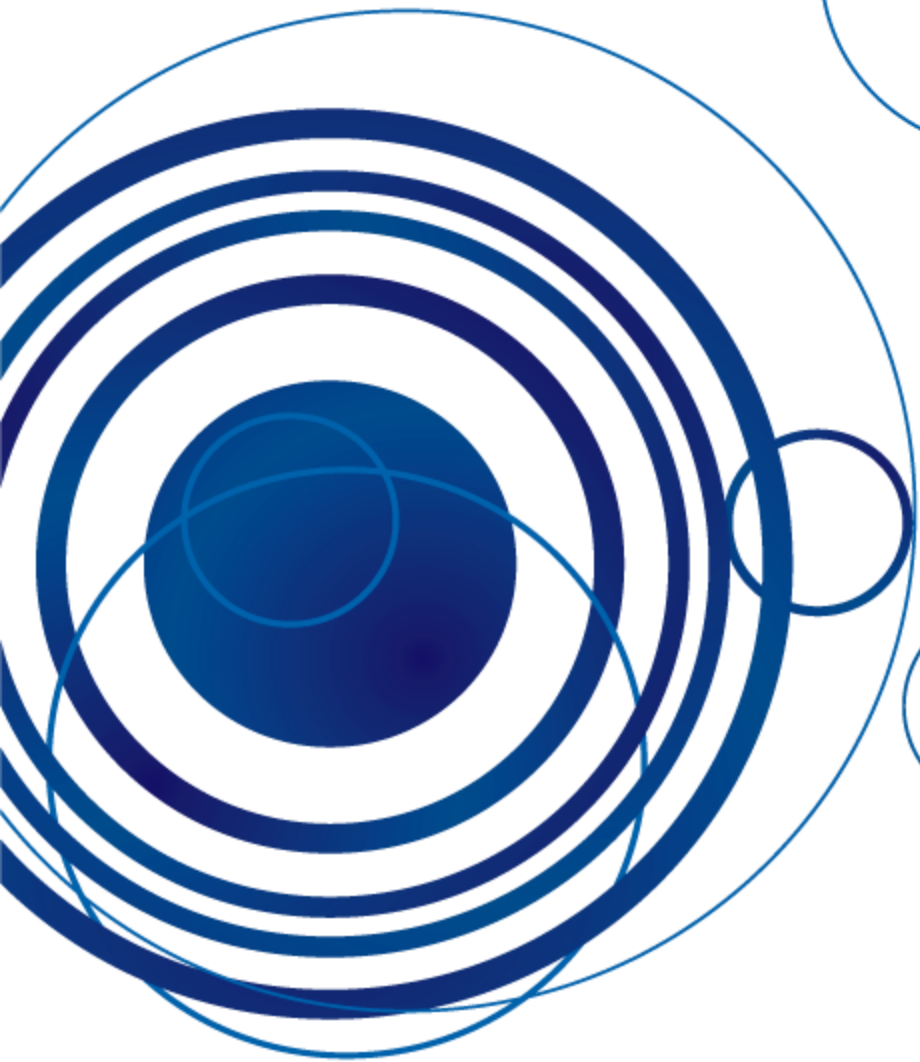


LOGO



공학설계입문

10주차: CSS 스타일시트
강의 : 신경섭

11101001110000111110101110010101010011001010011010111101001110000111110101110010101010011001010011010111101001110
111001100000110111010010111010111110101010100101011110011000001101110100101110101111101010101001010111100110000
11010011100110101010100101101111010100110101001010111010011100110101010100101101111010100110101001010111010011100





CSS의 개념

- 문서의 구조-> HTML
- 문서의 스타일 -> ?



문서의
스타일은
내가
책임집니다.



CSS3



문서의
내용은 내가
책임집니다.



HTML5



문서의
동작은 내가
책임집니다.



JavaScript





CSS의 역할

- 만약 CSS가 없다면

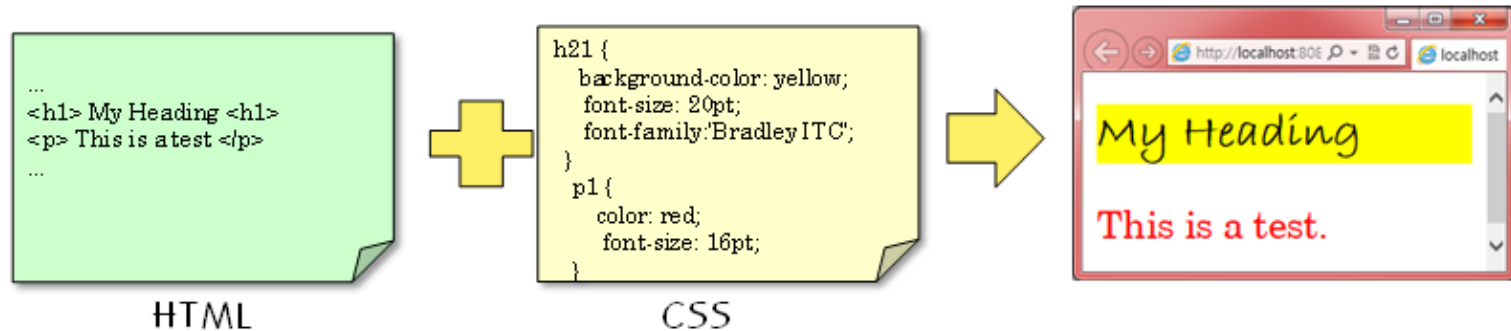
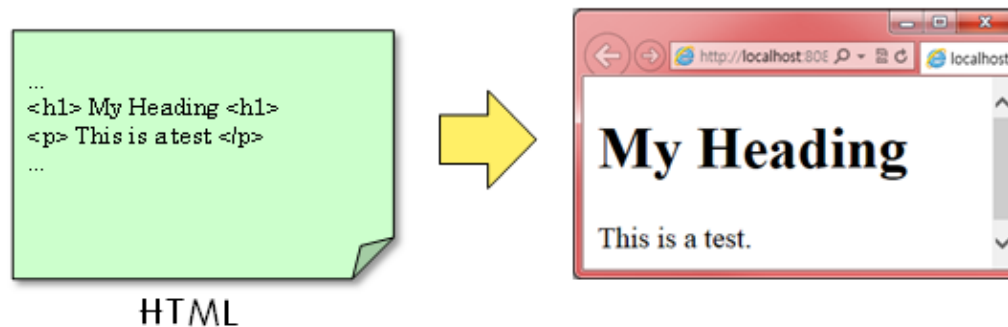
문서의 스타일 삭제





CSS

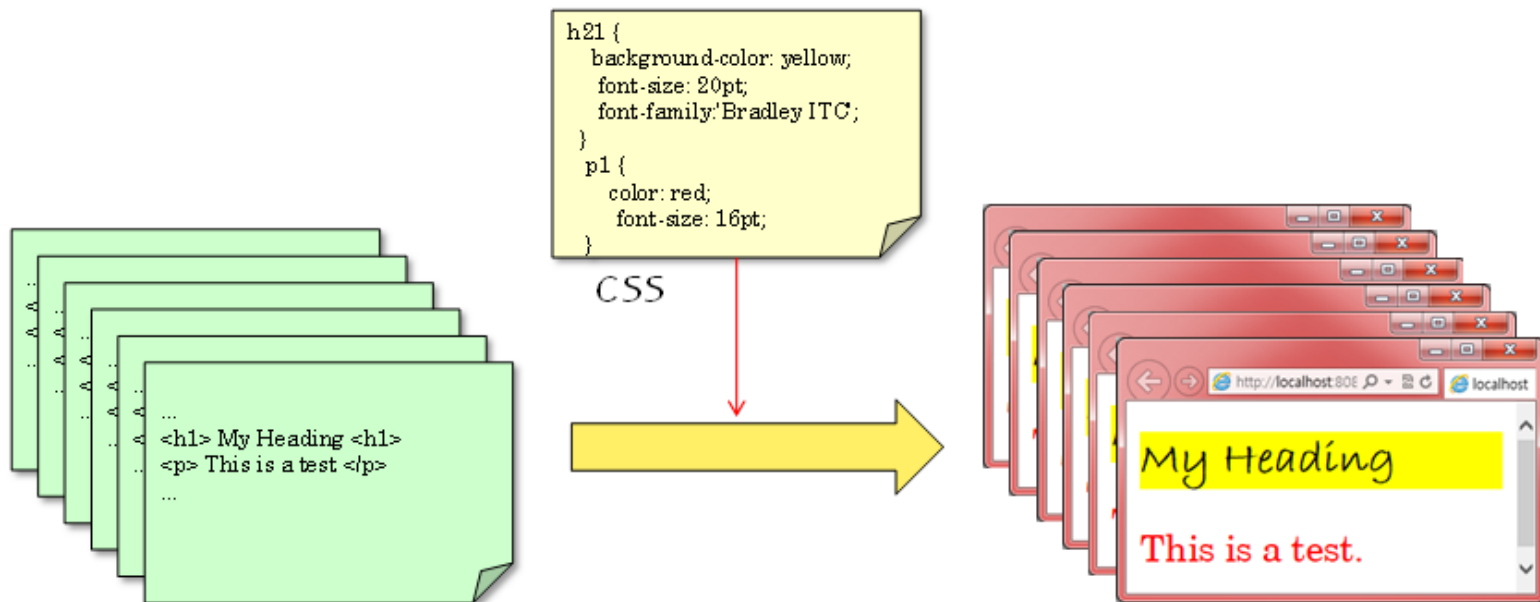
- CSS(Cascading Style Sheets): 문서의 스타일을 지정한다.





CSS의 장점

- 거대하고 복잡한 사이트를 관리할 때에 필요
- 모든 페이지들이 동일한 CSS를 공유
- CSS에서 어떤 요소의 스타일을 변경하면 관련되는 전체 페이지의 내용이 한꺼번에 변경





CSS3의 기능

- 선택자(selectors)
- 박스 모델(Box Model)
- 배경 및 경계선(Backgrounds and Borders)
- 텍스트 효과(Text Effects)
- 2차원 및 3차원 변환(2D/3D Transformations)
- 애니메이션(Animations)
- 다중 컬럼 레이아웃(Multiple Column Layout)
- 사용자 인터페이스(User Interface)





CSS3의 문법

- 선택자(selector) { 속성: 값; }
- 끝에 반드시 ;을 적어 준다.

```
p { background-color: yellow; }
```

↑ ↑ ↑
선택자(selector) 속성(property) 값(value)

- 주석: /* ... */

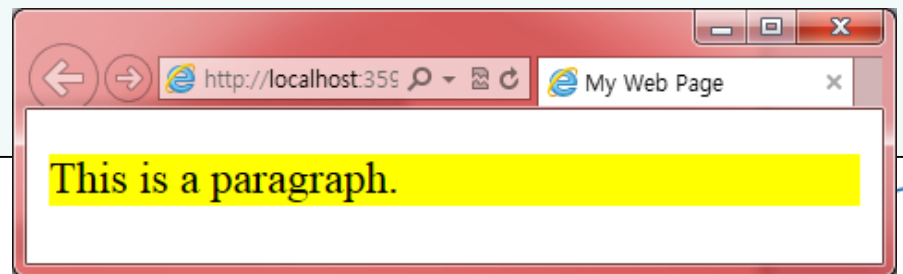


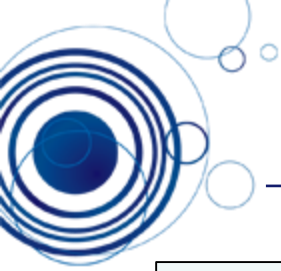


CSS의 위치

```
<!doctype html>
<html>
<head>

  <title>My Web Page</title>
  <style>
    p { background-color: yellow; }
  </style>
</head>
<body>
  <p>This is a paragraph.</p>
</body>
</html>
```





예제

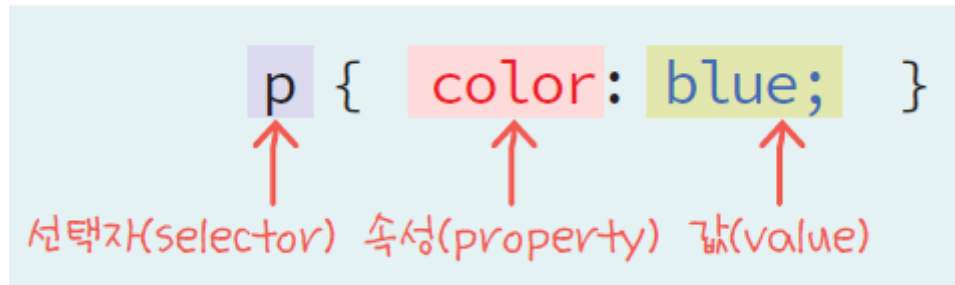
```
<!DOCTYPE html>
<html>
<head>
  <title>My Web Page</title>
  <style>
    h1 {
      background-color: yellow;
      border: 2px solid red;
    }
  </style>
</head>
<body>
  <h1>This is a heading.</h1>
</body>
</html>
```





선택자

- 선택자(selector): HTML 요소를 선택하는 부분



- 선택자는 jQuery에서도 사용
- 가장 많이 사용되는 것은 6가지 정도
- 선택자에 대한 W3C의 문서는
<http://www.w3.org/TR/css3-selectors/>





선택자의 종류

- 타입 선택자(type selector)
- 전체 선택자(universal selector)
- 클래스 선택자(class selector)
- 아이디 선택자(ID selector)
- 속성 선택자(attribute selector)
- 의사 선택자(pseudo-class)





타입 선택자

- 타입 선택자(type selector) : HTML 요소 이름을 사용

`h1` { *`color: green;`* }

모든 *h1* 요소를 선택한다.





전체 선택자

- 전체 선택자(universal selector): 페이지 안의 모든 요소를 선택

 ** { color: green; }*

전체 요소를 선택한다.





아이디 선택자

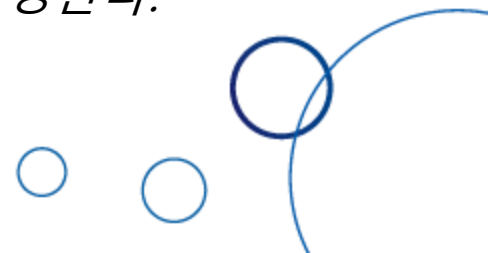
- 아이디 선택자(id selector): 특정한 요소를 선택

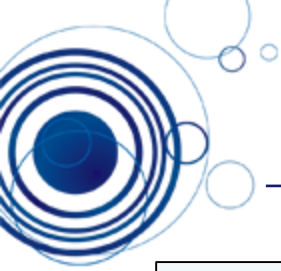
#target { color: red; }

id가 target인 요소를 선택한다.

<p id="target">Hello World!</p>

<p>요소의 id를 "target"로 지정한다.

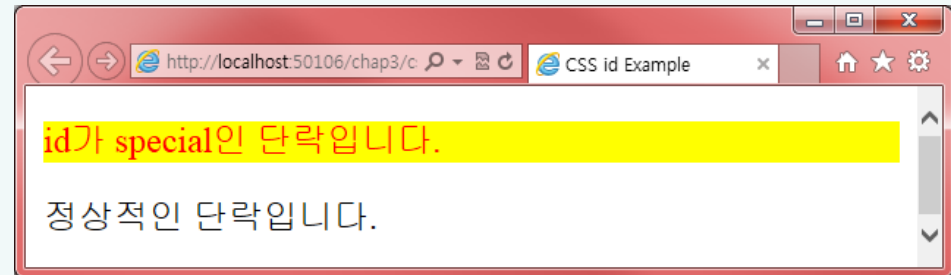




예제

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS id Example</title>
  <style>
    #special {
      background-color: yellow;
      color: red;
    }
  </style>
</head>

<body>
  <p id="special">id가 special인 단락입니다.</p>
  <p>정상적인 단락입니다.</p>
</body>
</html>
```





클래스 선택자

- 클래스 선택자(class selector)는 클래스가 부여된 요소를 선택한다.

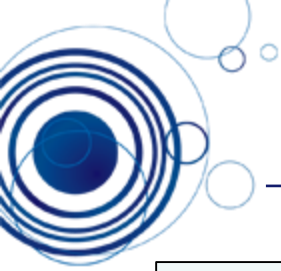
.target { color: red; }

클래스가 *target*인 요소를 선택한다.

<p class="target">Hello World!</p>

<p> 요소의 클래스를 "target"로 지정한다.

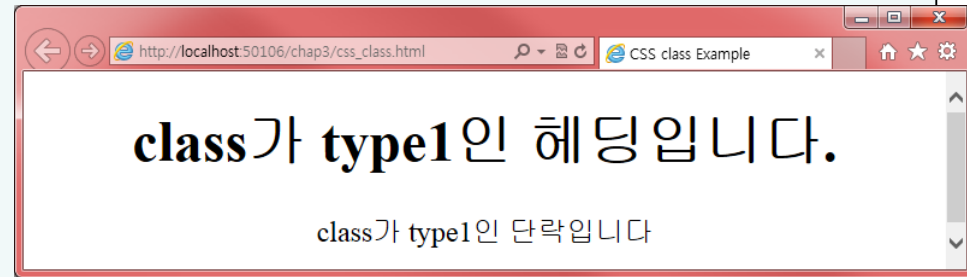




예제

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS class Example</title>
  <style>
    .type1 {
      text-align: center;
    }
  </style>
</head>

<body>
  <h1 class="type1">class가 type1인 헤딩입니다.</h1>
  <p class="type1">class가 type1인 단락입니다</p>
</body>
</html>
```





선택자 그룹

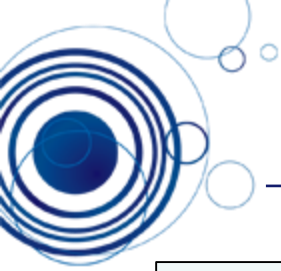
- 선택자를 콤마(,)로 분리하여 나열할 수 있다.

$h1, h2, h3$ { font-family: sans-serif; }

<h1>, <h2>, <h3>요소를 선택한다.

i slept like a top

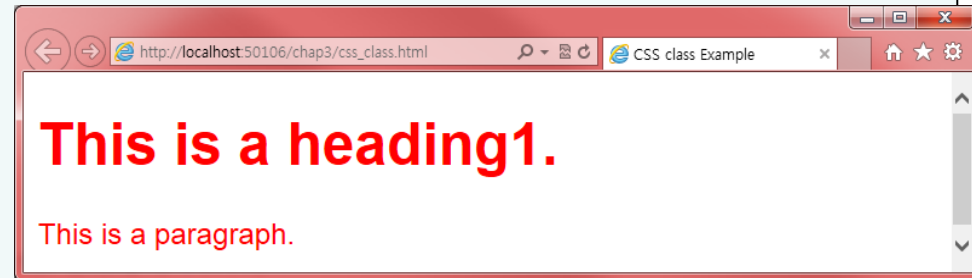




예제

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS selector Example</title>
  <style>
    h1, p {
      font-family: sans-serif;
      color: red;
    }
  </style>
</head>

<body>
  <h1>This is a heading1.</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```



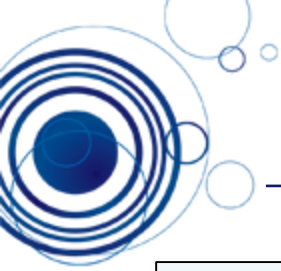


자손, 자식, 형제 결합자

선택자	설명
<code>s1 s2</code>	s1 요소에 포함된 s2 요소를 선택한다. (후손 관계)
<code>s1 > s2</code>	s1 요소의 직계 자식 요소인 s2를 선택한다.(자식 관계)

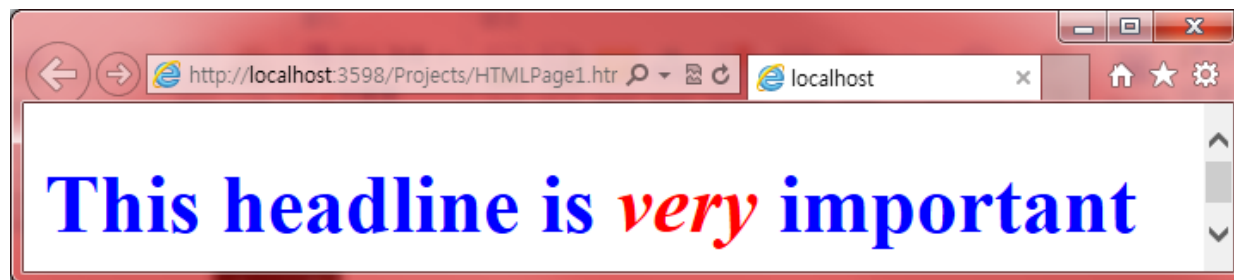
- `body em { color:red; }` `/* body 안의 em 요소 */`
- `body > h1 { color:blue; }` `/* body 안의 h1 요소 */`





예제

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body em { color: red; } /* body 안의 em 요소 */
    body > h1 { color: blue; } /* body 안의 h1 요소 */
  </style>
</head>
<body>
  <h1>This headline is <em>very</em> important</h1>
</body>
</html>
```

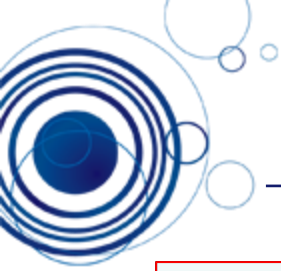




의사 클래스

- 의사 클래스(pseudo-class): 클래스가 정의된 것처럼 간주
 - a:link { color: blue; }
 - a:visited { color: green; }
 - a:hover { color: red; }





예제

```
a:link {  
  text-decoration: none;  
  color: blue;  
  background-color: white;  
}
```

```
a:visited {  
  text-decoration: none;  
  color: green;  
  background-color: silver;  
}
```

```
a:hover {  
  text-decoration: none;  
  color: white;  
  background-color: blue;  
}
```

...

겨울 왕국

눈의 여왕

라푼젤

라따뚜이





속성 선택자

- 속성 선택자: 특정한 속성을 가지는 요소를 선택한다.
 - `h1[title] { color: blue; }`
 - `p[class="example"] { color: blue; }`





CSS 삽입 위치

- 외부 스타일 시트(external style sheet)
- 내부 스타일 시트(internal style sheet)
- 인라인 스타일 시트(inline)

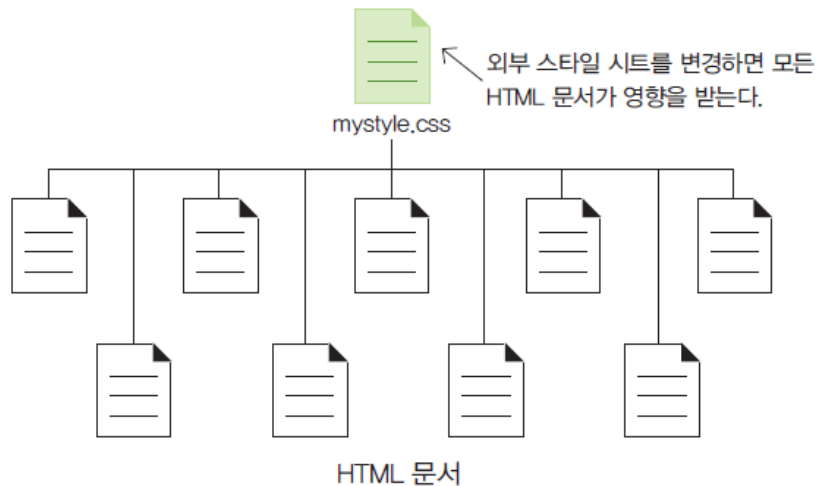




외부 스타일 시트

- 외부 스타일 시트는 스타일 시트를 외부에 파일로 저장하는 것
- 많은 페이지에 동일한 스타일을 적용하려고 할 때 좋은 방법

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



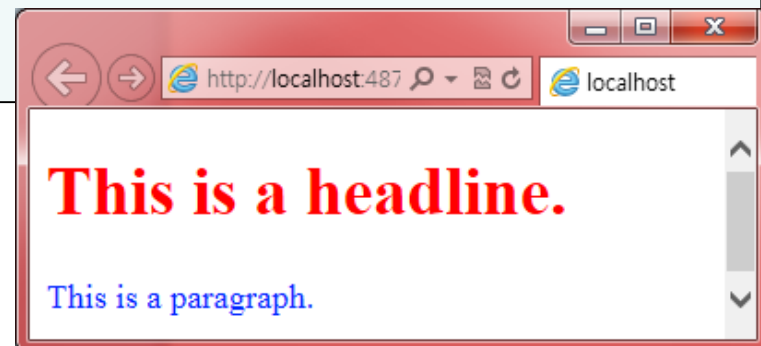


예제

mystyle.css

```
h1 { color: red; }  
p { color: #0026ff; }
```

```
<!DOCTYPE html>  
<html>  
<head>  
  <link type="text/css" rel="stylesheet" href="mystyle.css">  
</head>  
<body>  
  <h1>This is a headline.</h1>  
  <p>This is a paragraph.</p>  
</body>  
</html>
```

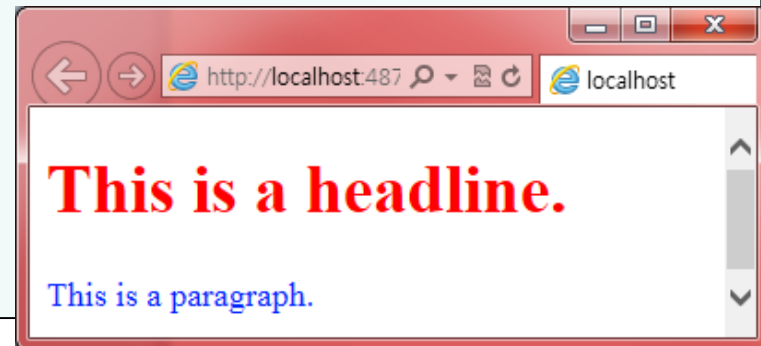




내부 스타일 시트

- 내부 스타일 시트는 HTML 안에 CSS를 정의하는 것이다.

```
<!DOCTYPE html>
<html>
<head>
  <style>
    h1 {      color: red;      }
    p {      color: #0026ff;    }
  </style>
</head>
<body>
  <h1>This is a headline.</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```





인라인 스타일 시트

- 각각의 HTML 요소마다 스타일을 지정하는 것
- 2개 이상의 선언이 있다면 반드시 끝에 ;을 적어 준다.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
</head>
```

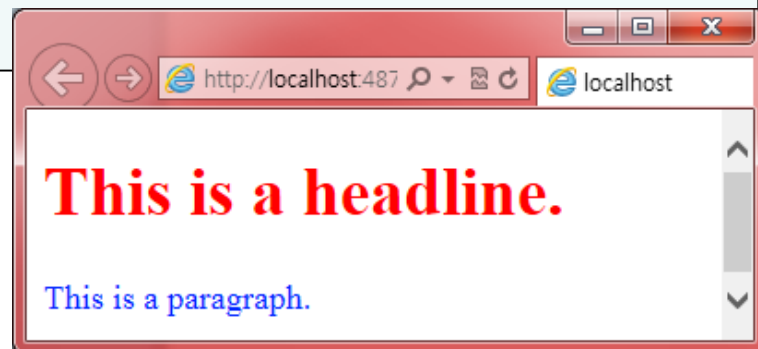
```
<body>
```

```
<h1 style="color: red">This is a headline.</h1>
```

```
<p style="color: #0026ff">This is a paragraph.</p>
```

```
</body>
```

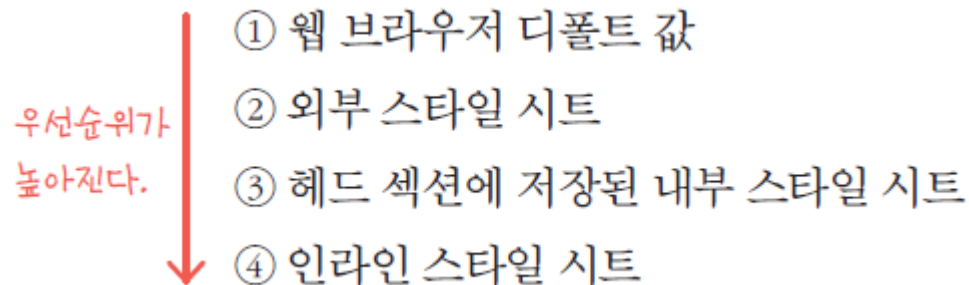
```
</html>
```



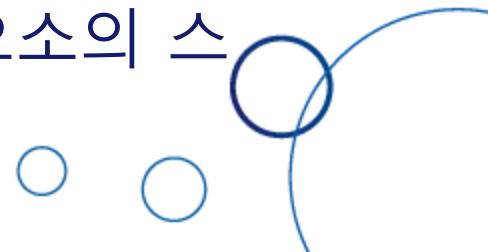


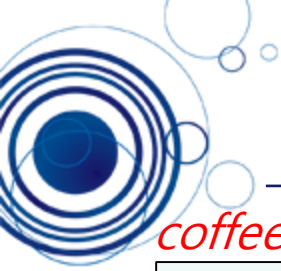
다중 스타일 시트

- 하나의 요소에 대하여 외부, 내부, 인라인 스타일이 서로 다르게 지정하고 있다면 어떤 스타일이 사용될까?



- 공통적으로 사용되는 스타일은 <body> 요소의 스타일에 정의하는 것이 편리하다.





예제

coffee.css

```
h1, p {  
    font-family: serif;  
    color:      black;  
}  
h1 {  
    border-bottom: 1px solid gray;  
    color:        red;  
}  
body {  
    background-color: yellow;  
}
```



```
<!DOCTYPE html>
<html>
<head>
  <title>Web Programming</title>
  <link type="text/css" rel="stylesheet" href="coffee.css">
</head>
<body>
  <h1>Welcome to Web Coffee!</h1>
  
  <p>
    하우스 로스팅 원두의 신선한 커피를 맛보세요!
    <em>공인 1급 Barista</em>가
    최고급 원두만을 직접 엄선하여 사용합니다.
  </p>
  <h2>메뉴</h2>
  <p>
    아메리카노,카페라떼,카푸치노,카페모카, ...
  </p>
</body>
</html>
```




CSS의 속성들

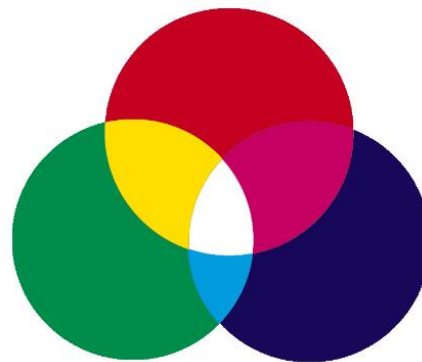
특성	설명
color	텍스트 색상
font-weight	볼드체 설정
padding	요소의 가장자리와 내용간의 간격
font-size	폰트의 크기
background-color	배경색
border	요소를 감싸는 경계선
font-style	이탤릭체 설정
background-image	배경 이미지
text-align	텍스트 정렬
list-style	리스트 스타일





색상

방법	설명
이름으로 표현	“red“
16진수로 표현	#FF0000
10진수로 표현	rgb(255, 0, 0)
퍼센트로 표현	rgb(100%, 0%, 0%)





16진수로 색상 나타내기

- 16진수 코드는 빨간색, 녹색, 청색 값을 각각 2자리의 16진수로 표시한 것

16진수 코드는 항상
#으로 시작한다.

ff 00 00
red green blue

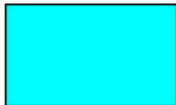















빨간색 성분이 ff, 녹색 성분이 00,
청색 성분이 00이므로 이것을 합친
색상은 빨간색이다.

```
body
{
  background-color: #ffd800;
}
```





색상의 이름으로 나타내기

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

```
body
{
  background-color: aqua;
}
```





RGB 값으로 표시하기

```
body
{
  background-color: rgb(60%, 40%, 10%);
}
```

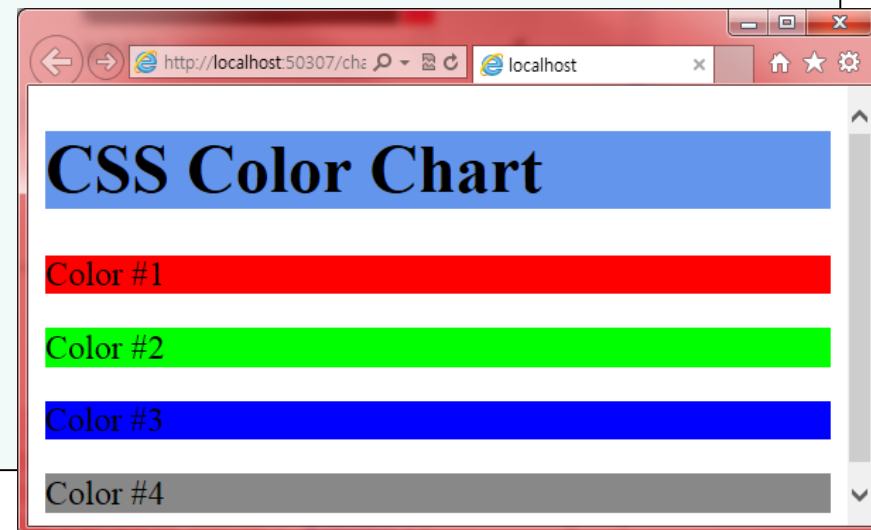
```
body
{
  background-color: rgb(153, 102, 25);
}
```



예제

```
<!DOCTYPE html>
<html>
<head>
  <style>
    h1 { background-color: #6495ed;    }
    p.a { background-color: #ff0000;   }
    p.b { background-color: #00ff00;   }
    p.c { background-color: #0000ff;   }
    p.d { background-color: #888888;   }
  </style>
</head>

<body>
  <h1>CSS Color Chart</h1>
  <p class="a">Color #1</p>
  <p class="b">Color #2</p>
  <p class="c">Color #3</p>
  <p class="d">Color #4</p>
</body>
</html>
```





폰트

속성	설명
<u>font</u>	한줄에서 모든 폰트 속성을 설정할 때 사용
<u>font-family</u>	폰트 패밀리 설정
<u>font-size</u>	폰트의 크기 설정
<u>font-style</u>	폰트 스타일 설정
<u>font-weight</u>	폰트의 볼드체 여부 설정



폰트 지정

```
body {  
  font-family: "Times New Roman", Times, serif;  
}
```

가장 선호하는 폰트
↓
font-family: "Times New Roman", Times, serif;
↑
폰트의 종류 속성
↑
마지막에는 항상 일반적인 폰트를 지정해야 한다.

Arial
Arial (Bold)
Arial (Italic)
Arial (Bold Italic)
Arial Black
Comic Sans MS
Comic Sans MS (Bold)
Courier New
Courier New (Bold)
Courier New (Italic)
Courier New (Bold Italic)
Georgia
Georgia (Bold)
Georgia (Italic)
Georgia (Bold Italic)
Impact
Times New Roman
Times New Roman (Bold)
Times New Roman (Italic)
Times New Roman (Bold Italic)



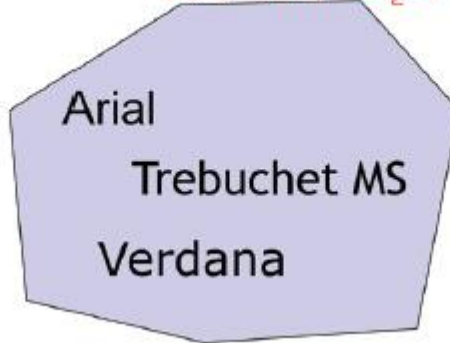
폰트 패밀리

- serif 폰트는 우아하고 전통적인 느낌
- sans-serif은 깔끔하고 가독성이 좋다.
- monospace는 타자기 서체
- cursive와 fantasy 폰트는 장난스러우며 스타일리쉬한 느낌

Serif 패밀리

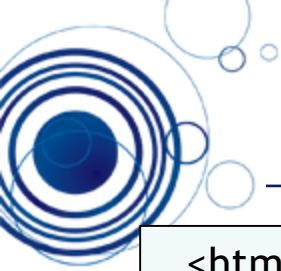


Sans Serif 패밀리



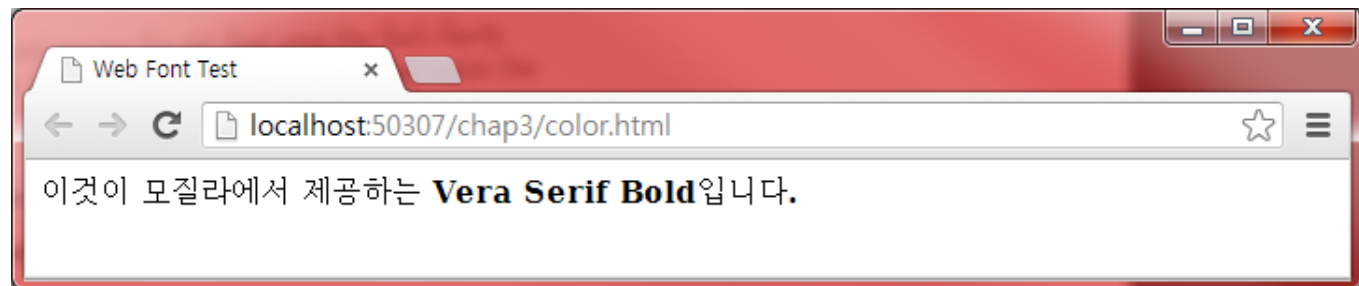
Monospace 패밀리





웹 폰트

```
<html>
<head>
  <title>Web Font Test</title>
  <style>
    @font-face {
      font-family: "Vera Serif Bold";
      src: url("http://developer.mozilla.org/@api/deki/files/2934/=VeraSeBd.ttf");
    }
    body { font-family: "Vera Serif Bold", serif }
  </style>
</head>
<body>
  이것이 모질라에서 제공하는 Vera Serif Bold입니다.
</body>
</html>
```





폰트 크기 설정

- 폰트의 단위
 - pt – 포인트
 - px - 픽셀
 - % - 퍼센트
 - em – 배수(scale factor)
 - 키워드 – xx-small, x-small, small, medium, large, x-large, xx-large

```
body {  
  font-size: 20pt;  
}
```

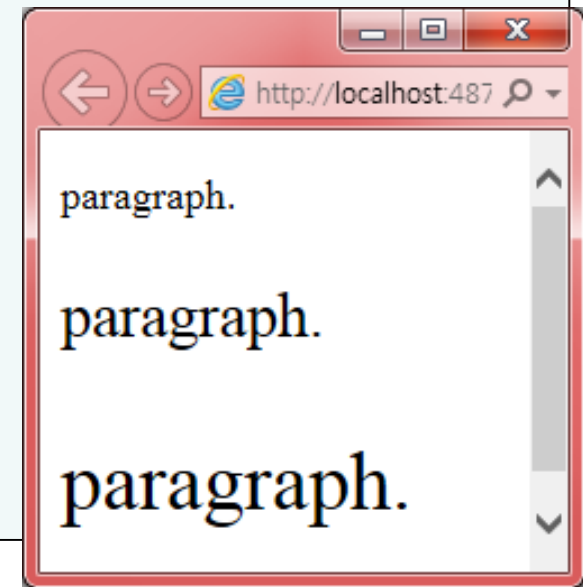
↑
폰트 크기를 나타내는 속성

↑
20포인트를 의미한다.



폰트 크기 예제

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {      font-size: medium;
    }
    p#t1 {      font-size: 1.0em;    }
    p#t2 {      font-size: 1.5em;    }
    p#t3 {      font-size: 2.0em;    }
  </style>
</head>
<body>
  <p id="t1">paragraph.</p>
  <p id="t2">paragraph.</p>
  <p id="t3">paragraph.</p>
</body>
</html>
```





폰트 속성

- font-weight – 볼드체 여부(normal, bold)
- font-style – 이탤릭체 여부(normal, italic, oblique)

```
h1#s1 { font-weight: normal; }  
h1#s2 { font-weight: bold; }
```

↑
볼드체로 설정

heading 1

heading 1

```
h1#s1 { font-style: normal; }  
h1#s2 { font-style: italic; }
```

↑
이탤릭체로 설정

heading 1

heading 1



폰트 축약 기법

```
<!DOCTYPE html>
<html>
<head>
  <style>
    p.style1 {
      font: italic 30px arial,sans-serif;
    }
    p.style2 {
      font: bold 40px Georgia,serif;
    }
  </style>
</head>
<body>
  <p class="style1">font: italic 30px arial,sans-serif</p>
  <p class="style2">font: bold 40px Georgia,serif</p>
</body>
</html>
```





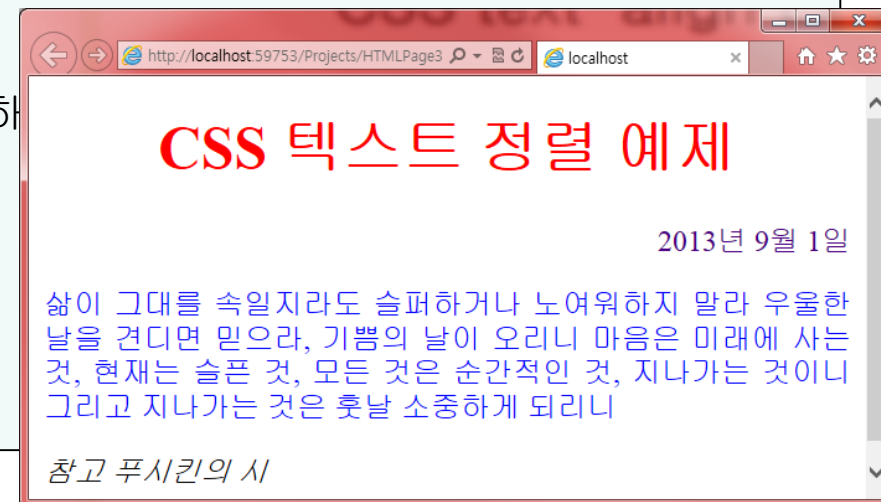
텍스트 스타일

속성	설명
<u>color</u>	텍스트의 색상을 지정한다.
<u>direction</u>	텍스트 작성 방향을 지정한다. (가로쓰기, 세로쓰기)
<u>letter-spacing</u>	글자간 간격을 지정한다.
<u>line-height</u>	텍스트 줄의 높이를 지정한다.
<u>text-align</u>	텍스트의 수평 정렬을 지정한다.
<u>text-decoration</u>	텍스트 장식을 지정한다.
<u>text-indent</u>	텍스트의 들여쓰기를 지정한다.
text-shadow	그림자 효과를 지정한다.
<u>text-transform</u>	대소문자 변환을 지정한다.



텍스트 정렬

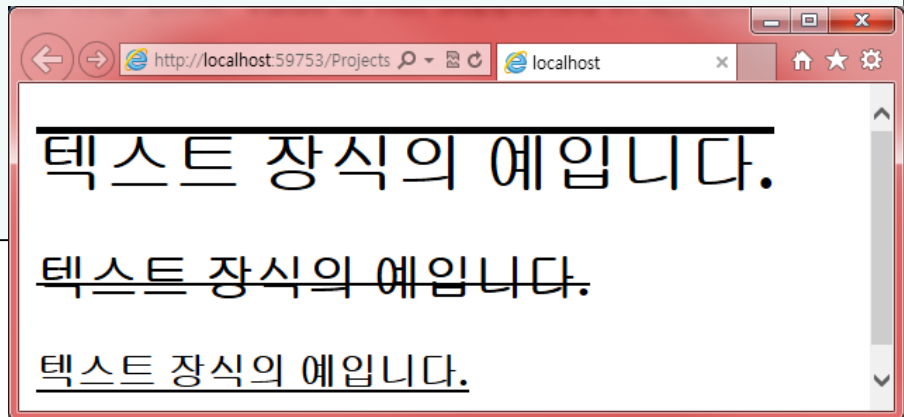
```
<!DOCTYPE html>
<html>
<head>
  <style>
    h1 {      text-align: center;      color: red;      } // 중앙정렬
    p.date {  text-align: right;      color: indigo;   } // 오른쪽정렬
    p.poet {  text-align: justify;    color: blue;     } // 양쪽정렬
  </style>
</head>
<body>
  <h1>CSS 텍스트 정렬 예제</h1>
  <p class="date">2013년 9월 1일</p>
  <p class="poet">
    삶이 그대를 속일지라도 슬퍼하거나 노여워하
    ...
  </p>
  <p><em>참고 푸시킨의 시</em> </p>
</body>
</html>
```



텍스트 장식

```
<!DOCTYPE html>
<html>
<head>
<style>
  h1 { text-decoration:overline; }
  h2 { text-decoration:line-through; }
  h3 { text-decoration:underline; }
</style>
</head>

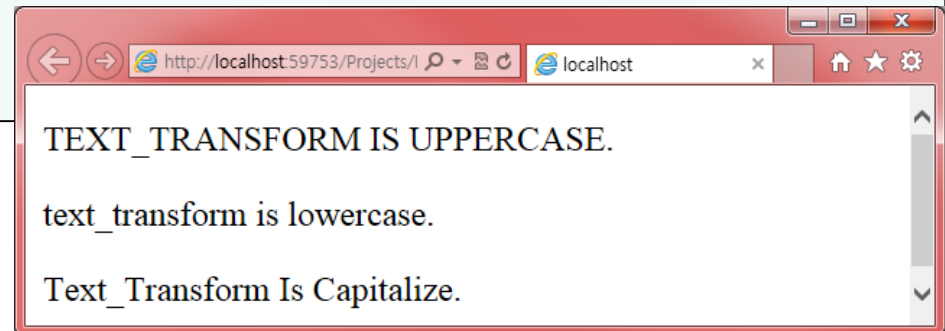
<body>
  <h1>텍스트 장식의 예입니다.</h1>
  <h2>텍스트 장식의 예입니다.</h2>
  <h3>텍스트 장식의 예입니다.</h3>
</body>
</html>
```



텍스트 변환

```
<!DOCTYPE html>
<html>
<head>
<style>
  p.upper { text-transform:uppercase; }
  p.lower { text-transform:lowercase; }
  p.capit { text-transform:capitalize; }
</style>
</head>

<body>
  <p class="upper">text_transform is uppercase.</p>
  <p class="lower">text_transform is lowercase.</p>
  <p class="capit">text_transform is capitalize.</p>
</body>
</html>
```



텍스트 그림자

```
<!DOCTYPE html>
<html>
<head>
  <style>
    h1 {
      text-shadow: 5px 5px 5px #FF0000;
    }
  </style>
</head>
<body>
  <h1>Text-shadow 처리!</h1>
</body>
</html>
```

```
h1 {
  text-shadow: 3px 5px 10px #000;
}
```

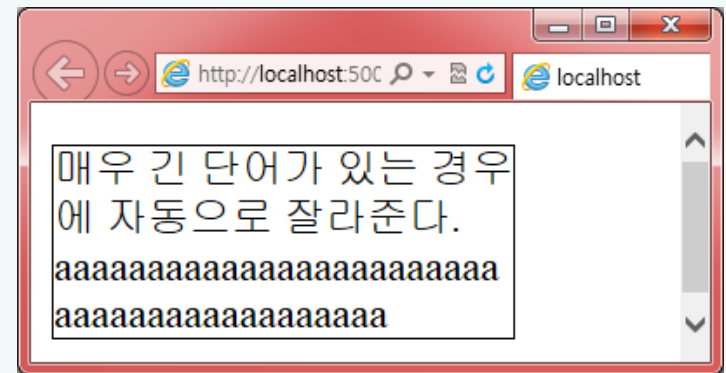
Diagram illustrating the text shadow properties in the CSS code:

- 3px**: x방향 이동거리 (x-direction movement distance)
- 5px**: y방향 이동거리 (y-direction movement distance)
- 10px**: 흐림 정도 (blur degree)
- #000**: 텍스트 그림자 색상 (text shadow color)

Text-shadow 처리!

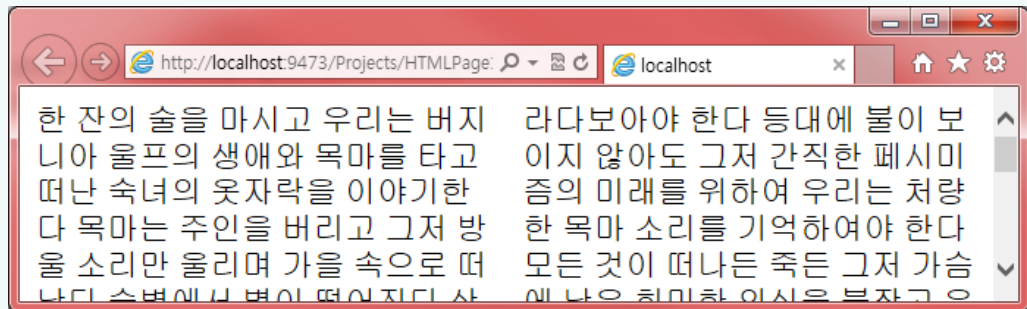
Word Wrapping

```
<!DOCTYPE html>
<html>
<head>
  <style>
    p.test {
      width: 11em;
      border: 1px solid #000000;
      word-wrap: break-word;
    }
  </style>
</head>
<body>
  <p class="test">
    매우 긴 단어가 있는 경우에 자동으로 잘라준다.
    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
  </p>
</body>
</html>
```



다중 컬럼

```
<!DOCTYPE html>
<html>
<head>
  <style>
    .poet {
      column-count: 2;
    }
  </style>
</head>
<body>
  <div class="poet">
    한 잔의 술을 마시고
    우리는 버지니아 울프의 생애와
    목마를 타고 떠난 숙녀의 옷자락을 이야기한다
    ...
    가을바람 소리는
    내 쓰러진 술병 속에서 목메어 우는데
  </div>
</body>
</html>
```



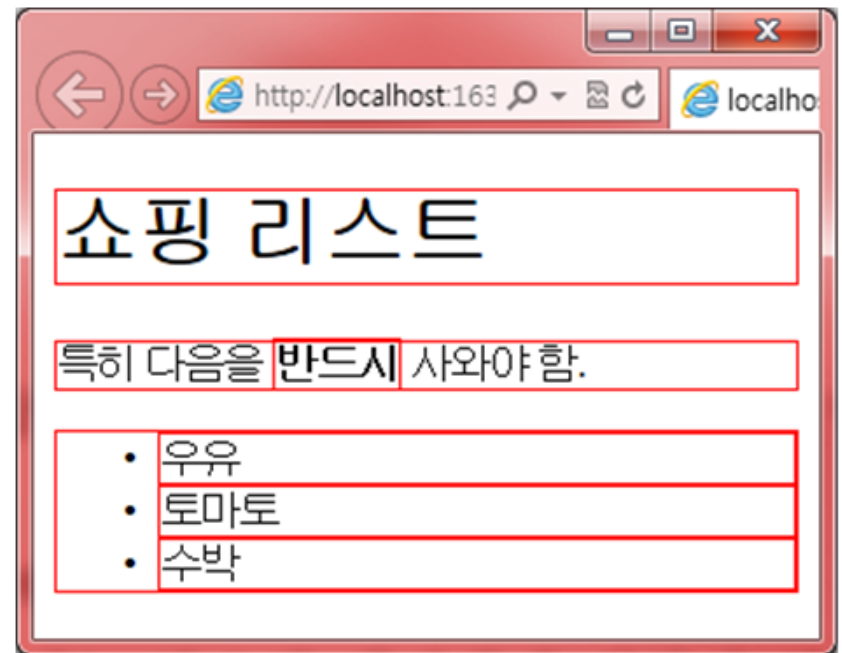


박스모델

- HTML 요소들을 박스(사각형) 형태로 그리는 것
- 박스는 배치, 색상, 경계 등의 속성을 가진다.

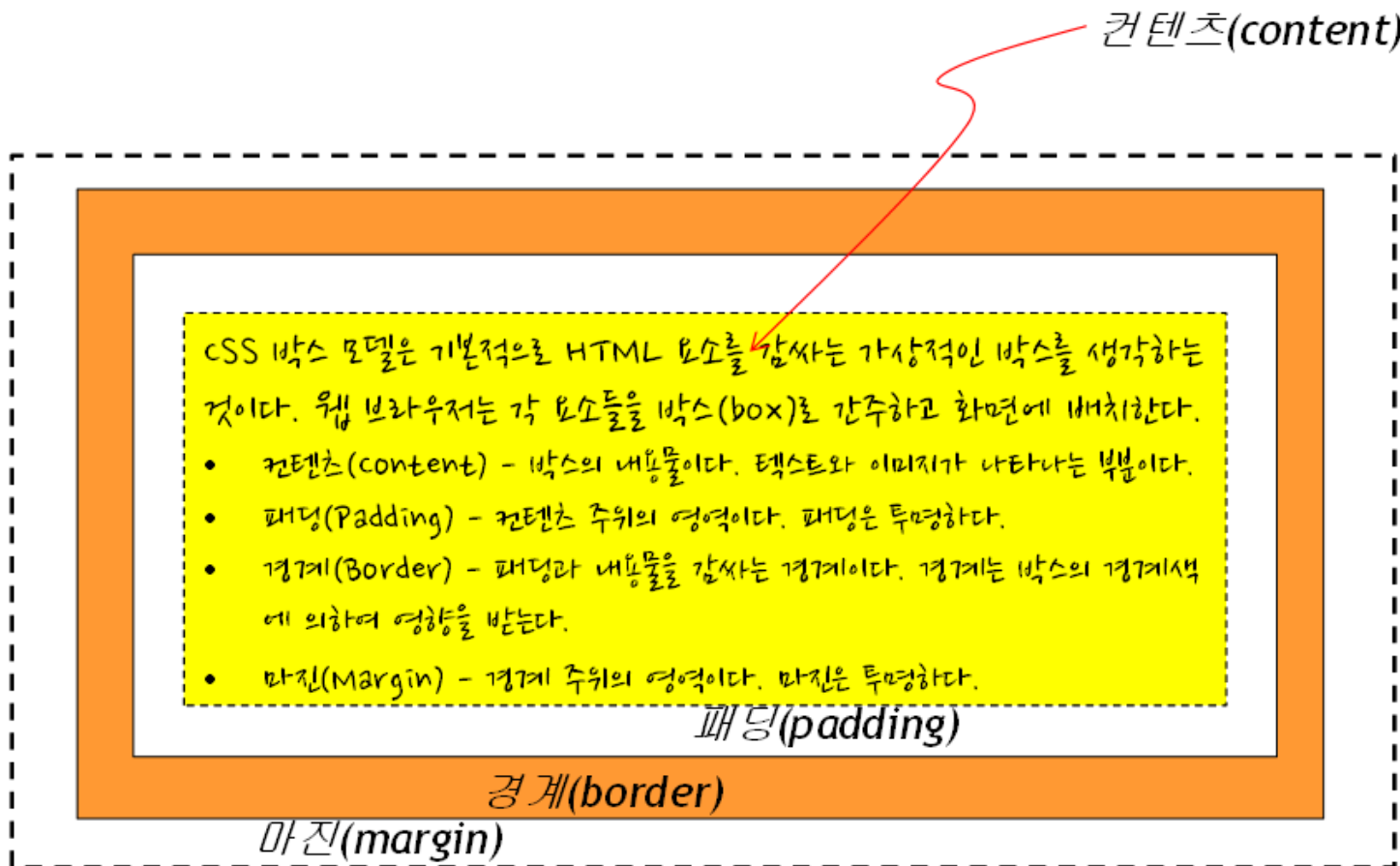
```
<!DOCTYPE html>
<html>
<body>
  <h1>쇼핑 리스트</h1>
  <p>특히 다음을 <strong>반드시</strong> 사와야 함.</p>
  <ul>
    <li>우유</li>
    <li>토마토</li>
    <li>수박</li>
  </ul>
</body>
</html>
```

->





박스모델의 속성

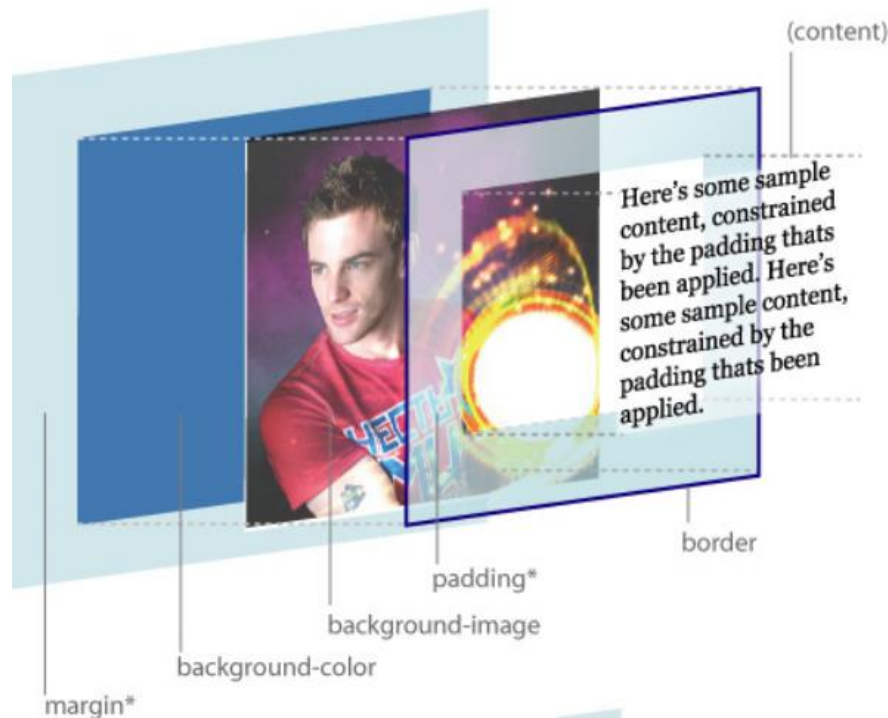




배경색과 배경 이미지

- HTML 요소에 배경색과 배경 이미지가 설정되어 있는 경우에, 패딩은 투명하므로 배경 이미지와 배경 색이 보이게 된다

THE CSS BOX MODEL HIERARCHY



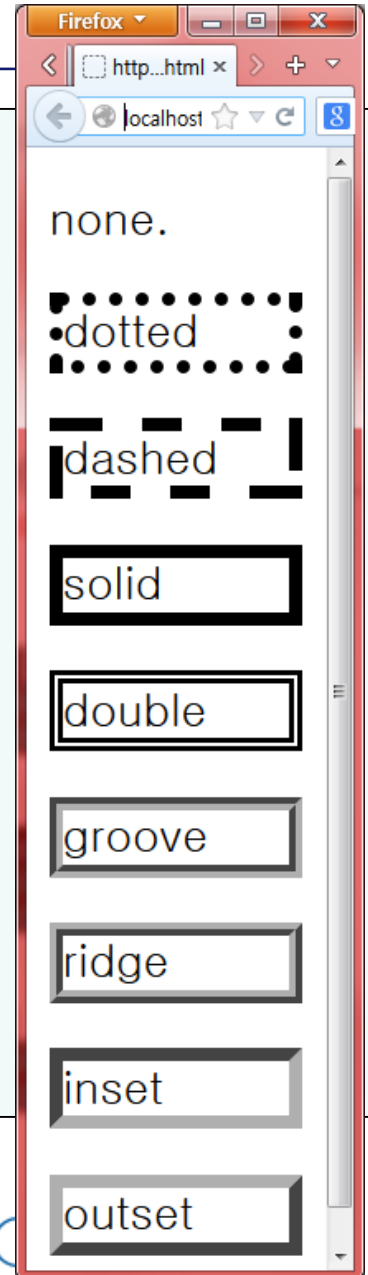
* transparent elements

(그림 출처: Jon Hicks)



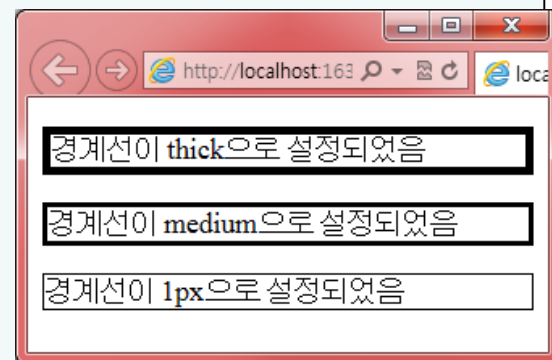
경계선 스타일

```
<!DOCTYPE html>
<html>
<body>
  <p style="border-style: none">none.</p>
  <p style="border-style: dotted">dotted</p>
  <p style="border-style: dashed">dashed</p>
  <p style="border-style: solid">solid</p>
  <p style="border-style: double">double</p>
  <p style="border-style: groove">groove</p>
  <p style="border-style: ridge">ridge</p>
  <p style="border-style: inset">inset</p>
  <p style="border-style: outset">outset</p>
</body>
</html>
```



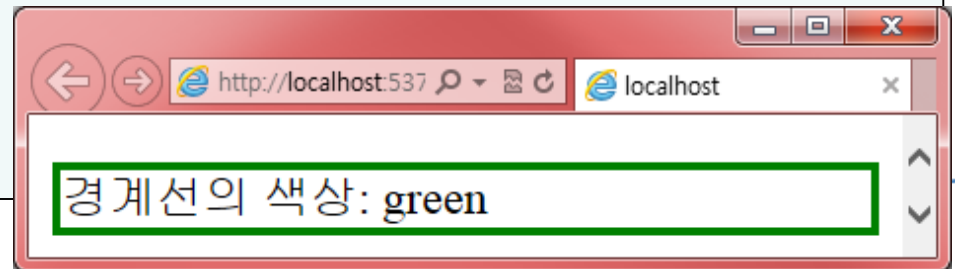
경계선의 폭

```
...
<head>
  <style>
    p.thick {
      border-style: solid;      border-width: thick;
    }
    p.medium {
      border-style: solid;      border-width: medium;
    }
    p.thin {
      border-style: solid;      border-width: 1px;
    }
  </style>
</head>
<body>
  <p class="thick">경계선이 thick으로 설정되었음</p>
  <p class="medium">경계선이 medium으로 설정되었음</p>
  <p class="thin">경계선이 1px으로 설정되었음</p>
</body>
</html>
```



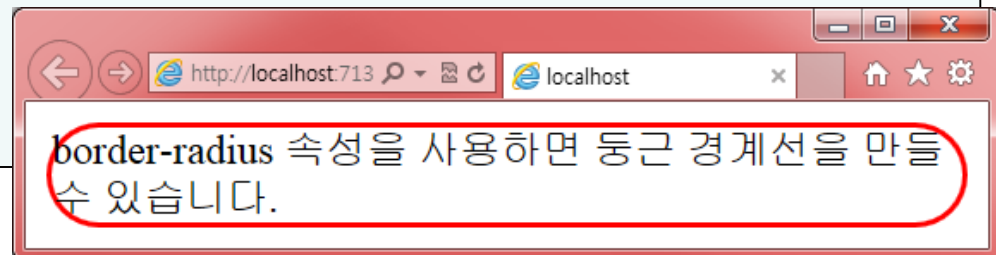
경계선의 색상

```
<!DOCTYPE html>
<html>
<head>
  <style>
    p.green {
      border-style: solid;
      border-color: green;
    }
  </style>
</head>
<body>
  <p class="green">경계선의 색상: green</p>
</body>
</html>
```



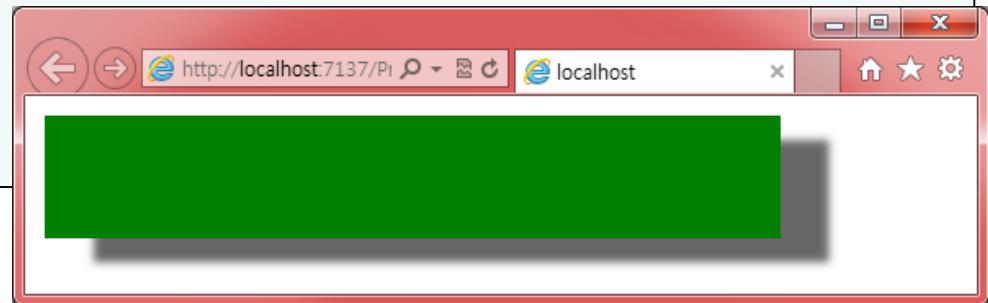
둥근 경계선

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      border: 2px solid red;
      border-radius: 25px;
    }
  </style>
</head>
<body>
  <div>border-radius 속성을 사용하면 둥근 경계선을 만들 수 있습니다.</div>
</body>
</html>
```



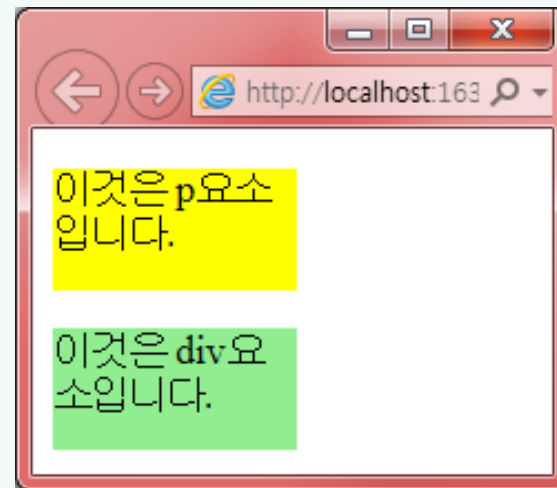
경계선 그림자

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width: 300px;
      height: 50px;
      background-color: green;
      box-shadow: 20px 10px 5px #666666;
    }
  </style>
</head>
<body>
  <div></div>
</body>
</html>
```



요소 크기 설정

```
<!DOCTYPE html>
<html>
<head>
  <style>
    #target1 {
      width: 100px;
      height: 50px;
      background-color: yellow;
    }
    #target2 {
      width: 100px;
      height: 50px;
      background-color: lightgreen;
    }
  </style>
</head>
<body>
  <p id="target1">이것은 p요소입니다. </p>
  <div id="target2">이것은 div요소입니다.</div>
</body>
</html>
```





마진과 패딩 설정하기

값	설명
auto	브라우저가 마진을 계산한다.
length	마진을 px, pt, cm 단위로 지정할 수 있다. 디폴트는 0px이다.
%	마진을 요소 폭의 퍼센트로 지정한다.
inherit	마진이 부모 요소로부터 상속된다.

- 마진의 각 변을 별도로 지정하는 방법.

```
margin-top:10px;  
margin-right:20px;  
margin-bottom:30px;  
margin-left:40px;
```

- 한 줄에서 모든 변에 대한 값을 지정하는 방법.

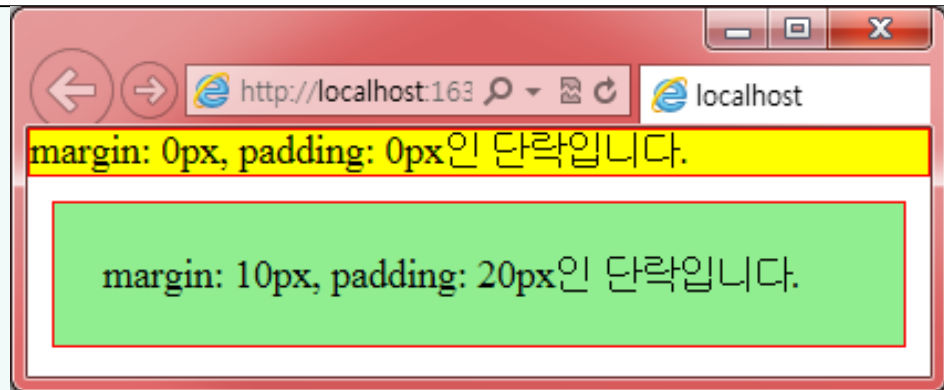
```
margin: 10px 20px 30px 40px;
```

The diagram illustrates the shorthand `margin` property. Below the code, four yellow boxes labeled `top`, `right`, `bottom`, and `left` have red arrows pointing to the corresponding values in the `margin` declaration: `10px` for top, `20px` for right, `30px` for bottom, and `40px` for left.



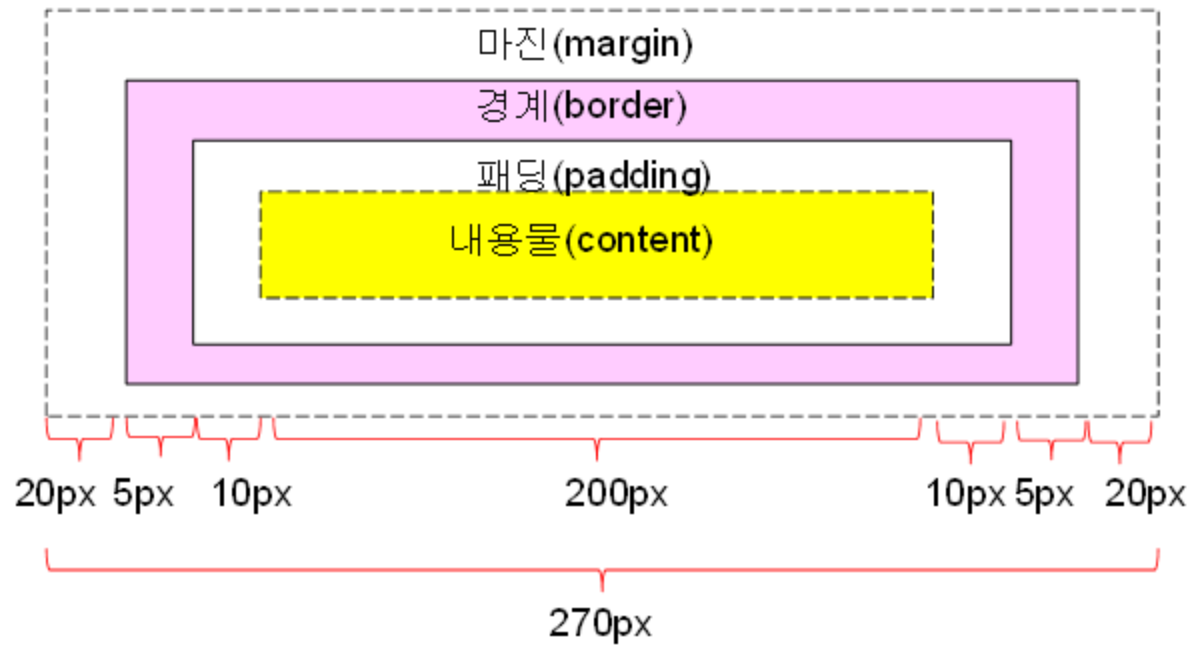
마진과 패딩 예제

```
...  
<style>  
  body {  
    margin: 0px;  
    padding: 0px;  
  }  
  p {  
    margin: 0px;  
    padding: 0px;  
    background-color: yellow; border: 1px solid red;  
  }  
  #target {  
    margin: 10px;  
    padding: 20px;  
    background-color: lightgreen; border: 1px solid red;  
  }  
</style>  
</head>  
<body>  
  <p>margin: 0px, padding: 0px인 단락입니다.</p>  
  <p id="target">margin: 10px, padding: 20px인 단락입니다.</p>  
</body>  
</html>
```





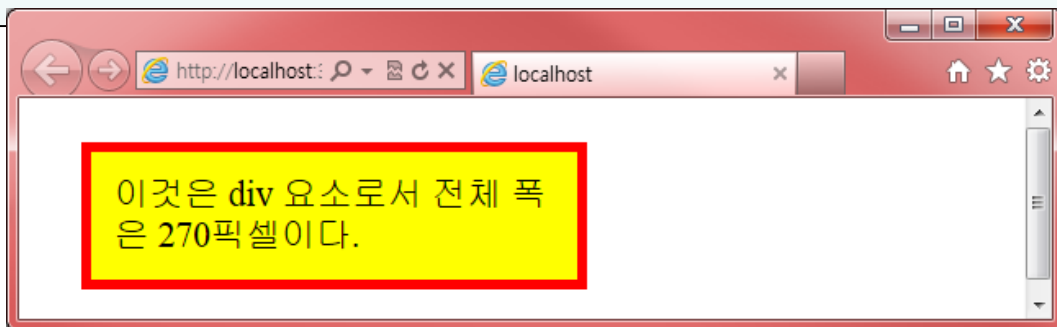
박스의 크기 계산



예제

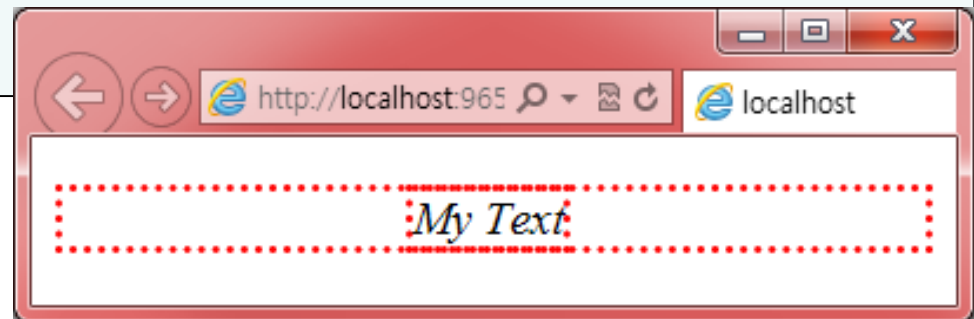
```
<!DOCTYPE html>
<html>
<head>
  <style>
    div.test {
      background-color: yellow;
      width: 200px;
      padding: 10px;
      border: 5px solid red;
      margin: 20px;
    }
  </style>
</head>

<body>
  <div class="test">이것은 div 요소로서 전체 폭은 270픽셀이다.</div>
</body>
</html>
```



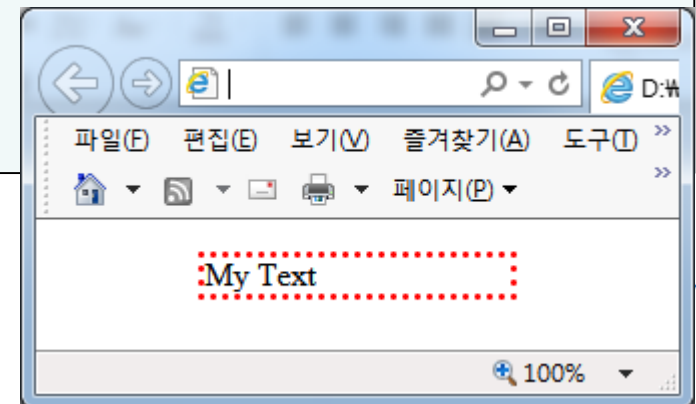
수평정렬(인라인 요소)

```
<!DOCTYPE html>
<html>
<head>
  <style>
    p, em, strong, div
    {
      border: dotted 3px red;
    }
  </style>
</head>
<body>
  <p style="text-align: center">
    <em>My Text </em>
  </p>
</body>
</html>
```



수평정렬(블록 요소)

```
<!DOCTYPE html>
<html>
<head>
  <style>
    p, h1
    {
      border: dotted 3px red;
    }
  </style>
</head>
<body>
  <p style="margin-left: auto; margin-right: auto; width: 50%">
    My Text
  </p>
</body>
</html>
```



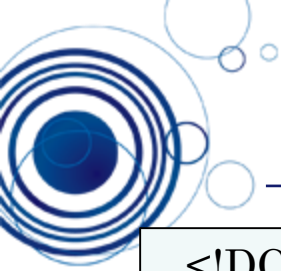


배경 설정하기

속성	설명
<u>background</u>	한줄에서 모든 배경 속성을 정의한다.
<u>background-attachment</u>	배경 이미지가 고정되어 있는지 스크롤되는지를 지정한다.
<u>background-color</u>	배경색을 정의한다.
<u>background-image</u>	배경 이미지를 정의한다.
<u>background-position</u>	배경 이미지의 시작위치를 지정한다.
<u>background-repeat</u>	배경 이미지의 반복 여부를 지정한다.

```
body { background-color: red; }           /* 배경을 빨간색으로 설정한다 */  
body { background-color: rgb(255,0,0); }  /* 배경을 빨간색으로 설정한다 */  
body { background-color: #ff0000; }      /* 배경을 빨간색으로 설정한다 */
```





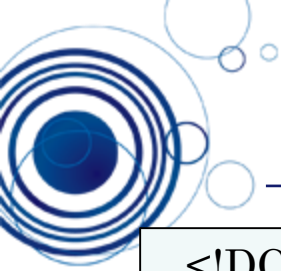
배경 이미지 설정

```
<!DOCTYPE html>
<html>

<head>
  <style>
    body {
      background-image: url('back1.jpg')
    }
  </style>
</head>

<body>
  <h1>삶이 그대를 속일지라도</h1>
  <p> 삶이 그대를 속일지라도
  슬퍼하거나 노하지 말아라.
  ...
  지나가 버린 것 그리움이 되리니.
  </p>
</body>
</html>
```



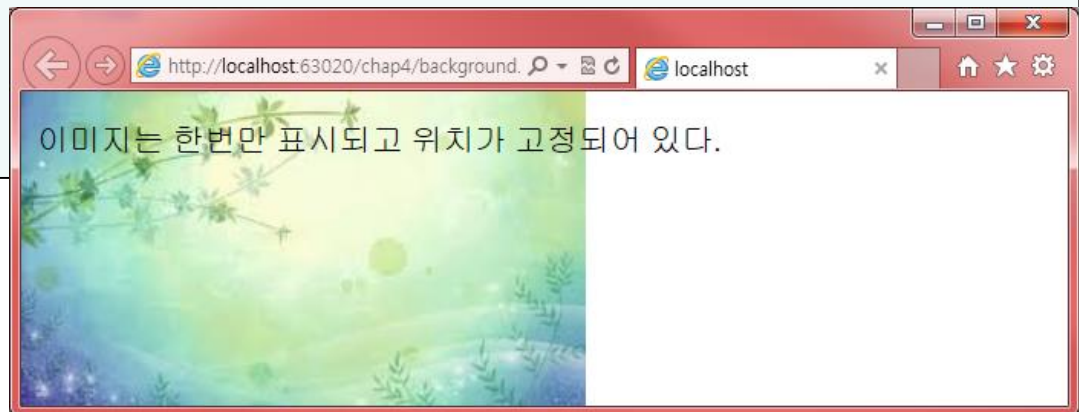


고정된 배경 이미지

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      background-image: url('back1.jpg');
      background-repeat: no-repeat;
      background-attachment: fixed;
    }
  </style>
</head>

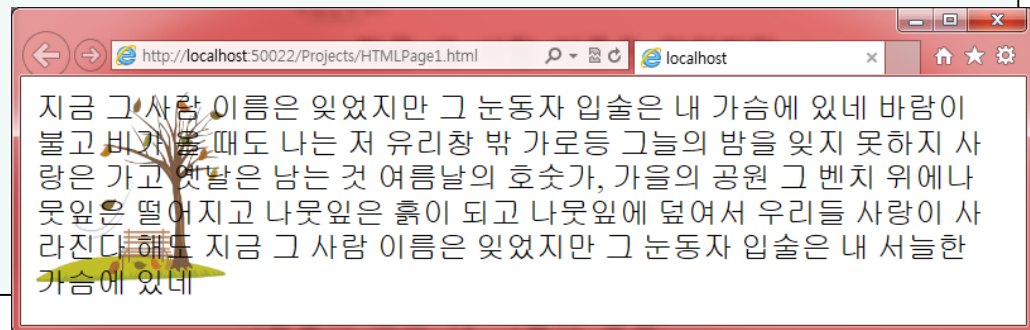
<body>
  <p>이미지는 한번만 표시되고 위치가 고정되어 있다.</p>

</body>
</html>
```



배경 이미지 크기

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width: 500px;
      height: 100px;
      background: url(back.jpg);
      background-repeat: no-repeat;
      background-size: 100px 100px;
    }
  </style>
</head>
<body>
  <div>
    지금 그 사람의 이름은 잊었지만
    그의 눈동자 입술은
    내 가슴에 있네
    ...
    내 가슴에 있네
  </div>
</body>
</html>
```





링크 스타일

- a:link - 방문되지 않은 링크의 스타일
- a:visited - 방문된 링크의 스타일
- a:hover - 마우스가 위에 있을 때의 스타일
- a:active - 마우스로 클릭되는 때의 스타일



링크 예제

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

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

```
  a:visited { color:green; }
```

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

```
  a:active { color:yellow; }
```

```
</style>
```

```
</head>
```

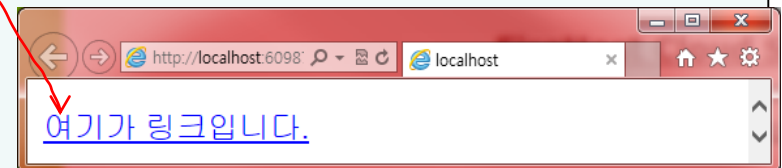
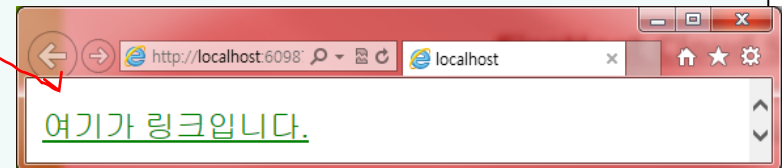
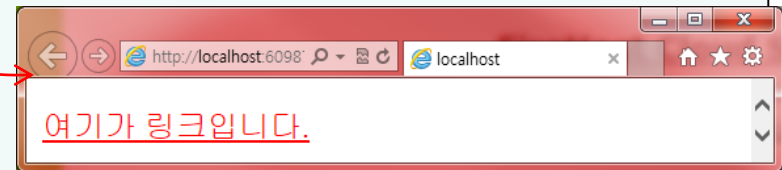
```
<body>
```

```
  <p><a href="" target="_blank">여기가  
  링크입니다.</a>
```

```
  </p>
```

```
</body>
```

```
</html>
```



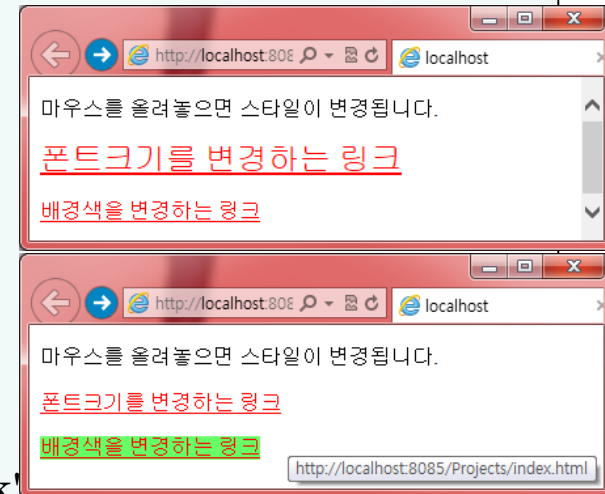
링크 예제

```
<!DOCTYPE html>

<html>
<head>
  <style>
    a.style1:link {      color: #ff0000;      }
    a.style1:visited {   color: #0000ff;      }
    a.style1:hover {     font-size: 150%;     }
    a.style2:link {      color: #ff0000;      }
    a.style2:visited {   color: #0000ff;      }

    a.style2:hover {     background: #66ff66;  }
  </style>
</head>

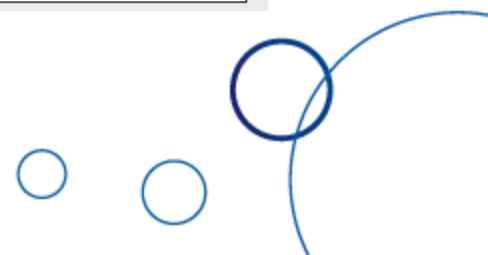
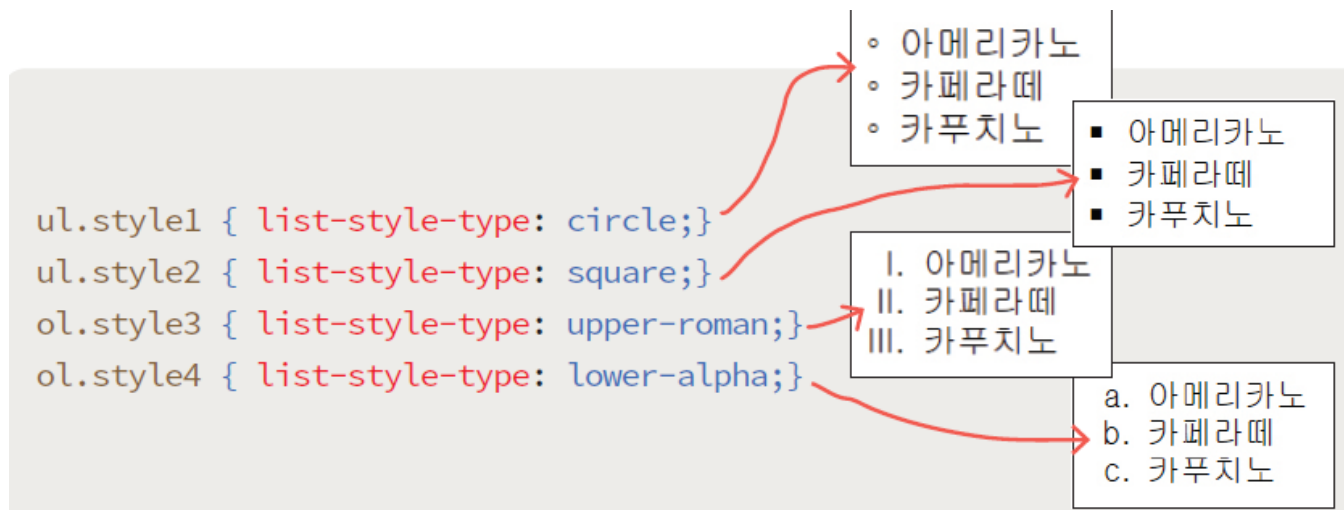
<body>
  <p>마우스를 올려놓으면 스타일이 변경됩니다.</p>
  <p><a class="style1" href="index.html" target="_blank">
폰트크기를 변경하는 링크</a></p>
  <p><a class="style2" href="index.html" target="_blank">
배경색을 변경하는 링크</a></p>
</body>
</html>
```

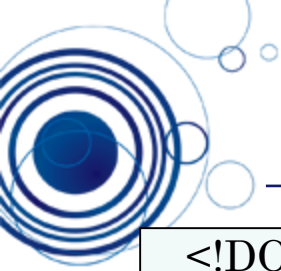




리스트 스타일

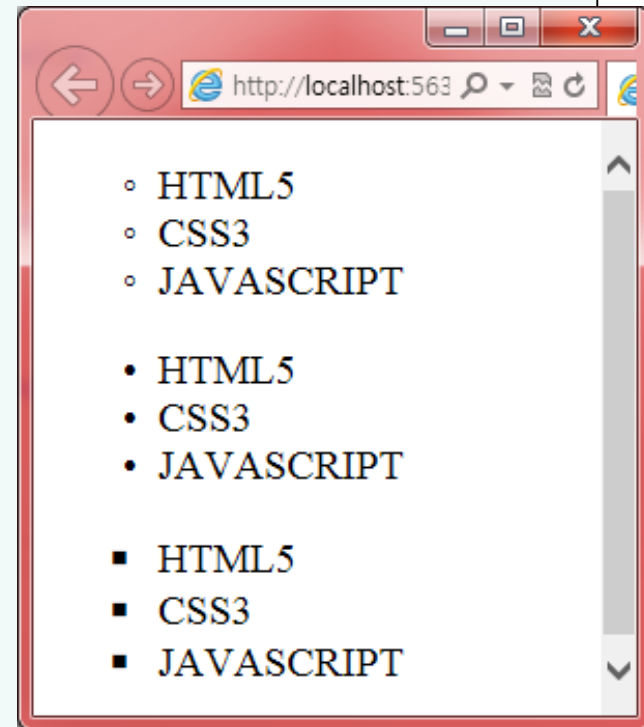
속성	설명
list-style	리스트에 대한 속성을 한줄로 설정한다.
list-style-image	리스트 항목 마커를 이미지로 지정한다.
list-style-position	리스트 마커의 위치를 안쪽인지 바깥쪽인지를 지정한다.
list-style-type	리스트 마커의 타입을 지정한다.





링크 예제

```
<!DOCTYPE html>
<html>
<head>
  <style>
    ul.a {      list-style-type: circle;    }
    ul.b {      list-style-type: disc;      }
    ul.c {      list-style-type: square;    }
  </style>
</head>
<body>
  <ul class="a">
    <li>HTML5</li>
    <li>CSS3</li>
    <li>JAVASCRIPT</li>
  </ul>
  <ul class="b">
    <li>HTML5</li>
    <li>CSS3</li>
    <li>JAVASCRIPT</li>
  </ul>
  <ul class="c">
    <li>HTML5</li>
    <li>CSS3</li>
    <li>JAVASCRIPT</li>
  </ul>
</body>
</html>
```





수평 리스트 예제


```
<!DOCTYPE html>

<html>

<head>
  <style>
    ul {
      list-style:none;
      text-align:center;
      border-top:1px solid red; border-bottom:1px solid red; padding:10px 0;

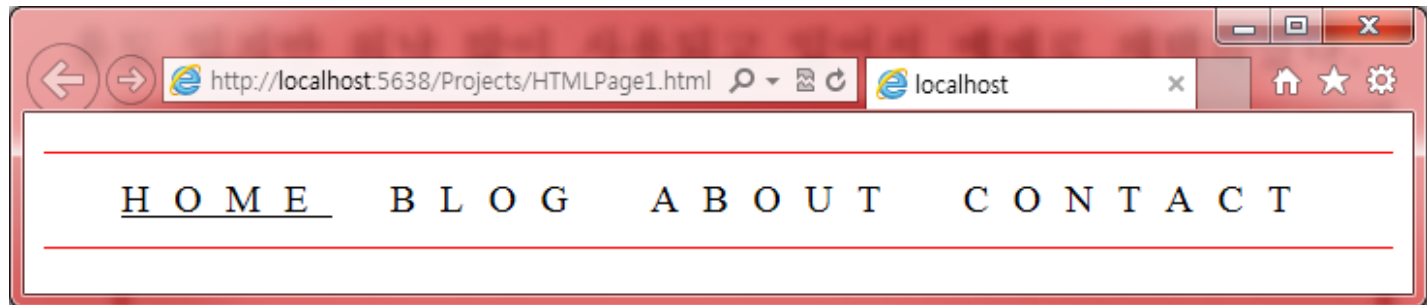
    }
    ul li {
      display:inline;
      text-transform:uppercase;
      padding:0 10px; letter-spacing:10px;
    }
    ul li a { text-decoration:none; color:black; }
    ul li a:hover { text-decoration:underline; }

  </style>
</head>
```



수평 리스트 예제

```
<body>
  <ul>
    <li><a href="#">Home</a></li>
    <li><a href="#">Blog</a></li>
    <li><a href="#">About</a></li>
    <li><a href="#">Contact</a></li>
  </ul>
</body>
</html>
```





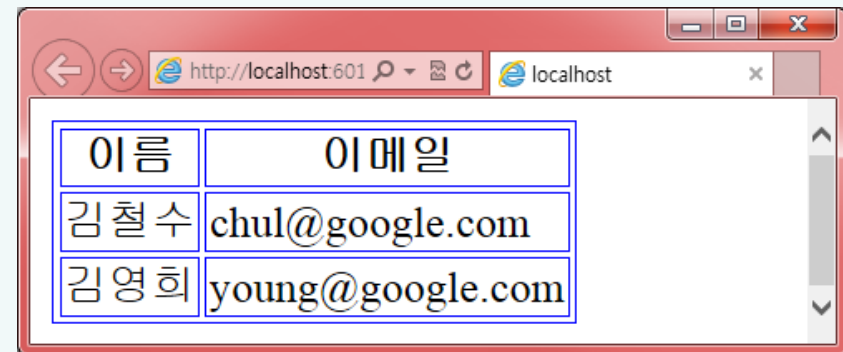
테이블 스타일

속성	설명
<code>border</code>	테이블의 경계선
<code>border-collapse</code>	이웃한 셀의 경계선을 합칠 것인지 여부
<code>width</code>	테이블의 가로 길이
<code>height</code>	테이블의 세로 길이
<code>border-spacing</code>	테이블 셀 사이의 거리
<code>empty-cells</code>	공백 셀을 그릴 것인지 여부
<code>table-align</code>	테이블 셀의 정렬 설정



테이블의 경계

```
<!DOCTYPE html>
<html>
<head>
  <style>
    table, td, th {
      border: 1px solid blue;
    }
  </style>
</head>
<body>
  <table>
    <tr>
      <th>이름</th>
      <th>이메일</th>
    </tr>
    <tr>
      <td>김철수</td>
      <td>chul@google.com</td>
    </tr>
    <tr>
      <td>김영희</td>
      <td>young@google.com</td>
    </tr>
  </table>
</body>
</html>
```



A screenshot of a web browser window displaying the rendered HTML table. The browser's address bar shows 'http://localhost:601' and the page title is 'localhost'. The table has two columns: '이름' (Name) and '이메일' (Email). The first row contains '김철수' (Kim Cheol-su) and 'chul@google.com'. The second row contains '김영희' (Kim Young-hee) and 'young@google.com'. The table is styled with a blue border.

이름	이메일
김철수	chul@google.com
김영희	young@google.com

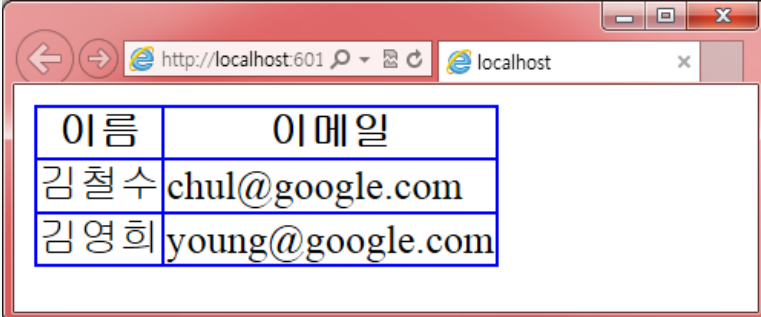


경계 통합

- collapse : 이웃하는 셀의 경계선을 합쳐서 단일선으로 표시한다.
- separate : 이웃하는 셀의 경계선을 합치지 않고 분리하여 표시한다.

```
<!DOCTYPE html>
<html>
<head>
  <style>
    table {
      border-collapse: collapse;
    }

    table, th, td {
      border: 1px solid blue;
    }
  </style>
</head>
...
```

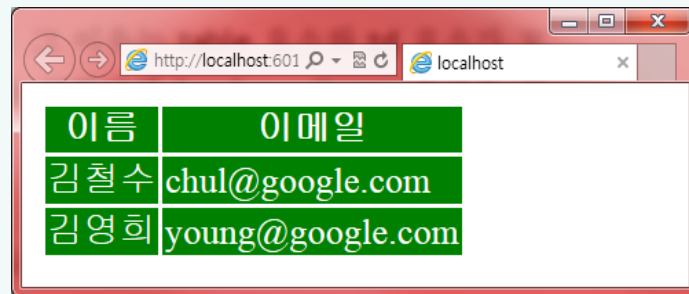


A screenshot of a web browser window displaying the rendered table. The browser's address bar shows 'http://localhost:601' and the page title is 'localhost'. The table has two columns: '이름' (Name) and '이메일' (Email). The first row contains '김철수' and 'chul@google.com'. The second row contains '김영희' and 'young@google.com'. The borders of the table and its cells are collapsed into a single blue line.

이름	이메일
김철수	chul@google.com
김영희	young@google.com

테이블 배경색

```
<!DOCTYPE html>
<html>
<head>
  <style>
    td, th {
      color: white;
      background-color: green;
    }
  </style>
</head>
<body>
  <table>
    <tr>
      <th>이름</th>
      <th>이메일</th>
    </tr>
    <tr>
      <td>김철수</td>
      <td>chul@google.com</td>
    </tr>
    <tr>
      <td>김영희</td>
      <td>young@google.com</td>
    </tr>
  </table>
</body>
</html>
```



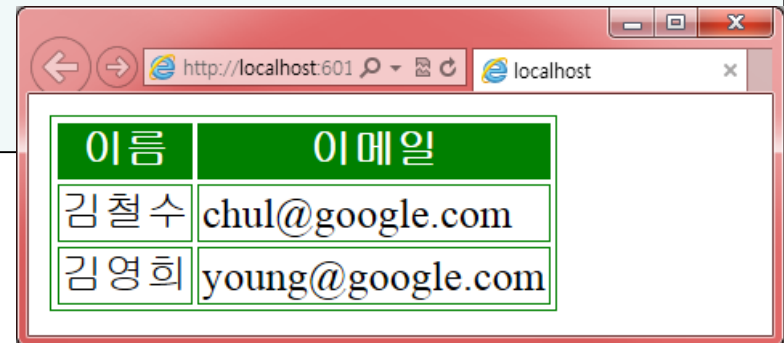
A screenshot of a web browser window displaying the rendered HTML table. The browser's address bar shows 'http://localhost:601' and the page title is 'localhost'. The table has a green background and white text, matching the CSS in the code block.

이름	이메일
김철수	chul@google.com
김영희	young@google.com

헤더와 데이터의 분리

```
<!DOCTYPE html>
<html>
<head>
  <style>
    table, td, th {
      border: 1px solid green;
    }

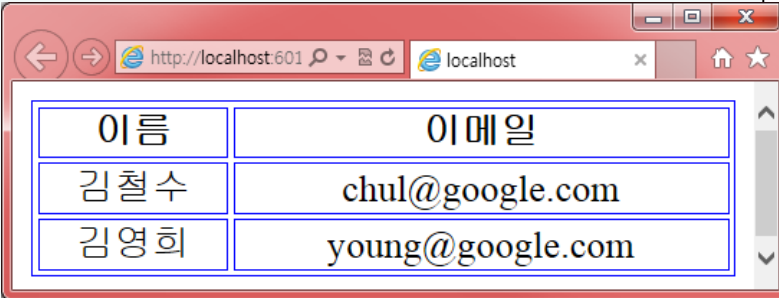
    th {
      background-color: green;
      color: white;
    }
  </style>
</head>
...
```



이름	이메일
김철수	chul@google.com
김영희	young@google.com

테이블 텍스트 정렬

```
<!DOCTYPE html>
<html>
<head>
<style>
  table, td, th { border: 1px solid blue; }
  table { width: 100%; }
  td { text-align: center; }
</style>
</head>
<body>
  <table>
    <tr>
      <th>이름</th>
      <th>이메일</th>
    </tr>
    <tr>
      <td>김철수</td>
      <td>chul@google.com</td>
    </tr>
    <tr>
      <td>김영희</td>
      <td>young@google.com</td>
    </tr>
  </table>
</body>
</html>
```



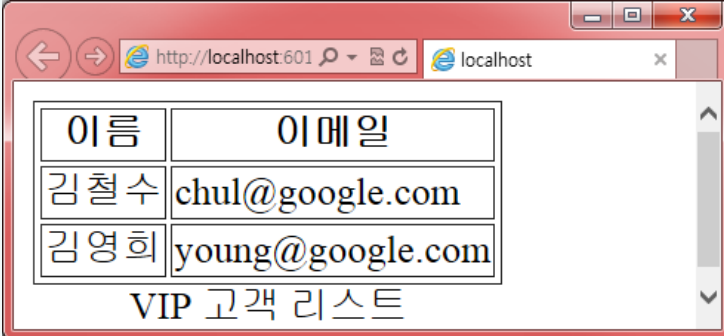
A screenshot of a web browser window showing the rendered HTML table. The browser's address bar displays 'http://localhost:601' and the page title is 'localhost'. The table has two columns: '이름' (Name) and '이메일' (Email). The first row contains the headers, and the subsequent two rows contain data for Kim Cheol-soo and Kim Young-hee.

이름	이메일
김철수	chul@google.com
김영희	young@google.com

테이블 캡션

```
<!DOCTYPE html>
<html>
<head>
  <style>
    caption { caption-side:bottom; }
  </style>
</head>

<body>
  <table border="1">
    <caption>VIP 고객 리스트</caption>
    <tr>
      <th>이름</th>
      <th>이메일</th>
    </tr>
    <tr>
      <td>김철수</td>
      <td>chul@google.com</td>
    </tr>
    <tr>
      <td>김영희</td>
      <td>young@google.com</td>
    </tr>
  </table>
</body>
</html>
```



A screenshot of a web browser window displaying the rendered HTML table. The browser's address bar shows 'http://localhost:601' and the page title is 'localhost'. The table has two columns: '이름' (Name) and '이메일' (Email). It contains two data rows: one for '김철수' (Kim Cheol-su) with email 'chul@google.com', and another for '김영희' (Kim Young-hee) with email 'young@google.com'. Below the table, the caption 'VIP 고객 리스트' (VIP Customer List) is displayed, demonstrating the effect of the CSS rule 'caption-side:bottom;'.

이름	이메일
김철수	chul@google.com
김영희	young@google.com

VIP 고객 리스트



짝수행과 홀수행 다르게 하기

```
<!DOCTYPE html>
<html>
<head>
  <style>
    #list {
      font-family: "Trebuchet MS",sans-serif;
      width: 100%;

    }

    #list td, #list th {
      border: 1px dotted gray;
      text-align: center;

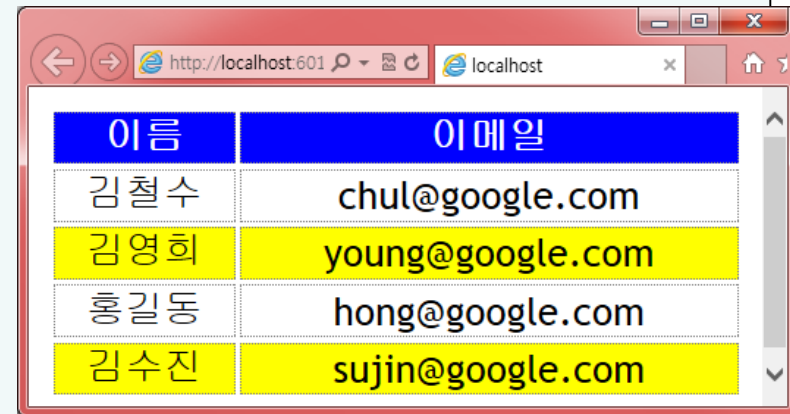
    }

    #list th {
      color: white;
      background-color: blue;
    }

    #list tr.alt td {
      background-color: yellow;
    }
  </style>
</head>
```

짝수행과 홀수행 다르게 하기

```
<body>
  <table id="list">
    <tr>
      <th>이름</th>
      <th>이메일</th>
    </tr>
    <tr>
      <td>김철수</td>
      <td>chul@google.com</td>
    </tr>
    <tr class="alt">
      <td>김영희</td>
      <td>young@google.com</td>
    </tr>
    <tr>
      <td>홍길동</td>
      <td>hong@google.com</td>
    </tr>
    <tr class="alt">
      <td>김수진</td>
      <td>sujin@google.com</td>
    </tr>
  </table>
</body>
</html>
```



이름	이메일
김철수	chul@google.com
김영희	young@google.com
홍길동	hong@google.com
김수진	sujin@google.com



레이아웃이란?

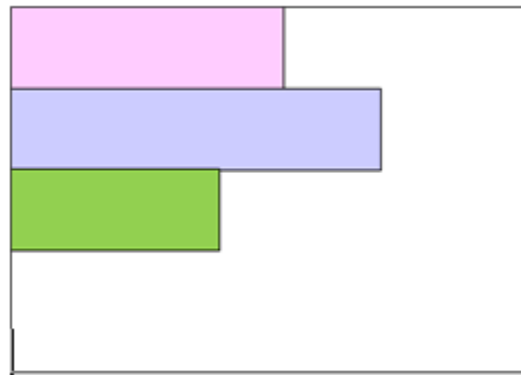
- 웹페이지에서 HTML 요소의 위치, 크기 등을 결정하는 것
 - 집안에서의 가구 배치와 비슷하다.



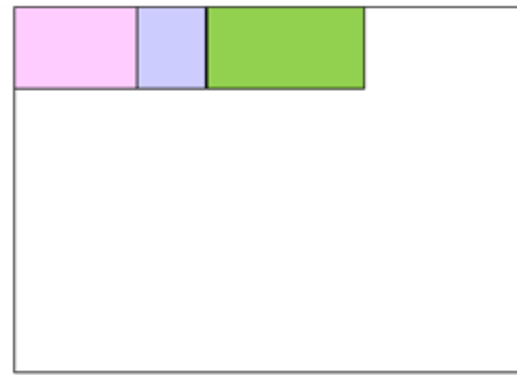


블록요소와 인라인 요소

- 블록(block) 요소 - 화면의 한 줄을 전부 차지한다.
- 인라인(inline) 요소 - 한 줄에 차례대로 배치된다.
현재 줄에서 필요한 만큼의 너비만을 차지한다.



블록(block)



인라인(inline)

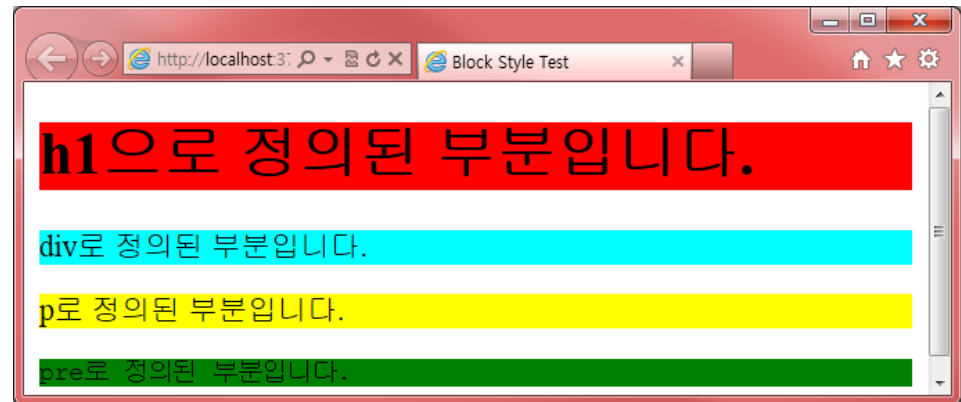




블록요소

- 한 줄을 전부 차지
- `<h1>`, `<p>`, ``, ``, `<table>`, `<blockquote>`, `<pre>`, `<div>`
`<form>` , `<header>`, `<nav>` 요소

```
<body>  
  <h1 style="background-color: red">h1으로 정의된 부분입니다.</h1>  
  <div style="background-color: aqua">div로 정의된 부분입니다.</div>  
  <p style="background-color: yellow">p로 정의된 부분입니다.</p>  
  <pre style="background-color: green">pre로 정의된 부분입니다.</pre>  
</body>
```

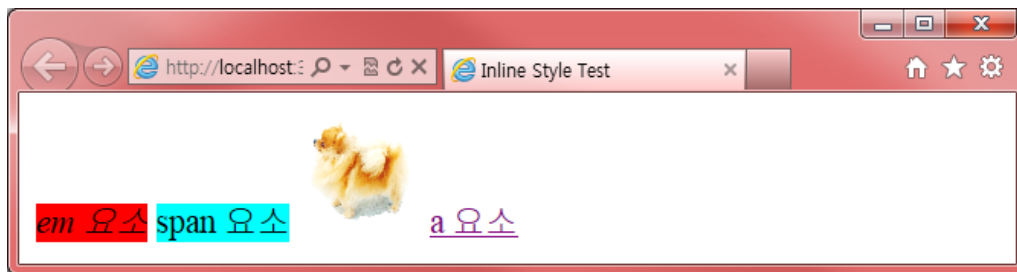




인라인요소

- 인라인 요소들은 한 줄 안에 차례대로 배치
- `<a>`, ``, ``, ``, `
`, `<input>`, `` 요소

```
<body>
  <em style="background-color: red">em 요소</em>
  <span style="background-color: aqua">span 요소</span>
  
  <a href="http://www.w3c.org">a 요소</a>
</body>
```

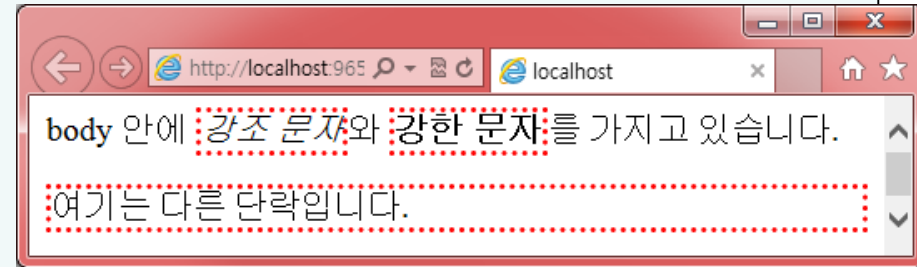




블록 요소와 인라인 요소의 혼합

```
<!DOCTYPE html>

<html>
<head>
  <style>
    p, em, strong {
      border: dotted 3px red;
    }
  </style>
</head>
<body>
  body 안에
  <em>강조 문자</em>와 <strong>강한 문자</strong>를 가지고 있습니다.
  <p>여기는 다른 단락입니다. </p>
</body>
</html>
```





CSS의 display 속성

- 속성 display를 block으로 설정하면 -> 블록 요소처럼 배치
- display를 inline으로 설정 -> 인라인 요소처럼 배치
 - display:block : 블록(block)
 - display:inline : 인라인(inline)
 - display:none : 없는 것으로 간주됨
 - display:hidden : 화면에서 감춰짐

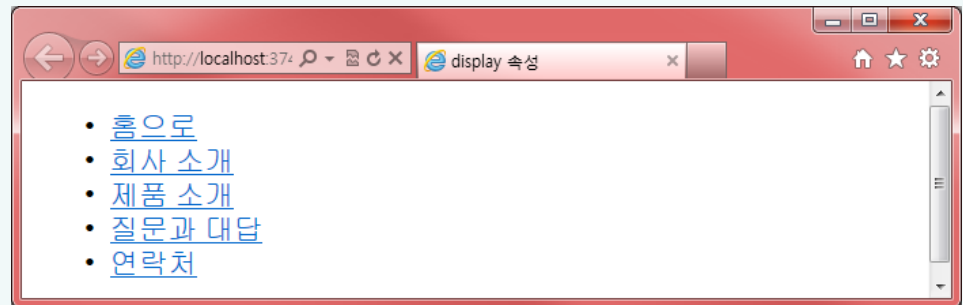


예제

```
<!DOCTYPE html>
<html>
<head>
  <title>display 속성</title>
  <style>

    .menubar li {
      display: inline;
      background-color: yellow;
      border: 1px solid;
      border-color: red;
      margin: 0;
      padding: .5em;
    }
  </style>
</head>

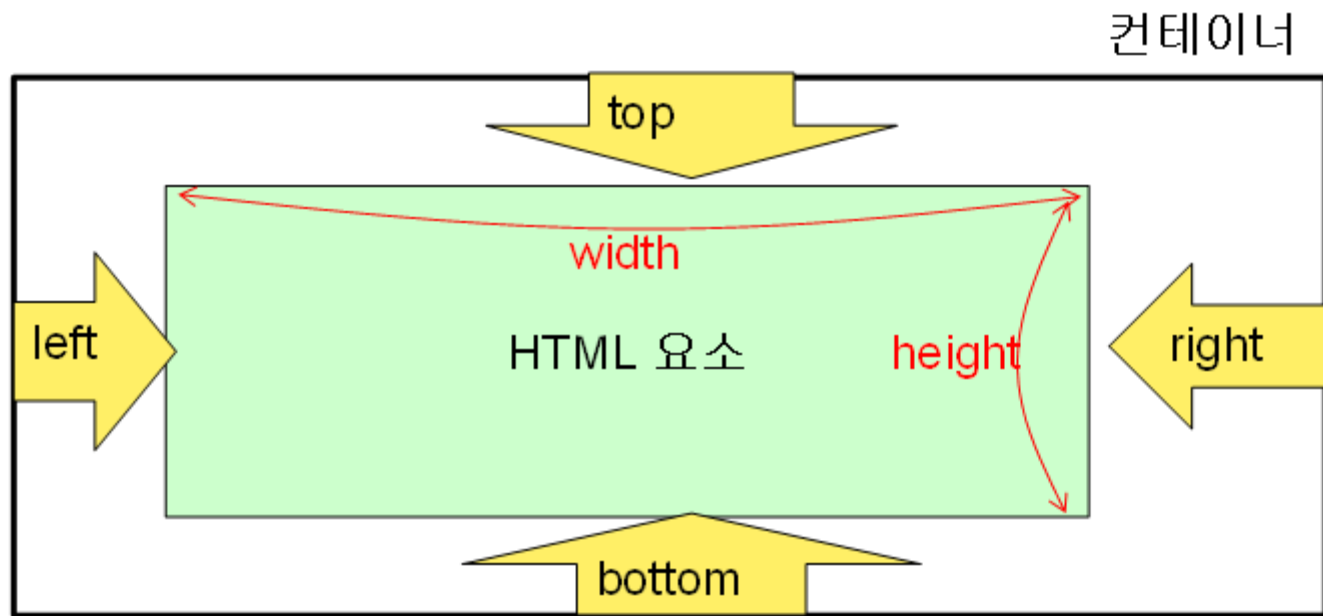
<body>
  <ul class="menubar">
    <li><a href="#">홈으로</a></li>
    <li><a href="#">회사 소개</a></li>
    <li><a href="#">제품 소개</a></li>
    <li><a href="#">질문과 대답</a></li>
    <li><a href="#">연락처</a></li>
  </ul>
</body>
</html>
```





요소의 위치

- top, bottom, left, right 속성으로 결정





위치 설정 방법

- 정적 위치 설정(static positioning)
 - 정상적인 흐름에 따른 배치
- 상대 위치 설정(relative positioning)
 - 정상적인 위치가 기준점이 된다.
- 절대 위치 설정(absolute positioning)
 - 컨테이너의 원점이 기준점이 된다.
- 고정 위치 설정(fixed positioning)
 - 윈도우의 원점이 기준점이 된다.





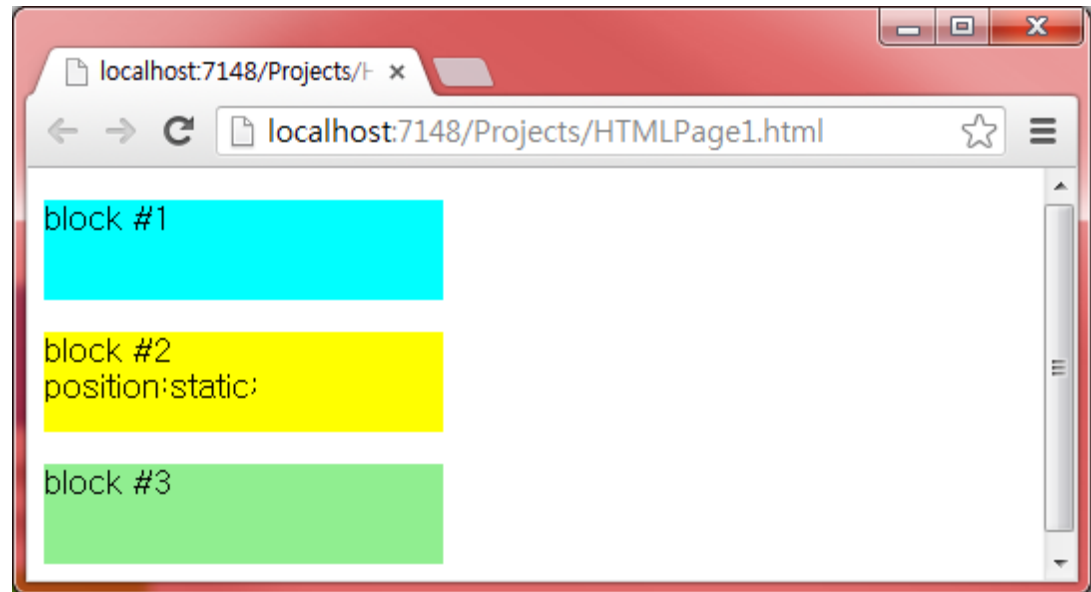
정적 위치 설정

- 정적 위치 설정(static positioning)
 - 블록 요소들은 박스처럼 상하로 쌓이게 되고 인라인 요소들은 한 줄에 차례대로 배치

```
<!DOCTYPE html>
<html>
<head>
  <style>
    #one {
      background-color: cyan;
      width: 200px;
      height: 50px;
    }
    #two {
      position: static;
      background-color: yellow;
      width: 200px;
      height: 50px;
    }
    #three {
      background-color: lightgreen;
      width: 200px;
      height: 50px;
    }
  </style>
```

예제

```
<body>
  <p id="one">block #1</p>
  <div id="two">
    block #2<br />
    position:static;<br />
  </div>
  <p id="three">block #3</p>
</body>
</html>
```





상대 위치 설정

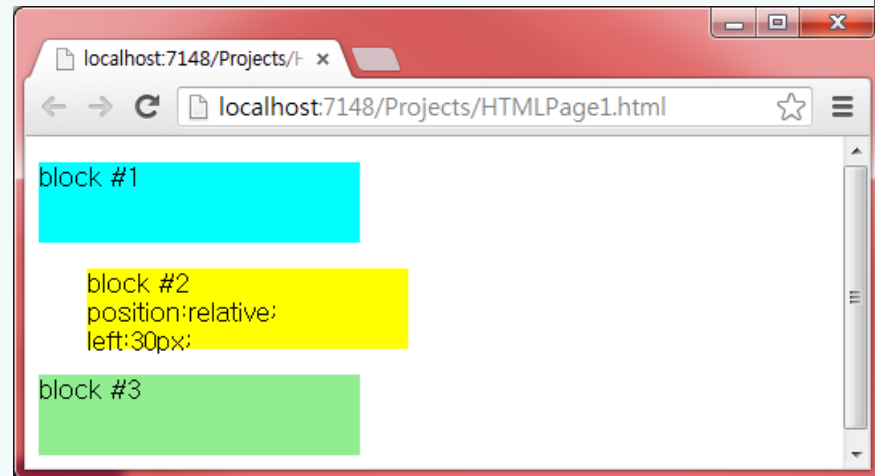
- 상대 위치 설정(relative positioning)
 - 정상적인 위치에서 상대적으로 요소가 배치

```
<style>
  #one {
    background-color: cyan;
    width: 200px;
    height: 50px;
  }

  #two {

    position: relative;
    left: 30px;
    background-color: yellow;
    width: 200px;
    height: 50px;
  }

  #three {
    background-color: lightgreen;
    width: 200px;
    height: 50px;
  }
</style>
```



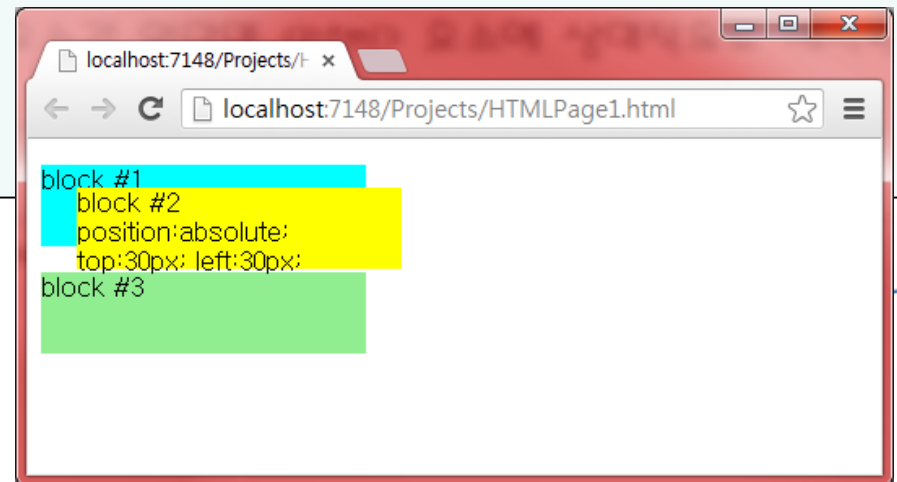


절대 위치 설정

- 절대 위치(absolute positioning)

- 전체 페이지를 기준으로 시작 위치에서 top, left, bottom, right 만큼 떨어진 위치에 배치

```
...  
#two {  
    position: absolute;  
  
    top: 30px;  
    left: 30px;  
  
    background-color: yellow;  
    width: 200px;  
    height: 50px;  
}  
...
```



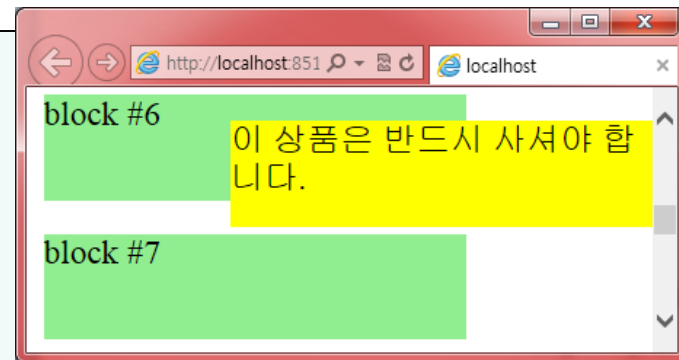


고정 위치 설정

- 고정 위치 설정(fixed positioning)

- 브라우저 윈도우에 상대적으로 요소의 위치를 잡는 것

```
<!DOCTYPE html>
<html>
<head>
  <style>
    p {
      background-color: lightgreen;
      width: 200px;
      height: 50px;
    }
    #two {
      background-color: yellow;
      position: fixed;
      top: 0px;
      right: 0px;
    }
  </style>
</head>
```

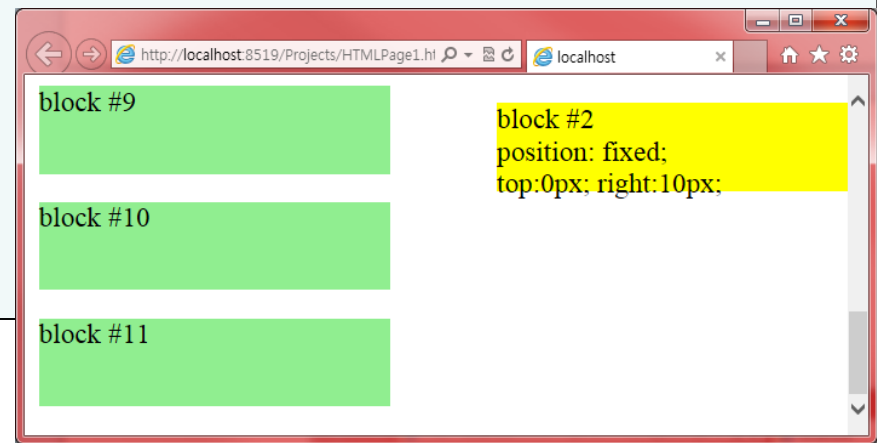
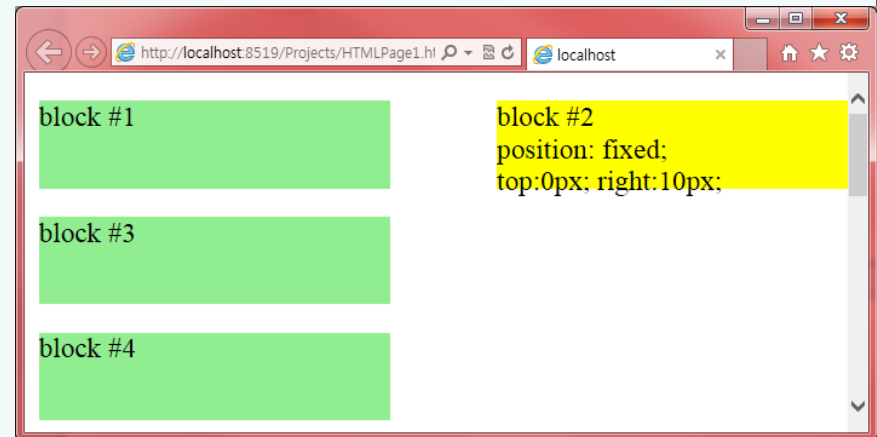




고정 위치 설정

```
<body>
  <p>block #1</p>
  <p id="two">
    block #2<br />
    position: fixed;<br />
    top:0px; right:10px;
  </p>

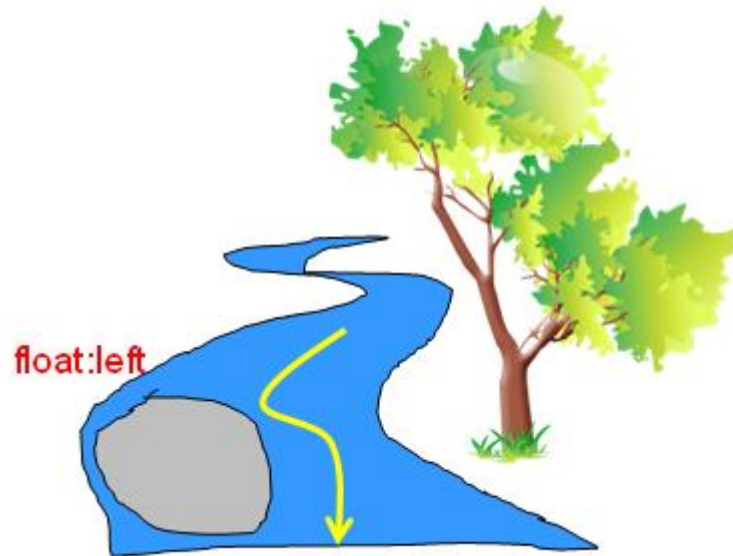
  <p>block #3</p>
  <p>block #4</p>
  <p>block #5</p>
  <p>block #6</p>
  <p>block #7</p>
  <p>block #8</p>
  <p>block #9</p>
  <p>block #10</p>
  <p>block #11</p>
</body>
</html>
```





float 속성

- 하나의 콘텐츠 주위로 다른 콘텐츠들이 물처럼 흘러가는 스타일 지정



예제

```
<!DOCTYPE html>
<html>
<head>
  <style>
    img.a {      float: left      }
  </style>
</head>

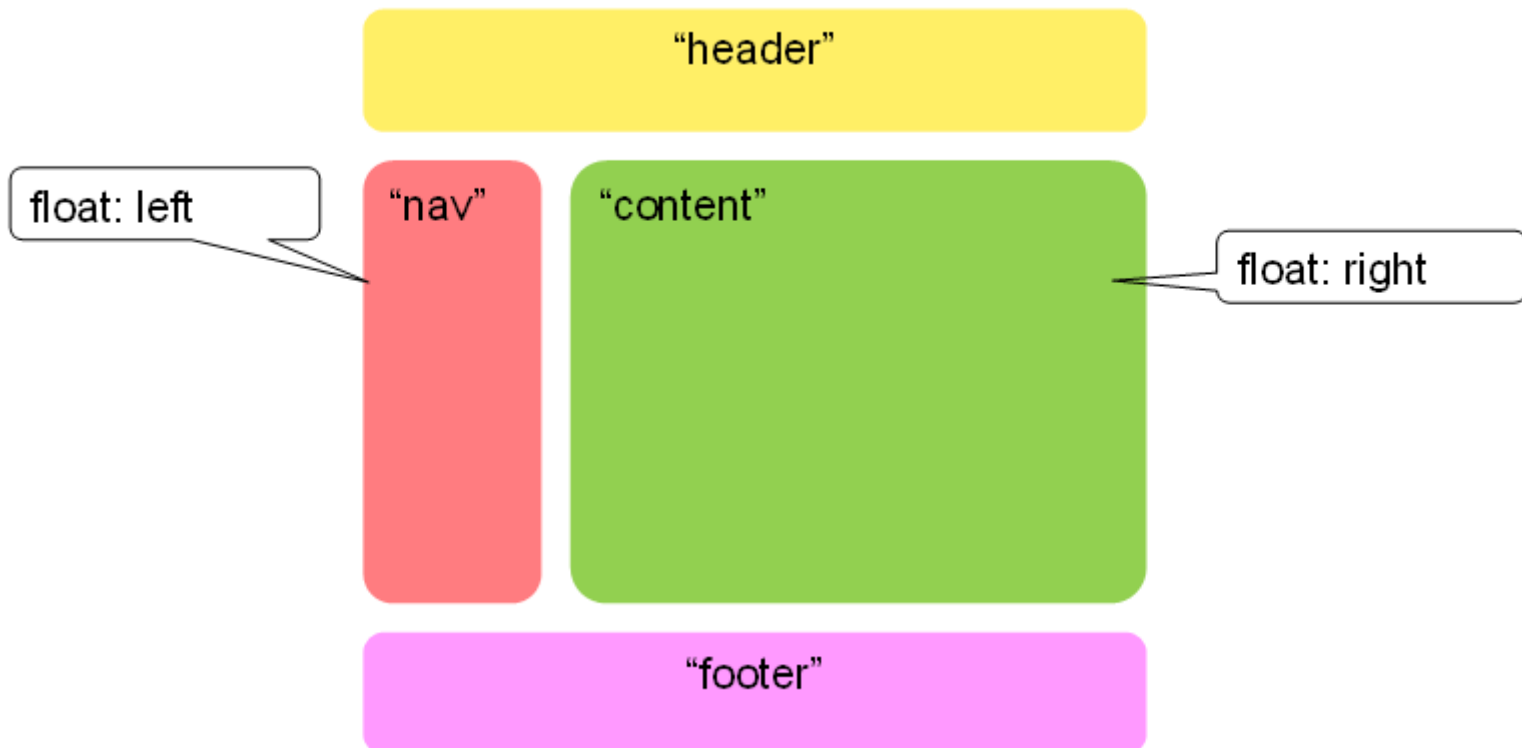
<body>
  
  <p>
    생활이 그대를 속일지라도
    슬퍼하거나 노여워 말라.
    ...
  </p>
</body>
</html>
```





float의 용도

- 레이아웃에 많이 사용된다.





clear 속성

- float 속성을 중단할 때 사용된다.

```
#footer {  
  clear: both;  
}
```

float: left

"content"

"nav"

float: right

"footer"

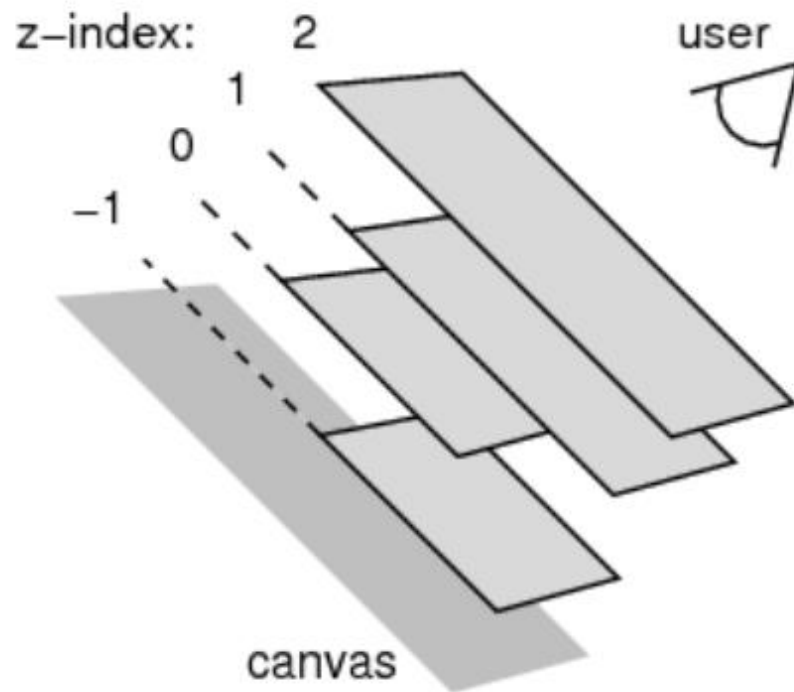
clear: both로
지정하면 빈칸을
채우지 않고
정상적으로
배치된다.





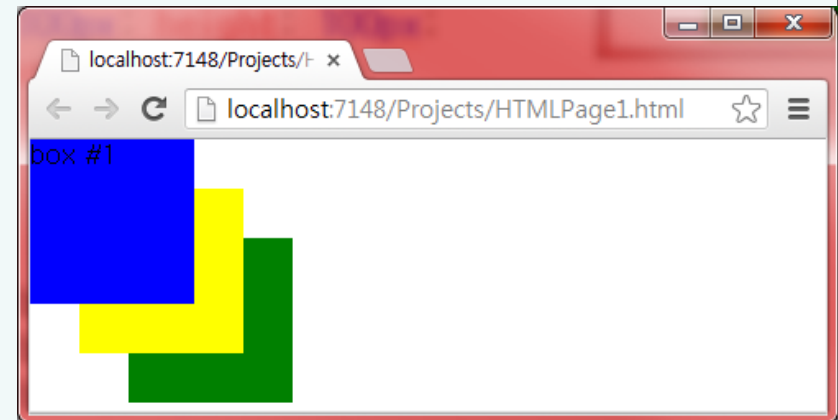
z-index

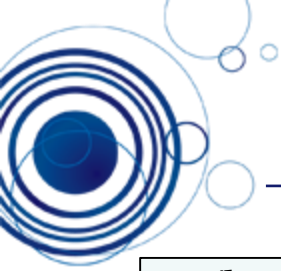
- 요소의 스택 순서를 지정



예제

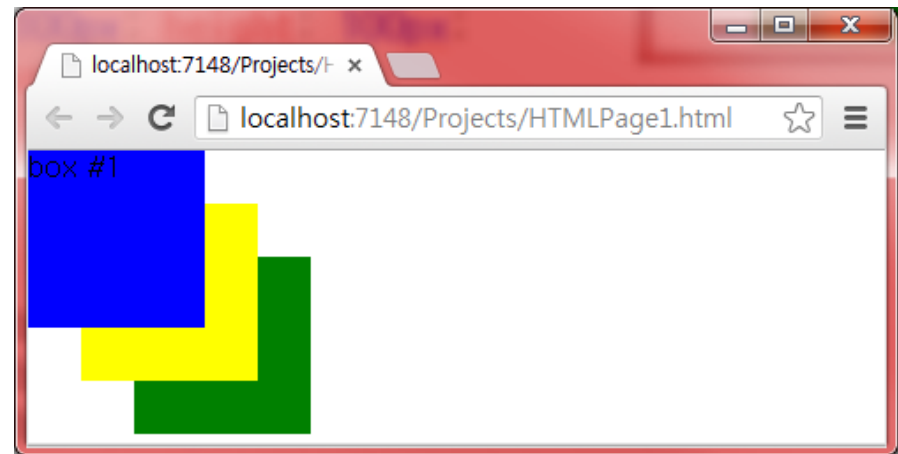
```
...  
<style>  
  #box1 {  
    position: absolute;  
    top: 0px; left: 0px;  
    width: 100px; height: 100px;  
    background: blue;  
    z-index: 200;  
  }  
  
  #box2 {  
    position: absolute;  
    top: 30px; left: 30px;  
    width: 100px; height: 100px;  
    background: yellow;  
    z-index: 100;  
  }  
  
  #box3 {  
    position: absolute;  
    top: 60px; left: 60px;  
    width: 100px; height: 100px;  
    background: green;  
    z-index: 0;  
  }  
</style>
```





예제

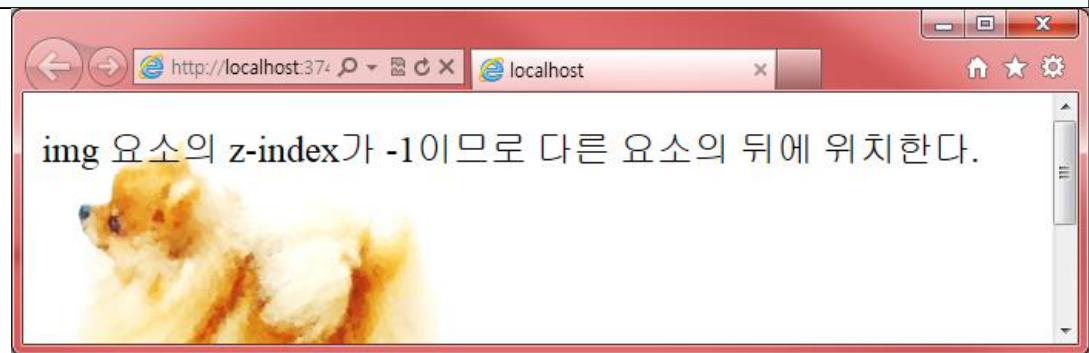
```
</head>
<body>
  <div id="box1">box #1 </div>
  <div id="box2">box #2 </div>
  <div id="box3">box #3 </div>
</body>
</html>
```



예제

```
<!DOCTYPE html>
<html>
<head>
  <style>
    img {
      position: absolute;
      left: 0px;
      top: 0px;
      z-index: -1;
    }
  </style>
</head>

<body>
  
  <p>img 요소의 z-index가 -1이므로 다른 요소의 뒤에 위치한다. </p>
</body>
</html>
```





요소의 크기 지정

- width, height – 요소의 크기
- min-width, min-height: 요소의 최소 크기
- max-width, max-height: 요소의 최대 크기





예제

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <style>
```

```
    p {
```

```
      min-width: 100px;
```

```
      min-height: 100px;
```

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

```
    }
```

```
  </style>
```

```
</head>
```

```
<body>
```

```
  <p>
```

이 요소는 min-width:100px;와 min-height:100px;으로 설정되었습니다.
요소의 크기는 width와 height 속성으로 결정된다.

만약 개발자가 요소의 width와 height를 명확하게 설정하지 않으면 브라우저가
요소 안의 콘텐츠의 크기를 계산하여서 요소의 크기를 결정한다.

```
  </p>
```

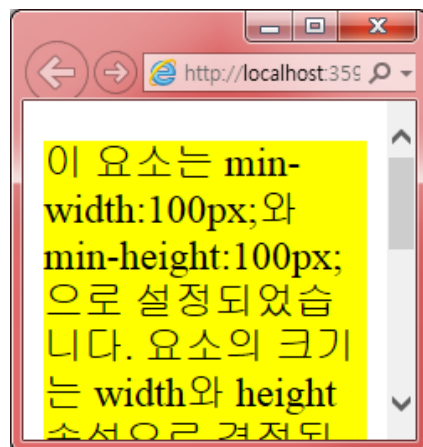
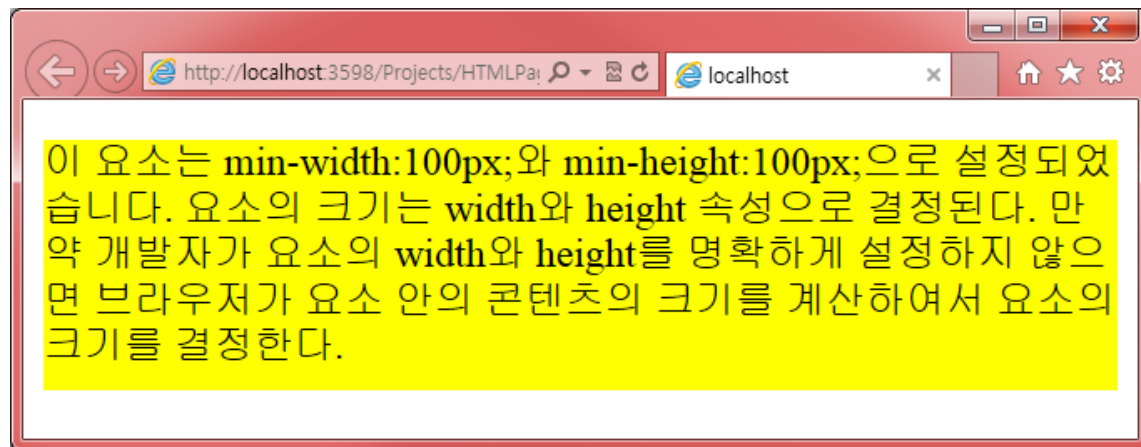
```
</body>
```

```
</html>
```





실행 결과

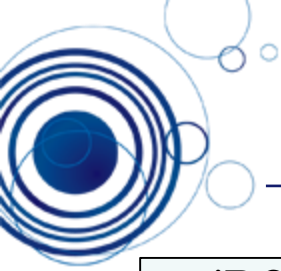




overflow 속성

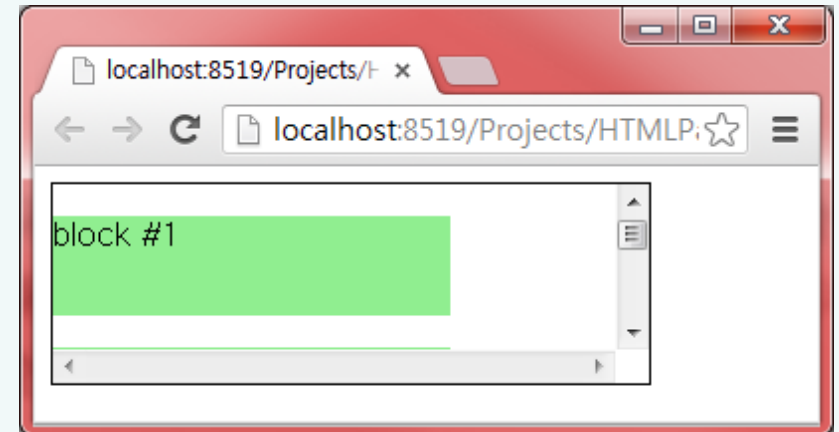
- overflow 속성: 자식 요소가 부모 요소의 범위를 벗어났을 때, 어떻게 처리할 것인지를 지정
 - hidden – 부모 영역을 벗어나는 부분을 보이지 않게 한다.
 - scroll – 부모 영역을 벗어나는 부분을 스크롤 할 수 있도록 한다.
 - auto – 자동으로 스크롤바가 나타난다.





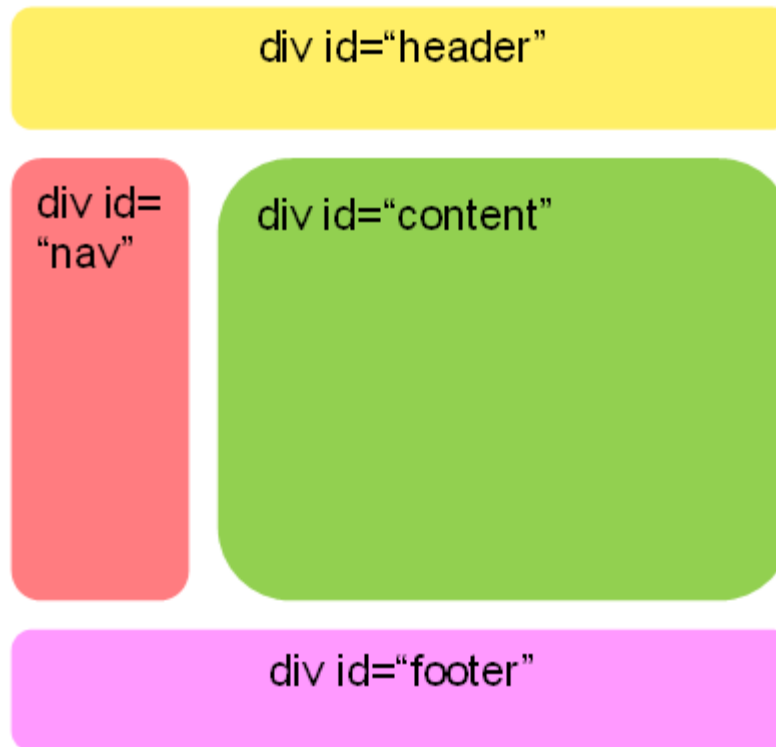
예제

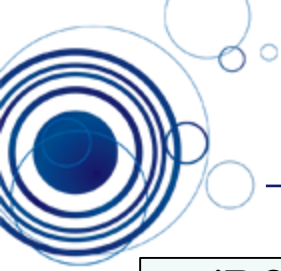
```
<!DOCTYPE html>
<html>
<head>
  <style>
    p {
      background-color: lightgreen;
      width: 200px;
      height: 50px;
    }
    #target {
      border: 1px solid black;
      width: 300px;
      height: 100px;
      overflow: scroll;
    }
  </style>
</head>
<body>
<div id=target>
  <p>block #1</p>
  <p>block #2</p>
  <p>block #3</p>
  <p>block #4</p>
  <p>block #5</p>
</div>
</body>
</html>
```





<div>를 이용한 레이아웃





예제

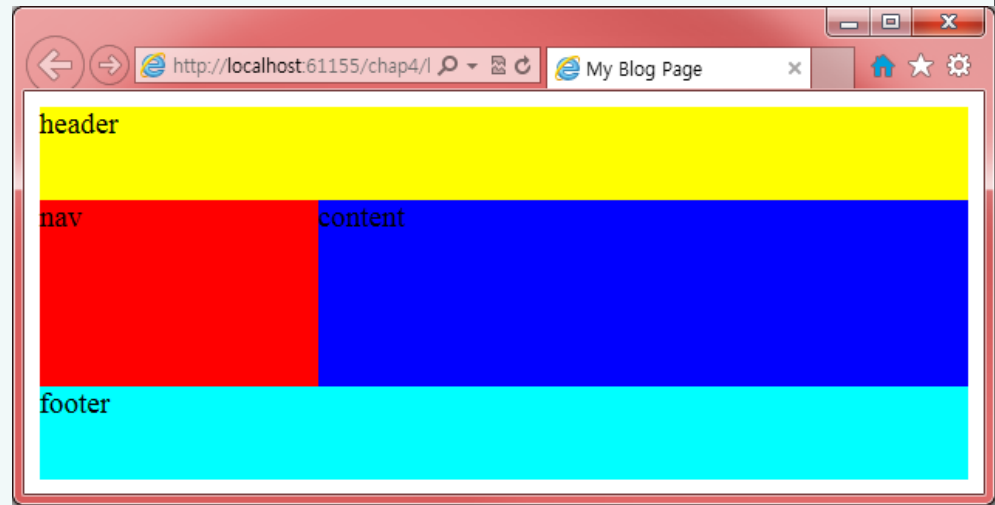
```
<!DOCTYPE html>
<html>
<head>
  <title>My Blog Page</title>
  <style>
#header {
  background-color: yellow;
  width: 100%;
  height: 50px;
}

#nav {
  width: 30%;
  background-color: red;
  height: 100px;
  float: left;
}
```



예제

```
#content {  
    width: 70%;  
    background-color: blue;  
    float: right;  
    height: 100px;  
}  
#footer {  
    background-color: aqua;  
    width: 100%;  
    height: 50px;  
    clear: both;  
}  
</style>  
</head>  
<body>  
    <div id="wrapper">  
        <div id="header"> header </div>  
        <div id="nav"> nav </div>  
        <div id="content"> content </div>  
        <div id="footer"> footer </div>  
    </div>  
</body>  
</html>
```





시맨틱 요소 레이아웃

div id="header"

div id="nav"

div id="content"

div id="footer"

<header>

<nav>

<section id="content">

<footer>





시맨틱 요소

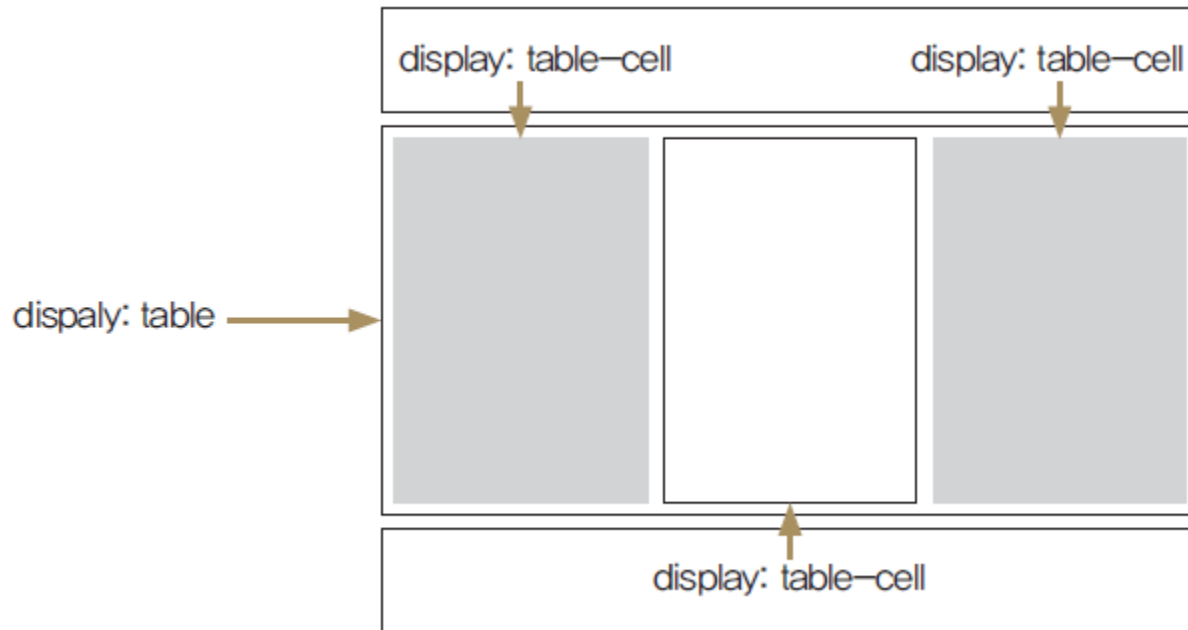
태그	설명
<code><header></code>	문서의 머리말(header)
<code><hgroup></code>	<code><h1></code> 에서 <code><h6></code> 요소들의 그룹
<code><nav></code>	내비게이션 링크
<code><article></code>	문서의 내용이나 블로그의 포스트
<code><section></code>	문서의 섹션을 의미한다.
<code><aside></code>	사이드바와 같이 옆에 위치하는 내용
<code><footer></code>	문서의 꼬리말/footer)
<code><figure></code>	그림이나 도표
<code><time></code>	날짜와 시간을 표시

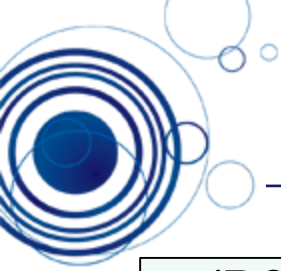




table-cell 속성

- display 속성에 table-cell을 하면 자식 요소들을 테이블의 셀처럼 배치하라는 의미가 된다.





CSS3 효과: 투명도

```
<!DOCTYPE html>

<html>
<head>
  <style>
    img {      opacity: 0.4;    }
    img:hover {  opacity: 1.0;  }
  </style>
</head>
<body>

  <h1>Opacity 속성</h1>
  
  
</body>
</html>
```



CSS3 효과: 가시성

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <style>
```

```
    #a {
```

```
      visibility: hidden;
```

```
      border: 1px dotted red;
```

```
    }
```

```
    #b {
```

```
      visibility: visible;
```

```
      border: 1px dotted red;
```

```
    }
```

```
  </style>
```

```
</head>
```

```
<body>
```

```
  <h1>Visibility 속성</h1>
```

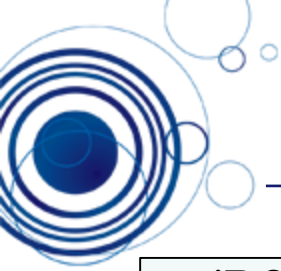
```
  
```

```
  
```

```
</body>
```

```
</html>
```

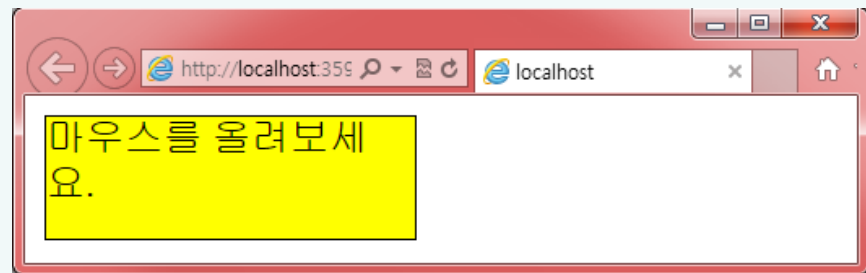




CSS3: 전환

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width: 100px;
      height: 50px;
      border: 1px solid black;
      background: yellow;
      transition: width 5s;
    }

    div:hover {
      width: 200px;
    }
  </style>
</head>
<body>
  <div>마우스를 올려보세요.</div>
</body>
</html>
```



CSS3: 전환

```
<!DOCTYPE html>
<html>

<head>
  <style>
    p {
      width: 100px;
      height: 50px;
      border: 1px solid black;
      background: yellow;
      transition: width 5s height 5s border 5s, transform 5s;
      -webkit-transition: width 5s, height 5s, border 5s, -webkit-transform 5s;
    }

    p:hover {
      width: 200px;
      height: 100px;
      border: 10px solid red;
      transform: rotate(180deg);
      -webkit-transform: rotate(180deg);
    }
  </style>
</head>
<body>
  <p>마우스를 올려보세요.</p>
</body>
</html>
```

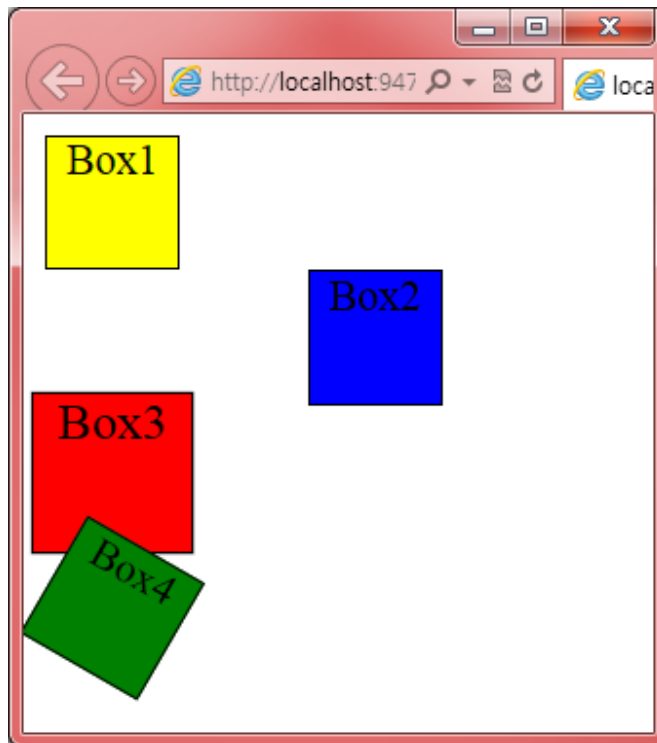


실행결과 보기



CSS3 변환

- 도형을 이동, 크기 변환, 회전
- 도형의 크기나 형태, 위치를 변환
- 2차원 또는 3차원적으로 변환





transform 속성

- transform: translate(10px, 10px) - 평행이동
- transform: rotate(45deg) - 회전
- transform: scale(2, 1.2) - 크기변환
- transform: skew(20deg, 10deg) - 비틀기 변환
- transform: matrix() - 일반적인 변환



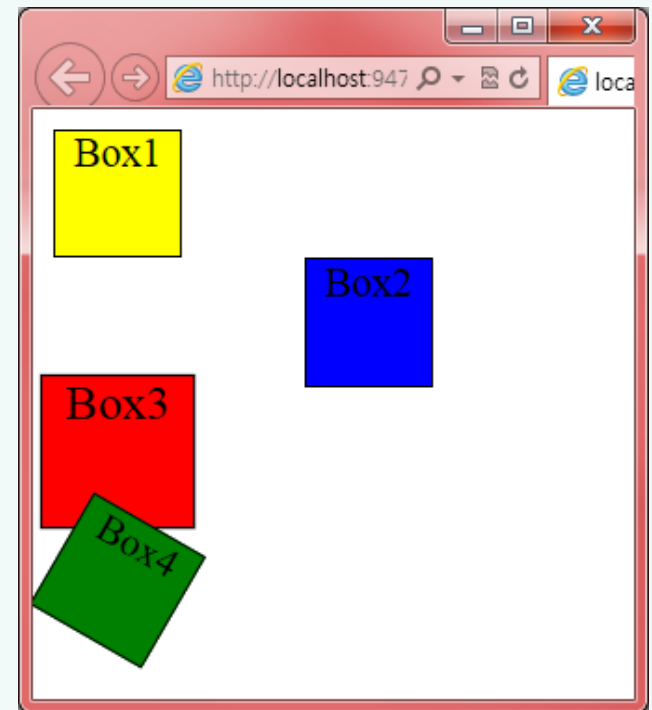
CSS3: 전환

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      width: 50px;
      height: 50px;
      background-color: yellow;
      border: 1px solid black;
      text-align: center;
    }

    div#box2 {
      transform: translate(100px, 0px);
      background-color: blue;
    }

    div#box3 {
      transform: scale(1.2, 1.2);
      background-color: red;
    }

    div#box4 {
      transform: rotate(30deg);
      background-color: green;
    }
  </style>
</head>
```



CSS3: 전환

```
<body>
```

```
  <div id="box1">Box1</div>
```

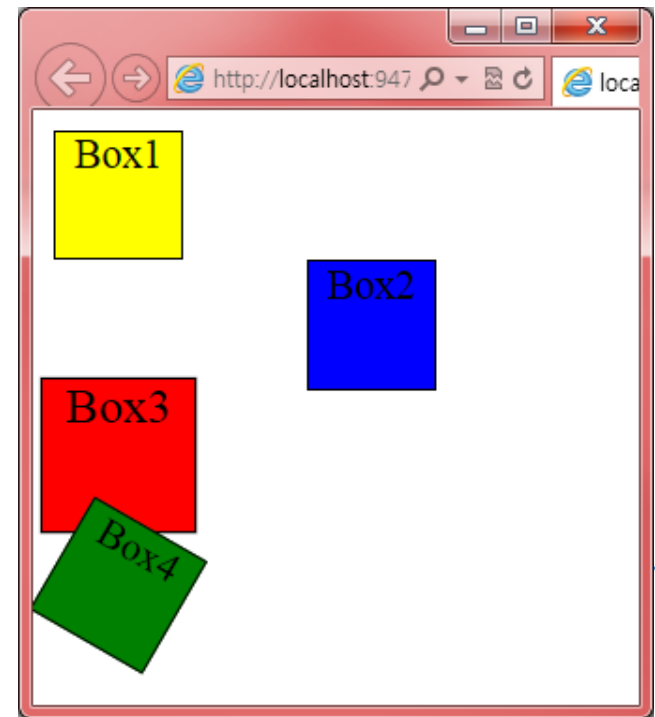
```
  <div id="box2">Box2</div>
```

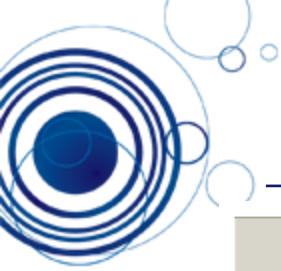
```
  <div id="box3">Box3</div>
```

```
  <div id="box4">Box4</div>
```

```
</body>
```

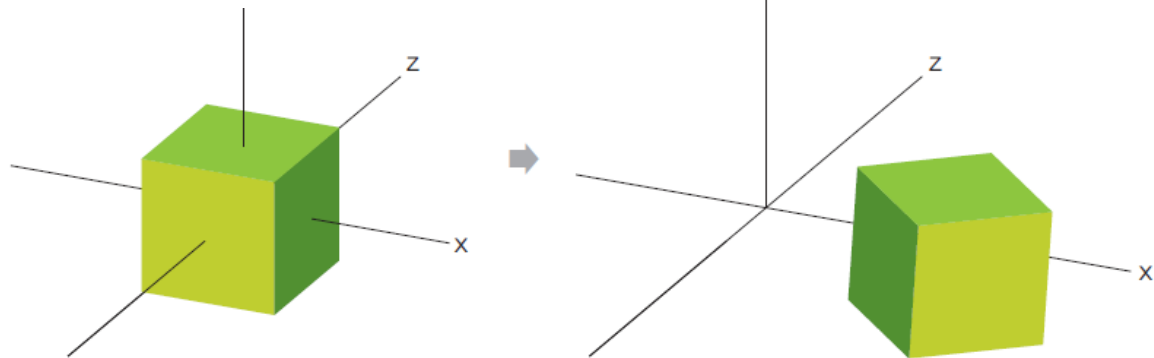
```
</html>
```





CSS3: 3차원 변환

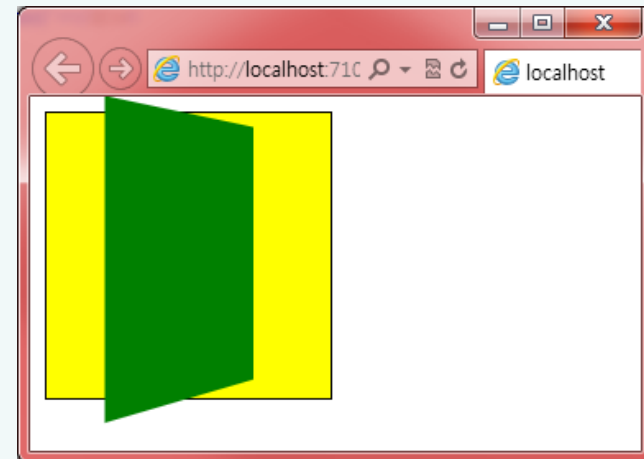
메서드	설명
<code>translate3d(x,y,z)</code>	3차원 평행 이동
<code>translateX(x)</code>	3차원 평행 이동(x축)
<code>translateY(y)</code>	3차원 평행 이동(y축)
<code>translateZ(z)</code>	3차원 평행 이동(z축)
<code>scale3d(x,y,z)</code>	3차원 크기 변환
<code>scaleX(x)</code>	3차원 크기 변환(x축)
<code>scaleY(y)</code>	3차원 크기 변환(y축)
<code>scaleZ(z)</code>	3차원 크기 변환(z축)
<code>rotate3d(x,y,z,angle)</code>	3차원 회전 변환
<code>rotateX(angle)</code>	3차원 회전 변환(x축)
<code>rotateY(angle)</code>	3차원 회전 변환(y축)
<code>rotateZ(angle)</code>	3차원 회전 변환(z축)
<code>perspective(n)</code>	원근 변환



예제

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div {
      background-color: green;
      height: 150px;

      width: 150px;
    }
    .container {
      background-color: yellow;
      border: 1px solid black;
    }
    .transformed {
      backface-visibility: visible;
      transform-origin: 50% 42%;
      transform: perspective(500px) rotateY(59deg) rotateX(0deg);
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="transformed"></div>
  </div>
</body>
</html>
```





CSS3 애니메이션

키프레임을 지정한다. 애니메이션의 이름을 지정한다.

↓ ↓
`@keyframes myanim`

{

0% {left:0px; top:0px;}

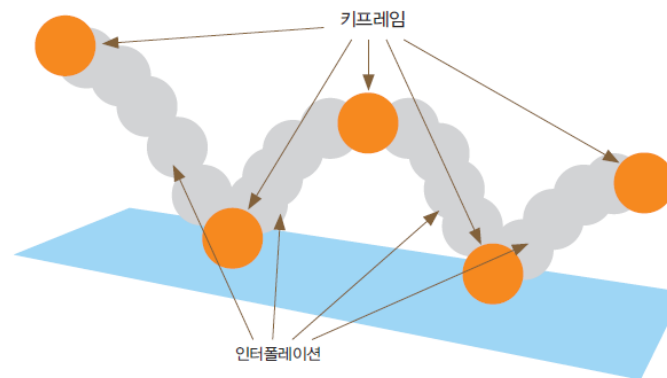
25% {left:100px; top:0px;} ← 25%일 때의 키프레임 지정

50% {left:200px; top:0px;}

75% {left:100px; top:0px;}

100% {left:0px; top:0px;}

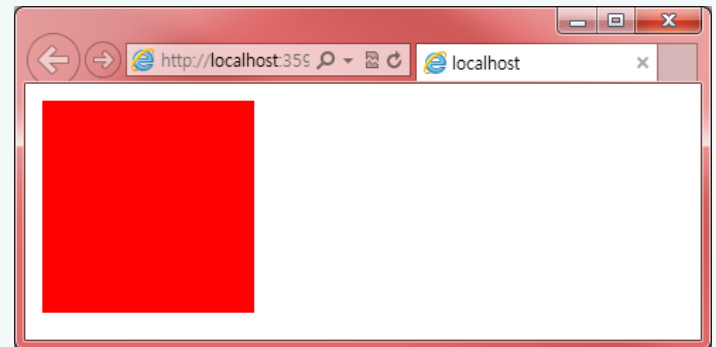
}



예제

```
<!DOCTYPE html>

<html>
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      background: red;
      position: relative;
      animation: 2s myanim;
      animation-iteration-count: 10;
    }
    @keyframes myanim
    {
      0%   {left:0px; top:0px;}
      25%  {left:100px; top:0px;}
      50%  {left:200px; top:0px;}
      75%  {left:100px; top:0px;}
      100% {left:0px; top:0px;}
    }
  </style>
</head>
<body>
  <div></div>
</body>
</html>
```



실습 2

- 아래와 같은 페이지를 만들어 본인의 github에 올려주세요.
- 다음주 실습과 동시에 채점합니다.
- 아래 페이지는 본인 선호도에 맞게 그림, 텍스트, 크기, 색상은 변경가능합니다.

