

# CSS와 박스 모델

## 블록 레벨 요소

- 요소를 삽입했을 때 혼자 한 줄을 차지하는 요소
  - 요소의 너비가 100%
- 예) <div>, <p> 등



### Do it! 블록 레벨 요소 사용하기

예제 파일 08\block.html

(... 생략 ...)

```
<h1>시간이란...</h1>
```

```
<div>내일 죽을 것처럼 <p class="accent">오늘</p>을 살고</div>
```

```
<p>영원히 살 것처럼 <br>내일을 꿈꾸어라.</p>
```

(... 생략 ...)

시간이란...

내일 죽을 것처럼

오늘

을 살고

영원히 살 것처럼  
내일을 꿈꾸어라.

## 인라인 레벨 요소

- 줄을 차지하지 않는 요소
  - 화면에 표시되는 콘텐츠만큼만 영역을 차지하고 나머지 공간에는 다른 요소가 올 수 있음
- 예) <img>, <strong> 등

### Do it! 인라인 레벨 요소

예제 파일 08\inline.html

(... 생략 ...)

```
<h1>시간이란...</h1>
```

```
<div>내일 죽을 것처럼 <span class="accent">오늘</span>을 살고</div>
```

```
<p>영원히 살 것처럼 <br>내일을 꿈꾸어라.</p>
```

(... 생략 ...)

시간이란...

내일 죽을 것처럼 오늘을 살고

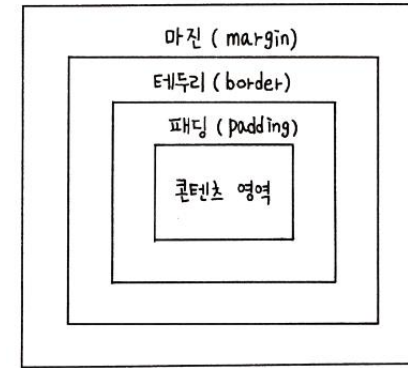
영원히 살 것처럼  
내일을 꿈꾸어라.



# CSS와 박스 모델

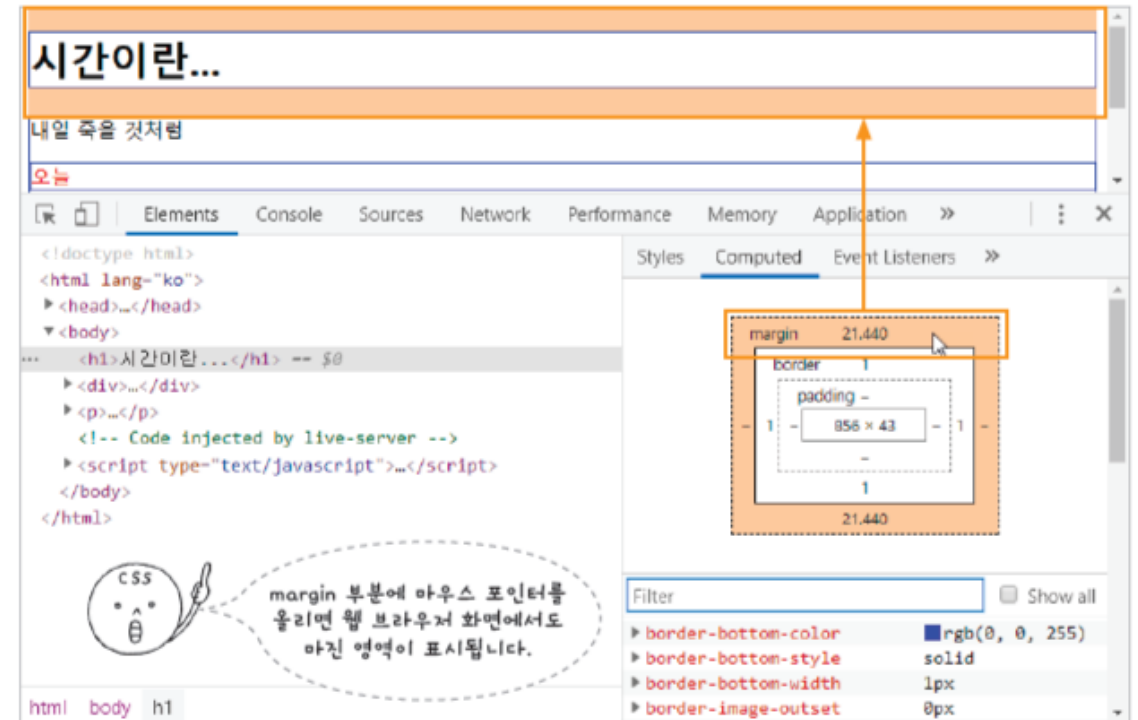
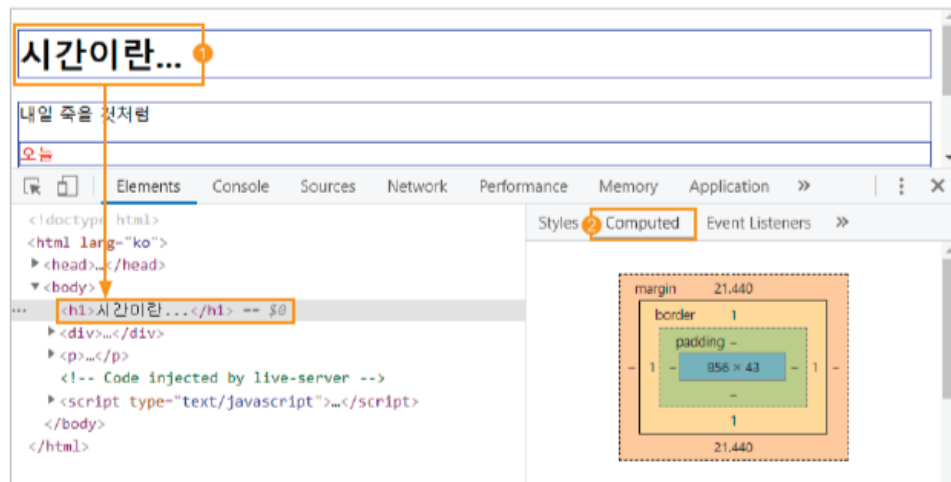
## 박스 모델

실제 콘텐츠 영역, 패딩(padding), 박스의 테두리(border), 그리고 마진(margin) 등의 요소로 구성됨.



## 개발자 도구 창에서 박스 모델 확인 가능

- ① 웹 브라우저에서 웹 문서 열기
- ② 박스 모델 확인할 부분을 마우스 오른쪽 버튼으로 누른 뒤 [검사] 선택
- ③ 개발자 도구 창 위에서 [Computed] 탭 클릭
- ④ 해당 부분의 박스 모델이 그림으로 표시됨



# CSS와 박스 모델

## width, height 속성

실제 콘텐츠 영역의 크기 지정

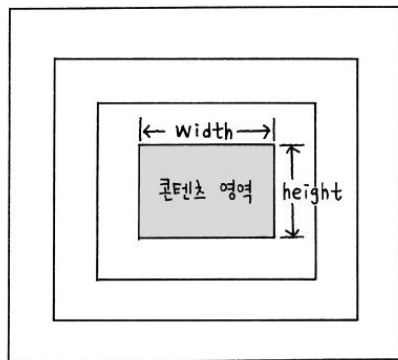


표 8-1 width와 height의 속성값

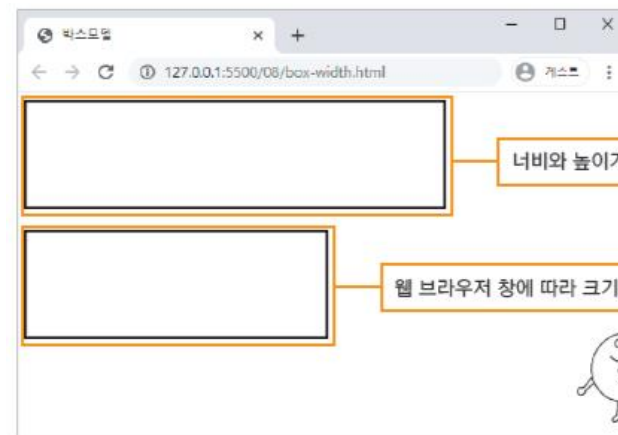
종류	설명
<크기>	너비나 높이의 값을 px이나 em 단위로 지정합니다.
<백분율>	박스 모델을 포함하는 부모 요소를 기준으로 너비값이나 높이값을 백분율(%)로 지정합니다.
auto	박스의 너비값과 높이값이 콘텐츠 양에 따라 자동으로 결정됩니다. 기본값입니다.



### Do it! 박스 모델 너비 지정하기

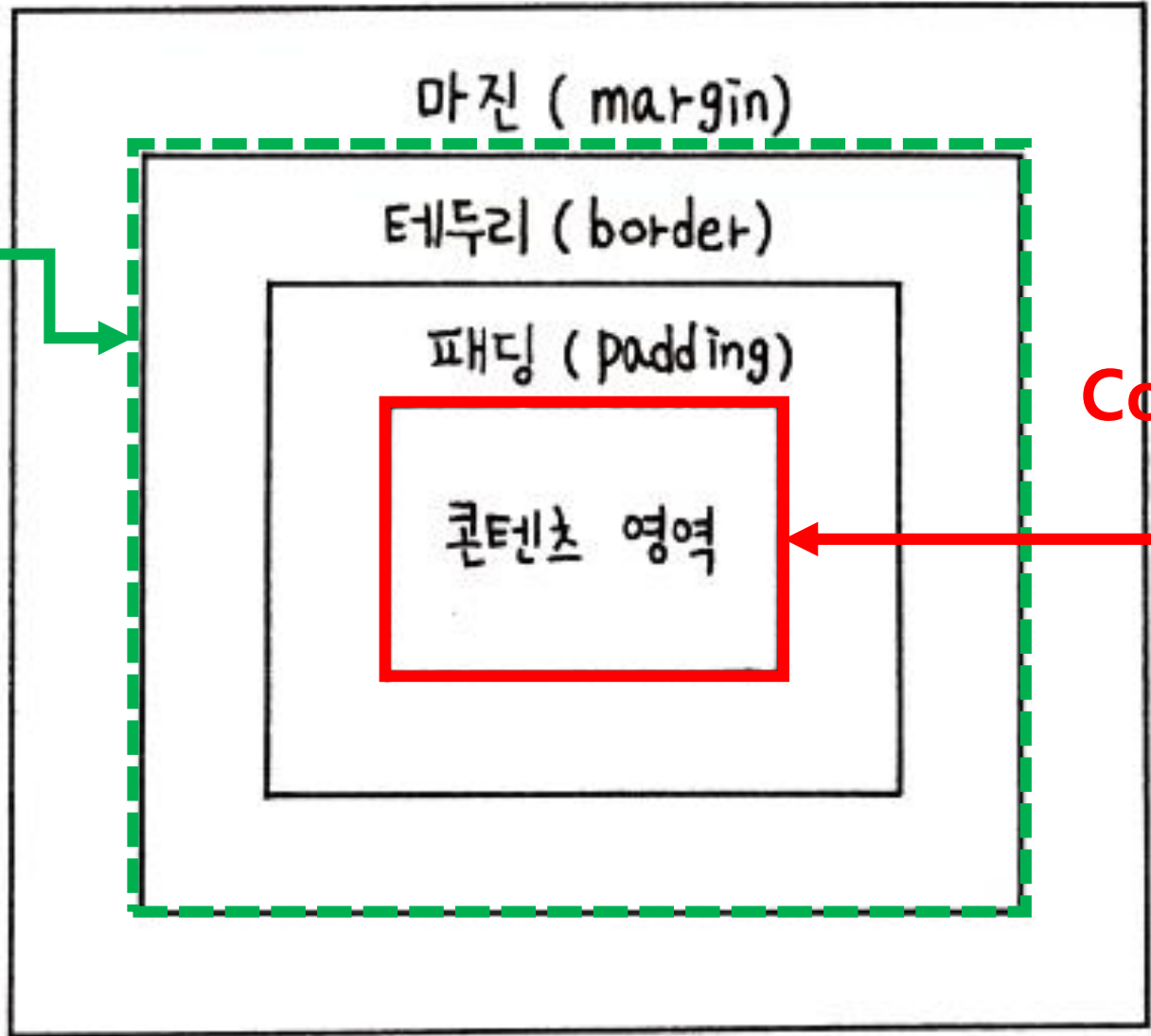
```
(... 생략 ...)
.box1 {
  width: 400px; /* 고정 너비 */
  height: 100px; /* 고정 높이 */
}
.box2 {
  w : 50%; /* 가변 너비 - 웹 브라우저의 50% */
  h : 100px; /* 고정 높이 */
}
</style>

.....
<div class="box1"></div>
<div class="box2"></div>
(... 생략 ...)
```



웹 브라우저 창을  
줄이면 박스가  
어떻게 변할까요?

Border-box 구간



Contents-box 구간

# CSS와 박스 모델

## box-sizing 속성

실제 박스 모델의 너비를 계산할 때 어디까지 포함할지 결정하는 속성

종류	설명
border-box	테두리까지 포함해서 너비값을 지정합니다.
content-box	콘텐츠 영역만 너비값을 지정합니다. 기본값입니다.



**Do it!** 박스 모델에 그림자 효과 주기

예제 파일 08\box-shadow.html

(... 생략 ...)

```
.box1 { box-shadow: 2px -2px 5px 0px; } ❶
```

```
.box2 { box-shadow: 5px 5px 15px 5px blue; } ❷
```

</style>

(... 생략 ...)

모든 그림자



## box-shadow 속성

선택한 요소에 그림자 효과 내기

수평 거리와 수직 거리는 필수,  
기타 속성은 옵션

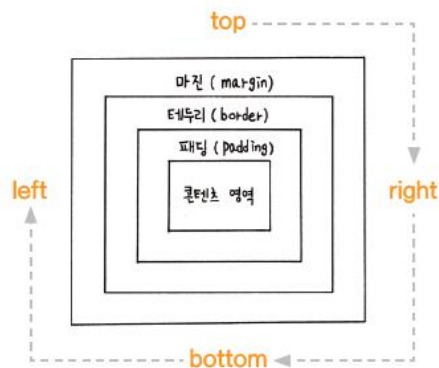
기본형 box-shadow: <수평 거리> <수직 거리> <흐림 정도> <번짐 정도> <색상> inset

종류	설명
<수평 거리>	그림자가 가로로 얼마나 떨어져 있는지를 나타냅니다. 양숫값은 요소의 오른쪽에, 음숫값은 요소의 왼쪽에 그림자를 만듭니다. 필수 속성입니다.
<수직 거리>	그림자가 세로로 얼마나 떨어져 있는지를 나타냅니다. 양숫값은 요소의 아래쪽에, 음숫값은 요소의 위쪽에 그림자를 만듭니다. 필수 속성입니다.
<흐림 정도>	이 값을 생략하면 0을 기본값으로 하여 진한 그림자를 표시합니다. 이 값이 커질수록 부드러운 그림자를 표시하며, 음숫값은 사용할 수 없습니다.
<번짐 정도>	양숫값을 사용하면 모든 방향으로 그림자가 퍼져서 박스보다 그림자가 크게 표시됩니다. 반대로 음숫값은 모든 방향으로 그림자가 축소되어 보입니다. 기본값은 0입니다.
<색상>	한 가지만 지정할 수도 있고, 공백으로 구분해서 여러 개의 색상을 지정할 수도 있습니다. 기본값은 현재 검은색입니다.
inset	이 키워드를 함께 표시하면 안쪽 그림자로 그림니다.

# 테두리 스타일 지정하기

## 박스 모델의 값 지정 방향









- 4개 방향의 값을 한꺼번에 지정할 때는 방향 순서를 지켜야 함
- top → right → bottom → left



## border-style 속성

- 기본 값이 none → 화면에 테두리 표시안됨
- 테두리를 그리기 위해서는 맨 먼저 테두리 스타일부터 지정

기본형 border-style: none | hidden | dashed | dotted | double | groove | inset | outset | ridge | solid

종류	설명	예시
none	테두리가 없습니다. 기본값입니다.	
hidden	테두리를 감춥니다. 표에서 border-collapse: collapse일 경우 다른 테두리도 표시되지 않습니다.	
solid	테두리를 실선으로 표시합니다.	
dotted	테두리를 점선으로 표시합니다.	
dashed	테두리를 짧은 직선으로 표시합니다.	
double	테두리를 이중선으로 표시합니다. 두 선 사이의 간격이 border-width값이 됩니다.	
groove	테두리를 창에 조각한 것처럼 표시합니다. 홈이 파인 듯 입체 느낌이 납니다.	
inset	표에서 border-collapse: separate일 경우 전체 박스 테두리가 창에 박혀 있는 것처럼 표시되고, 표에서 border-collapse: collapse일 경우 groove와 똑같이 표시됩니다.	
outset	표에서 border-collapse: separate일 경우 전체 박스 테두리가 창에서 튀어나온 것처럼 표시되고, 표에서 border-collapse: collapse일 경우 ridge와 똑같이 표시됩니다.	
ridge	테두리를 창에서 튀어나온 것처럼 표시합니다.	

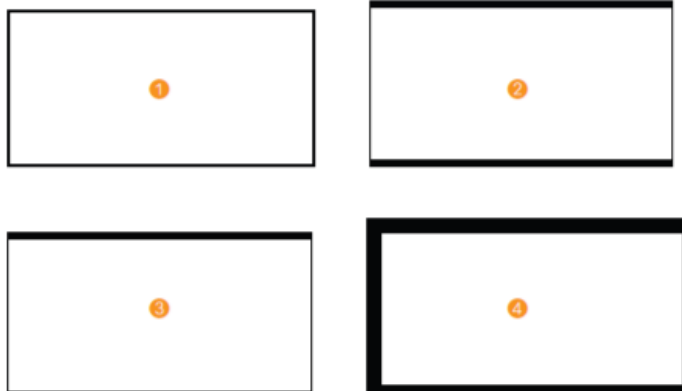
# 테두리 스타일 지정하기

## border-width 속성

테두리 두께 지정

thin < medium < thick 순으로 두꺼워짐

기본형 border-width: <크기> | thin | medium | thick



### Do it! 박스 모델의 테두리 두께 지정하기

(... 생략 ...)

```
#box1 { border-width: 2px; } ①
```

```
#box2 { border-width: thick thin; } ②
```

```
#box3 { border-width: thick thin thin; } ③
```

```
#box4 { border-width: 10px 5px 5px 10px; } ④
```


(... 생략 ...)

- ① 4개 방향 테두리 모두 같은 값이 적용됨
- ② top → right → bottom으로 적용하다가  
마지막 left 속성값이 빠짐.  
left 속성값은 마주 보는 right 속성값과 똑같이 적용됨
- ③ 첫 번째 값인 thick이 top, bottom 값이 되고,  
두 번째 값 thin은 left, right 값이 됨
- ④ top → right → bottom → left 순으로 적용됨

# 테두리 스타일 지정하기

## border-color 속성

테두리 색상 지정

 **Do it!** 박스 모델의 테두리 색상 지정하기 예제 파일 08\border-3.html

```
(... 생략 ...)  
#box1 {  
    border-color: red;           /* 전체 테두리 빨강 */  
}  
#box2 {  
    border-top-color: blue; /* 위쪽 테두리 파랑 */  
    border-left-color: red;  /* 왼쪽 테두리 빨강 */  
}  
(... 생략 ...)
```

doit:김윤



## border 속성

- 테두리 스타일과 두께, 색상 등을 묶어 표기
- border-top이나 border-right처럼 방향을 함께 써서 4개 방향의 스타일을 따로 지정할 수 있음
- 순서는 상관없음

 **Do it!** 테두리 스타일 묶어서 지정하기 예제 파일 08\border-4.html

```
(... 생략 ...)  
h1 {  
    padding-bottom: 5px;  
    border-bottom: 3px solid rgb(75, 70, 70); /* 아래쪽 테두리 3px 회색 실선 */  
}  
p {  
    padding: 10px;  
    border: 3px dotted blue; /* 모든 테두리를 3px 파란색 점선 */  
}  
(... 생략 ...)
```

bottom:김윤

## 박스 모델

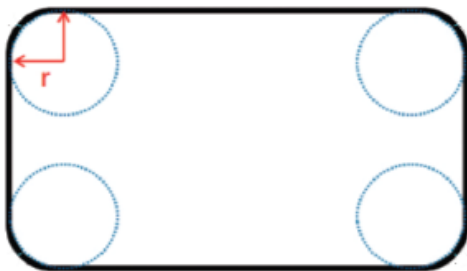
박스 모델은 실제 콘텐츠 영역, 박스와 콘텐츠 영역 사이의 여백인 패딩(padding), 박스의 테두리(border), 그리고 여러 박스 모델 간의 여백인 마진(margin) 등의 요소로 구성됩니다.



# 테두리 스타일 지정하기

## border-radius 속성

- 박스 모델의 테두리를 둥글게 처리
- 박스 모델의 꼭짓점 부분에 원(반지름  $r$ )이 있다고 가정해서 둥글게 처리



기본형 `border-radius: <크기> | <백분율>`

종류	설명
<크기>	반지름 크기를 px, em의 단위와 함께 수치로 표시합니다
<백분율>	현재 요소의 크기를 기준으로 비율(%)로 지정합니다.



**Do it!** 이미지의 꼭짓점을 둥글게 표시하기

(... 생략 ...)

```
#round { border-radius: 25px; }  
</style>
```

.....

```
  

```

(... 생략 ...)



**Do it!** 이미지를 원형으로 표시하기

(... 생략 ...)

```
#circle { border-radius: 50%; }  
</style>
```

.....

```
  

```

(... 생략 ...)

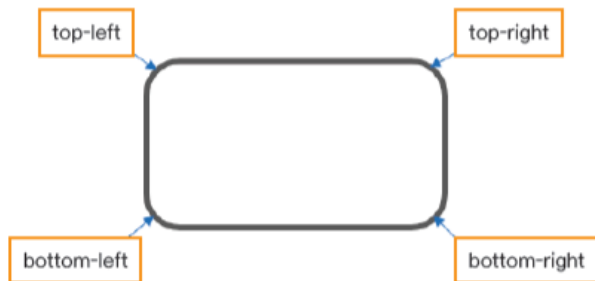


# 테두리 스타일 지정하기

## border-radius 속성

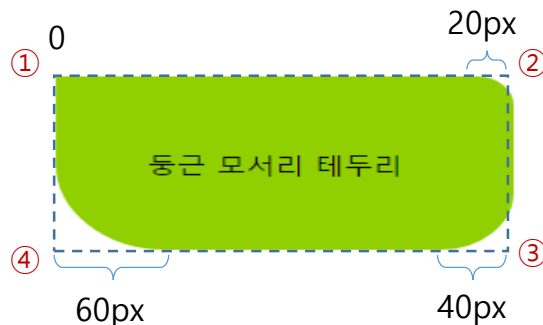
border와 radius 사이에 위치를 나타내는 예약어를 사용하면 꼭짓점마다 다르게 처리 가능

각 꼭짓점을 나타내는 예약어



border-radius : 0px 20px 40px 60px;

(① ~ ④의 시계방향 순으로  
반지름 적용  
④, ③이 생략되면 ②, ①과  
같은 반지름의 대칭 구조)



Do it! 원하는 꼭짓점만 둥글게 처리하기

(... 생략 ...)

```
#round1 {  
  border: 2px solid blue;  
  border-top-left-radius: 20px;  
  border-top-right-radius: 20px;  
}
```

(... 생략 ...)



# 다양한 둥근 모서리 테두리(BorderCase.html)

```
<!DOCTYPE html>
<html>
<head> <title>둥근 모서리 테두리 </title>
<style>
p {
  background : #90D000;
  width : 300px;
  padding : 20px;
}
#round1 { border-radius : 50px; }
#round2 { border-radius : 0px 20px 40px 60px; }
#round3 { border-radius : 0px 20px 40px; }
#round4 { border-radius : 0px 20px; }
#round5 { border-radius : 50px;
          border-style : dotted; }
</style>
</head>
<body>
<h3>둥근 모서리 테두리</h3>
<hr>
<p id="round1">반지름 50픽셀의 둥근 모서리</p>
<p id="round2">반지름 0, 20, 40, 60 둥근 모서리</p>
<p id="round3">반지름 0, 20, 40, 20 둥근 모서리</p>
<p id="round4">반지름 0, 20, 0, 20 둥근 모서리</p>
<p id="round5">반지름 50의 둥근 점선 모서리</p>
</body>
</html>
```

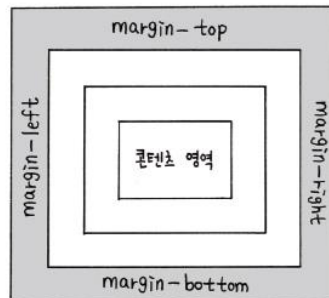


# 여백을 조절하는 속성

## margin 속성

- 현재 요소 주변의 여백
- 마진을 이용하면 요소와 요소 간의 간격 조절 가능

기본형 margin: <크기> | <백분율> | auto



종류	설명	예시
<크기>	너비값이나 높이값을 px이나 em 같은 단위와 함께 수치로 지정합니다.	margin: 50px;
<백분율>	박스 모델을 포함한 부모 요소를 기준으로 너비값이나 높이값을 퍼센트(%)로 지정합니다.	margin: 0.1%;
auto	display 속성에서 지정한 값에 맞게 적절한 값을 자동으로 지정합니다.	



## margin 속성을 사용해 가운데 정렬하기

- 배치할 요소의 너비값이 정해져 있어야 함
- margin-left와 margin-right의 속성값을 auto로 지정



Do it! 웹 문서 전체를 웹 브라우저 화면 중앙에 배치하기

예제 파일 08\margin-3.html

```
(... 생략 ...)  
<style>  
  body { background-color: #222; } /* 배경색은 어둡게 지정 */  
  #container {  
    background-color: #fff;  
    width: 600px; /* 가로 너비 600px */  
    margin: 20px auto; /* 위아래 마진은 20px씩, 좌우 마진은 자동 계산 */  
    padding: 20px;  
  }  
  .....  
<body>  
  <div id="container">  
    <h1>레드향</h1>  
    <p>껍질에 붉은 빛이 돌아 <b>레드향</b>이라 불린다.</p>  
    .....  
  </div>  
(... 생략 ...)
```

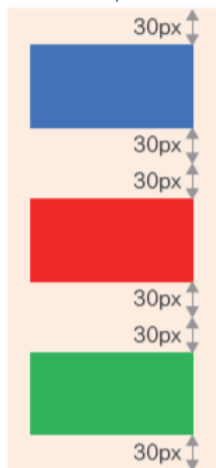
margin: 20px auto

# 여백을 조절하는 속성

## 마진 중첩 현상

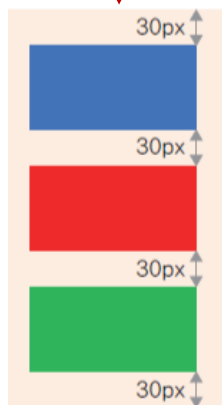
- 요소를 세로로 배치할 경우,  
마진과 마진이 만날 때 마진 값이 큰 쪽으로 겹쳐지는 것
- 요소를 가로로 배치할 경우에는 상관없음

예상 모습



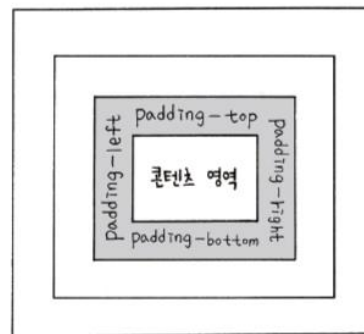
```
div {  
  width:200px; /* 너비 */  
  height:100px; /* 높이 */  
  margin:30px; /* 마진 */  
}  
#box1 { background:rgb(0, 77, 243); }  
#box2 { background:rgb(255, 72, 0); }  
#box3 { background:rgb(18, 219, 0); }  
  
<div id="box1"></div>  
<div id="box2"></div>  
<div id="box3"></div>
```

실제 화면



## padding 속성

- 콘텐츠 영역과 테두리 사이의 여백
- 마진을 지정하는 방법과 같다



# 웹 문서 레이아웃 만들기

## display 속성

사용할 수 있는 속성 값이 아주 많다

요소의 배치 방법 결정.

블록 레벨 요소, 인라인 레벨 요소를 바꿀 때 사용하는 속성 값

종류	설명
block	인라인 레벨 요소를 블록 레벨 요소로 만듭니다.
inline	블록 레벨 요소를 인라인 레벨 요소로 만듭니다.
inline-block	인라인 레벨 요소와 블록 레벨 요소의 속성을 모두 가지고 있으며 마진과 패딩을 지정할 수 있습니다.
none	해당 요소를 화면에 표시하지 않습니다.

<style>

```
* { box-sizing: border-box; } /* 모든 요소에 테두리까지 포함한 너비값 지정 */
nav ul { list-style: none; } /* 불릿이나 숫자 번호 없앴 */
nav ul li {
    display: inline-block; /* 인라인 레벨 요소와 블록 레벨 요소 모두 지정 */
    padding: 20px; /* 패딩값 20px */
    margin: 0 20px; /* 마진값 20px */
    border: 1px solid #222; /* 메뉴의 테두리 지정 */
}
```

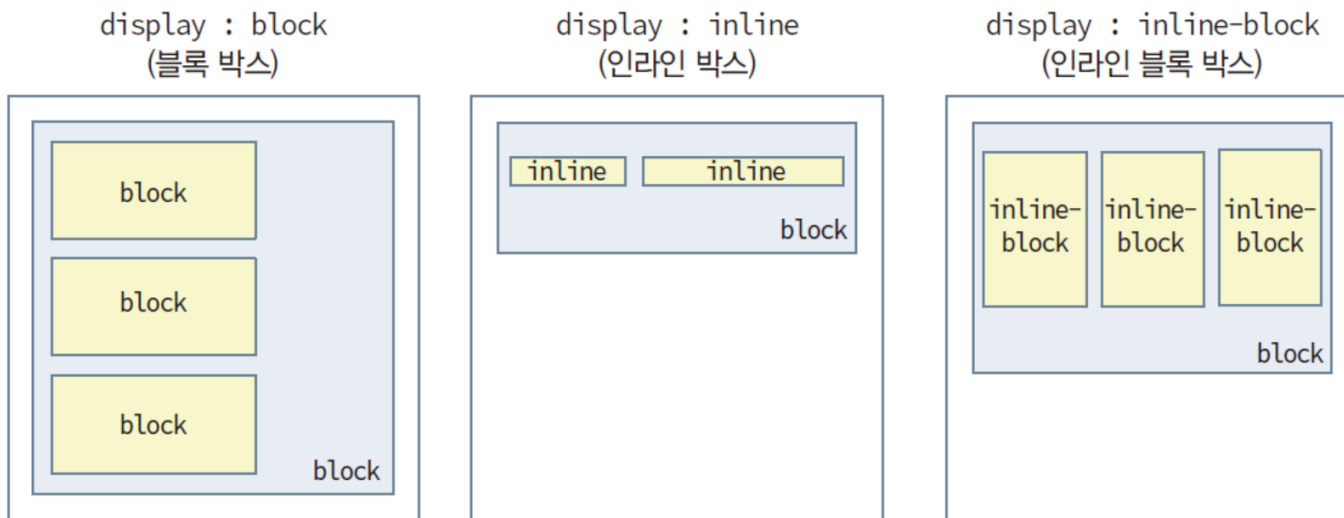
menu 1

menu 2

menu 3

menu 4

# 박스의 유형 제어 : display



| 블록박스<br>(display:block)       | 인라인 박스<br>(display:inline)        | 인라인 블록 박스<br>(display:inline-block) |
|-------------------------------|-----------------------------------|-------------------------------------|
| 항상 새 라인에서 시작                  | 새 라인에서 시작 못함.<br>라인 안(inline)에 있음 | 새 라인에서 시작 못함.<br>라인 안(inline)에 있음   |
| 블록 박스 내에만 배치                  | 모든 박스 내 배치 가능                     | 모든 박스 내 배치 가능                       |
| 옆에 다른 요소 배치 불가능               | 옆에 다른 요소 배치 가능                    | 옆에 다른 요소 배치 가능                      |
| width와 height으로 크기 조절         | width와 height으로 크기 조절 불가능         | width와 height 크기 조절 가능              |
| padding, border, margin 조절 가능 | margin-top, margin-bottom 조절 불가능  | padding, border, margin의 조절 가능      |

inline 속성

block 속성

# display : block

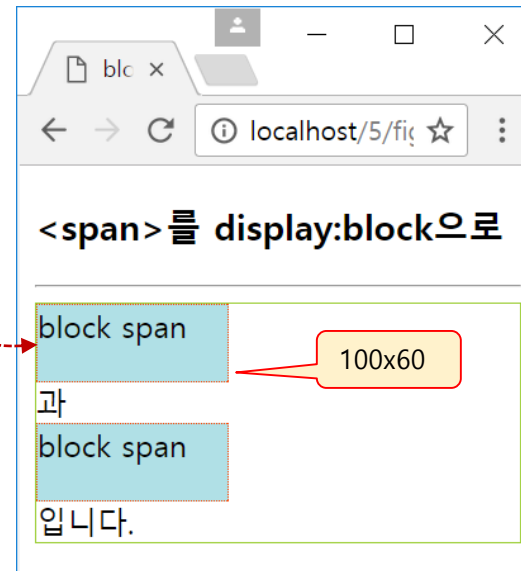
예 : <span>을 블록 박스로 수정

CSS 스타일

```
span {  
  display : block;  
  width : 100px;  
  height : 60px;  
}
```

HTML 코드

```
<div>  
  <span>block span</span>과  
  <span>block span</span>입니다.  
</div>
```



\* <span>을 block 박스로 지정하고,  
폭과 높이를 각각 100px, 60px로 지정  
한 사례.  
한 줄을 독점적으로 차지하여 옆에  
다른 태그 배치되지 않음



# display : inline

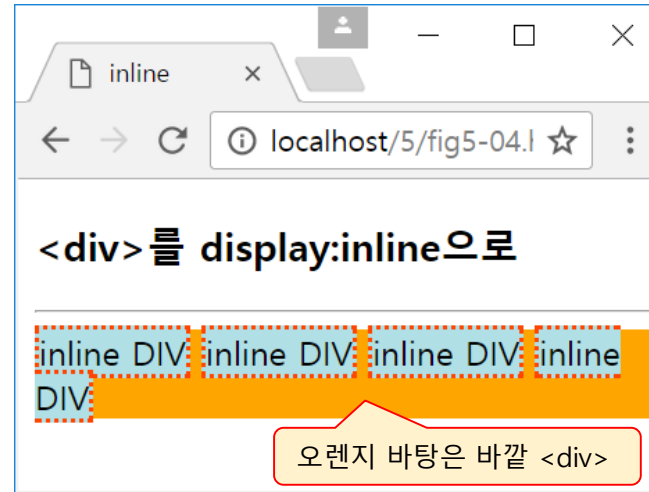
예 : <div>를 인라인 박스로 수정

CSS 스타일

```
div div {  
  display : inline;  
}
```

HTML 코드

```
<div style="background : orange">  
  <div>inline DIV</div>  
  <div>inline DIV</div>  
  <div>inline DIV</div>  
</div>
```



\* <div>가 inline박스로 지정됨.  
inline 박스는 라인 안에 다른 요소들과 함께 배치.  
공간이 좁으면 남은 부분이 다음 라인으로 넘어감

# display : inline-block

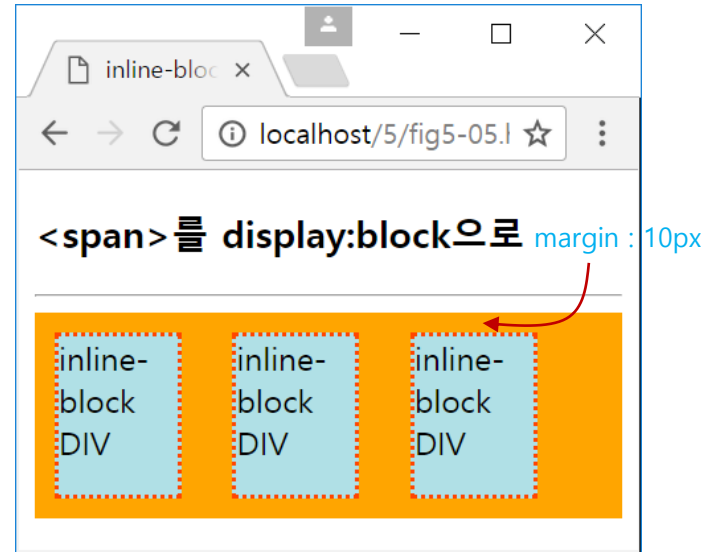
예 : <div>를 인라인 블록 박스로 수정

CSS 스타일

```
div div{  
  display : inline-block;  
  border : 2px dotted orangered ;  
  background : powderblue;  
  margin : 10px;  
  width : 60px; height : 80px;  
}
```

HTML 코드

```
<div style="background : orange">  
  <div>inline-block DIV</div>  
  <div>inline-block DIV</div>  
  <div>inline-block DIV</div>  
</div>
```



\* inline-block 박스는 라인 안에 다른 요소들과 함께 배치. 동시에 width, height, margin 으로 크기 조절 가능

# display 프로퍼티로 박스 유형 설정 (DisplayEx.html)

```
<!DOCTYPE html>
<html> <head> <title>display 프로퍼티 </title>
<style>
div {
  border : 2px solid yellowgreen;
  color : blue;
  background : aliceblue;
}
span {
  border : 3px dotted red;
  background : yellow;
}
</style> </head>
<body>
<h3>인라인, 인라인 블록, 블록</h3>
<hr>
나는 <div style="display:none">
      div(none)</div>입니다.<br><br>
나는 <div style="display:inline">
      div(inline)</div>입니다.<br><br>
나는 <div style="display:inline-block; height:50px">
      div(inline-block)</div>입니다.<br><br>
나는 <div>div<span style="display:block">
      span(block)</span></div>
      </div>
      </body>
</html>
```

display:none으로 지정되어 텍스트가 보이지 않음



# 웹 문서 레이아웃 만들기

## float 속성

요소를 왼쪽이나  
오른쪽에 떠 있게 만들

기본형 float: left | right | none

속성 값	설명
left	해당 요소를 문서의 왼쪽으로 배치합니다.
right	해당 요소를 문서의 오른쪽으로 배치합니다.
none	좌우 어느 쪽으로도 배치하지 않습니다.



**Do it!** 텍스트 왼쪽에 이미지 배치하기

```
(... 생략 ...)
<style>
  img {
    float: left; /* 왼쪽에 떠 있게 */
    margin-right: 40px; /* 오른쪽 마진 40px */
  }
.....

<p>Ex et adipisicing .....</p>
(... 생략 ...)
```



Ex et adipisicing voluptate aliqua cupidatat nulla. Laboris est sint sit aliqua enim. Aute Lorem eu sint aute sunt proident. Do culpa consectetur elit dui laboris reprehenderit incididunt nulla. Irure exercitation tempor aliqua laboris cupidatat anim in non officia aliquip excepteur fugiat labore.

Lorem ipsum reprehenderit adipisicing exercitation enim velit veniam incididunt sit consectetur elit exercitation. Commodore veniam sit quis nisi ea. Ipsum do aliqua nostrud laboris elit dui adipisicing id Lorem qui. Labore dolor ipsum anim incididunt. Velit qui cillum sunt labore incididunt dui aute Lorem nulla et. Sint commodore aute amet laboris ullamco exercitation Lorem dolore veniam ut reprehenderit incididunt. Laborum nulla eiusmod cillum irure anim aute.

Excepteur voluptate ad irure ipsum dui. Deserunt cupidatat commodore proident eu mollit cillum commodore quis quis et ad. Incidunt adipisicing enim laboris voluptate.

## clear 속성

float 속성을 무효화 시키는 속성

기본형 clear: none | left | right | both

종류	설명
left	float: left를 해제합니다.
right	float: right를 해제합니다.
both	float: left와 float: right를 해제합니다.

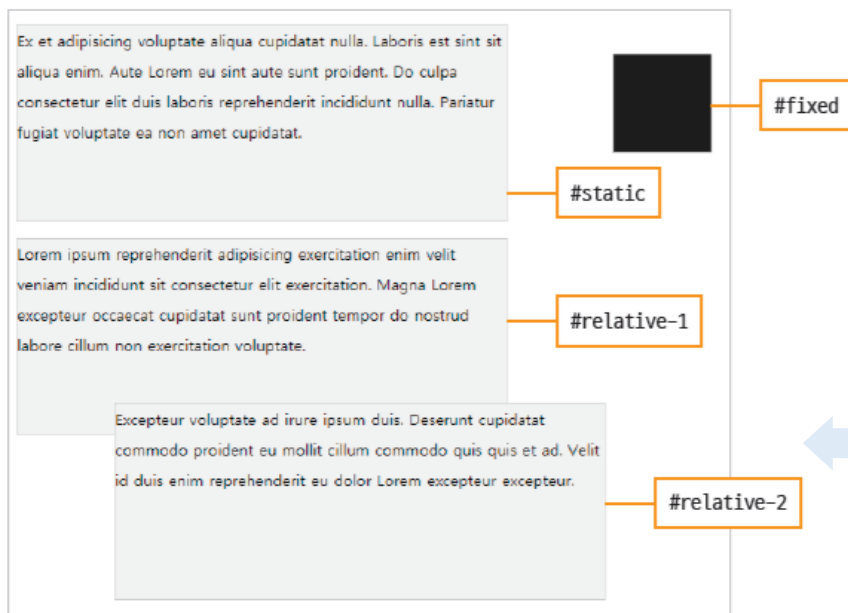
# 웹 요소의 위치 지정하기

## position 속성

웹 문서 안에 요소들을 배치하기 위한 속성

기본형 position: static | relative | absolute | fixed

속성 값	설명
static	요소를 문서의 흐름에 맞추어 배치합니다.*
relative	이전 요소에 자연스럽게 연결해 배치하되 위치를 지정할 수 있습니다.
absolute	원하는 위치를 지정해 배치합니다.
fixed	지정한 위치에 고정해 배치합니다. 화면에서 요소가 잘릴 수도 있습니다.



Do it! position 속성으로 요소 위치 지정하기

예제 파일 08\position-1.html

(... 생략 ...)

```
#static { position: static; }
#relative-1 { position: relative; }
#relative-2 {
  p : relative; /* 포지셔닝 - absolute */
  left: 100px; /* 왼쪽에서 100px 떨어지게 */
  top: -50px; /* 위쪽에서 -50px 떨어지게(위로 이동) */
}
#fixed {
  width: 100px;
  height: 100px;
  background-color: #222;
  position: fixed; /* 포지셔닝 - fixed */
  right: 30px; /* 오른쪽에서 30px 떨어지게 */
  top: 30px; /* 위쪽에서 30px 떨어지게 */
}
</style>
```

```
<p id="static">Ex et adipisicing ..... </p>
<p id="relative-1">Lorem ipsum reprehenderit ..... </p>
<p id="relative-2">Excepteur voluptate ad ..... </p>
<p id="fixed"></p>
```

(... 생략 ...)

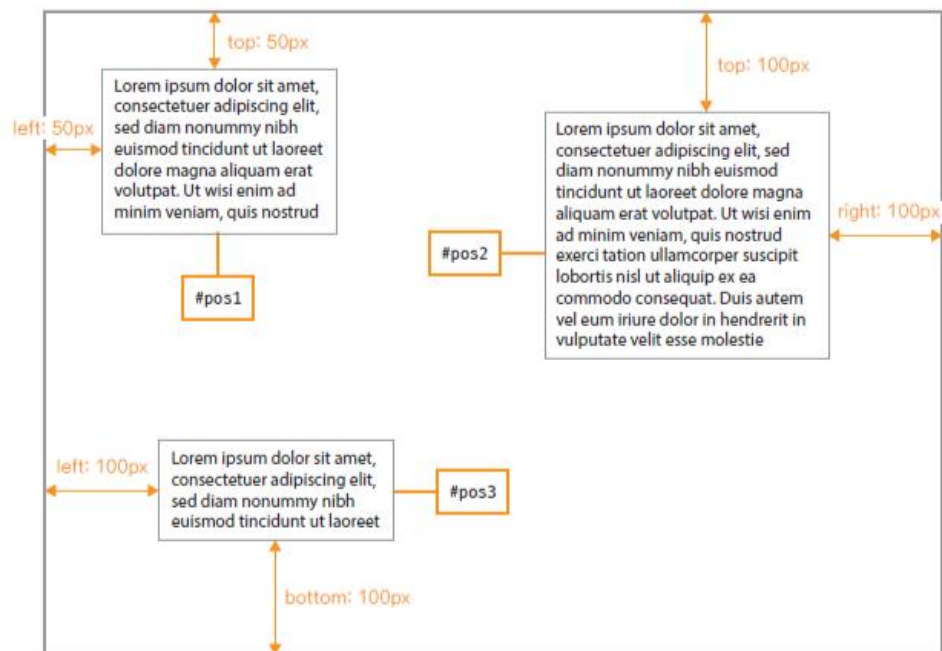
uonjsod:呂윤

# 웹 요소의 위치 지정하기

## left, right, bottom, top 속성

웹 요소의 위치를 지정하는 속성

종류	설명
left	기준 위치와 요소 사이에 왼쪽으로 얼마나 떨어져 있는지 지정합니다.
right	기준 위치와 요소 사이에 오른쪽으로 얼마나 떨어져 있는지 지정합니다.
top	기준 위치와 요소 사이에 위쪽으로 얼마나 떨어져 있는지 지정합니다.
bottom	기준 위치와 요소 사이에 아래쪽으로 얼마나 떨어져 있는지 지정합니다.



Do it! 텍스트 요소 자유롭게 배치하기

예제 파일 08\left-top.html

(... 생략 ...)

```
#pos1 {  
    position: absolute; /* 포지셔닝 - absolute */  
    left: 50px;        /* 왼쪽에서 50px 떨어지게 */  
    top: 50px;         /* 위쪽에서 50px 떨어지게 */  
}  
  
#pos2 {  
    position: absolute; /* 포지셔닝 - absolute */  
    right: 100px;       /* 오른쪽에서 100px 떨어지게 */  
    top: 100px;        /* 위쪽에서 100px 떨어지게 */  
}  
  
#pos3 {  
    position: absolute; /* 포지셔닝 - absolute */  
    left: 100px;       /* 왼쪽에서 100px 떨어지게 */  
    bottom: 100px;     /* 아래쪽에서 100px 떨어지게 */  
}
```

```
.....  
<p id="pos1">Ex et adipisicing .....</p>  
<p id="pos2">Lorem ipsum reprehenderit .....</p>  
<p id="pos3">Excepteur voluptate .....</p>
```

(... 생략 ...)

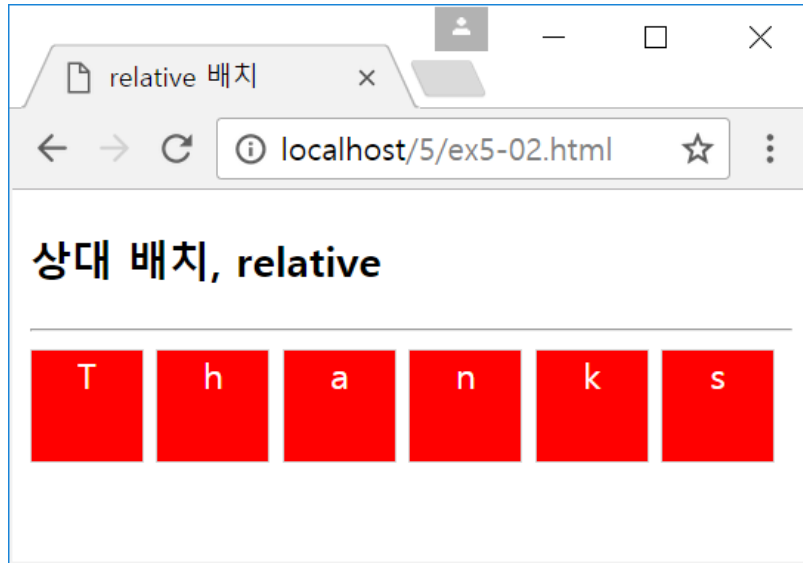
# 쉽게 이해하자면(test2.html)

```
*{
  margin: 0;
  padding:0;
  box-sizing: border-box;
}
.item-wrap{
  position:relative;
  background-color: #eee;
  width: 600px;
  height: 1800px;
  padding:20px;
  margin:auto;
  margin-top:100px;
  border-radius: 8px;
  border: 1px solid #667949;
}
.item{
  width: 560px;
  color: white;
  padding:20px 10px;
  border-radius: 4px;
  text-align: center;
  background-color: blue;
  line-height: 50px;
  margin-top:10px;
  border-color: rgb(8, 6, 2);
  box-shadow: 0 4px 16px rgba(0,79,255,0.2);
}
```

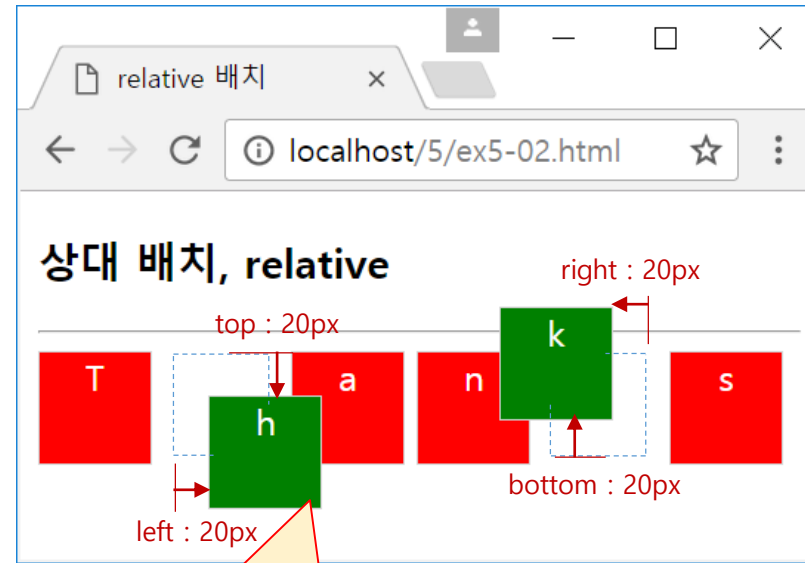
```
.static{
  position:static;
  left:60px;
}
.relative{
  position:relative;
  left:60px;
}
.absolute{
  position:absolute;
  left:60px;
}
.fixed{
  position:fixed;
  top:430px;
  left:60px;
}
```



# Relative 상대배치



초기 상태



상대 배치(position:relative)

h와 k 블록에 마우스를 올린 경우,  
상대 배치에 따라 위치가 변함