



## Chapter 6

# 박스모델과 색상지정



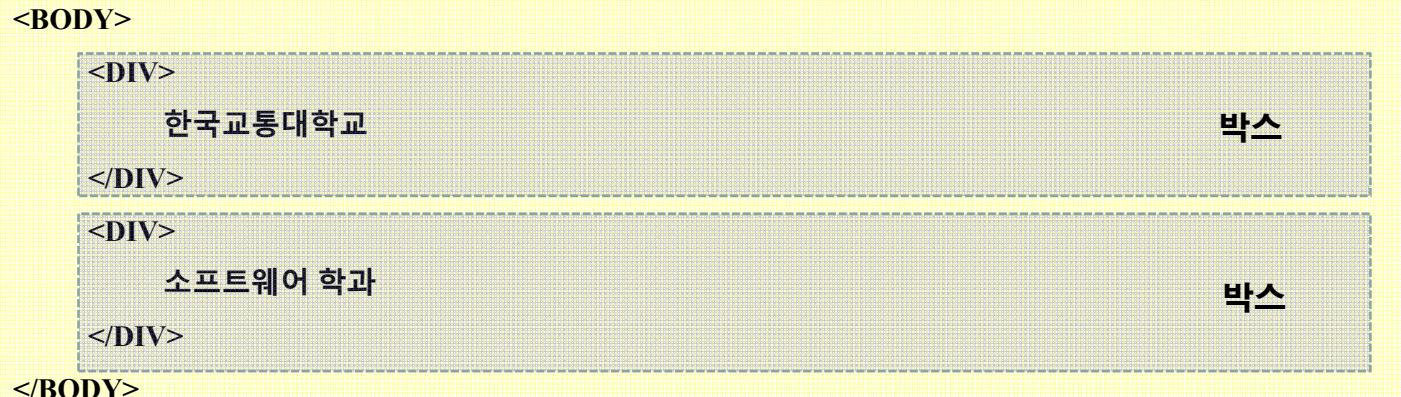
## 박스 모델 (BOX MODEL)



## ■ 박스의 정의

- 웹 문서에 표현하는 사각형 형태의 논리적인 영역을 의미
  - 테이블과는 다른 개념임
- 주로 <DIV> 태그를 사용해 박스라는 논리적인 영역을 작성
  - <DIV> 태그는 특정 내용을 논리적으로 구문할 때 사용하기 때문
    - : CSS에서는 주로 <DIV> 태그를 사용해 논리적인 영역을 생성하고 스타일을 적용함
    - : <SPAN> 태그도 사용 가능
- <DIV> 태그로 생성된 논리적인 영역에 테두리를 지정하면 박스가 형성됨

```
<BODY>
  <DIV>
    한국교통대학교
  </DIV>
  <DIV>
    소프트웨어 학과
  </DIV>
</BODY>
```



<DIV>

박스

</DIV>

박스

</DIV>

</BODY>





## ■ 박스에 지정할 수 있는 스타일

- **박스의 표현**

- `display` 속성을 사용해 박스를 표현하는 방법 지정

- **크기 지정**

- 박스의 너비와 높이를 지정

- **여백 지정**

- 박스 외부의 다른 콘텐츠와의 간격을 의미
  - 박스의 상단, 하단 좌측, 우측 여백 지정

- **테두리선 지정**

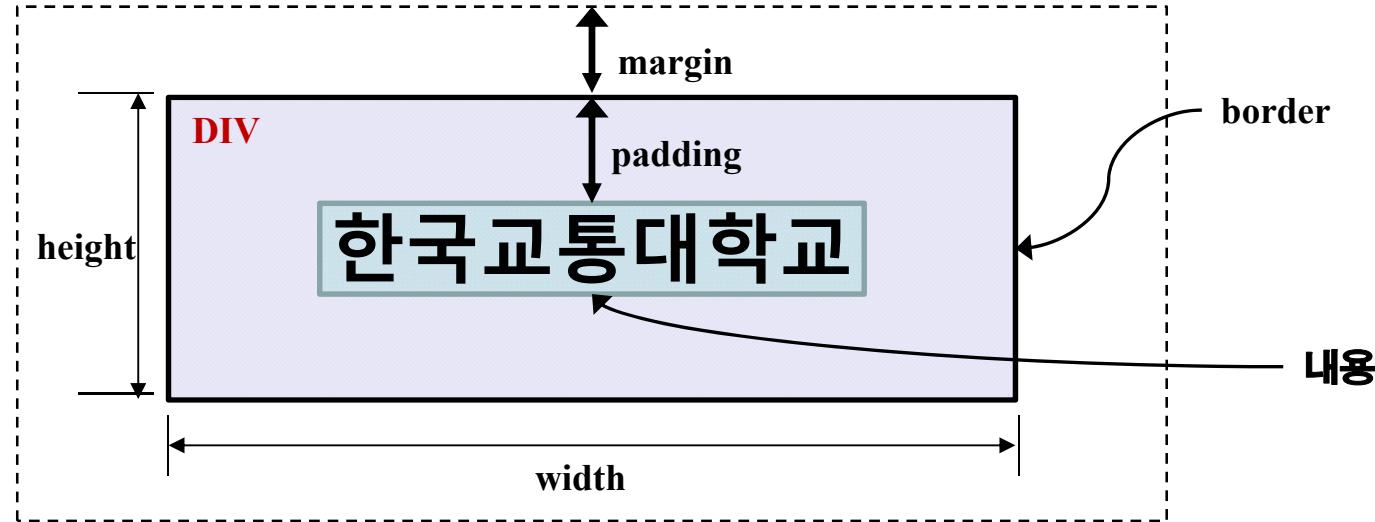
- 선의 종류, 선의 굵기, 선의 색상을 지정

- **위치 지정**

- 상대 위치와 절대 위치 지정



# 박스 모델의 개요



- **border**
  - 박스의 테두리를 의미(선의 종류, 굵기, 색상지정 가능)
- **padding**
  - border와 내용 사이의 간격 (상/하/좌/우 같거나 서로 다른 여백지정 가능)
- **margin**
  - border와 박스외부의 내용 사이 간격(상/하/좌/우 같거나 서로 다른 여백지정 가능)





## ■ display 속성

- HTML 요소의 표현 방식을 지정하기 위한 속성

**display : block | inline | inline-block**

속성값	의미
block	요소를 block 형식으로 표현 ( 아래 위 줄 바꿈 발생 ) 너비와 높이 지정 가능 <DIV> 요소는 기본적으로 block 속성 값을 가짐
inline	요소를 inline 형식으로 표현 ( 줄 바꿈 없이 일렬로 배치 ) 너비와 높이 지정 불가능 <SPAN> 요소는 기본적으로 inline 속성 값을 가짐
inline-block	요소를 inline 형식으로 표현하고 내부는 block으로 표현 너비와 높이 지정 불가능

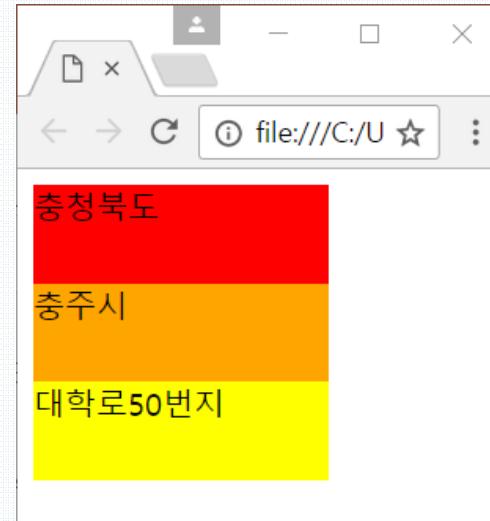




## ■ <DIV>에 display 속성 지정하는 경우의 예

- **display : block**을 지정한 경우

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>display 속성</title>
    <style type="text/css">
        div {
            display:block;
            width:150px; height:50px;
        }
        #div_a { background-color:red }
        #div_b { background-color:orange }
        #div_c { background-color:yellow }
    </style>
</head>
<body>
    <div id=div_a>충청북도</div>
    <div id=div_b>충주시</div>
    <div id=div_c>대학로50번지</div>
</body>
</html>
```

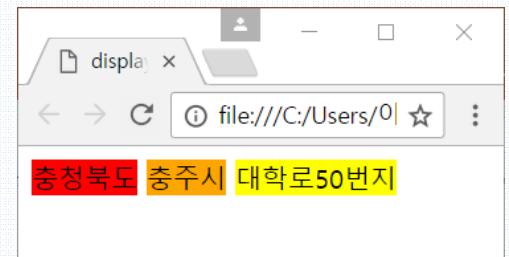




## ■ <DIV>에 display 속성 지정하는 경우의 예

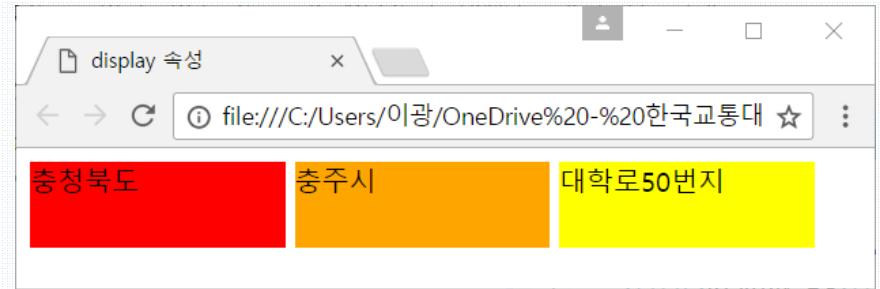
- **display : inline을 지정한 경우**

```
div {  
    display:inline;  
    width:150px; height:50px;  
}
```



- **display : inline-block을 지정한 경우**

```
div {  
    display:inline-block;  
    width:150px; height:50px;  
}
```

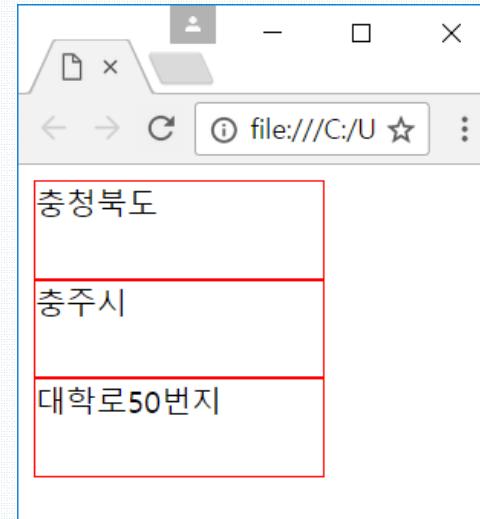




## ■ <SPAN>에 display 속성 지정하는 경우의 예

- **display : block을 지정한 경우**
  - <DIV>에 적용하는 것과 유사함

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>display 속성</title>
    <style type="text/css">
        span {
            display:block;
            width:150px; height:50px;
        }
        .span_a { border:1px solid red }
    </style>
</head>
<body>
    <span class=span_a>충청북도</span>
    <span class=span_a>충주시</span>
    <span class=span_a>대학로50번지</span>
</body>
</html>
```

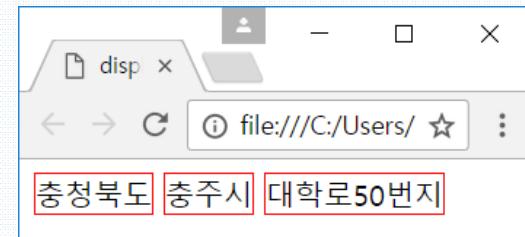




## ■ <SPAN>에 display 속성 지정하는 경우의 예

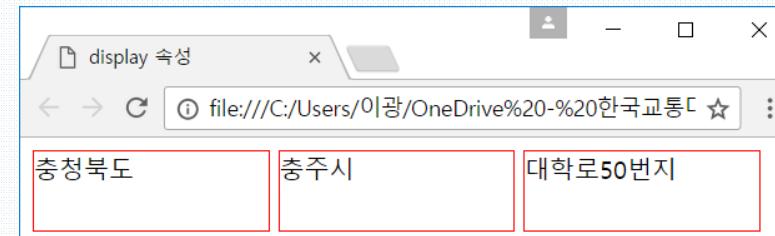
- **display : inline을 지정한 경우**

```
span {  
    display:inline;  
    width:150px; height:50px;  
}
```



- **display : inline-block을 지정한 경우**

```
span {  
    display:inline-block;  
    width:150px; height:50px;  
}
```





## ■ margin 속성

- 박스의 외부 여백을 지정하기 위한 속성

```
margin : [ margin-top값 margin-right값 margin-bottom값 margin-left값 ]
```

- 상/하/좌/우 margin을 일괄 지정

- 위쪽 여백부터 시계방향으로 지정 ( 여백 지정의 순서가 바뀌면 안됨 )

- 일부 margin 요소를 생략할 수 있음

- 하나의 값만 입력할 경우

: 상/하/좌/우 여백을 동일한 값으로 지정

- 두 개의 값을 지정할 경우

: 첫 번째 값으로 상/하 여백을 지정하고, 두 번째 값으로 좌/우 여백을 지정함

- 세 개의 값을 입력할 경우

: 첫 번째 값은 상단여백, 두 번째 값은 좌/우 여백, 세 번째 값은 하단여백으로 지정함





## ■ margin 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>display 속성</title>
    <style type="text/css">
        div {
            background-color:gray;
            color:white;
            font-size:50px;
        }
        .margin1 { margin:50px; }
        .margin2 { margin:50px 100px; }
        .margin3 { margin:20px 40px 60px; }
        .margin4 { margin:30px 60px 90px 120px; }
    </style>
</head>
<body>
    <P>박스의 상단 내용입니다.</P>
    <DIV class="margin1">한국교통대학교</DIV>
    <P>박스의 하단 내용입니다.</P>
</body>
</html>
```





## ▣ 외부 여백의 세부 지정

- **margin-top 속성**
  - 상단 외부 여백 지정
- **margin-left 속성**
  - 좌측 외부 여백 지정
- **margin-right 속성**
  - 우측 외부 여백 지정
- **margin-bottom 속성**
  - 하단 외부 여백 지정
- **일부 여백은 생략 가능**
- **단위는 픽셀을 사용**



```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>margin 속성</title>
<style type="text/css">
.margin1{
    background-color:gray;
    color:white;
    font-size:50px;
    margin-top:10px;
    margin-bottom:10px;
    margin-left:50px;
    margin-right:50px;
}
</style>
</head>
<body>
<P>박스의 상단 내용입니다.</P>
<DIV class="margin1">한국교통대학교</DIV>
<P>박스의 하단 내용입니다.</P>
</body>
</html>
```





## ■ padding 속성

- 박스의 내부(테두리선과 콘텐츠 사이)의 여백을 지정하기 위한 속성

padding : [ padding-top 값 padding-right 값 padding-bottom 값 padding-left 값 ]

- 상/하/좌/우 padding을 일괄 지정
  - 위쪽 여백부터 시계방향으로 지정 (여백 지정의 순서가 바뀌면 안됨)

- 일부 padding 요소를 생략할 수 있음

- 하나의 값만 입력할 경우

- : 상/하/좌/우 여백을 동일한 값으로 지정

- 두 개의 값을 지정할 경우

- : 첫 번째 값으로 상/하 여백을 지정하고, 두 번째 값으로 좌/우 여백을 지정함

- 세 개의 값을 입력할 경우

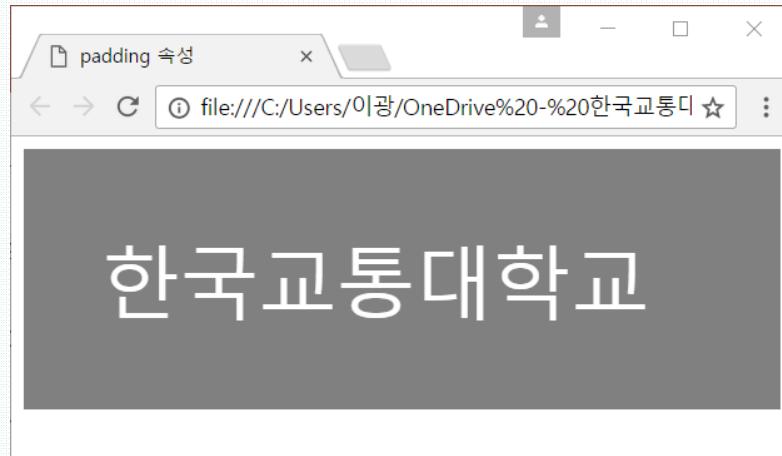
- : 첫 번째 값은 상단여백, 두 번째 값은 좌/우 여백, 세 번째 값은 하단여백으로 지정함





## ■ padding 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>padding 속성</title>
    <style type="text/css">
        div {
            background-color:gray;
            color:white;
            font-size:50px;
        }
        .padding1 { padding:50px; }
        .padding2 { padding:50px 200px; }
        .padding3 { padding:20px 40px 60px; }
        .padding4 { padding:30px 60px 90px 120px; }
    </style>
</head>
<body>
    <div class="padding1">한국교통대학교</div>
</body>
</html>
```



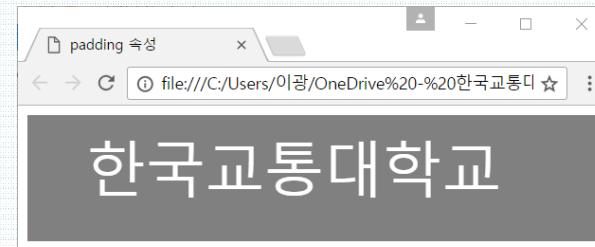


## ■ 내부 여백의 세부 지정

- **padding-top 속성**
  - 상단 내부 여백 지정
- **padding-left 속성**
  - 좌측 내부 여백 지정
- **padding-right 속성**
  - 우측 내부 여백 지정
- **padding-bottom 속성**
  - 하단 내부 여백 지정
- **일부 여백은 생략 가능**
  - 단위는 픽셀을 사용



```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>padding 속성</title>
    <style type="text/css">
        .padding1{
            background-color:gray;
            color:white;
            font-size:50px;
            padding-top:10px;
            padding-bottom:30px;
            padding-left:50px;
            padding-right:70px;
        }
    </style>
</head>
<body>
    <div class="padding1">한국교통대학교</div>
</body>
</html>
```





## ■ width 속성, height 속성

- 박스 영역의 크기를 지정하는 속성

**width : px | % | auto**

**height : px | % | auto**

- % - 화면의 크기에 대한 비율
- pt - 포인트
- px - 픽셀 (주로 사용함)

- [예]

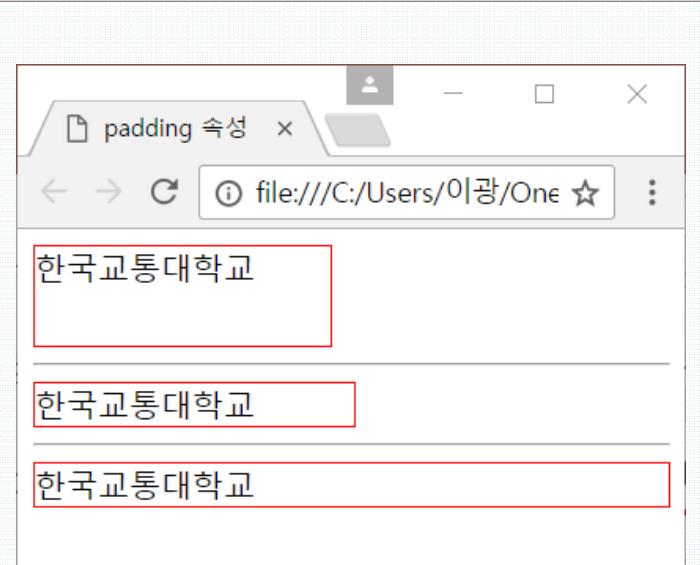
: <DIV style="width:500; height:300;">한국교통대학교</DIV>





## ■ width 속성, height 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>padding 속성</title>
    <style type="text/css">
        div { display:block; border:1px solid red }
        .size_1{ width:150px; height:50px }
        .size_2{ width:50%; height:100% }
        .size_3{ width:auto; height:auto }
    </style>
</head>
<body>
    <div class="size_1">한국교통대학교</div>
    <hr>
    <div class="size_2">한국교통대학교</div>
    <hr>
    <div class="size_3">한국교통대학교</div>
</body>
</html>
```





## ■ border-style 속성

- 테두리 선의 종류를 지정하기 위한 속성

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

- 지정된 스타일로 상/ 하/ 좌/ 우 테두리의 모든 스타일이 동시에 적용됨

- 선의 종류

- : none - 표시하지 않음
- : hidden - ( 테두리 선은 존재하지만 출력되지 않음 )
- : dotted - 점으로 표현
- : dashed - 점선으로 표현
- : solid - 실선으로 표현
- : double - 이중실선으로 표현
- : groove - 오른쪽 아래에서 빛을 받는 스타일로 표현
- : ridge - 왼쪽 위에서 빛을 받는 스타일로 표현
- : outset - 바깥쪽으로 돌출된 스타일로 표현
- : inset - 안쪽으로 함몰된 스타일로 표현





## ■ 테두리선 종류의 세부 지정

- 모든 테두리 선의 종류를 지정해야 하는 것은 아님
  - 지정하지 않으면 선은 출력되지 않음
- 
- **border-top-style 속성**
    - 상단 테두리선의 종류를 지정
  - **border-left-style 속성**
    - 좌측 테두리선의 종류를 지정
  - **border-right-style 속성**
    - 우측 테두리선의 종류를 지정
  - **border-bottom-style 속성**
    - 하단 테두리선의 종류를 지정





## ■ border-color 속성

- 테두리 선의 색상을 지정하기 위한 속성

border-color : 선의 색상

- 지정된 색상으로 상단, 하단, 좌측, 우측의 선 색상이 일괄적으로 적용
- 색상의 지정은 컬러의 이름, 16진수 RGB값, rgb() 함수 모두 사용 가능

- [예]

- - : <DIV style="border-color : red">한국교통대학교</DIV>
  - : <DIV style="border-color : #FF0000">한국교통대학교</DIV>
  - : <DIV style="border-color : rgb(255, 0, 0)">한국교통대학교</DIV>

- 선의 종류가 반드시 지정되어야 선의 색상을 지정할 수 있음
  - border-color 속성을 사용하지 위해서는 border-style 속성이 먼저 지정되어야 함





## ■ 테두리선 색상의 세부 지정

- **border-top-color 속성**
  - 상단 테두리선의 색상을 지정
- **border-left-color 속성**
  - 좌측 테두리선의 색상을 지정
- **border-right-color 속성**
  - 우측 테두리선의 색상을 지정
- **border-bottom-color 속성**
  - 하단 테두리선의 색상을 지정





## ■ border-width 속성

- 테두리선의 두께를 지정하기 위한 속성

border-width : 선의 굵기

- 상/ 하/ 좌/ 우 테두리의 굵기가 지정된 값으로 지정
- 굵기 지정 방법
  - thin - 얇게 지정
  - medium - 중간으로 지정(기본값)
  - thick - 굵게 지정
  - px - 지정한 픽셀 값으로 지정 ( 주로 픽셀을 사용함 )
- 선의 종류가 반드시 지정되어야 선의 두께를 지정할 수 있음
  - border-width 속성을 사용하지 위해서는 border-style 속성이 먼저 지정되어야 함





## ■ 테두리선 두께의 세부 지정

- **border-top-width 속성**
  - 상단 테두리선의 굵기를 지정
- **border-left-width 속성**
  - 좌측 테두리선의 굵기를 지정
- **border-right-width 속성**
  - 우측 테두리선의 굵기를 지정
- **border-bottom-width 속성**
  - 하단 테두리선의 굵기를 지정





## ■ border 속성

- 테두리 선의 종류, 굵기, 색상을 일괄 지정하기 위한 속성

border : 선의 스타일 선의 굵기 선의 색상

- 모든 값이 지정되어야 하는 것은 아님
  - 단, 선의 스타일은 반드시 지정되어야 함
  - 선의 스타일이 지정되지 않으면 굵기와 색상을 지정할 수 없음
- [예]
  - : DIV { border : solid 10px red; }





## ■ 테두리 효과의 세부 지정

- **border-top 속성**

- 테두리선의 위쪽 선에 대해 선의 종류, 선의 색상, 선의 굵기를 동시에 지정

- **border-left 속성**

- 테두리선의 왼쪽 선에 대해 선의 종류, 선의 색상, 선의 굵기를 동시에 지정

- **border-right 속성**

- 테두리선의 오른쪽 선에 대해 선의 종류, 선의 색상, 선의 굵기를 동시에 지정

- **border-bottom 속성**

- 테두리선의 하단 선에 대해 선의 종류, 선의 색상, 선의 굵기를 동시에 지정





## ■ 테두리 효과 지정의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>테두리 효과 지정 속성</title>
    <style type="text/css">
        body{
            font-size:11pt;
            font-weight:bold;
            text-align:center;
        }
        div{
            padding:10px;
            background-color:#FFCC33;
            color:blue;
            border-width:5px;
            border-color:#FF0000;
        }
    </style>
</head>
<body>
    <h3><U>문단 테두리 스타일</U></h3>
    <br>
    <div> 임베디드 스타일 시트 방식에 의해 적용된 문단 (DIV 태그) </div>
    <br><hr><br>
```



# 테두리 효과 지정



```
<div style="border-width:5px; border-color: #FF9900 ;border-style:dashed;">  
    border-width, border-color, border-style을 사용한 지정<br>  
    두께:5픽셀, 색:#FF9900, 스타일:dashed  
</div>  
<br><hr><br>  
    <div style="border:5px dashed #FF9900;">  
        border 속성을 사용해 동시 지정<br>  
        두께:5픽셀,스타일:dashed,색:#FF9900</div>  
<br><hr><br>  
    <div style="border:3px solid; border-bottom-color:#CC6600; border-top-color:#FF0066; ">  
        선의 색상을 다르게 표현 :<br>아래쪽은 #CC6600, 위쪽은 #FF0066  
</div>  
<br><hr><br>  
    <div style="border-top:5px solid red; border-bottom:10px solid blue">  
        border-top과 border-bottom를 사용한 상단 선과 하단 선만 속성 지정  
</div>  
<br><hr><br>  
    <h3>border-style의 종류 비교</h3>  
<br>  
    <div style="border-style:none">none</div><br>  
    <div style="border-style:solid">solid</div><br>  
    <div style="border-style:double">double</div><br>  
    <div style="border-style:groove">groove</div><br>  
    <div style="border-style:ridge">ridge</div><br>  
    <div style="border-style:inset">inset</div><br>  
    <div style="border-style:outset">outset</div><br>
```





## ■ 테두리 효과 지정의 예

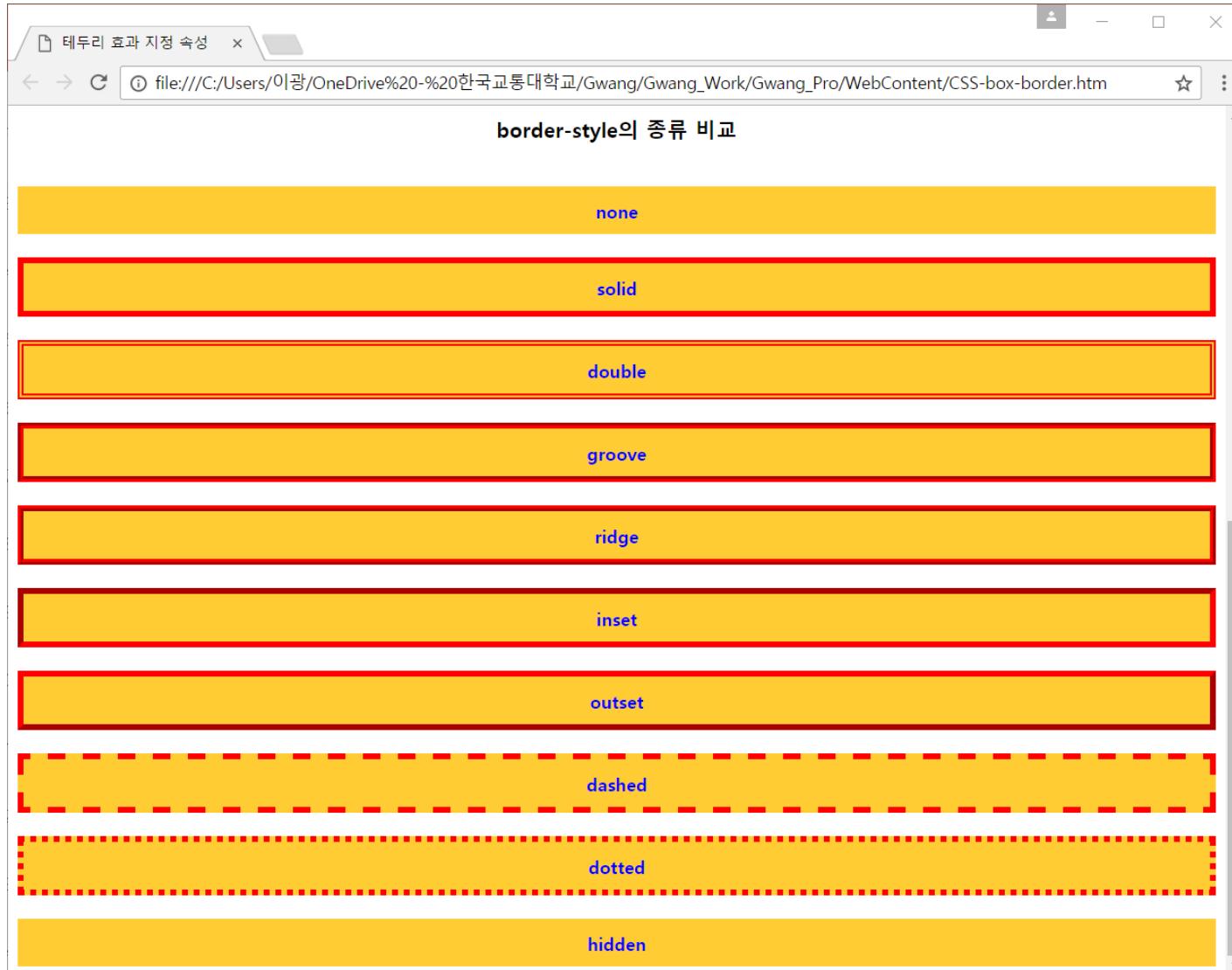
```
<div style="border-style:dashed">dashed</div><br>
<div style="border-style:dotted">dotted</div><br>
<div style="border-style:hidden">hidden</div>
</body>
</html>
```

The screenshot shows a web browser window with the title "테두리 효과 지정 속성". The address bar shows the file path: file:///C:/Users/이광/OneDrive%20-%20한국교통대학교/Gwang/Gwang\_Work/Gwang\_Pro/WebContent/CSS-box-border.h. The content area displays five horizontal bars, each demonstrating a different border property:

- 임베디드 스타일 시트 방식에 의해 적용된 문단 (DIV 태그)**: A solid yellow bar.
- border-width, border-color, border-style을 사용한 지정  
두께:5픽셀, 색:#FF9900, 스타일:dashed**: A yellow bar with a dashed border.
- border 속성을 사용해 동시에 지정  
두께:5픽셀, 스타일:dashed, 색:#FF9900**: A yellow bar with a dashed border.
- 선의 색상을 다르게 표현 :  
아래쪽은 #CC6600, 위쪽은 #FF0066**: A yellow bar with a blue top border.
- border-top과 border-bottom를 사용한 상단 선과 하단 선만 속성 지정**: A yellow bar with a red top border.



# 테두리 효과 지정





## ▣ float 속성

- 한 행에서 박스가 왼쪽이나 오른쪽에 위치하게 하는 속성

**float : left | right | none**

- none : 플로팅되지 않음
- left : 왼쪽 방향으로 플로팅
- right : 오른쪽 방향으로 플로팅

속성값	의미
none	플로팅되지 않음
left	왼쪽 방향으로 플로팅
right	오른쪽 방향으로 플로팅





## ▣ float 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>플로팅 속성</title>
    <style type="text/css">
        img { float:left; margin:5px; }
    </style>
</head>
<body>
    <div>
        
        <p>현재 행의 왼쪽 또는 오른쪽으로 이동되는 박스를 플로트(float, 또는 플로팅 박스(floating box)라고 부른다.</p>
        <p>가장 흥미로운 특징은 콘텐츠가 측면을 따라 흐를 수 있다는 것이다. 즉, 일반적인 콘텐츠의 흐름에서 분리되어 부모 요소의 영역 내에서 새롭게 배치되기 때문에, 왼쪽에 위치한 플로팅 박스의 오른쪽 측면과 오른쪽에 위치한 플로팅 박스의 왼쪽 측면 아래로 흐르게 된다.</p>
    </div>
</body>
</html>
```

현재 행의 왼쪽 또는 오른쪽으로 이동되는 박스를 플로트(float, 또는 플로팅 박스(floating box)라고 부른다.



가장 흥미로운 특징은 콘텐츠가 측면을 따라 흐를 수 있다는 것이다. 즉, 일반적인 콘텐츠의 흐름에서 분리되어 부모 요소의 영역 내에서 새롭게 배치되기 때문에, 왼쪽에 위치한 플로팅 박스의 오른쪽 측면과 오른쪽에 위치한 플로팅 박스의 왼쪽 측면 아래로 흐르게 된다.





## ■ clear 속성

- **float 속성이 적용되는 부분을 해제하는 속성**
  - float속성이 영향을 주지 않아야 하는 부분을 지정하는데 사용

**clear : left | right | both**

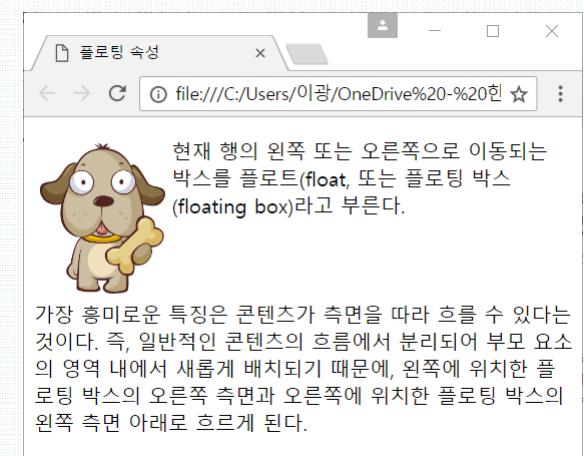
속성값	의미
left	clear:left가 미치는 영향을 해제하는 속성값
right	clear:right가 미치는 영향을 해제하는 속성값
both	clear:left와 clear:right에 관계없이 미치는 영향을 해제하는 속성값





## ■ clear 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>플로팅 속성</title>
    <style type="text/css">
        img { float:left; margin:5px; }
    </style>
</head>
<body>
    <div>
        
        <p>현재 행의 왼쪽 또는 오른쪽으로 이동되는 박스를 플로트(float, 또는 플로팅 박스(floating box)라고 부른다.</p>
        <p style="clear:left">가장 흥미로운 특징은 콘텐츠가 측면을 따라 흐를 수 있다는 것이다. 즉, 일반적인 콘텐츠의 흐름에서 분리되어 부모 요소의 영역 내에서 새롭게 배치되기 때문에, 왼쪽에 위치한 플로팅 박스의 오른쪽 측면과 오른쪽에 위치한 플로팅 박스의 왼쪽 측면 아래로 흐르게 된다.</p>
    </div>
</body>
</html>
```





## ▣ overflow-x 속성, overflow-y 속성

- 수평방향과 수직방향으로 오버플로우가 발생되는 경우 이를 처리하는 방법을 지정
  - 오버플로우(overflow)란 콘텐츠의 내용이 요소의 영역을 벗어나는 경우를 의미

**overflow-x : visible | hidden | scroll | auto | no-display | no-content**

**overflow-y : visible | hidden | scroll | auto | no-display | no-content**

## ▣ overflow 속성

- overflow-x와 overflow-y를 동시에 지정하는 속성

**overflow : visible | hidden | scroll | auto | no-display | no-content**





## ▣ overflow 관련 속성의 값

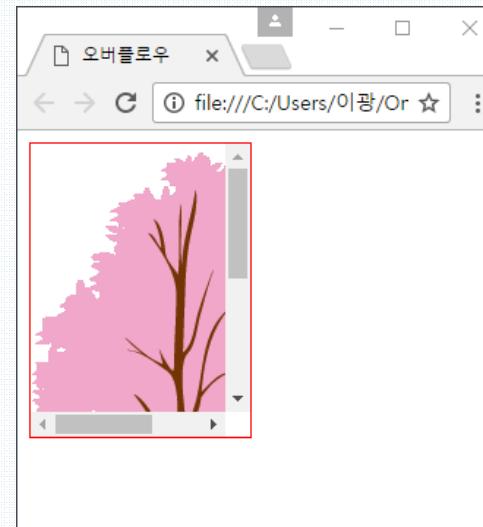
속성값	의미
visible	요소가 영역을 벗어난다 할지라도 화면에 출력 overflow-x나 overflow-y에 지정해도 동일한 효과 발생(방향에 상관없이 모두 출력됨)
inline	요소가 영역을 벗어난 경우 화면에 출력하지 않음 - overflow-x에 적용할 경우, 수평방향으로 벗어난 이미지는 출력되지 않음 - overflow-y에 적용할 경우, 수직방향으로 벗어난 이미지는 출력되지 않음 - overflow에 적용할 경우, 영역의 크기 만큼만 출력하고 나머지는 짤림
scroll	요소의 영역 우측과 하단에 스크롤바를 생성 - overflow-x나 overflow-y에 지정해도 동일한 효과 발생(방향에 상관없이 모두 스크롤바 생성) - 요소의 크기가 영역보다 작을 경우에도 스크롤바가 생성됨 (활성화되지는 않음)
auto	브라우저 기본 설정에 따름 - 요소가 영역을 벗어난 경우에만 영역의 우측과 하단에 스크롤바를 생성 - overflow-x나 overflow-y에 지정해도 동일한 효과 발생
no-display	요소의 크기가 영역보다 작을 경우에 박스를 제거함 - display:none과 동일함 (현재는 브라우저가 지원하지 않음)
no-content	요소의 크기가 영역보다 작을 경우에 박스를 표현하지 않음 - visibility:hidden과 동일 (현재는 브라우저가 지원하지 않음)





## ▣ overflow 관련 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>오버플로우</title>
    <style type="text/css">
        div {
            width:150px; height:200px;
            border:1px solid red;
            overflow-x: auto;
            overflow-y: auto;
        }
    </style>
</head>
<body>
    <div>
        
    </div>
</body>
</html>
```





## ■ visibility 속성

- 요소에 의해 작성된 박스를 화면에 표시할 지의 여부를 지정하는 속성

**visibility : visible | hidden | collapse**

속성값	의미
visible	요소가 영역을 벗어난다 할지라도 화면에 출력
hidden	요소가 영역을 벗어난 경우 화면에 출력하지 않음 - 레이아웃은 존재하며, 단지 보이지만 않음  [참고] display:none을 적용할 경우 레이아웃이 존재하지 않음
collapse	hidden과 유사

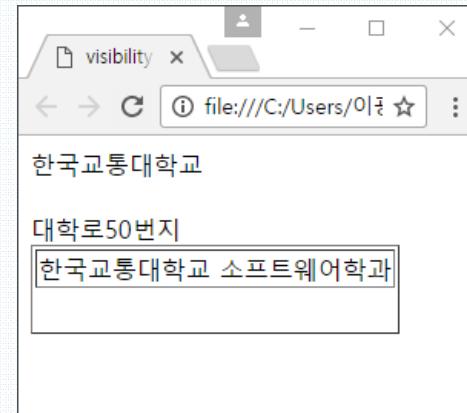




## ■ visibility 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>visibility 속성</title>
    <style type="text/css">

        div#a1 { visibility:hidden; }
        .tab1 { visibility:hidden; }
    </style>
</head>
<body>
    <div>한국교통대학교</div>
    <div id=a1>소프트웨어학과</div>
    <div>대학로50번지</div>
    <table border=1>
        <tr>
            <td>한국교통대학교 소프트웨어학과</td>
        </tr>
        <tr class="tab1" >
            <td>소프트웨어학과</td>
        </tr>
    </table>
</body>
</html>
```





## ■ position 속성

- 레이어의 위치를 지정하기 위한 속성

**position: 기준위치 위치값**

- position 속성은 기준위치를 나타내는 값과 위치를 지정하는 값이 필요함

- 기준위치 값

- : static      - 위치를 지정하지 않음(기본값)
- : relative    - 요소의 원래 위치를 기준
- : absolute    - 브라우저의 좌측 상단을 기준

- 위치 값

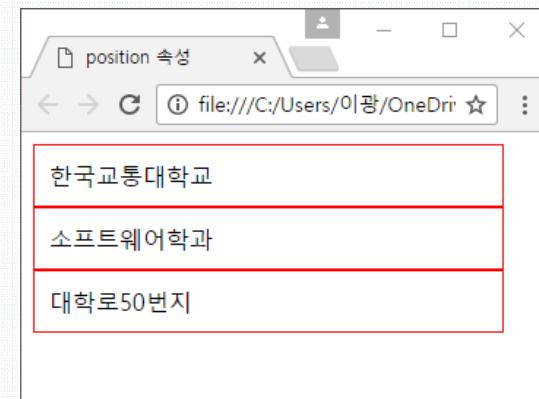
- : left / top      - 기준위치를 기준으로 오른쪽(left)과 아래쪽(top) 옵션 지정
- : right / bottom   - 기준위치를 기준으로 왼쪽(bottom)과 아래쪽(top) 옵션 지정





## ■ position 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>position 속성</title>
    <style type="text/css">
        div { border:solid 1px red;
               margin:0px; padding:10px; width:300px;
        }
    </style>
</head>
<body>
    <div style="position:static">한국교통대학교</div>
    <div style="position:relative">소프트웨어학과</div>
    <div style="position:absolute">대학로50번지</div>
</body>
</html>
```

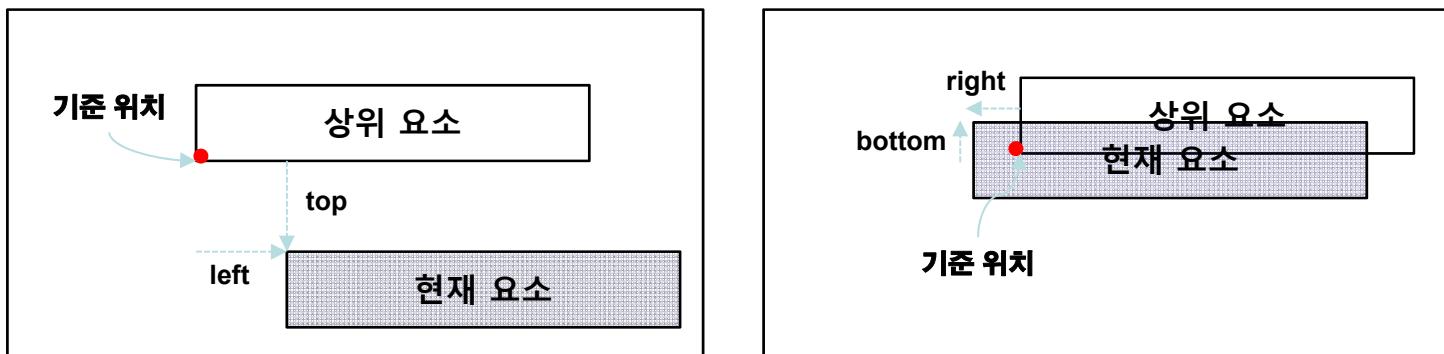




## ■ relative일 경우의 위치지정

- 만약 position 속성을 사용하지 않았을 경우, 현재요소가 나타나야 할 원래의 그 위치가 기준 위치임

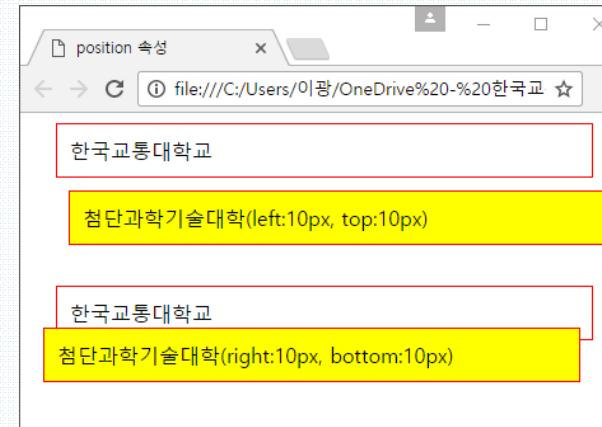
위치값	의미	단위	
left	기준위치를 기준으로 박스의 좌측 상단 모서리 시작점의 오른쪽 옵셋	픽셀(px) 비율(%)	
top	기준위치를 기준으로 박스의 좌측 상단 모서리 시작점의 아래쪽 옵셋		
right	기준위치를 기준으로 박스의 좌측 상단 모서리 시작점의 왼쪽 옵셋		
bottom	기준위치를 기준으로 박스의 좌측 상단 모서리 시작점의 위쪽 옵셋		





## ▣ relative일 경우의 위치지정 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>position 속성</title>
    <style type="text/css">
        div { border:solid 1px red;
               margin-left:20px; padding:10px;
               width:400px;
        }
    </style>
</head>
<body>
    <div style="position:relative;">한국교통대학교</div>
    <div style="position:relative; left:10px; top:10px; background-color:yellow;">
        첨단과학기술대학(left:10px, top:10px)
    </div>
    <br><br>
    <div style="position:relative;">한국교통대학교</div>
    <div style="position:relative; right:10px; bottom:10px; background-color:yellow;">
        첨단과학기술대학(right:10px, bottom:10px)
    </div>
</body>
</html>
```

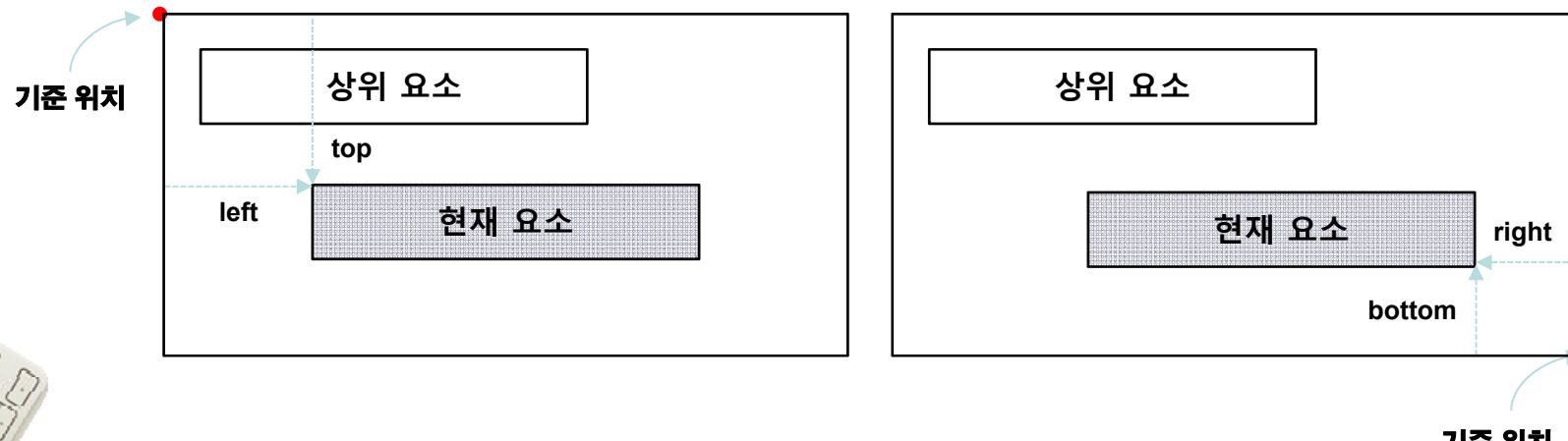




## ■ absolute 일 경우의 위치지정

- 현재요소가 나타나야 할 위치에 관계없이 브라우저의 모서리를 기준으로 함

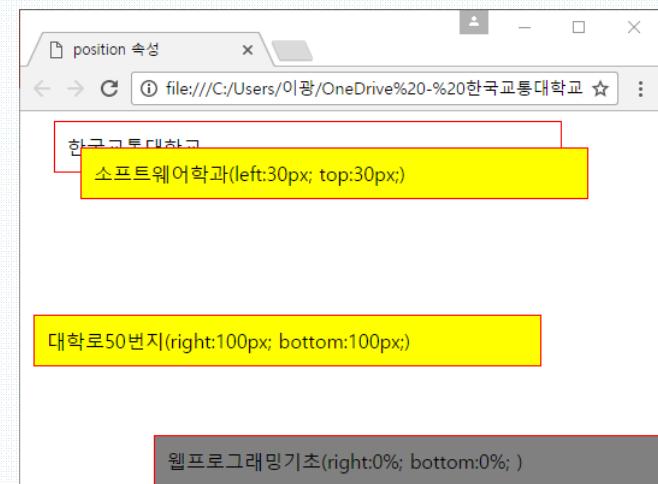
위치값	의미	단위
left	브라우저의 좌측 상단을 기준으로 박스의 좌측 상단 모서리 시작점의 오른쪽 옵셋	픽셀(px) 비율(%)
top	브라우저의 좌측 상단을 기준으로 박스의 좌측 상단 모서리 시작점의 아래쪽 옵셋	
right	브라우저의 우측 하단을 기준으로 박스의 우측 하단 모서리 시작점의 왼쪽 옵셋	
bottom	브라우저의 우측 하단을 기준으로 박스의 우측 하단 모서리 시작점의 위쪽 옵셋	





## ■ absolute 일 경우의 위치지정 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>position 속성</title>
    <style type="text/css">
        div { border:solid 1px red;
               margin-left:20px; padding:10px;
               width:400px;
        }
    </style>
</head>
<body>
    <div style="position:relative;">한국교통대학교</div>
    <div style="position:absolute; left:30px; top:30px; background-color:yellow;">
        소프트웨어학과(left:30px; top:30px; )
    </div>
    <div style="position:absolute; right:100px; bottom:100px; background-color:yellow;">
        대학로50번지(right:100px; bottom:100px; )
    </div>
    <div style="position:absolute; right:0%; bottom:0%; background-color:gray;">
        웹프로그래밍기초(right:0%; bottom:0%; )
    </div>
</body>
</html>
```





## ■ z-index 속성

- 레이어가 겹쳐 표현될 경우 상단에 출력할 레이어의 위치를 지정하는 속성

**z-index : 숫자**

- 출력할 순서는 1부터 시작하는 숫자로 표현

: 숫자가 클수록 위에 위치함

- 예

: `div { position:absolute; width:50px; height:50px; border:2px solid; z-index:2 }`

## ■ direction 속성

- 박스 내 문자의 좌우 정렬 방식을 지정하기 위한 속성

**direction : rtl | ltr**

- rtl : 우측정렬을 수행

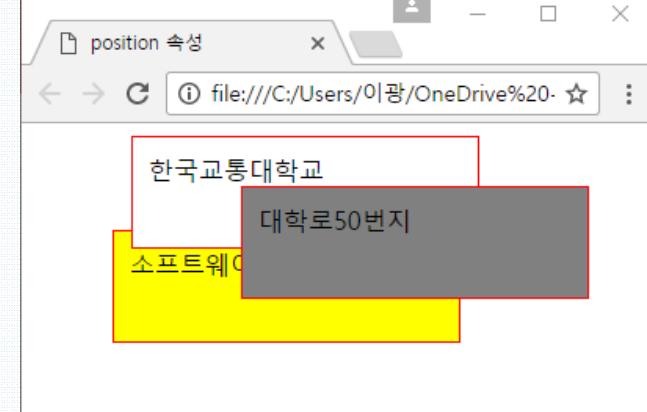
- ltr : 좌측정렬을 수행





## ■ z-index 속성, direction 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>position 속성</title>
    <style type="text/css">
        div { border:solid 1px red;
               margin-left:20px; padding:10px;
               width:200px; height:50px;
        }
    </style>
</head>
<body>
    <div style="position:absolute; left:50px; background-color:white; z-index:2">한국교통대학교</div>
    <div style="position:relative; left:30px; top:60px; background-color:yellow; z-index:1">
        소프트웨어학과
    </div>
    <div style="position:absolute; left:120px; top:40px; background-color:gray; z-index:3">
        대학로50번지
    </div>
</body>
</html>
```



# 박스의 예제



```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>position 속성</title>
    <style type="text/css">

        body{
            margin:0;
            color:white;
            font-family:impact, arial black;
        }
        div{
            font-size:80pt;
            word-spacing:-10pt;
            position:relative; top:70px; left:50px;
        }
        span{
            font-size:30pt;
            position:relative;
        }
    </style>
</head>
<body>
```



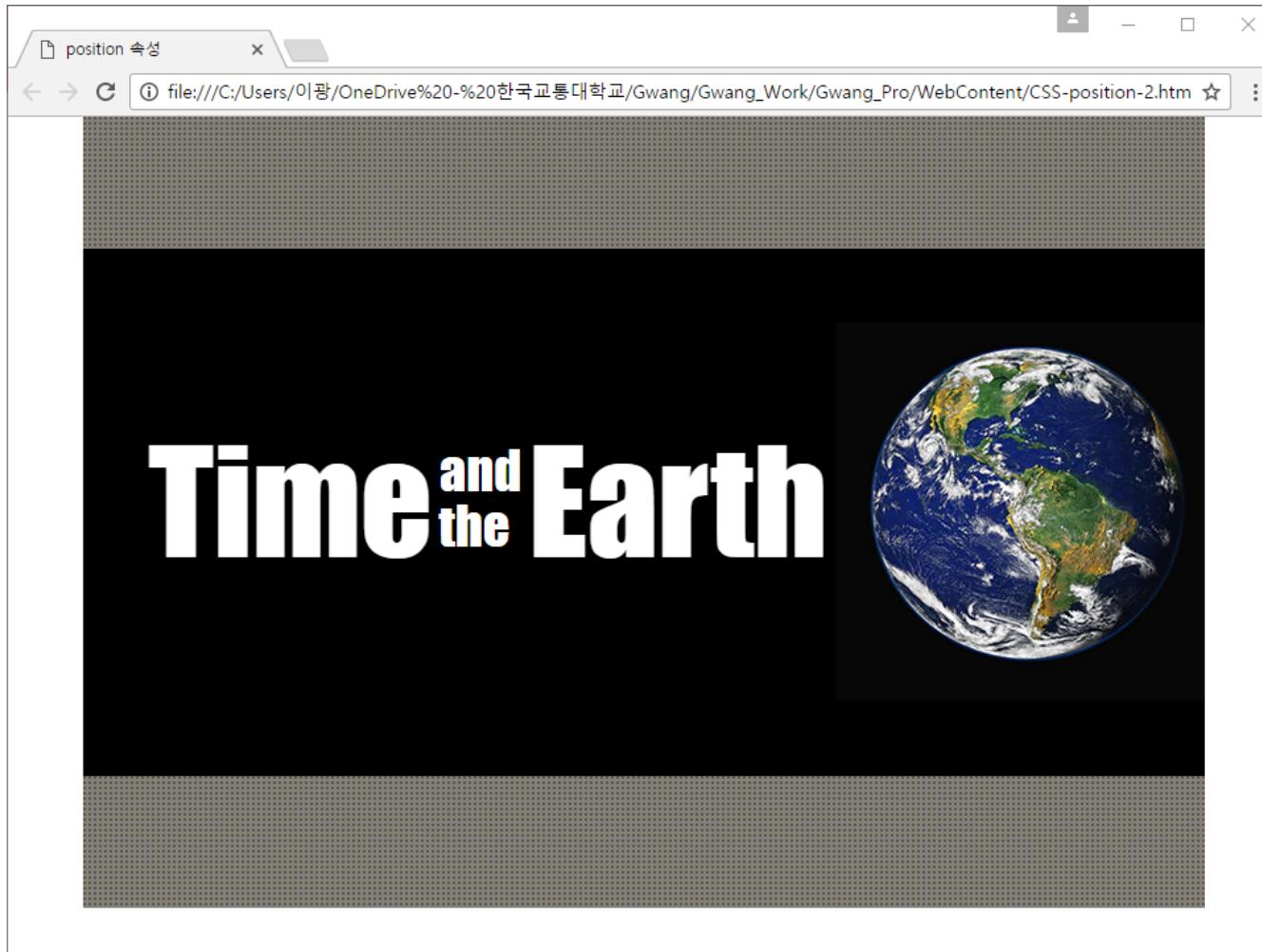
## 박스의 예제



```
<TABLE align=center BORDER=0 CELLPADDING=0 CELLSPACING=0 WIDTH=850px HEIGHT=600px  
background="images/bg.gif">  
  
<TR>  
  <TD HEIGHT=100px>&nbsp;</TD>  
</TR>  
  
<TR>  
  <TD BGCOLOR=BLACK>  
    <IMG SRC="images/earth.jpg" ALIGN=RIGHT>  
    <DIV>Time <SPAN STYLE="top:-50px;">and</SPAN> Earth</DIV>  
    <SPAN STYLE="left:270px">the</SPAN>  
  </TD>  
</TR>  
  
<TR>  
  <TD HEIGHT=100px>&nbsp;</TD>  
</TR>  
  
</TABLE>  
</body>  
</html>
```



## 박스의 예제





박스와 배경 및 색상



## ■ background-color 속성

- 문서 또는 박스 배경의 색상을 지정하기 위한 속성

**background-color : color | transparent**

속성값	의미
transparent	박스 배경이 투명하게 지정됨
color	박스의 배경색을 지정

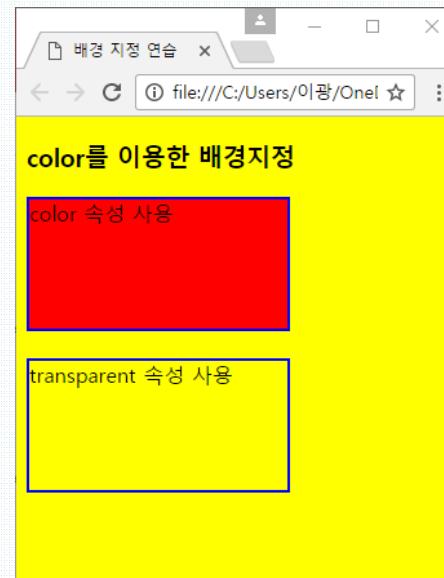




## ■ background-color 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>배경 지정 연습</title>
    <style>
        body { background-color:yellow; }
        .box1 {
            width:200px; height:100px;
            border: 2px solid blue;
            background-color:red;
        }
        .box2 {
            width:200px; height:100px;
            border: 2px solid blue;
            background-color:transparent;
        }
    </style>
</head>

<body>
    <h3>color를 이용한 배경지정</h3>
    <div class="box1">color 속성 사용</div><br>
    <div class="box2">transparent 속성 사용</div>
</body>
</html>
```





## ■ background-image 속성

- 문서 또는 박스의 배경에 이미지를 지정하기 위한 속성

```
background-image : url(이미지이름)
```

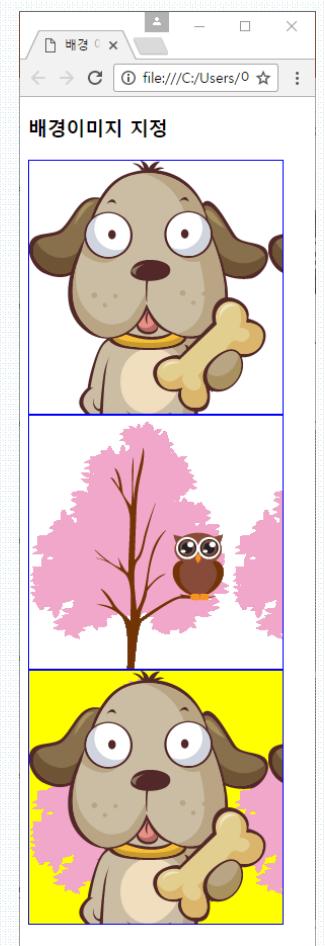
- 반드시 url()을 사용해 지정해야 함
  - [예] background-image : url(dog.png);
- 배경에는 하나 이상의 이미지를 동시에 지정할 수 있음
  - 이 경우 먼저 지정한 이미지들이 상위 계층에 존재하게 됨
  - [예] background-image : url(dog.png), url(tree.gif);  
: dog.png 아래에 tree.gif 이미지가 존재함





## ■ background-image 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>배경 이미지 지정 연습</title>
    <style>
        div {
            width:250px; height:250px;
            border : 1px solid blue;
        }
        .code1 { background-image : url(images/dog.png) }
        .code2 { background-image : url(images/tree.gif); }
        .code3 {
            background-image : url(images/dog.png), url(images/tree.gif);
            background-color:yellow;
        }
    </style>
</head>
<body>
    <h3>배경이미지 지정</h3>
    <div class="code1"></div>
    <div class="code2"></div>
    <div class="code3"></div>
</body>
</html>
```





## ■ background-repeat 속성

- 문서 또는 박스의 배경에 이미지의 반복 출력 방법을 지정하기 위한 속성

**background-repeat : repeat-x | repeat-y | [repeat | space | round | no-repeat]**

속성값	의미
repeat	배경 이미지를 가로 및 세로 방향으로 반복하도록 지정
no-repeat	배경 이미지를 반복하지 않도록 지정
repeat-x	배경 이미지를 가로 방향으로 반복하도록 지정
repeat-y	배경 이미지를 세로 방향으로 반복하도록 지정
space	배경을 채우는 영역의 경계에서 배경 이미지가 잘려 보이지 않도록 반복 이미지간의 공백을 조정 <ul style="list-style-type: none"><li>- background-position 속성 값의 방향은 무시됨</li><li>만약, 2개의 이미지가 반복되기에 충분한 공간이 없을 경우<ul style="list-style-type: none"><li>- 배경 이미지는 한번만 출력됨</li><li>- background-position 속성에 의해 배경 이미지의 위치가 결정됨</li></ul></li></ul>
round	space 속성과 가능한 유사 <ul style="list-style-type: none"><li>- 배경 이미지가 잘려 보이지 않게 처리하기 위하여 배경 이미지의 크기가 재설정</li></ul>





## ■ background-repeat 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>배경 이미지 지정 연습</title>

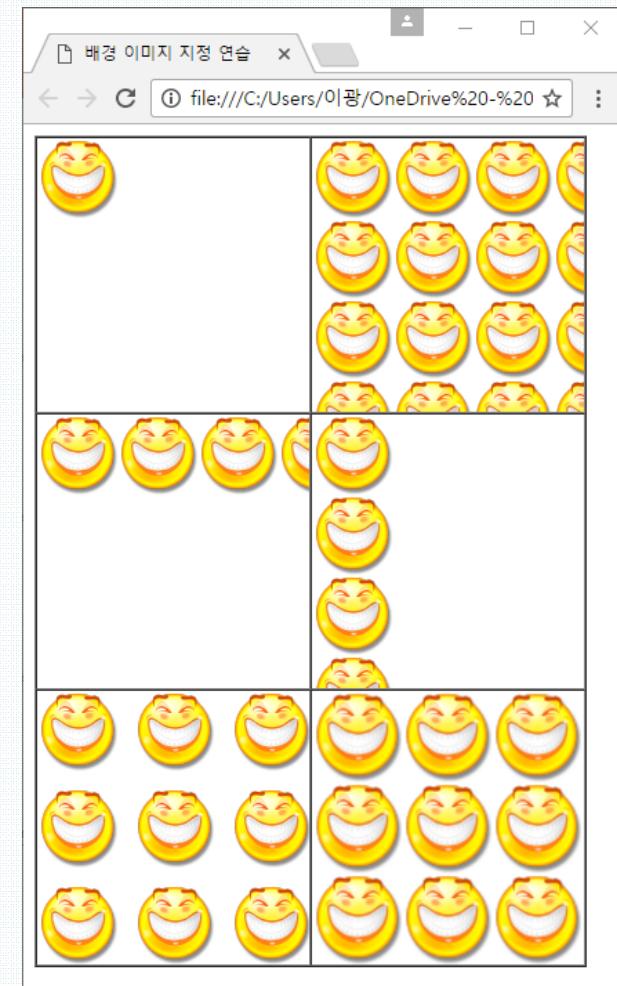
    <style>
        td { background-image:url(images/smile.png); }
        td#code1 { background-repeat : no-repeat; }
        td#code2 { background-repeat : repeat; }
        td#code3 { background-repeat : repeat-x; }
        td#code4 { background-repeat : repeat-y; }
        td#code5 { background-repeat : space; }
        td#code6 { background-repeat : round; }
    </style>
</head>
<body>
    <table width=400 border=1 cellspacing=0 cellpadding=0>
        <tr height=200>
            <td width=200 id="code1"></td>
            <td width=200 id="code2"></td>
        </tr>
        <tr height=200>
            <td id="code3"></td>
            <td id="code4"></td>
        </tr>
    </table>
</body>
```



## 박스의 배경 이미지 반복



```
<tr height=200>
  <td id="code5"></td>
  <td id="code6"></td>
</tr>
</table>
</body>
</html>
```





## ■ background-attachment 속성

- 페이지 스크롤 발생 시 배경 이미지를 스크롤할지 고정 할지를 지정하기 위한 속성

**background-attachment : scroll | attachment**

속성값	의미
scroll	콘텐츠와 배경 이미지가 함께 스크롤됨
fix	콘텐츠가 스크롤되더라고 배경 이미지는 고정됨





## ■ background-position 속성

- 문서 또는 박스에서 배경 이미지의 위치를 지정하기 위한 속성

**background-position : 가로위치 세로위치**

속성값	의미
가로위치	<p>이미지의 수평방향 위치 지정</p> <ul style="list-style-type: none"><li>- left, center, right를 가질 수 있음</li><li>- left(왼쪽 정렬), center(중앙 정렬), right(우측 정렬)</li></ul> <p>필셀 값이나 퍼센트 값을 가질 수 있음</p>
세로위치	<p>이미지의 수직방향 위치 지정</p> <ul style="list-style-type: none"><li>- top, center, bottom 를 가질 수 있음</li><li>- top(상단 정렬), center(중앙 정렬), bottom(하단 정렬)</li></ul> <p>필셀 값이나 퍼센트 값을 가질 수 있음</p>



# 박스의 배경 이미지 위치



- **background-position 속성 값 지정**

- 위치를 0% 0%로 지정할 경우

- : 이미지의 좌측 상단 모서리가 영역의 좌측 상단에 위치

- 위치를 100% 100%로 지정할 경우

- : 이미지의 우측 하단 모서리가 영역의 좌측 상단에 위치

- 가로위치 값과 세로위치 값의 순서는 실행과 무관함

- : background-position : right bottom = background-position : bottom right

- 가로위치 값과 세로위치 값 중 하나만 입력할 경우

- : 가로 위치를 지정한 것으로 인식

- : 세로 위치는 중앙(center)로 인식

- : 이미지를 정 중앙에 위치시킬 경우 center center 대신 center만 입력 가능

- 100%를 이상의 값을 사용하거나, 실제 모니터의 해상도보다 큰 픽셀 지정 가능

- : 이 경우 브라우저를 벗어나는 이미지의 부분이 잘림



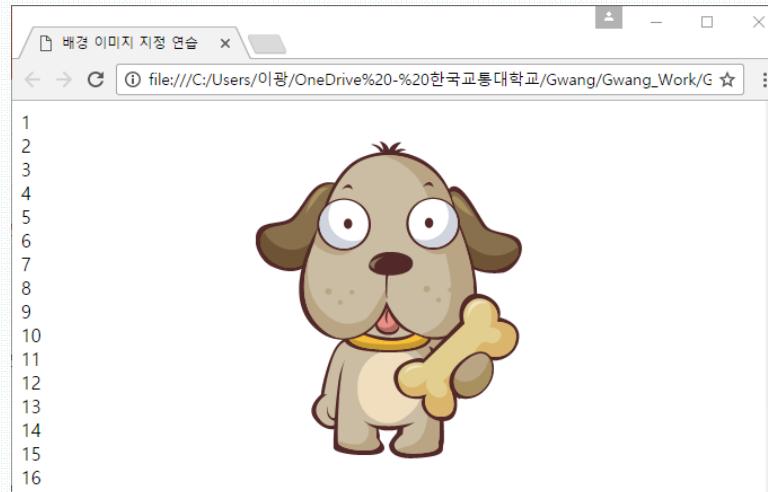


## ■ background-attachment, background-position 속성의 예

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>배경 이미지 지정 연습</title>

<style>
    body {
        background-repeat:no-repeat;
        background-attachment:fixed;
        background-position: center center;
    }
</style>

</head>
<body style="background-image:url(images/dog.png)">
    1<br>2<br>3<br>4<br>5<br>6<br>7<br>8<br>9<br>10<br>
    11<br>12<br>13<br>14<br>15<br>16<br>17<br>18<br>19<br>20<br>
</body>
</html>
```





## ■ background-clip 속성

- 박스에서 배경이 적용되는 영역을 지정하기 위한 속성

**background-clip : border-box | padding-box | content-box**

속성값	의미
border-box	배경 속성이 적용되는 영역을 테두리 영역까지 표시하도록 지정
padding-box	배경 속성이 적용되는 영역을 패딩 영역까지 표시하도록 지정
content-box	배경 속성이 적용되는 영역을 콘텐츠 영역까지 표시하도록 지정

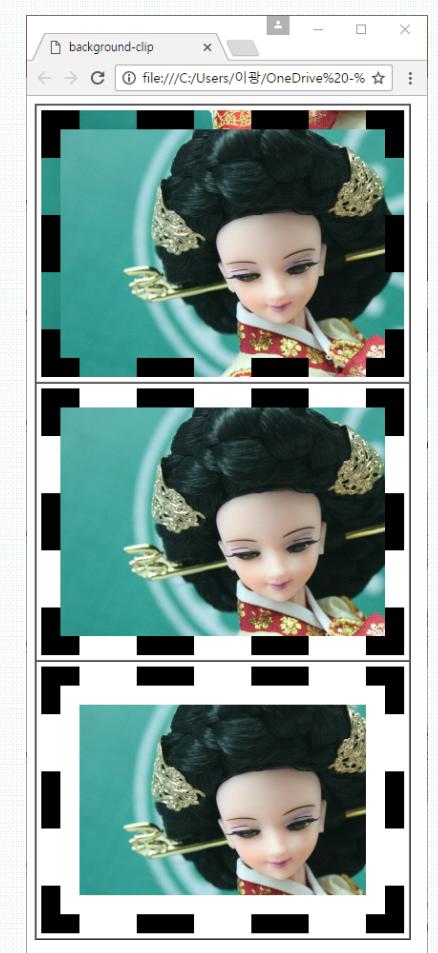




## ■ background-clip 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>background-clip</title>
    <style type="text/css">
        div {width: 300px; height: 200px;
            border: 20px dashed; padding: 20px;
            background: url(images/doll.jpg) 0 0 repeat;
        }

        div#border { background-clip: border-box; }
        div#padding { background-clip: padding-box; }
        div#content { background-clip: content-box; }
    </style>
</head>
<body>
    <table border=1 cellpadding=5 cellspacing=0>
        <tr><td><div id="border"></div></td></tr>
        <tr><td><div id="padding"></div></td></tr>
        <tr><td><div id="content"></div></td></tr>
    </table>
</body>
</html>
```





## ■ background-origin 속성

- 박스에서 배경 이미지의 시작 위치를 지정하기 위한 속성

**background-origin : border-box | padding-box | content-box**

속성값	의미
border-box	배경 이미지의 시작 위치를 테두리 박스의 상대적인 위치로 지정
padding-box	배경 이미지의 시작 위치를 패딩 박스의 상대적인 위치로 지정
content-box	배경 이미지의 시작 위치를 콘텐츠 박스의 상대적인 위치로 지정



# 박스의 배경 이미지 시작 위치



## ■ background-origin 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>background-origin</title>
    <style type="text/css">
        div {width: 300px; height: 200px;
            border: 20px dashed; padding: 20px;
            background: url(images/doll.jpg);
            background-position: 0% 0%;
            background-repeat: no-repeat;
        }

        div#border {background-origin: border-box; }
        div#padding {background-origin: padding-box; }
        div#content {background-origin: content-box; }
    </style>
</head>
<body>
    <table border=1 cellpadding=5 cellspacing=0>
        <tr><td><div id="border"></div></td></tr>
        <tr><td><div id="padding"></div></td></tr>
        <tr><td><div id="content"></div></td></tr>
    </table>
</body>
</html>
```





## ■ background-size 속성

- 박스 내 배경 이미지의 크기를 지정하기 위한 속성

**background-size : 가로크기 세로크기**

**background-size : auto | cover | contain**

- 가로크기와 세로크기를 지정하기 위해 pixel, percentage를 사용할 수 있음

속성값	의미
auto	이미지를 기본 크기로 지정(기본값)
length	길이를 지정하여 이미지의 크기를 직접 지정 (단위로 픽셀을 사용)
percentage	이미지의 크기를 영역의 비율로 지정 (영역에 가득 채우려면 모두 100%로 지정)
cover	요소 박스 영역 전체에 배경 이미지를 채우도록 지정 배경 이미지가 가지고 있는 원래 크기의 비율을 무시
contain	요소 박스 영역 전체에 배경 이미지를 채우도록 지정 배경 이미지가 가지고 있는 원래 크기의 비율을 유지한





## ■ background-size 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>background-size</title>
    <style type="text/css">
        div {width: 200px; height: 150px;
            border: 1px solid blue;
            background: url(images/dog.png) 0 0 no-repeat;
        }
        div#auto { background-size: auto; }
        div#length { background-size: 50px 50px; }
        div#psnt { background-size: 50% 50%; }
        div#cover { background-size: cover; }
        div#contain { background-size: contain; }

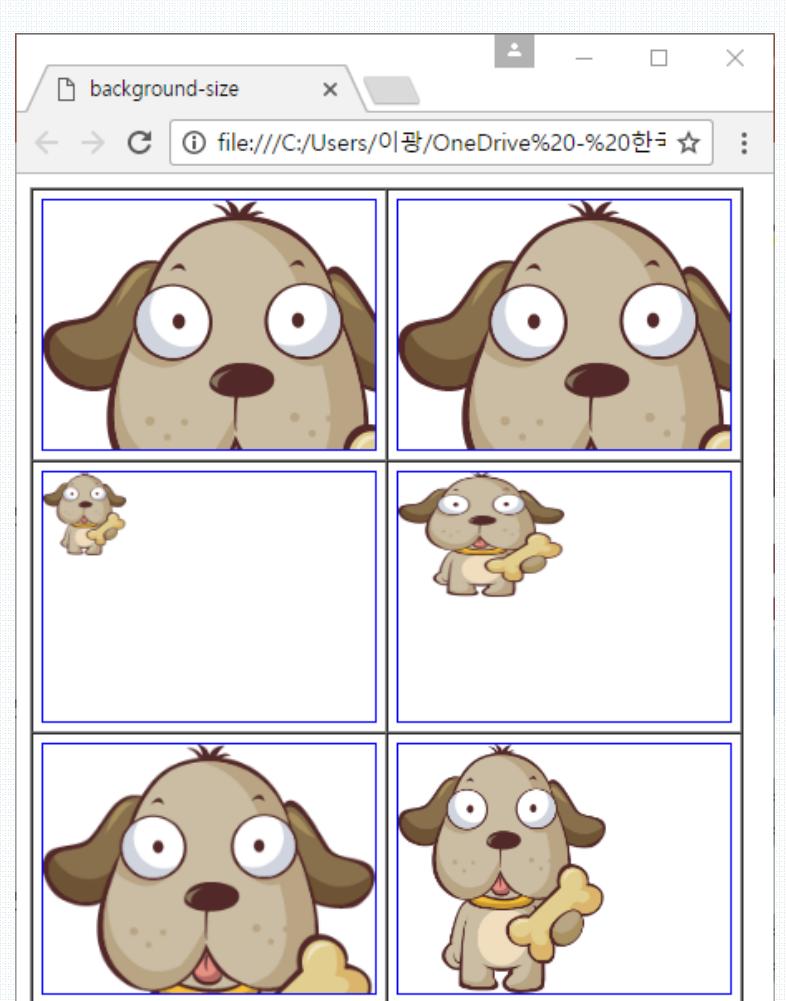
    </style>
</head>
<body>
    <table border=1 cellpadding=5 cellspacing=0>
        <tr>
            <td><div></div></td>
            <td><div id="auto"></div></td>
        </tr>
    </table>
</body>
</html>
```



# 박스의 배경 이미지 크기



```
<tr>
  <td><div id="length"></div></td>
  <td><div id="psnt"></div></td>
</tr>
<tr>
  <td><div id="cover"></div></td>
  <td><div id="contain"></div></td>
</tr>
</table>
</body>
</html>
```





## ■ background 속성

- 배경 이미지의 모든 속성을 일괄 지정하기 위한 속성

background : 이미지 위치 / 크기 반복여부 스트롤여부

- 여러 개의 이미지를 동시에 지정할 수 있음

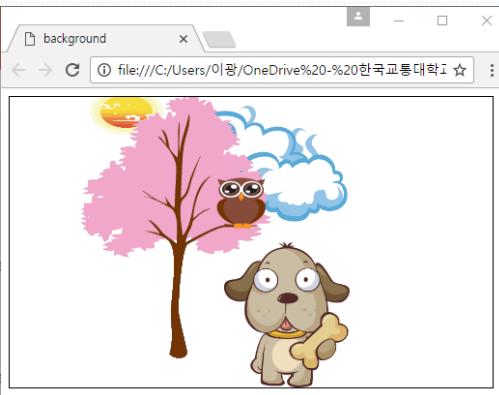
```
div {  
    background : url('images/dog.png') 60% bottom / 130px 150px no-repeat scroll,  
            url('images/img_tree.gif') 50% 15% / 200px 280px no-repeat fixed,  
            url('images/cloud.png') center 5% / 40% 40% no-repeat scroll,  
            url('images/moon.png') 40% 5% / 15% 15% no-repeat fixed;  
}
```

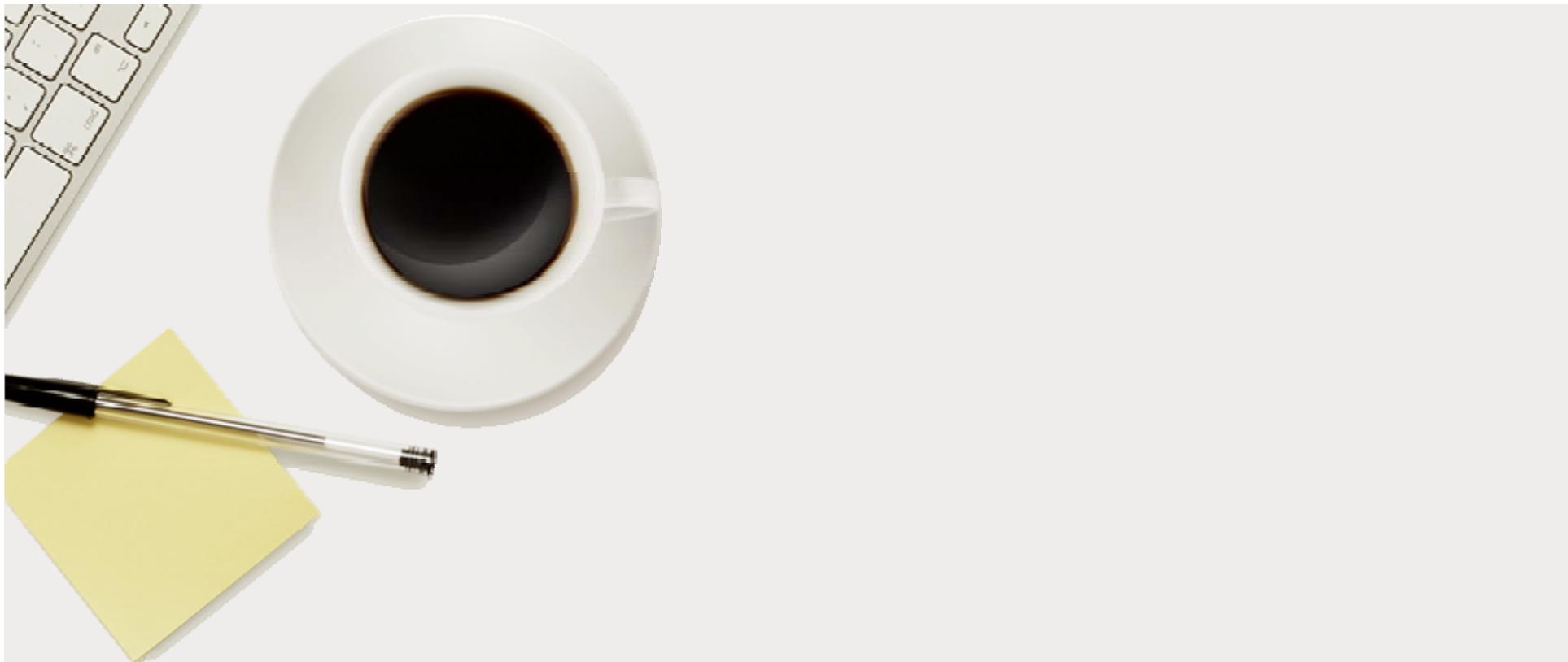




## ■ background 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>background</title>
    <style type="text/css">
        div{
            height:300px; width:500px;
            border:1px solid #000;
            background : url(images/dog.png) 60% bottom / 130px 150px no-repeat scroll,
                        url(images/tree.gif) 25% 5% / 200px 280px no-repeat fixed,
                        url(images/cloud.png) center 5% / 40% 40% no-repeat scroll,
                        url(images/moon.png) 20% 0% / 15% 15% no-repeat fixed;
        }
    </style>
</head>
<body>
    <div></div>
</body>
</html>
```





## 박스 모서리 모양 지정



## ■ border-radius 속성

- 박스의 각 모서리를 둥글게 지정하는 속성
  - 값은 라운드의 반지름을 의미하므로 값이 클수록 라운드도 커짐
    - : 값은 픽셀이나 퍼센트를 사용

border-radius : 값

- border-radius에 값을 하나만 지정할 경우
  - : 박스의 모든 모서리가 같은 값으로 지정됨

border-radius : 값

- border-radius에 네개의 값을 하나만 지정할 경우
  - : 박스의 각 모서리를 별도의 값으로 지정함
  - : 왼쪽상단모서리 → 오른쪽상단모서리 → 오른쪽하단모서리 → 왼쪽하단모서리





## ▣ 서로 다른 둥근 모서리를 지정하는 방법

- [방법-1] border-radius에 4개 모서리의 값을 스페이스로 구분하여 지정
  - 왼쪽 상단 모서리부터 시계방향으로 값을 지정함
- [방법-2] 각 모서리를 지정하기 위한 별도의 속성 사용
  - border-top-left-radius : 왼쪽 상단 모서리의 지정
  - border-bottom-left-radius : 왼쪽 하단 모서리의 지정
  - border-top-right-radius : 오른쪽 상단 모서리의 지정
  - border-bottom-right-radius : 오른쪽 하단 모서리의 지정

## ▣ 원형이 아닌 형태(타원 형태)의 모서리 지정 방법

- border-radius 가로크기/세로크기
  - [예] border-radius 10px / 20px

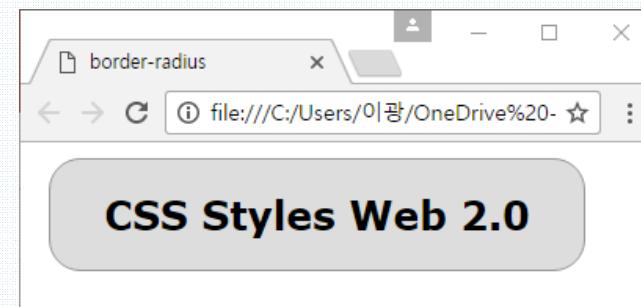




## ■ border-radius 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>border-radius</title>
    <style>
        body { text-align: center; }
        #mainbox {
            width:300px;
            margin:10px; padding:20px;
            text-align: center;
            border: 1px solid #999999;
            background: #DDDDDD;
            border-radius: 20px;
        }
        #title { font: bold 25px verdana, sans-serif; }
    </style>
</head>

<body>
    <header id="mainbox">
        <div id="title">CSS Styles Web 2.0</div>
    </header>
</body>
</html>
```

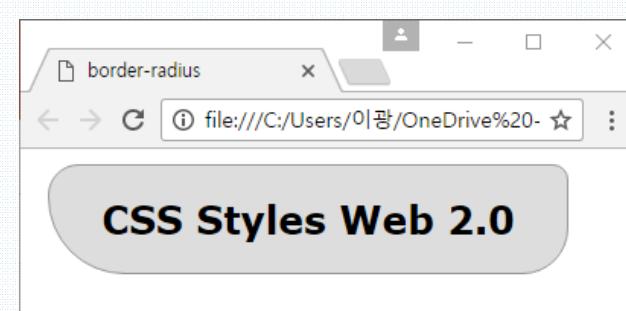




## ■ border-radius 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>border-radius</title>
    <style>
        body { text-align: center; }
        #mainbox {
            width:300px;
            margin:10px; padding:20px;
            text-align: center;
            border: 1px solid #999999;
            background: #DDDDDD;
            border-radius: 20px 10px 30px 50px; ;
        }
        #title { font: bold 25px verdana, sans-serif; }
    </style>
</head>

<body>
    <header id="mainbox">
        <div id="title">CSS Styles Web 2.0</div>
    </header>
</body>
</html>
```

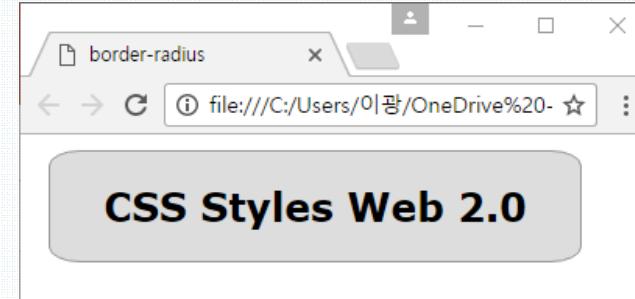




## ■ border-radius 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>border-radius</title>
    <style>
        body { text-align: center; }
        #mainbox {
            width:300px;
            margin:10px; padding:20px;
            text-align: center;
            border: 1px solid #999999;
            background: #DDDDDD;
            border-radius: 20px / 10px;
        }
        #title { font: bold 25px verdana, sans-serif; }
    </style>
</head>

<body>
    <header id="mainbox">
        <div id="title">CSS Styles Web 2.0</div>
    </header>
</body>
</html>
```





## ■ box-shadow 속성

- 박스에 그림자 효과를 지정하기 위한 속성
  - 그림자의 색상, 가로 위치, 세로 위치, 놓도 방향을 지정

box-shadow : 색상 가로offset 세로offset 번짐정도 [inset]

속성값	의미
색상	그림자의 색상을 지정 ( HTML 색상지정방식 사용 가능 )
가로offset	그림자의 가로 옵셋 지정 ( 그림자의 크기를 픽셀 값으로 지정 )
세로offset	그림자의 가로 옵셋 지정 ( 그림자의 크기를 픽셀 값으로 지정 )
번짐 정도	그림자의 경계선의 흐림 정도를 픽셀로 표현
inset	내부 그림자 효과를 지정할 때 사용





## ■ box-shadow 속성의 예

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>border-shadow</title>
    <style>
        div {
            border-radius : 10px;
            width:200px; height:100px; border: 2px solid blue;
            background-color:red;
        }
        .box1 { box-shadow : rgb(150,150,150) 10px 10px; }
        .box2 { box-shadow : rgb(150,150,150) 10px 10px 10px; }
        .box3 { box-shadow : rgb(150,150,150) 10px 10px inset; }
    </style>
</head>
<body>
    <div class="box1"></div><br>
    <div class="box2"></div><br>
    <div class="box3"></div>
</body>
</html>
```

