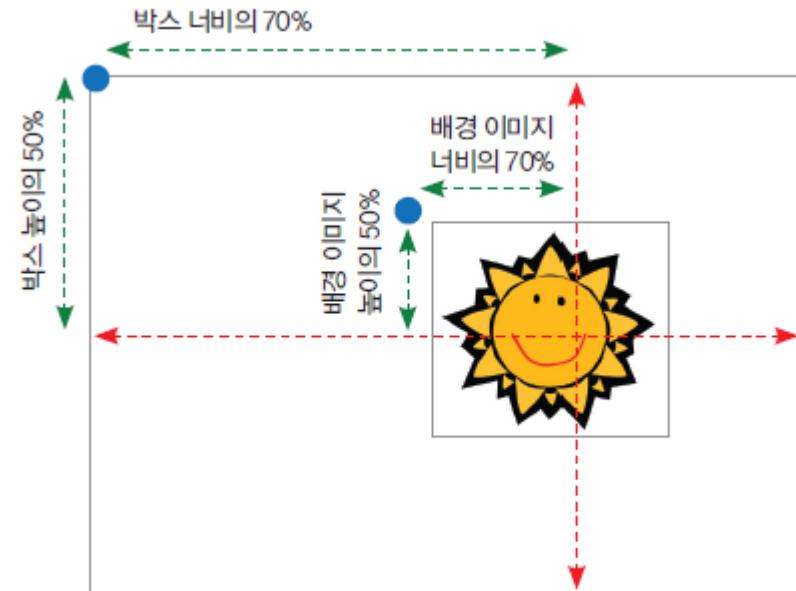
- 배경 이미지의 고정 여부를 지정

속성 값	scroll   fixed   local
기본 값	scroll
상속 여부	상속되지 않음
지원 웹 브라우저	     ※ local 속성 값은 파이어폭스에서 지원하지 않음

# background-position

- 배경 이미지의 위치를 지정

속성 값	left   center   right   top   bottom   <percentage>   <length>
기본 값	0 0
상속 여부	상속되지 않음
지원 웹 브라우저	   



▲ 가로 방향으로 70%, 세로 방향으로 50% 지점에 배경 이미지를 배치한 예

# background-clip

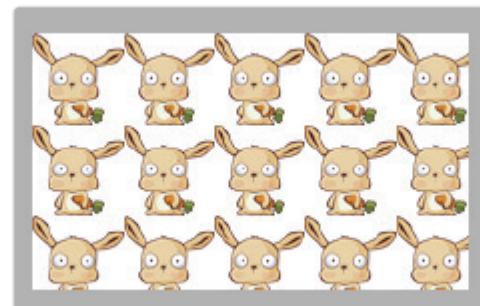
- 배경 속성이 적용되는 영역을 지정

속성 값	border-box   padding-box   content-box
기본 값	border-box
상속 여부	상속되지 않음
지원 웹 브라우저	    

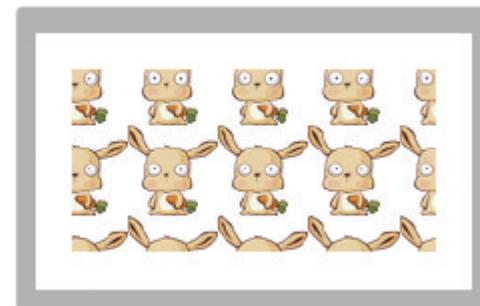
border-box



padding-box



content-box



▲ background-clip 속성의 적용 예

# background-origin

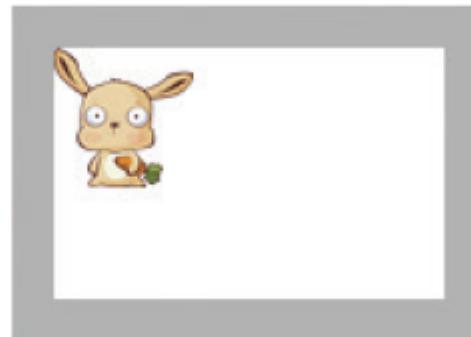
- 배경 이미지의 시작점을 지정

속성 값	border-box   padding-box   content-box
기본 값	border-box
상속 여부	상속되지 않음
지원 웹 브라우저	    

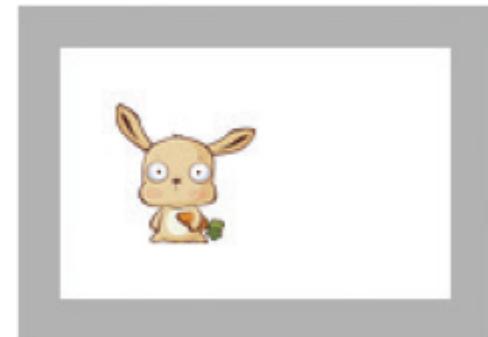
border-box



padding-box



content-box



▲ background-origin 속성의 적용 예

# background-size

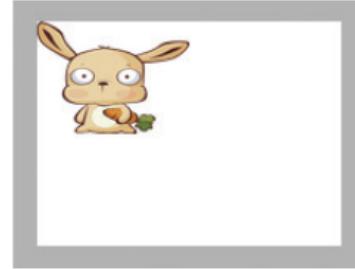
- 배경 이미지의 크기를 지정

속성 값	<code>&lt;length&gt;   &lt;percentage&gt;   auto </code> [1,2]   cover   contain
기본 값	auto
상속 여부	상속되지 않음
지원 웹 브라우저	    

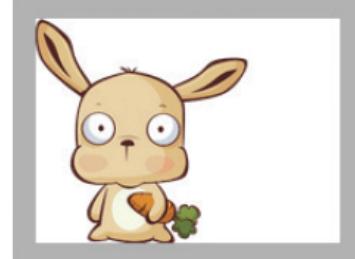
가로 세로 50px



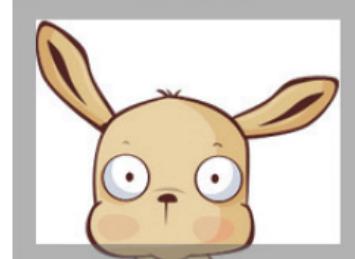
가로 세로 50%



contain



cover



▲ background-size 속성의 적용 예

# background

- 배경 관련 속성을 일괄적으로 지정

속성 값	[ <bg-layer> , ]* <final-bg-layer>  <bg-layer> : <bg-image>    <position> [ / <bg-size> ]?    <repeat-style>    <attachment>    <box>{1,2} <final-bg-layer> : <bg-image>    <position> [ / <bg-size> ]?    <repeat-style>    <attachment>    <box>{1,2}    <background-color>
기본 값	세부 속성 참조
상속 여부	상속되지 않음
지원 웹 브라우저	    

# border-color

- 요소 박스의 테두리 선 색상을 지정

속성 값	color
기본 값	currentColor
상속 여부	상속되지 않음
지원 웹 브라우저	    

# border-style

- 요소 박스 테두리의 선 스타일을 지정

속성 값	none   hidden   dotted   dashed   solid   double   groove   ridge   inset   outset
기본 값	세부 속성 참조(none)
상속 여부	상속되지 않음
지원 웹 브라우저	    

# border-width

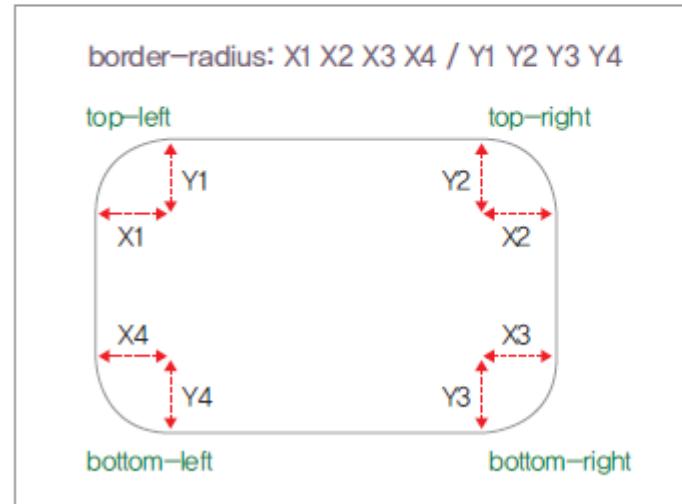
- 요소 박스 테두리의 선 굵기를 지정

속성 값	<code>&lt;length&gt;   thin   medium   thick</code>
기본 값	세부 속성 참조( <code>none</code> )
상속 여부	상속되지 않음
지원 웹 브라우저	    

# border-radius

- 요소 박스의 테두리 선을 둥근 모서리 형태로 지정

속성 값	[ <length>   <percentage> ][1,2]
기본 값	세부 속성 참조
상속 여부	상속되지 않음
지원 웹 브라우저	   

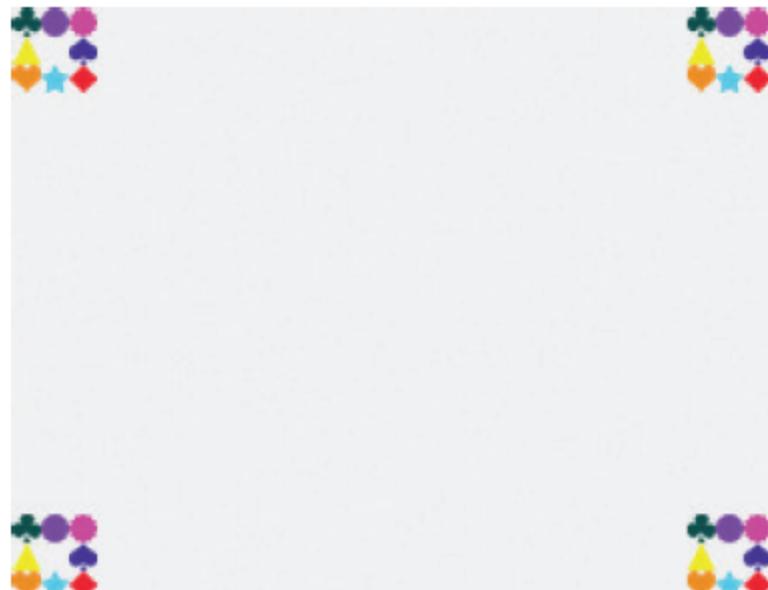


▲ border-radius 속성 값의 개수와 모서리 방향과의 관계

# border-image-source

- 요소 박스 테두리에 표시될 이미지를 지정

속성 값	none   <image>
기본 값	none
상속 여부	상속되지 않음
지원 웹 브라우저	   

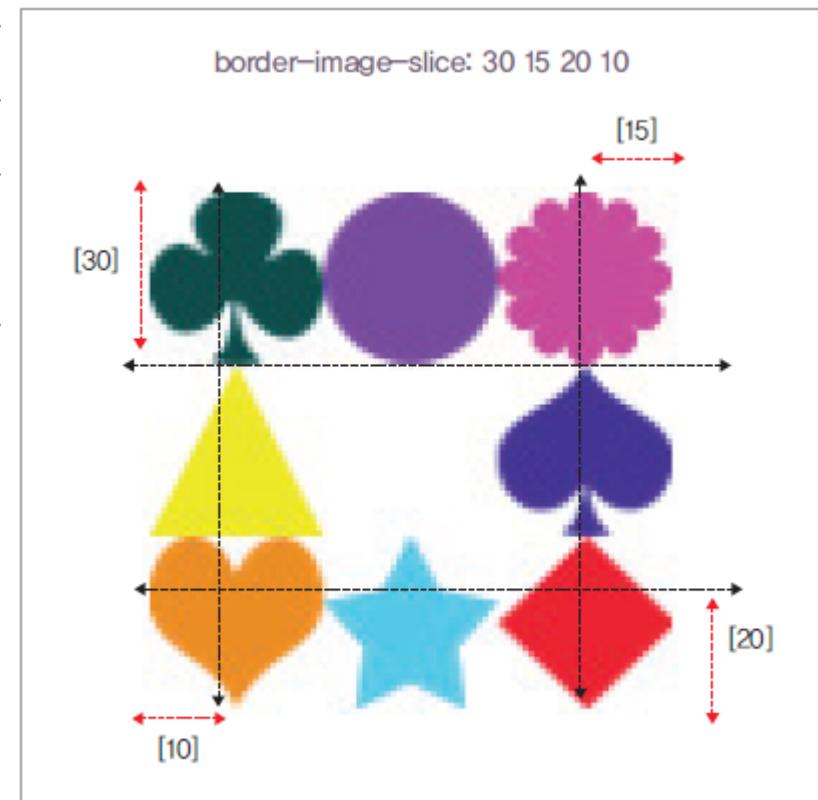


▲ border-image-source 속성의 적용 예

# border-image-slice

- 테두리에 지정할 이미지가 요소 박스 상하좌우 테두리 영역에 보여질 크기를 지정

속성 값	<code>[&lt;number&gt;   &lt;percentage&gt;]{1,4} &amp;&amp; fill?</code>
기본 값	none
상속 여부	상속되지 않음
지원 웹 브라우저	

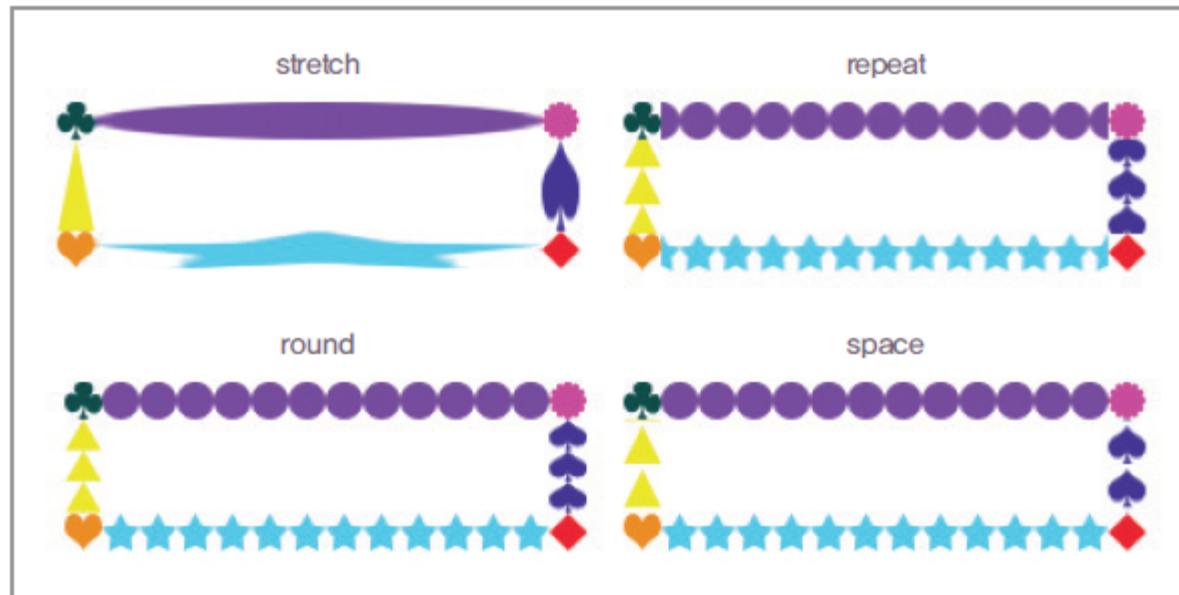


▲ border-image-slice 속성을 사용하여 9등분으로 분할한 결과

# border-image-repeat

- 요소 박스에 지정된 이미지가 반복되는 스타일을 지정

속성 값	[ stretch   repeat   round   space ][1,2]
기본 값	stretch
상속 여부	상속되지 않음
지원 웹 브라우저	   

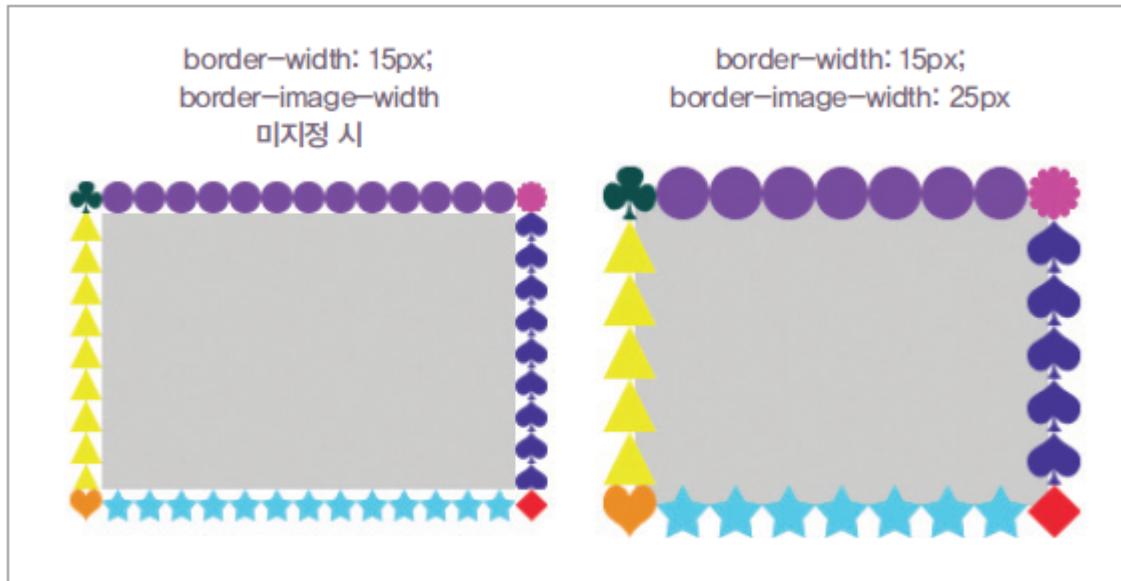


▲ border-image-repeat 속성 적용 예

# border-image-width

- 각 방향별로 border 이미지 표시 영역의 길이를 지정

속성 값	[ <length>   <percentage>   <number>   auto ]{1,4}
기본 값	1
상속 여부	상속되지 않음
지원 웹 브라우저	   

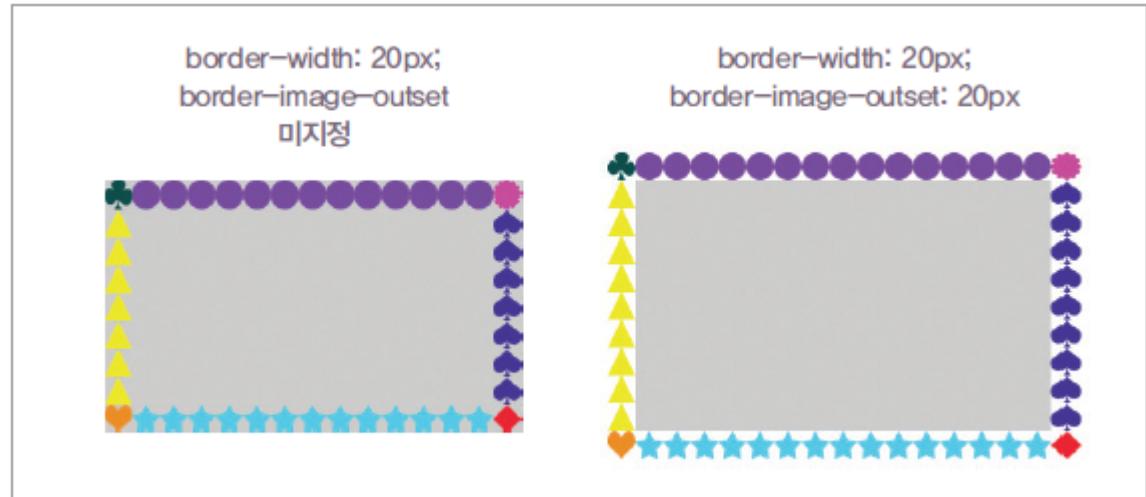


▲ border-width와 border-image-width를 각각 다르게 지정한 경우

# border-image-outset

- 테두리에 지정한 이미지가 border 영역에서 벗어나 요소 박스 외부에 표시되는 이미지 영역을 지정

속성 값	[<length>   <number> ][1,4]
기본 값	0
상속 여부	상속되지 않음
지원 웹 브라우저	   



▲ border-image-outset 속성의 비교

# border-image

- 요소 박스의 테두리 이미지 지정과 관련된 속성을 일괄 적용하는 대표 속성

속성 값	<code>&lt;border-image-source&gt;    &lt;border-image-slice&gt; [ / &lt;border-image-width&gt;   / &lt;border-image-width&gt;? / &lt;border-image-outset&gt; ]?    &lt;border-image-repeat&gt;</code>
기본 값	세부 속성 값 참조
상속 여부	상속되지 않음
지원 웹 브라우저	   

# box-decoration-break

- 요소 박스가 페이지나 단으로 인해 나누어졌거나 인라인 요소가 자동 줄바꿈 때문에 나누어졌을 경우, border나 padding을 추가할 것인지의 여부를 지정

속성 값	slice   clone
기본 값	slice
상속 여부	상속되지 않음
지원 웹 브라우저	 

box-decoration-break는 요소 박스가 페이지나 단으로 인해 나뉘어졌거나 인라인 요소가 자동 줄바꿈 때문에 나뉘어졌을 경우 border나 padding을 추가할지 여부를 지정하는 속성입니다.

box-decoration-break는 요소 박스가 페이지나 단으로 인해 나뉘어졌거나 인라인 요소가 자동 줄바꿈 때문에 나뉘어졌을 경우 border나 padding을 추가할지 여부를 지정하는 속성입니다.

----- slice를 적용한 경우 -----

----- clone을 적용한 경우 -----

▲ 인라인 요소에 box-decoration-break 속성을 지정한 결과

# box-shadow

- 요소 박스의 그림자를 지정

속성 값	none   [ inset ? <offset-x> <offset-y> <blur-radius> <spread-radius> <color> ]
기본 값	none
상속 여부	상속되지 않음
지원 웹 브라우저	    

# float

- 일반적인 흐름에서 분리된 요소를 부모 영역을 기준으로 배치하는 속성

속성 값	left   right   none   <page-floats>
기본 값	none
상속 여부	상속되지 않음
지원 웹 브라우저	    

# clear

- float 속성의 선언으로 요소의 배치 위치에 영향을 받게 된 경우, 이를 해제하고자 할 때 사용하는 속성

속성 값	none   left   right   both
기본 값	None
상속 여부	상속되지 않음
지원 웹 브라우저	    

# position

- 요소 박스의 배치 방식을 지정

속성 값	static   relative   absolute   center   page   fixed
기본 값	static
상속 여부	상속되지 않음
지원 웹 브라우저	     ※ center, page 속성 값은 현재 지원하는 웹 브라우저가 없음

# top, right, bottom, left

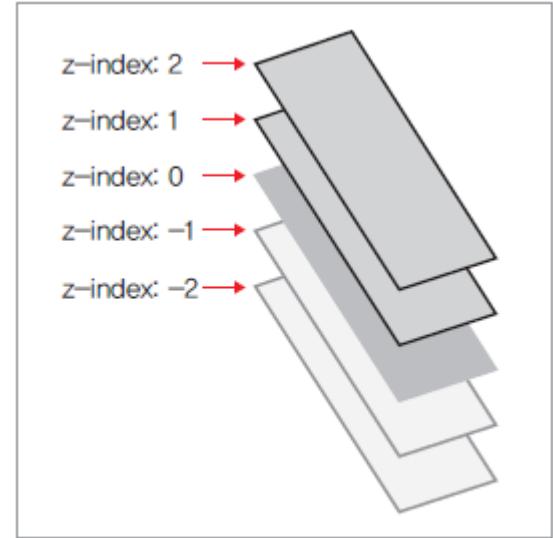
- position 속성 값이 static이 아닌 경우, 해당 요소의 위치를 지정할 때 사용 할 수 있는 offset 관련 속성

속성 값	auto   <length>   <percentage>
기본 값	auto
상속 여부	상속되지 않음
지원 웹 브라우저	    

# z-index

- position 속성 값이 static이 아닌 요소의 offset 관련 속성 값과 겹쳤을 경우, 겹치는 순서를 지정하는 속성

속성 값	auto   <integer>
기본 값	auto
상속 여부	상속되지 않음
지원 웹 브라우저	    

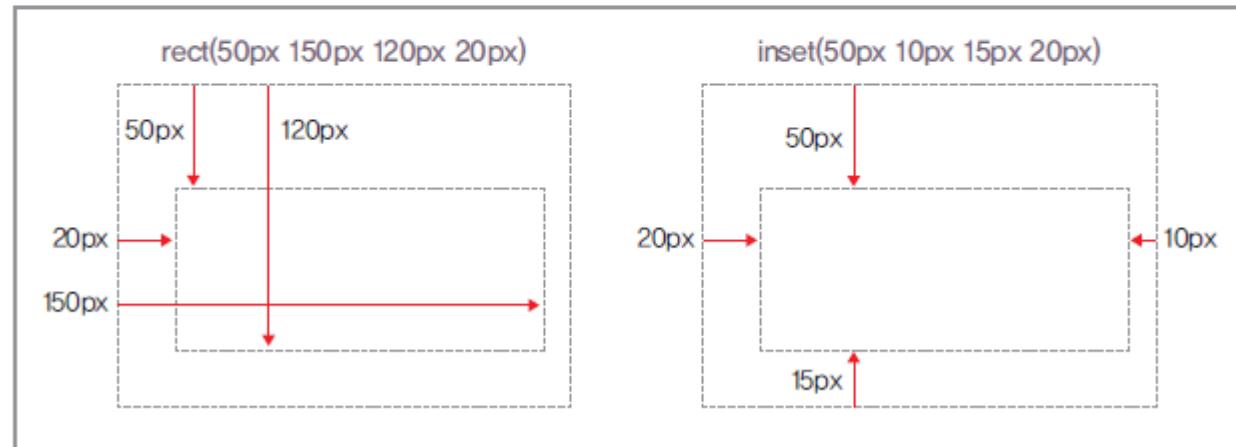


▲ z-index 속성 값에 따른 요소 박스의 겹침 순서

# clip

- 요소 박스의 특정 영역만 나타나도록 지정

속성 값	auto   <shape>
기본 값	auto
상속 여부	상속되지 않음
지원 웹 브라우저	    



▲ rect VS inset

# column-width

- 다단 구성 시 단의 너비를 지정

속성 값	<code>&lt;length&gt;   auto</code>
기본 값	<code>auto</code>
상속 여부	상속되지 않음
지원 웹 브라우저	    

### 웹 표준 기술

웹 표준 기술은 정치 운영 체제 독립적입니다. 웹은 다양한 환경에서 정보를 제공하고자 하는 목적을 가지고 탄생했습니다. W3C의 웹 표준은 웹이 가진 목적을 구현하고, 웹이 가진 모든 가능성을 풀어내기 위해 제정되었습니다.

HTML에는 웹이 전달하고자 하는 정보가 담겨 있습니다. 따라서 웹을 통해 전달하고자 하는 주요한 정보들은 가능한 한 모두 HTML 안에 담아야 합니다. 현재는 HTML 4.0과 마을 XML로 재구성한 XHTML 1.0이 최신 표준이며, 차세대 표준으로 HTML 5가 작성 중입니다.

CSS는 웹이 어떻게 보여질 것인가를 담당합니다. 다시 말해서 CSS는 웹의 표현을 제어하는 언어입니다. CSS level 1이 최신 표준이지만, 권고 후보인 CSS level 2가 사실상의 표준으로 널리 사용되고 있습니다. 현재는 CSS level 3가 작성 중이며, HTML 5와 함께 차세대 웹 표준으로 주목받고 있습니다.

javascript는 웹 브라우저상에서 해석되거나 동작하는 프로그램 언어로, 사용자 층의 동작 요소를 담당합니다. 문서를 기반으로 하는 기존의 웹에서는 무기적인 요소였지만, 웹이 애플리케이션화되어 가면서 역할이 점차 중요해지고 있습니다. 웹 표준 기술 요소는 서로 분리된 구조로 사용할 때 장점을 극대화시킬 수 있습니다.

▲ column-width 속성 적용 예

# column-count

- 다단 구성 시 단의 개수를 지정

속성 값	<code>&lt;integer&gt;   auto</code>
기본 값	<code>auto</code>
상속 여부	상속되지 않음
지원 웹 브라우저	    

# columns

- 단의 너비와 개수를 일괄 지정하기 위한 대표 속성

속성 값	<code>&lt;column-width&gt;    &lt;column-count&gt;</code>
기본 값	개별 속성 참조
상속 여부	상속되지 않음
지원 웹 브라우저	    

# column-gap

- 단과 단 사이의 간격을 지정

속성 값	<code>&lt;length&gt;   normal</code>
기본 값	<code>normal</code>
상속 여부	상속되지 않음
지원 웹 브라우저	    

# column-rule

- 단과 단 사이의 구분선을 표현하기 위한 속성

속성 값	<code>&lt;column-rule-width&gt;    &lt;column-rule-style&gt;    [&lt;column-rule-color&gt;   transparent ]</code>
기본 값	세부 속성 값 참조
상속 여부	상속되지 않음
지원 웹 브라우저	    

웹 표준 기술은 장치 운영 체제 독립적입니다.

웹은 다양한 환경에서 정보를 제공하고자 하는 목적을 가지고 탄생했습니다. W3C의 웹 표준은 웹이 가진 목적을 구현하고, 웹이 가진 모든 가능성을 끌어내기 위해 제정되었습니다.

## 웹 표준 기술

HTML에는 웹이 전달하고자 하는 정보가 담겨 있습니다.

따라서 웹을 통해 전달하고자 하는 주요한 정보들은 가능한 한 모두 HTML 안에 담아야 합니다. 현재는 HTML 4.01과 이를 XML로 재구성한 XHTML 1.0이 최신 표준이며, 차세대 표준으로 HTML 5가 작성 중입니다.

CSS는 웹이 어떻게 보여질 것인가를 담당합니다. 다시 말해서 CSS는 웹의 표현을 제어하는 언어입니다. CSS level 1이 최신 표준이지만, 권고 후보인 CSS level 2가 사실상의 표준으로 널리 사용되고 있습니다. 현재는

CSS level 3가 작성 중이며, HTML 5와 함께 차세대 웹 표준으로 주목받고 있습니다.

javascript는 웹 브라우저상에서 해석되거나 동작하는 프로그램 언어로, 사용자 측의 동작 요소를 담당합니다. 문서를 기반으로 하는 기준의 웹에서는 부가적인 요소였지만, 웹이 애플리케이션화되어 가면서 역할이 점차 중요해지고 있습니다. 웹 표준 기술 요소는 서로 분리된 구조로 사용할 때 장점을 극대화시킬 수 있습니다.

# column-span

- 다단 구성 시 여러 단을 차지하는 요소를 지정

속성 값	none   all
기본 값	none
상속 여부	상속되지 않음
지원 웹 브라우저	    ※ 인터넷 익스플로러와 사파리는 none 값을 지원하지 않음

웹 표준 기술은 장치 운영 체제 독립적입니다.

웹은 다양한 환경에서 정보를 제공하고자 하는 목적을 가지고 탄생했습니다. W3C의 웹 표준은 웹이 가진 목적을 구현하고, 웹이 가진 모든 가능성을 끌어내기 위해 제정되었습니다.

### 웹 표준 기술

HTML에는 웹이 전달하고자 하는 정보가 담겨 있습니다. 따라서 웹을 통해 전달하고자 하는 주요한 정보들은 가능한 한 모두 HTML 안에 담아야 합니다. 현재는 HTML 4.01과 이를 XML로 재구성한 XHTML 1.0이 최신 표준이며, 차세대 표준으로 HTML 5가 작성 중입니다.

CSS는 웹이 어떻게 보여질 것인가를 담당합니다. 다시 말해서 CSS는 웹의 표현을 제어하는 언어입니다. CSS level 1이 최신 표준이지만, 권고 후보인 CSS level 2가 사실상의 표준으로 널리 사용되고 있습니다. 현재는 CSS level 3가 작성 중이며, HTML 5와 함께 차세대 웹 표준으로 주목받고 있습니다.

javascript는 웹 브라우저에서 해석되거나 동작하는 프로그램 언어로, 사용자 측의 동적 요소를 담당합니다. 문서를 기반으로 하는 기준의 웹에서는 무가적인 요소였지만, 웹이 애플리케이션화되어 가면서 역할이 점차 중요해지고 있습니다. 웹 표준 기술 요소는 서로 분리된 구조로 사용할 때 장점을 극대화시킬 수 있습니다.

▲ column-span 속성 적용 예

# column-fill

- 다단 영역 콘텐츠 흐름의 영향을 받는 방식을 지정

속성 값	auto   balance
기본 값	balance
상속 여부	상속되지 않음
지원 웹 브라우저	 

## 웹 표준 기술

웹 표준 기술은 장치 운영

체제 독립적입니다.

웹은 다양한 환경에서 정보를 제공하고자 하는 목적을 가지고 탄생했습니다. W3C

의 웹 표준은 웹이 가진 목적을 구현하고, 웹이 가진 모든 가능성을 끌어내기 위해 제정되었습니다.

## 웹 표준 기술

웹 표준 기술은 장치 운영  
체제 독립적입니다.

웹은 다양한 환경에서 정보를 제공하고자 하는 목적을 가지고 탄생했습니다. W3C의 웹 표준은 웹이 가진 목적을 구현하고, 웹이 가진 모든 가능성을 끌어내기 위해 제정되었습니다.

▲ column-fill 속성 적용 예

# break-before, break-after, break-inside

- 다단의 분할 방식을 지정

※ break-before, break-after

속성 값	auto   always   avoid   left   right   page   column   avoid-page   avoid-column
기본 값	auto
상속 여부	상속되지 않음
지원 웹 브라우저	 10+

※ break-inside

속성 값	auto   avoid   avoid-page   avoid-column
기본 값	auto
상속 여부	상속되지 않음
지원 웹 브라우저	 10+

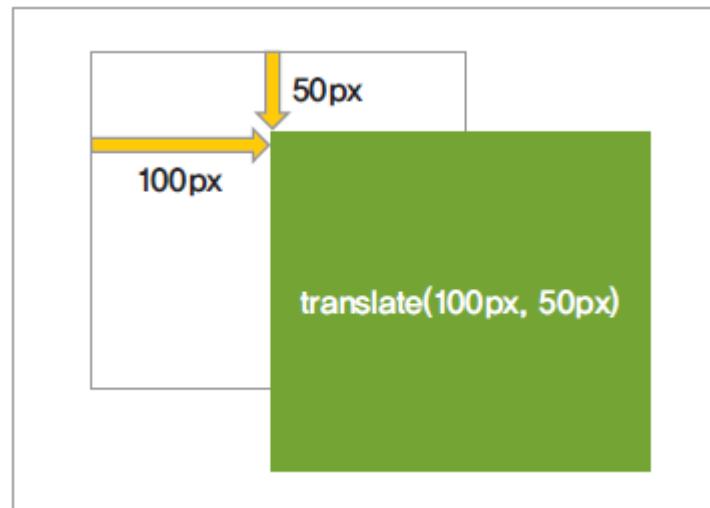
# transform

- 요소 박스를 변형하는 속성으로, 2차원 및 3차원 변형이 가능하며 변형 형태 별로 함수 타입의 속성 값을 지정. 이때 속성 값을 공란으로 구분하여 복수의 속성 값을 지정할 수 있음.

속성 값	none   <transform-function> [ <transform-function> ]*
기본 값	None
상속 여부	상속되지 않음
지원 웹 브라우저	     ※ IE9의 경우 2D Transform만 지원

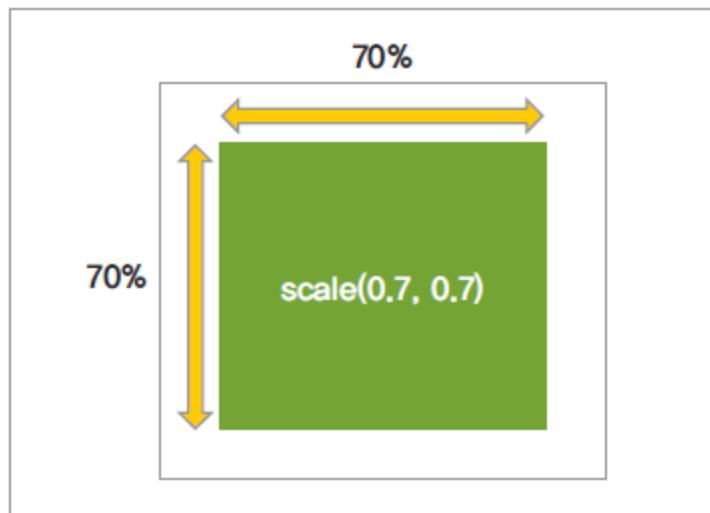
# translate()

- HTML 요소 박스를 평면상에서 수평 이동하는 기능



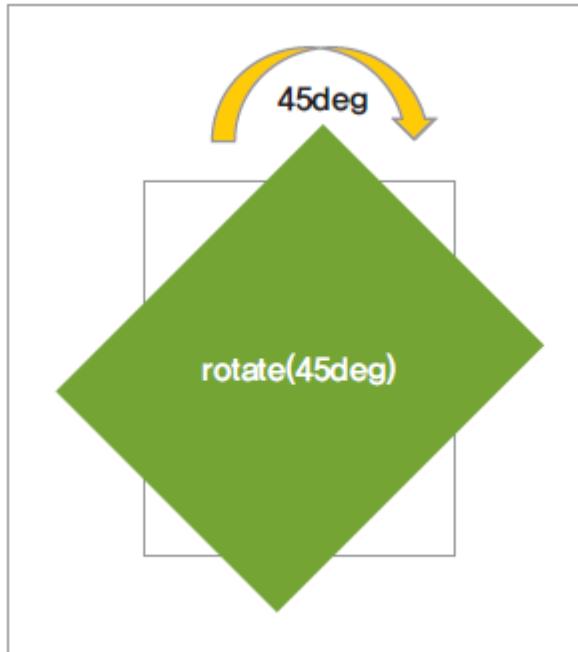
# scale( )

- HTML 요소 박스의 크기를 변형하는 기능



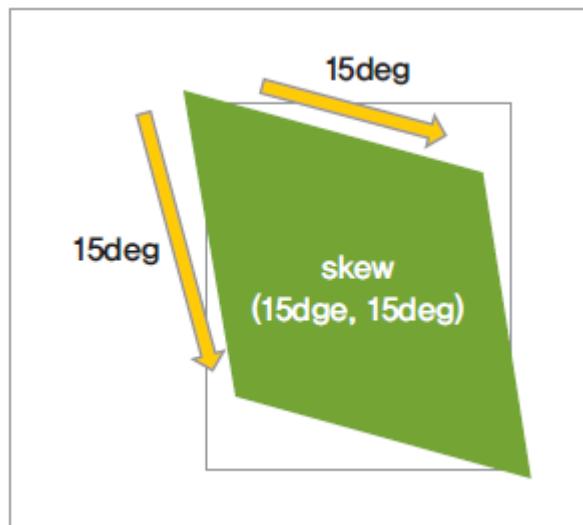
# rotate()

- HTML 요소 박스를 평면상에서 회전하는 기능으로, 함수의 값으로 회전 각도(deg)를 지정할 수 있음.



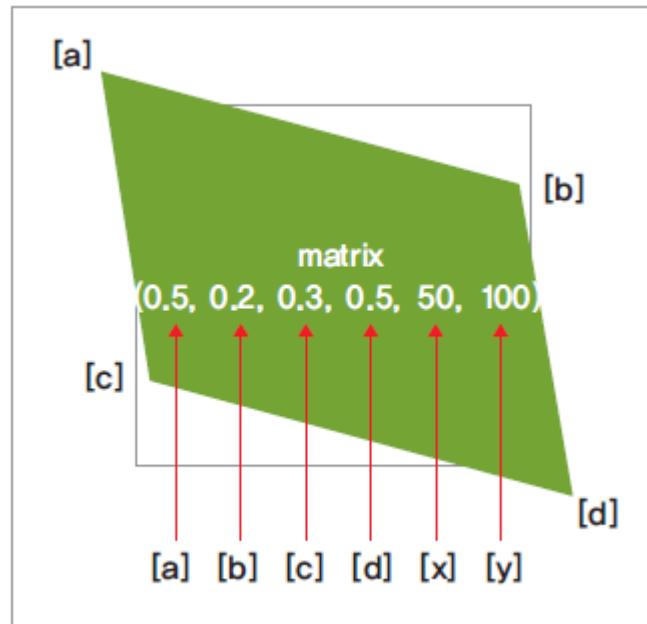
# skew( )

- HTML 요소 박스의 기울임을 지정하는 기능



# matrix( )

- HTML 요소 박스에 이동 및 회전 크기 변화와 기울임 등을 복합적으로 적용 할 수 있는 기능.
- 6개의 값은 순서대로 scaleX, skewX, skewY, scaleY, translateX, translateY 임.



# transform-style

- 3D 공간에서 변경된 3차원 좌표를 하위 요소에게 전달할 것인지의 여부를 결정하는 속성

속성 값	flat   preserve-3d
기본 값	Flat
상속 여부	상속되지 않음
지원 웹 브라우저	     ※ IE의 경우 preserve-3d 값을 지원하지 않음

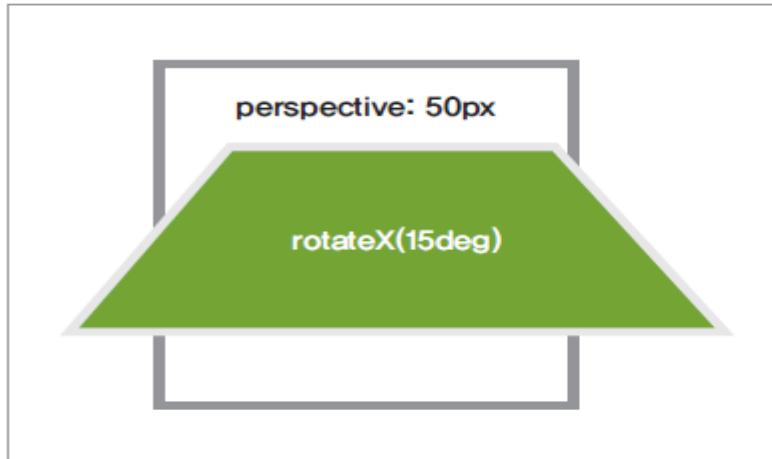


# perspective

- 3D 공간에서 요소와 관측 점과의 거리, 즉 원근감을 지정

속성 값	none   <length>
기본 값	none
상속 여부	상속되지 않음
지원 웹 브라우저	   

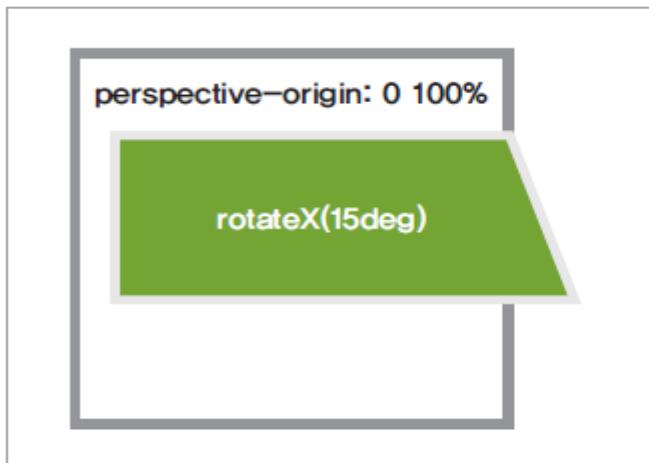
※ 파이어폭스의 경우 0 값은 지원하지 않음



# perspective—origin

- 3D 공간상에서 원근감의 방향을 지정하는 속성으로, 요소의 좌측 상단을 기준으로 지정된 위치 값부터 원근감의 관측점이 적용 됨.

속성 값	<code>&lt;percentage&gt;   &lt;length&gt;   left   center   right   top   bottom</code>
기본 값	<code>none</code>
상속 여부	상속되지 않음
지원 웹 브라우저	    



# backface-visibility

- 요소 박스가 뒤집혔을 경우 박스 뒷면의 표시 여부를 지정

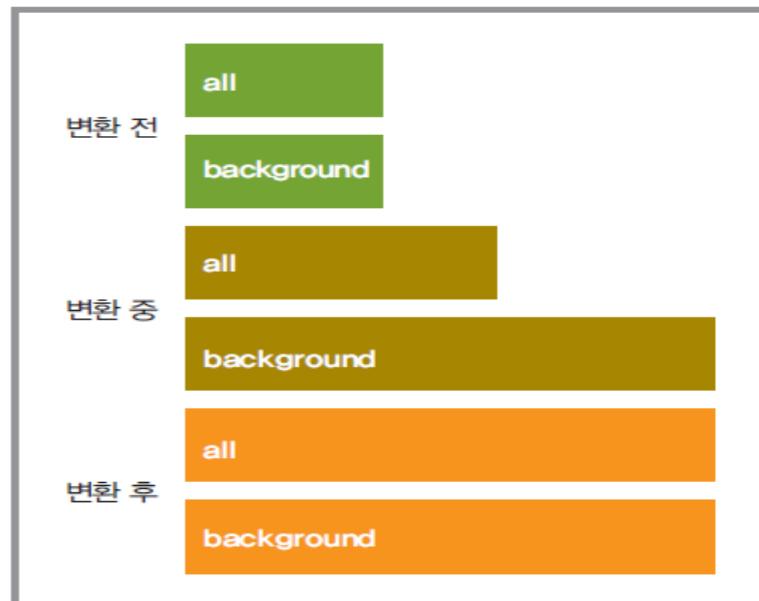
속성 값	visible   hidden
기본 값	visible
상속 여부	상속되지 않음
지원 웹 브라우저	    



# transition-property

- 요소에 지정된 속성을 변환하고자 할 때 사용

속성 값	none   all   <single-transition-property> [ '.' <single-transition-property> ]*
기본 값	all
상속 여부	상속되지 않음
지원 웹 브라우저	   



# transition-duration

- 변환이 진행되는 시간을 지정

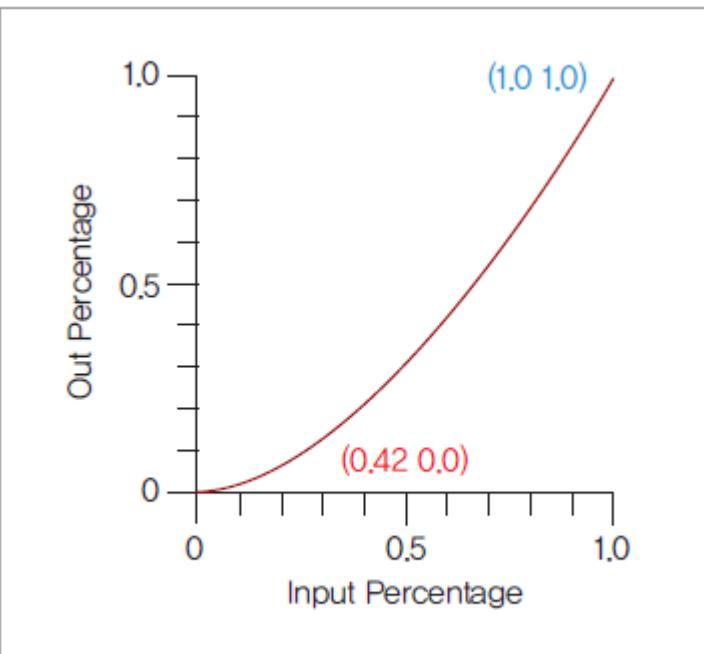
속성 값	<code>&lt;time&gt; [, &lt;time&gt;]*</code>
기본 값	0s
상속 여부	상속되지 않음
지원 웹 브라우저	   

# transition-timing-function

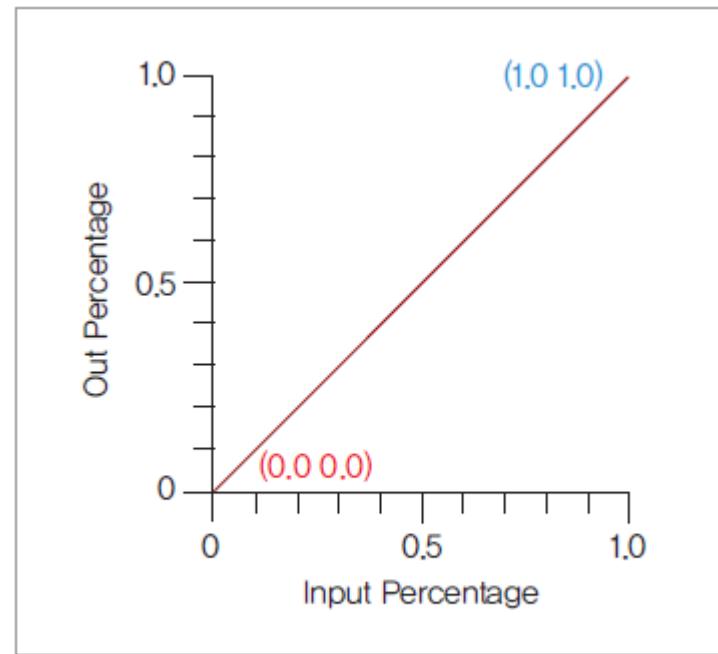
- 속성이 변환될 때 진행 속도의 형태를 지정

속성 값	ease   linear   ease-in   ease-out   ease-in-out   step-start   step-end   steps(<integer>[, [ start   end ] ]?)   cubic-bezier(<number>, <number>, <number>, <number>)
기본 값	ease
상속 여부	상속되지 않음
지원 웹 브라우저	    

# transition-timing-function 속성 값

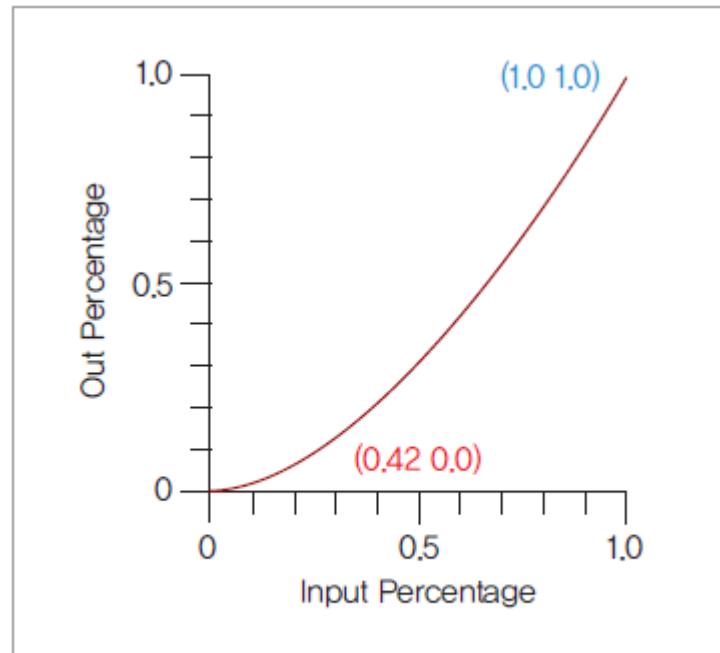


▲ ease의 속도 변화를 나타낸 그래프

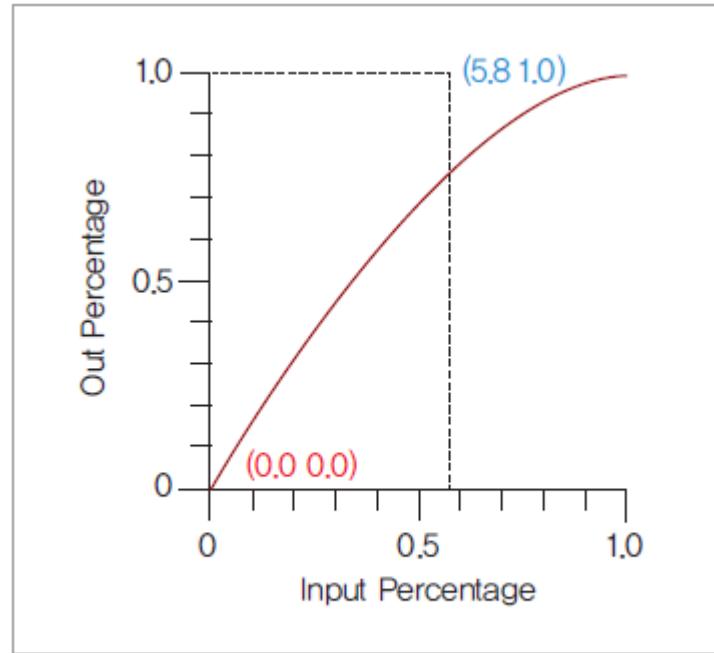


▲ linear의 속도 변화를 나타낸 그래프

# transition-timing-function 속성 값

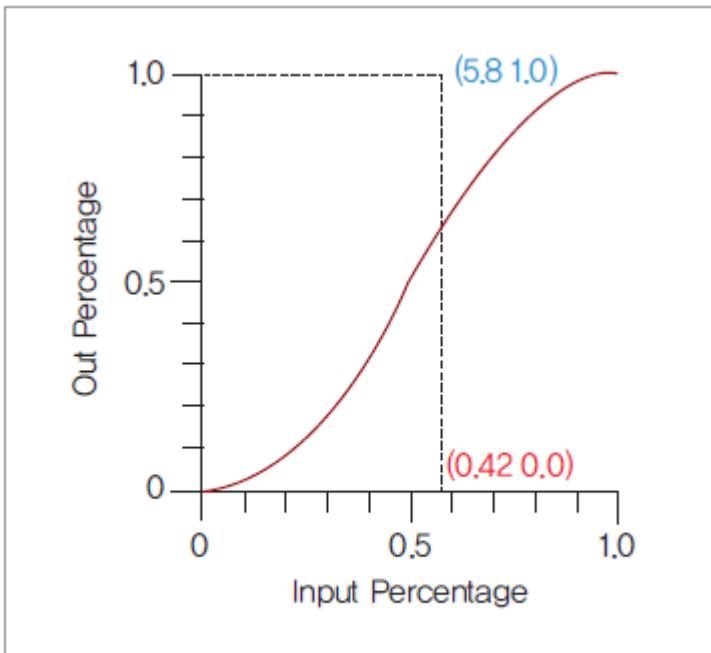


▲ ease-in의 속도 변화를 나타낸 그래프

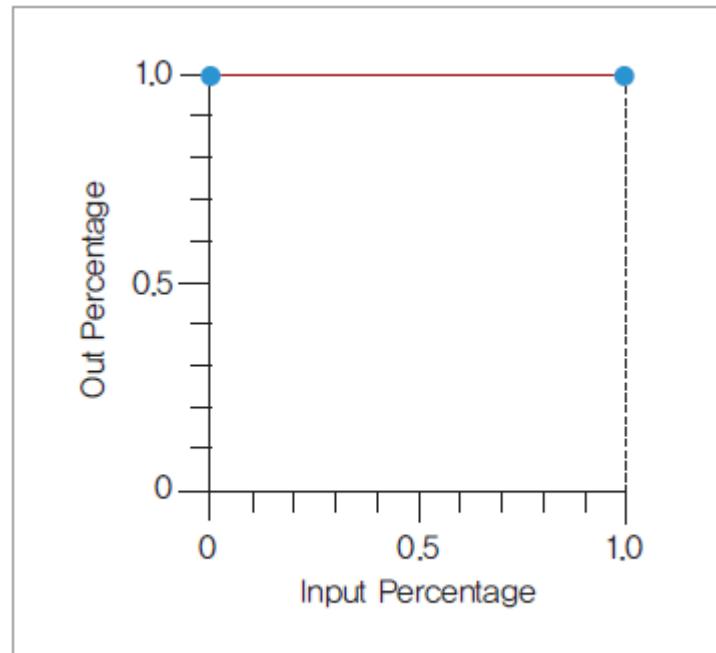


▲ ease-out의 속도 변화를 나타낸 그래프

# transition-timing-function 속성 값

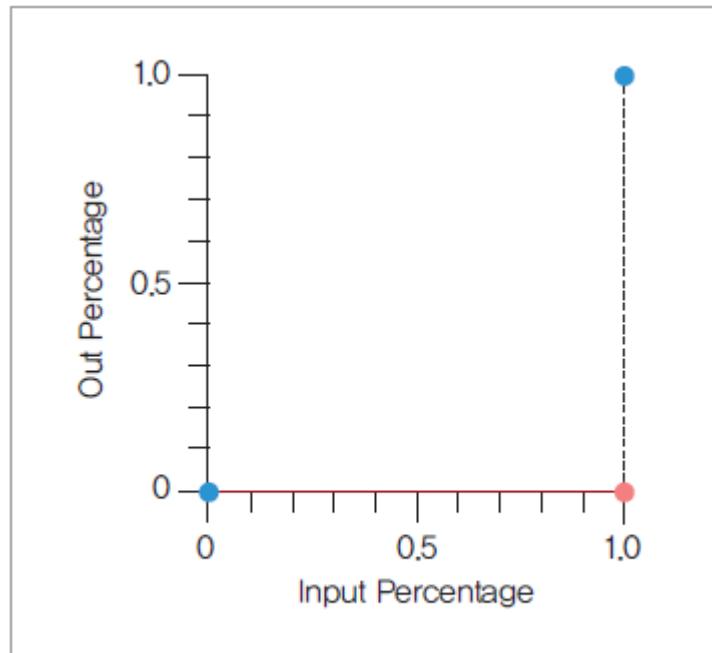


▲ ease-in-out의 속도 변화를 나타낸 그래프



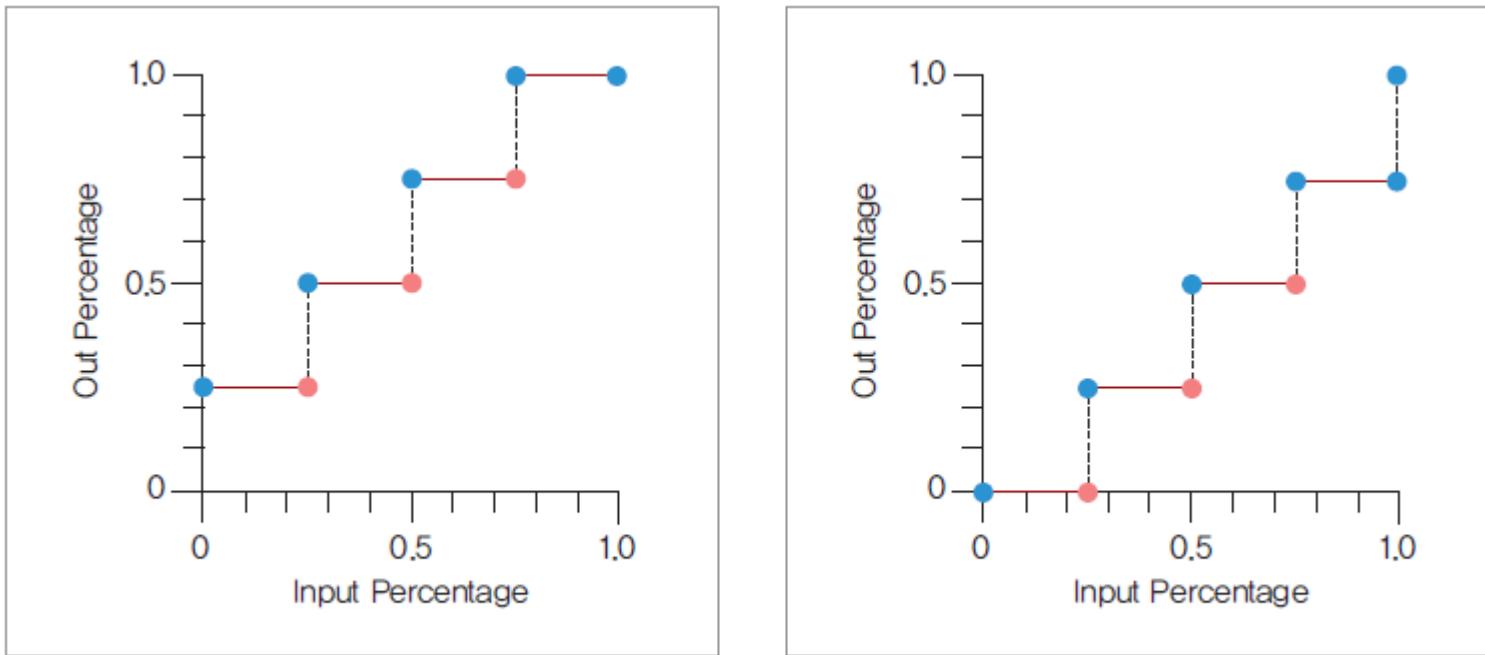
▲ step-start의 속도 변화를 나타낸 그래프

# transition-timing-function 속성 값



▲ step-start의 속도 변화를 나타낸 그래프

# transition-timing-function 속성 값



▲ 각각 step(4,start)와 step(4,end)의 속도 변화를 나타낸 그래프

# transition-delay

- 변환이 진행되기 전 지연되는 시간을 지정

속성 값	$\langle\text{time}\rangle$ [, $\langle\text{time}\rangle$ ]*
기본 값	0s
상속 여부	상속되지 않음
지원 웹 브라우저	    

# transition

- transition 관련 속성 값들을 일괄 적용하는 대표 속성

속성 값	none   <single-transition-property> ]    <transition-duration time>    <single-transition-timing-function>    <transition-delay time>
기본 값	개별 속성 값 참조
상속 여부	상속되지 않음
지원 웹 브라우저	   

# @keyframes

- animation 속성에 적용할 키프레임을 생성하기 위한 규칙.

지원 웹 브라우저



```
@keyframes 애니메이션 이름 {  
    from {  
        [ CSS 속성 : 값 ; ]  
    }  
    [percentage] {  
        [ CSS 속성 : 값 ; ]  
    }  
    to {  
        [ CSS 속성 : 값 ; ]  
    }  
}
```

# animation-name

- @keyframes 규칙으로 생성한 애니메이션 이름을 지정하여 해당 애니메이션이 실행되도록 하는 속성

속성 값	<code>&lt;single-animation-name&gt; [',' &lt;single-animation-name&gt; ]*</code>
기본 값	none
상속 여부	상속되지 않음
지원 웹 브라우저	    

# animation-duration

- `animation-name` 속성으로, 실행된 애니메이션 진행 시간을 지정

속성 값	<code>&lt;time&gt; [, &lt;time&gt;]*</code>
기본 값	0s
상속 여부	상속되지 않음
지원 웹 브라우저	    

# animation-timing-function

- 애니메이션 진행 속도의 변화 형태를 지정하는 속성으로, 앞서 살펴본 transition-timing-function 속성 값과 동일함.

속성 값	ease   linear   ease-in   ease-out   ease-in-out   step-start   step-end   steps(<integer>[, [ start   end ] ]?)   cubic-bezier(<number>, <number>, <number>, <number>)
기본 값	ease
상속 여부	상속되지 않음
지원 웹 브라우저	    

# animation-iteration-count

- 애니메이션의 반복 횟수를 지정

속성 값	infinite   <number>
기본 값	1
상속 여부	상속되지 않음
지원 웹 브라우저	    

# animation-direction

- 애니메이션의 진행 방향을 지정

속성 값	normal   reverse   alternate   alternate-reverse
기본 값	normal
상속 여부	상속되지 않음
지원 웹 브라우저	    

# animation-play-state

- 애니메이션의 진행 및 정지 상태를 지정

속성 값	running   paused
기본 값	running
상속 여부	상속되지 않음
지원 웹 브라우저	    

# animation-delay

- 애니메이션이 실행되기 전 지연 시간을 초 단위(s)로 지정

속성 값	<code>&lt;time&gt; [, &lt;time&gt;]*</code>
기본 값	0s
상속 여부	상속되지 않음
지원 웹 브라우저	    

# animation-fill-mode

- 애니메이션 실행 이전이나 이후에 애니메이션 효과를 표시할지의 여부를 지정하는 속성

속성 값	none   forwards   backwards   both
기본 값	none
상속 여부	상속되지 않음
지원 웹 브라우저	    

# animation

- 애니메이션과 관련된 속성 값을 일괄 지정하기 위한 대표 속성

속성 값	<code>&lt;single-animation-name&gt;    &lt;time&gt;    &lt;single-animation-timing-function&gt;    &lt;time&gt;    &lt;single-animation-iteration-count&gt;    &lt;single-animation-direction&gt;    &lt;single-animation-fill-mode&gt;    &lt;single-animation-play-state&gt;</code>
기본 값	개별 속성 값 참조
상속 여부	상속되지 않음
지원 웹 브라우저	    

# display : flex, display : inline-flex

- CSS3에 추가된 유연한 박스 레이아웃을 사용하려면 가장 먼저 기존에 사용했던 display 속성에 추가된 새로운 값인 flex 또는 inline-flex를 지정해야 함.
- 두 값의 차이점은 블록 수준의 유연한 박스를 생성할 것인지, 인라인 수준의 유연한 박스를 생성할 것인지를 여부임.



# flex-direction

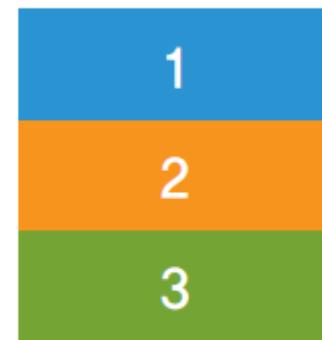
- 요소 박스의 배치 방향을 지정

속성 값	row   row-reverse   column   column-reverse
기본 값	row
상속 여부	상속되지 않음
지원 웹 브라우저	   

row



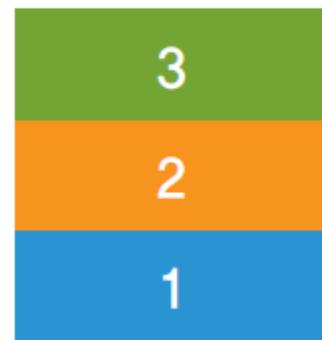
column



row-reverse



column-reverse



# flex-wrap

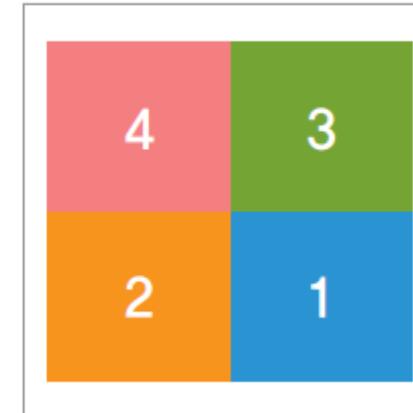
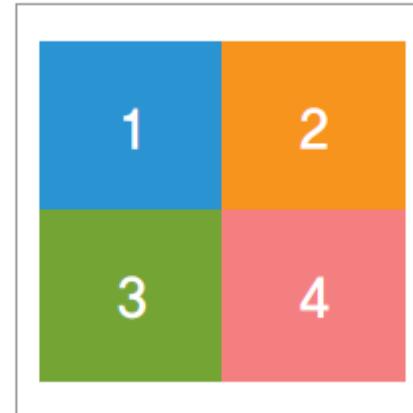
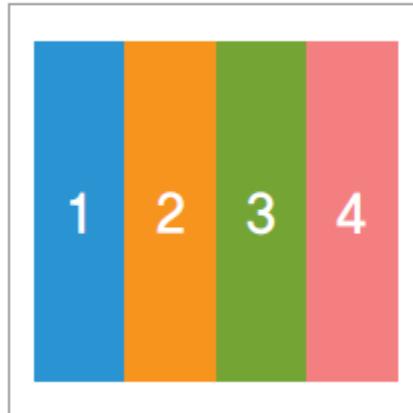
- flex로 지정된 요소에 포함된 자식 요소 박스가 한 줄 또는 여러 줄로 배치될 것인지를 지정.

속성 값	nowrap   wrap   wrap-reverse
기본 값	nowrap
상속 여부	상속되지 않음
지원 웹 브라우저	  

none

wrap

wrap-reverse



# flex-flow

- flex-direction 속성과 flex-flow 속성을 일괄적으로 지정할 수 있는 대표 속성

속성 값	⟨'flex-direction'⟩    ⟨'flex-wrap'⟩
기본 값	세부 속성 참조
상속 여부	상속되지 않음
지원 웹 브라우저	  

# order

- 요소 박스의 배치 순서를 지정

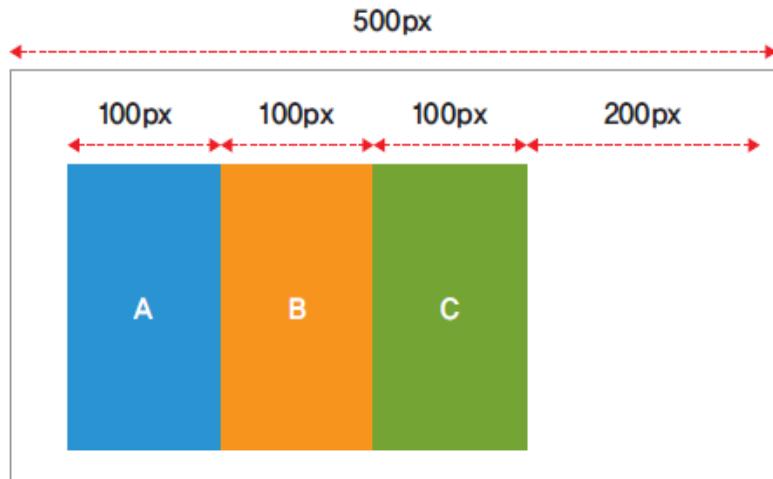
속성 값	<code>⟨integer⟩</code>
기본 값	0
상속 여부	상속되지 않음
지원 웹 브라우저	  

# flex-grow

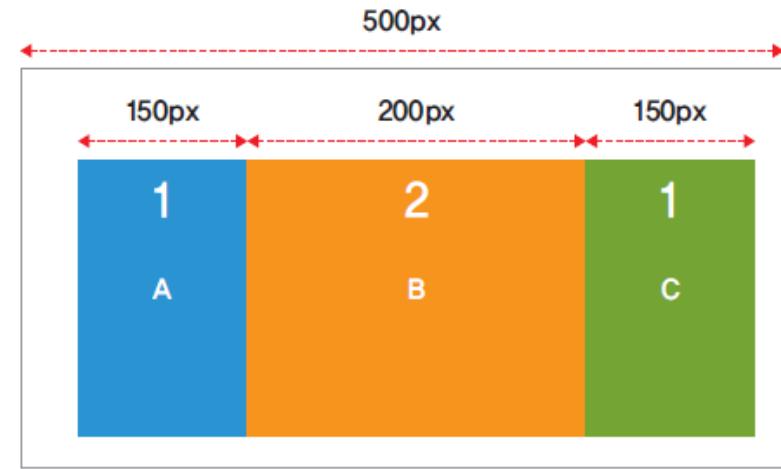
- flex로 정의된 부모 요소가 자식 요소보다 크거나 작을 경우, 자식 요소의 크기를 조정하는 방법으로 자식 요소에 확대 비율을 지정하여 부모 요소의 크기에 맞게 자동으로 크기를 조정

속성 값	<code>&lt;number&gt;</code>
기본 값	0
상속 여부	상속되지 않음
지원 웹 브라우저	  

# flex-grow



▲ `flex-grow` 속성을 지정하지 않은 경우

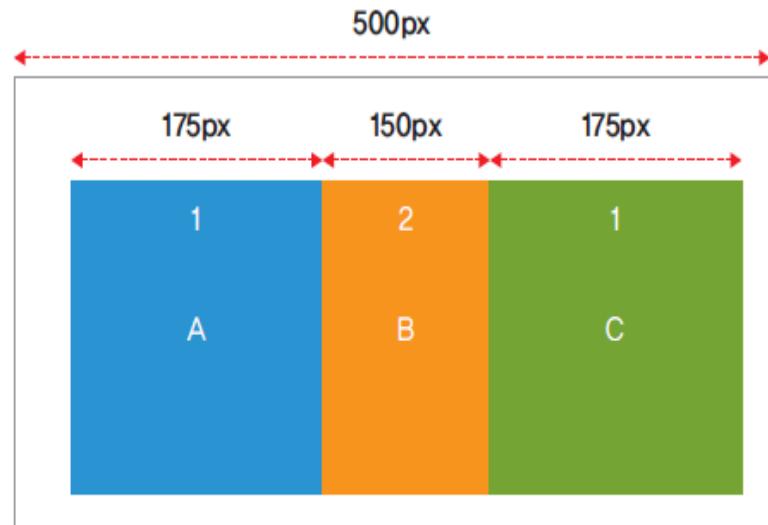


▲ `flex-grow` 속성을 지정한 경우

# flex-shrink

- flex-grow 속성과 마찬가지로 flex로 정의된 부모 요소가 자식 요소보다 크거나 작을 경우 자식 요소의 크기를 조정하는 방법으로, 자식 요소에 축소 비율을 지정하여 부모 요소의 크기에 맞게 자동으로 조정

속성 값	<code>&lt;number&gt;</code>
기본 값	1
상속 여부	상속되지 않음
지원 웹 브라우저	  



▲ flex-shrink 속성을 지정한 경우

# flex-basis

- 유연하게 조절되는 박스의 기준 크기를 지정하는 속성으로, width 속성에서 사용할 수 있는 값과 동일함.

속성 값	<'width'>
기본 값	auto
상속 여부	상속되지 않음
지원 웹 브라우저	  

# flex

- flex-grow 속성과 flex-shrink 속성 그리고 flex-basis 속성을 일괄적으로 지정할 수 있는 대표 속성

속성 값	none   [ <'flex-grow'> <'flex-shrink'>?    <'flex-basis'> ]
기본 값	세부 속성 참조
상속 여부	상속되지 않음
지원 웹 브라우저	   

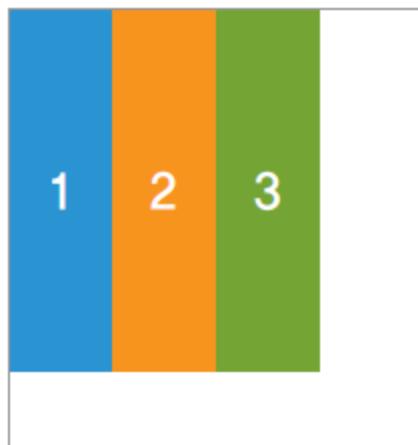
# justify-content

- flex로 지정된 부모 요소 내에서 자식 요소의 정렬 방식을 지정하기 위한 속성.
- 해당 정렬은 flex-direction 속성 값이 행 방향이냐, 열 방향이냐에 따라 start와 end 영역이 결정됨.

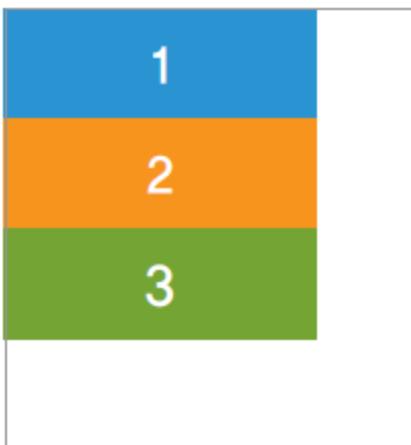
속성 값	flex-start   flex-end   center   space-between   space-around
기본 값	flex-start
상속 여부	상속되지 않음
지원 웹 브라우저	  

# justify-content

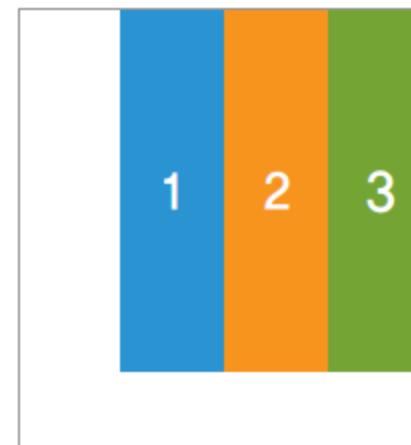
row



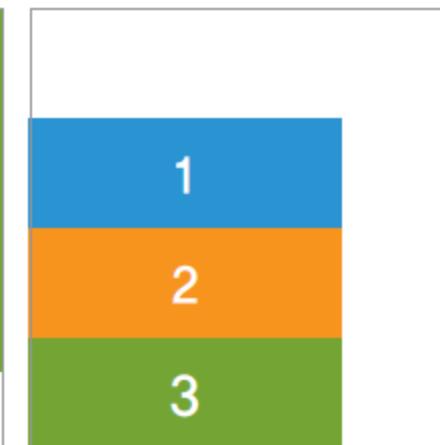
column



row



column



▲ flex-start로 지정된 경우

▲ flex-end로 지정된 경우

# justify-content

row

column

row

column

1 2 3

1  
2  
3

1 2 3

1  
2  
3

▲ center로 지정된 경우

▲ space-between으로 지정된 경우

row

column

1 2 3

1  
2  
3

▲ space-around로 지정된 경우

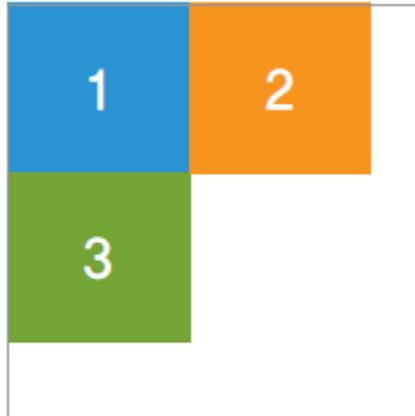
# align-content

- flex로 지정된 부모 요소 내에서 여러 개의 행이나 열로 구성된 요소 박스의 정렬을 지정하기 위한 속성

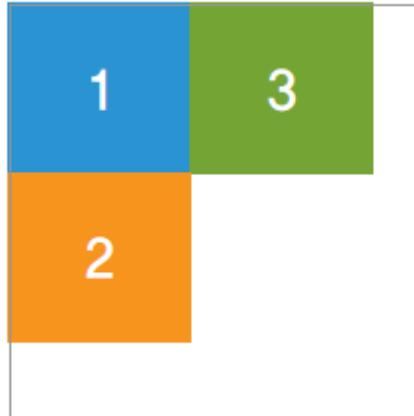
속성 값	flex-start   flex-end   center   space-between   space-around   stretch
기본 값	stretch
상속 여부	상속되지 않음
지원 웹 브라우저	 

# align-content

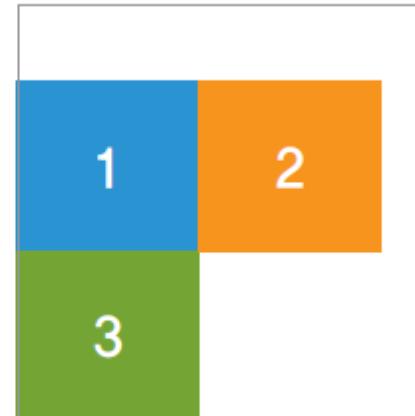
row wrap



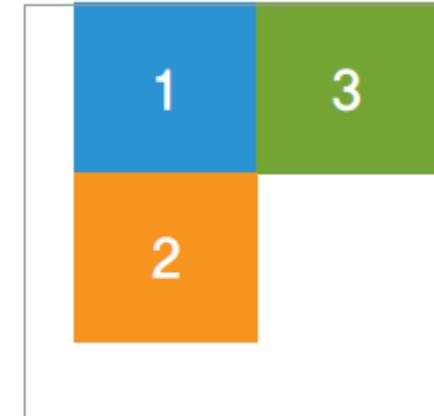
column wrap



row wrap



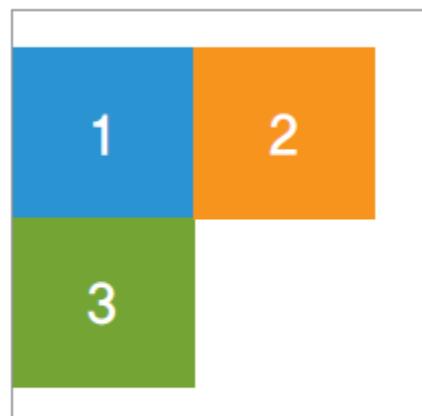
column wrap



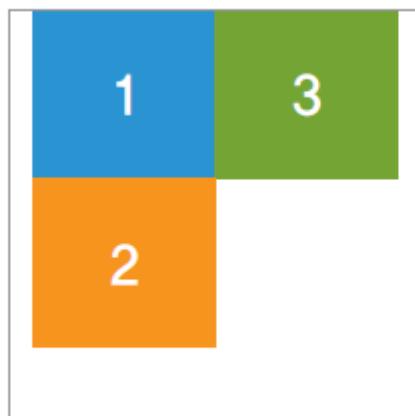
▲ flex-start로 지정된 경우

▲ flex-end로 지정된 경우

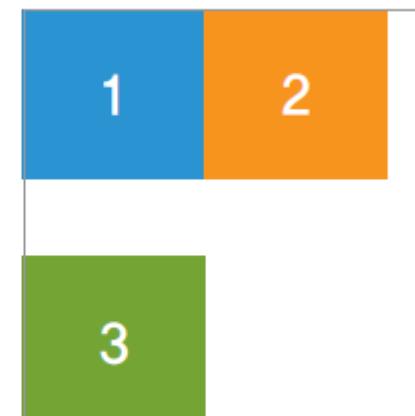
row wrap



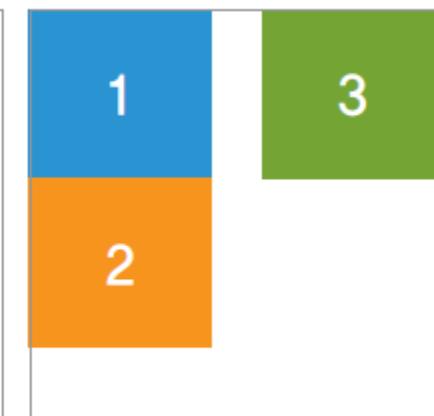
column wrap



row wrap



column wrap

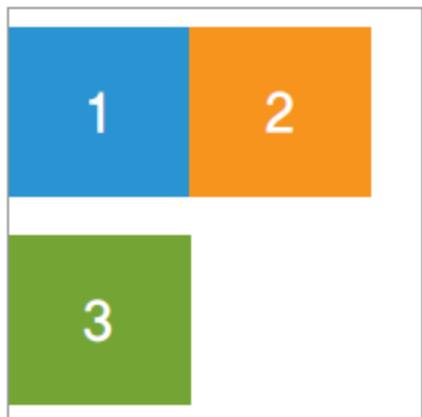


▲ center로 지정된 경우

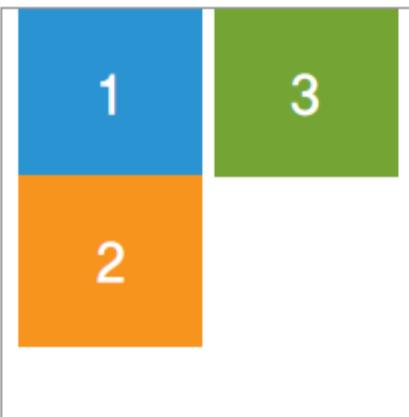
▲ space-between으로 지정된 경우

# align-content

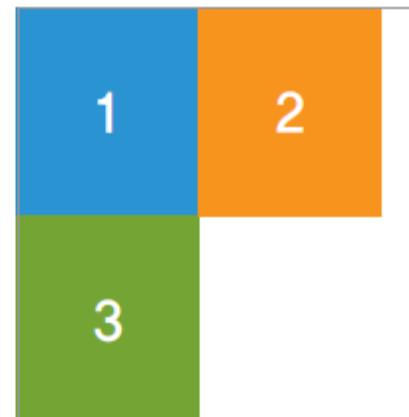
row wrap



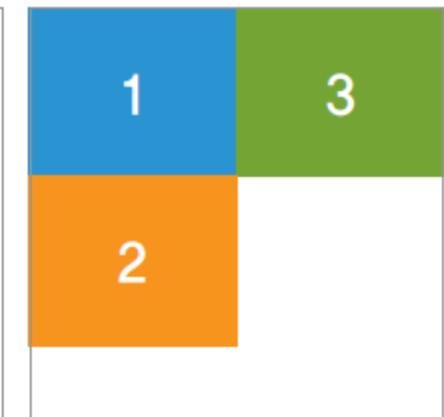
column wrap



row wrap



column wrap



▲ space-around로 지정된 경우

▲ stretch로 지정된 경우

# align-items, align-self

- 라인 박스의 높이에 따라 배치되는 요소 박스에 배치에 영향을 주는 교차 축 (cross axis)을 지정하는 속성

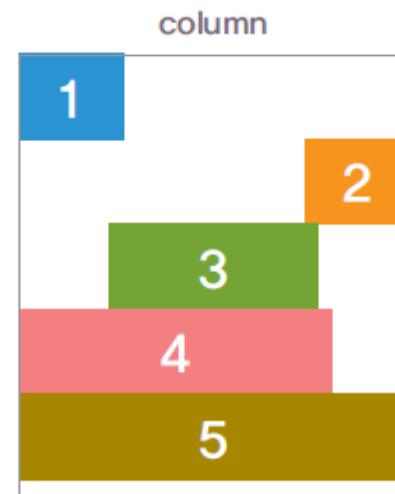
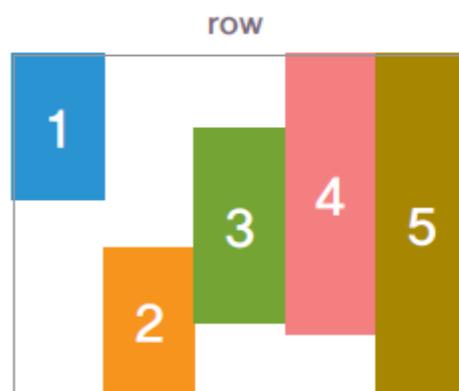
## align-items

속성 값	flex-start   flex-end   center   baseline   stretch
기본 값	stretch
상속 여부	상속되지 않음
지원 웹 브라우저	  

# align-items, align-self

## align-self

속성 값	auto   flex-start   flex-end   center   baseline   stretch
기본 값	auto
상속 여부	상속되지 않음
지원 웹 브라우저	   



▲ align-items, align-self 속성 값에 따른 요소 박스의 배치  
(① flex-start, ② flex-end, ③ center, ④ baseline, ⑤ stretch)

# content

- 가상 요소 선택자인 :before나 :after를 선택자로 지정한 후, 특정 요소 앞이나 뒤에 콘텐츠를 추가할 수 있도록 하는 기능

속성 값	normal   none   [ <string>   <uri>   <counter>   attr(<identifier>) ]+   open-quote   close-quote   no-open-quote   no-close-quote ]+   inherit
기본 값	세부 속성 참조(none)
상속 여부	상속되지 않음
지원 웹 브라우저	    

# box-sizing

- 요소 박스의 크기를 결정하는 방식을 지정

속성 값	content-box   padding-box   border-box   inherit
기본 값	content-box
상속 여부	상속되지 않음
지원 웹 브라우저	    

# outline

- outline-width, outline-style, outline-color 속성을 일괄적으로 지정하는 대표 속성으로 border 속성과는 별개로 요소의 외곽선을 표시

속성 값	[ <'outline-color'>    <'outline-style'>    <'outline-width'> ]   inherit
기본 값	medium
상속 여부	상속되지 않음
지원 웹 브라우저	    

# outline-offset

- 요소의 바깥쪽 영역과 외곽선 사이의 간격을 지정하는 속성.

속성 값	⟨length⟩   inherit
기본 값	0
상속 여부	상속되지 않음
지원 웹 브라우저	   

# resize

- 사용자에 의해 요소 박스의 크기를 조정할 것인지의 여부를 지정할 수 있는 속성

속성 값	none   both   horizontal   vertical   inherit
기본 값	none
상속 여부	상속되지 않음
지원 웹 브라우저	   

# text-overflow

- 요소 박스의 너비가 고정되어 있고, 요소 박스 내 텍스트가 박스의 크기를 넘는 경우 overflow된 텍스트를 처리하는 방식을 지정하는 속성.

속성 값	clip   ellipsis   <string> ){1,2}   inherit
기본 값	clip
상속 여부	상속되지 않음
지원 웹 브라우저	   

text-overflow는 요소 박스의

text-overflow는 요소 …

▲ text-overflow 속성의 적용 예

# cursor

- HTML 요소 위에 마우스가 오버될 때 마우스 포인터의 모양을 지정하는 속성.

속성 값	<pre>[[&lt;uri&gt; [&lt;x&gt; &lt;y&gt;]?]* [ auto   default   none   context-menu   help   pointer   progress   wait   cell   crosshair   text   vertical-text   alias   copy   move   no-drop   not-allowed   e-resize   n-resize   ne-resize   nw-resize   s-resize   se-resize   sw-resize   w-resize   ew-resize   ns-resize   nesw-resize   nwse-resize   col-resize   row-resize   all-scroll   zoom-in   zoom-out ] ]   inherit</pre>
기본 값	auto
상속 여부	상속되지 않음
지원 웹 브라우저	    

# Media Query

- Media Query는 CSS3에서 새롭게 추가된 속성이라기보다는 CSS2.1의 미디어 기능이 확장된 것으로 볼 수 있음.
- HTML4.01과 CSS2.1에서는 "aural(청각)", "braille(점자)", "handheld(휴대 기기)", "print(인쇄)", "projection(프로젝터)", "screen(화면)", "tty(전신 인자기)", "tv(텔레비전)", "embossed(양각)", "speech(음성)", "all(모든 미디어)"의 미디어 타입을 지정함으로써 사용자 환경에 따라 CSS를 선택적으로 적용할 수 있도록 정의하고 있음.

지원 웹 브라우저



# Media Query

- <link> 요소의 media 속성을 사용하여 미디어 타입에 따라 선택적으로 CSS를 적용하는 방법

```
<link rel="stylesheet" type="text/css" media="screen" href="screen.css">
<link rel="stylesheet" type="text/css" media="print" href="print.css">
```

# Media Query

- <style> 요소의 media 속성을 사용하여 미디어 타입에 따라 선택적으로 CSS를 적용하는 방법

```
<style type="text/css" media="screen">
    /* 미디어 타입이 screen일 때 적용되는 CSS */
    h1 {
        color : #ff0000 ;
    }
    p {
        font-size :1.2em ;
    }
</style>
<style type="text/css" media="print">
    /* 미디어 타입이 print 때 적용되는 CSS */
    h1, p {
        color : #000000 ;
        font-size : 1em ;
    }
</style>
```

# Media Query

- @media 규칙을 사용하여 미디어 타입에 따라 선택적으로 CSS를 적용하는 방법

```
/* 모든 매체에 공통적으로 적용되는 CSS */
h1 {
    color: #ff0000;
}
p {
    font-size: 1.2em;
}
@media print {
    /* 미디어 타입이 print일 때 적용되는 CSS */
    h1, p {
        color: #000000;
        font-size: 1em;
    }
}
```

## Media Query에서 매체의 해상도나 특징을 구분할 수 있는 방법

- width : 출력 매체의 너비의 따라 정의
- height : 출력 매체의 높이의 따라 정의
- device-width : 출력 장치의 디스플레이 너비에 따라 정의
- device-height : 출력 장치의 디스플레이 높이에 따라 정의
- orientation : 화면의 가로 세로 방향에 따라 미디어의 유형을 결정
- device-aspect-ratio : 출력 장치 디스플레이의 가로, 세로 비율에 따라 미디어 유형을 정의
- color : 색상 표시가 가능한 매체의 색상 깊이 값으로 정의
- color-index : 매체가 표시할 수 있는 색상의 개수로 정의
- monochrome : 흑백 매체의 음영 깊이 값으로 정의
- resolution : 매체의 출력 해상도의 값으로 정의
- scan : 화면의 주사 방식에 따라 정의
- grid : 점자 출력기와 같은 물리적 격자 형태의 출력 형태의 유형을 정의

# Media Query의 조건식

- not / only :  
not과 only는 미디어 쿼리의 조건을 제한하거나 부정

```
<!--link 요소의 사용 -->
<link rel="stylesheet" media="not screen" href="style.css">
<link rel="stylesheet" media="only screen" href="style.css">

<!--style 요소의 사용 -->
<style type="text/css" media="not screen">
    /* CSS 코드 */
</style>
<style type="text/css" media="only screen">
    /* CSS 코드 */
</style>

/* @media 규칙을 사용 */
@media not screen{
    /* CSS 코드 */
}
@media only screen{
    /* CSS 코드 */
}
```

## Media Query에서 매체의 해상도나 특징을 구분할 수 있는 방법

- and / ", "  
미디어 쿼리의 조건을 모두 만족할 때를 의미하며 콤마(,)로 적용할 매체에 따른 조건식을 명시할 수도 있습니다

```
<!--link 요소의 사용 -->
<link rel="stylesheet" media="screen and (min-width:320px)
and (max-width:800px)" href="style.css">

<!--style 요소의 사용 -->
<style type="text/css" media="screen and (min-width:320px)
and (max-width:800px)">
    /* CSS 코드 */
</style>

/* @media 규칙을 사용 */
@media screen and (min-width:320px) and (max-width:800px) {
    /* CSS 코드 */
}
```