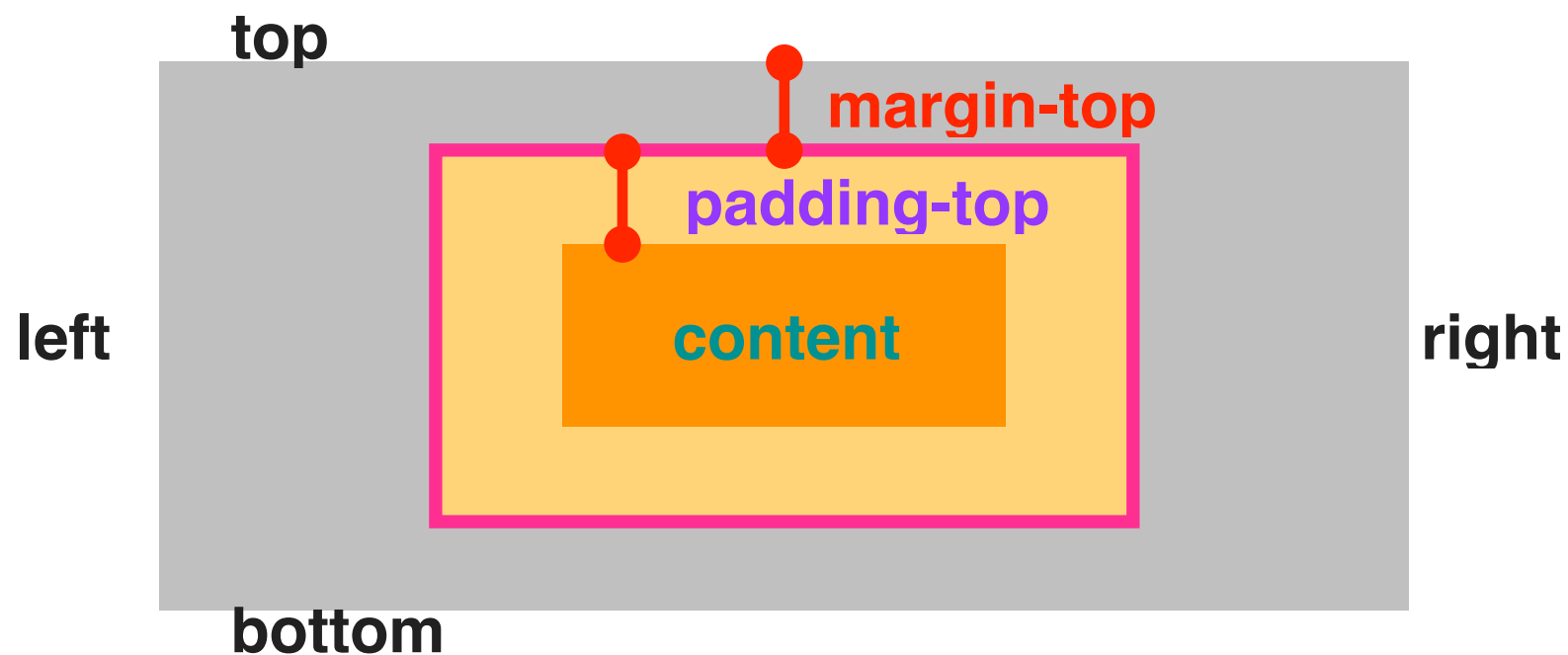


❖ 박스 모델

- 웹 문서에 텍스트, 이미지, 테이블 등의 요소를 배치하기 위해 사용
- 웹 문서의 전체 레이아웃을 정의
- 각종 요소들을 원하는 위치에 배치

❖ 박스의 속성

- **content** : 실제 내용이 표현되는 곳
- **padding** : 콘텐츠와 테두리 사이의 여백
- **border** : 박스의 테두리 두께
- **margin** : 테두리와 박스의 최종 경계 사이의 여백



❖ 박스 모델

box01.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Box Model</title>
  <style>
    div {
      background-color: yellow;
      width: 300px;
      padding: 25px;
      border: 15px solid navy;
      margin: 25px;
    }
  </style>
</head>
<body>
  <p>CSS3 박스 모델은 content, padding, border, margin으로 구
성되어 있다.</p>
  <div>박스 모델의 padding, border, margin 속성의 기본 값은 0이
며, 상하좌우 네 가지 방향을 top, bottom, left,
right를 이용하여 지정할 수 있습니다.</div>
</body>
</html>
```

❖ Content

- 실제 내용이 표현되는 곳
- 속성: width(너비 지정), height(높이 지정)

속성	설명
value	실제로 측정한 데이터 값이다.
min, max	데이터가 인식하는 최소값과 최대값이다. 기본값은 0~1이다.
low, high	허용되는 범위의 최소값과 최대값이다. low~high 값은 항상 min~max 값 범위 내에 있다.

❖ Content

content01.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Box Model</title>
  <style>
    p {
      background-color: yellow;
      color: red;
      font-weight: bold;
      font-size: 16pt;
    }
    p.c1 {
      width: 700px;
      height: 80px;
      color: green;
    }
    p.c2 {
      width: 50%;
      height: 150%;
      color: purple;
    }
    p.c3 {
      width: 10cm;
      height: 3cm;
      color: blue;
    }
  </style>
</head>
<body>
  <p>박스 모델의 내용 영역 크기 지정</p>
  <p class="c1">(1) 박스 모델의 크기를 픽셀(px) 단위로 지정</p>
  <p class="c2">(2) 박스 모델의 크기를 퍼센트(%) 단위로 지정</p>
  <p class="c3">(3) 박스 모델의 크기를 센티미터(cm) 단위로 지정</p>
</body>
```

❖ padding, margin

- padding : 박스의 안쪽 여백 설정
- Margin : 박스의 바깥쪽 여백 설정

속성값	설 명
수 치	여백을 픽셀(px), 포인트(pt), 센티미터(cm), 배수(em) 같은 단위로 지정
백분율	여백을 부모 요소를 기준으로 하여 백분율로 지정
auto	여백을 웹 브라우저가 자동으로 지정 / 기본값

❖ Content

content01.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Content</title>
  <style>
    p {
      background-color: yellow;
      color: red;
      font-weight: bold;
      font-size: 16pt;
    }
    p.pad {
      color: purple;
      padding: 20px;
    }
    p.mar {
      color: green;
      margin: 30px;
    }
    p.mp {
      color: blue;
      padding: 5%;
      margin: 5%;
    }
  </style>
</head>
<body>
  <p>박스의 안쪽 여백과 바깥쪽 여백 지정</p>
  <p class="pad">(1) 안쪽 여백 지정 - padding 20px</p>
  <p class="mar">(2) 바깥쪽 여백 지정 - margin 30px</p>
  <p class="mp">(3) 안쪽, 바깥쪽 여백 지정 - padding 5%, margin 5%</p>
</body>
</html>
```

❖ 속성값의 개수에 따라 적용되는 위치

```
h1 { margin: 5px      10px      5px      10px; }
```

top right bottom left

```
h1 { margin: 5px      10px      5px ; }
```

top right & left bottom

```
h1 { margin: 5px      10px; }
```

top & bottom right & left

```
h1 { margin: 5px; }
```

top & bottom & right & left

❖ Content

content02.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Content02</title>
  <style>
    body { font-weight: bold; font-size: 16pt; }
    .mp1 { background-color: aqua;
           color: red; margin-bottom: 40px;
           padding-left: 50px; }
    .mp2 { background-color: silver; color: green;
           margin: 20px; padding: 10px, 20px; }
    .mp3 { background-color: gray; color: purple;
           margin: 50px, 15px, 20px; padding: 20px; }
  </style>
</head>
<body style="background-color: lightyellow;">
  <p>박스 모델의 네 방향 여백 지정</p>
  <p class="mp1">mp1</p>
  <p class="mp2">mp2</p>
  <p class="mp3">mp3</p>
</body>
</html>
```


❖ border-width

- 테두리 두께 설정
- Top, bottom, left, right를 이용하여 네 방향으로 설정 가능

속성값	설 명
수 치	여백을 픽셀(px), 포인트(pt), 센티미터(cm) 같은 단위로 지정
thin	얇은(1px) 두께의 테두리 지정
medium	중간(3px) 두께의 테두리 지정
thick	굵은(5px) 두께의 테두리 지정

❖ border-color

- ◆ 테두리 색상 지정 (W3C의 칼라 코드 표 참고)

❖ border-style

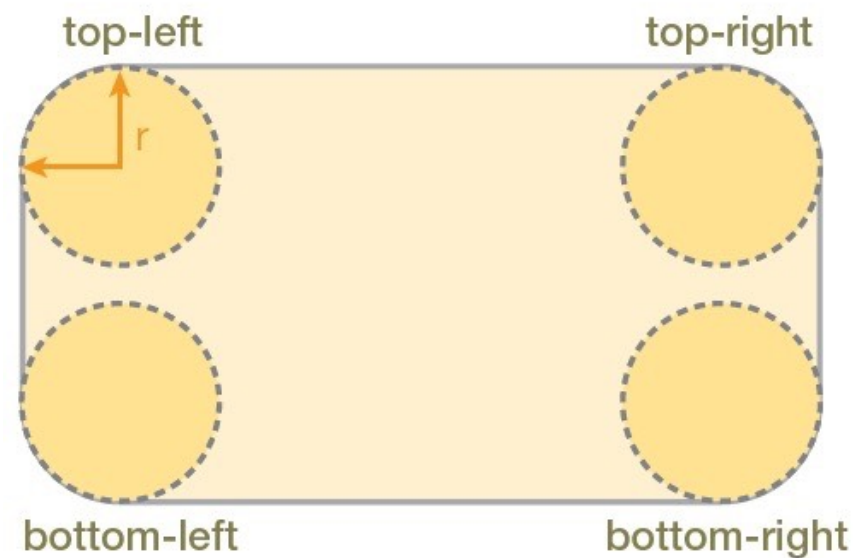
- 테두리 스타일 지정

속성값	설 명
none	테두리가 나타나지 않는다. 기본값
hidden	테두리를 감춘다.
dotted	테두리를 점선으로 지정
dashed	테두리를 파선으로 지정
solid	테두리를 실선으로 지정
double	테두리를 이중선으로 지정
groove	테두리를 오목한 선(홈이 파인듯 입체적으로) 지정
ridge	테두리를 볼록한 선(튀어 나온듯 입체적으로) 지정
inset	테두리 안쪽이 오목한 선으로 지정
outset	테두리 안쪽이 볼록한 선으로 지정

❖ border-radius

- 테두리의 모서리를 둥글게 설정

형 식	설 명
none	테두리가 나타나지 않는다. 기본값
hidden	테두리를 감춘다.
dotted	테두리를 점선으로 지정
dashed	테두리를 파선으로 지정
solid	테두리를 실선으로 지정



❖ Border

border01.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Border01</title>
  <style>
    body { font-weight: bold; font-size: 12pt; }
    .br1 { background-color: lime; color: red;
          border-style: solid; border-width: thick;
          border-color: green; border-radius: 30%; }
    .br2 { background-color: maroon; color: yellow;
          border-style: dotted; border-width: 2px;
          border-color: white; border-radius: 15px 35px; }
    .br3 { background-color: teal; color: aqua;
          border-style: dashed; border-width: 5px;
          border-color: red; border-radius: 5px 15px 25px 35px; }
    .br4 { border: 3px solid red;
          border-top-left-radius: 30px; }
  </style>
</head>
<body>
  <h1 class="br1">둥근 모서리 스타일링 예제 1</h1>
  <h1 class="br2">둥근 모서리 스타일링 예제 2</h1>
  <h1 class="br3">둥근 모서리 스타일링 예제 3</h1>
  <h1 class="br4">둥근 모서리 스타일링 예제 4</h1>
</body>
```

❖ box-shadow

- 박스에 그림자 효과 적용
- 수평 그림자(h-shadow): 그림자의 수평 거리 지정
- 수직 그림자(v-shadow): 그림자의 수직 거리 지정
- 그림자 흐림(blur): 그림자의 흐림 정도 지정
- 그림자 번짐(spread): 그림자의 번짐 정도 지정
- 그림자 색상(color): 그림자의 색상 지정
- 삽입 효과(inset): 박스 외부로 표현되는 그림자를 박스 안쪽으로 표현하는 효과

```
{ box-shadow: 수평그림자(필수) | 수직그림자(필수) | 그림자흐림 |  
              그림자번짐 | 그림자 색상 | 삽입효과; }
```

❖ Box-shadow

box-shadow01.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Box Shadow 1</title>
  <style>
    body { font-weight: bold; font-size: 12pt; }
    .boxshadow1 { background-color: yellow;
      border: 5px solid blue; box-shadow: 10px 10px 0px teal; }
    .boxshadow2 { background-color: orange;
      border: 5px solid red; box-shadow: 20px 20px 50px red; }
    .boxshadow3 { background-color: silver;
      border: 5px solid black;
      box-shadow: 20px 20px 30px -20px maroon; }
    .boxshadow4 { background-color: lime;
      border: 5px solid olive;
      box-shadow: 10px 10px 0px 10px fuchsia inset;
    }
  </style>
</head>
<body>
  <h1 class="boxshadow1">박스 그림자 스타일링 예제 1</h1>
  <h1 class="boxshadow2">박스 그림자 스타일링 예제 2</h1>
  <h1 class="boxshadow3">박스 그림자 스타일링 예제 3</h1>
  <h1 class="boxshadow4">박스 그림자 스타일링 예제 4</h1>
</body>
```

❖ Box-shadow

box-shadow02.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Box Shadow 2</title>
  <style>
    img {
      padding: 20px;
      margin: 20px;
    }
    .shadow1 { /* Bottom right coner */
      box-shadow: 5px 5px 10px #000;
    }
    .shadow2 { /* Up right coner */
      box-shadow: 5px -5px 10px #000;
    }
  </style>
</head>
<body>
  
  
</body>
</html>
```

❖ position

- 텍스트, 이미지, 표 등의 요소를 웹 문서에 배치할 때 사용하는 속성

구분	속성값	설명
정적 위치 설정	position: static;	각종 요소를 웹 문서의 흐름에 따라 배치한다.
상대 위치 설정	position: relative;	웹 문서의 정상적인 위치에서 상대적으로 얼마나 떨어져 있는지 표시하여 배치하는 방법이다.
절대 위치 설정	position: absolute;	전체 페이지를 기준으로 top, right, bottom, left의 속성을 이용하여 원하는 위치에 배치하는 방법이다.
고정 위치 설정	position: fixed;	요소의 위치를 '절대 위치 설정'과 똑같은 방법으로 배치하되, 창이 스크롤을 움직여도 사라지지 않고 고정된 위치에 그대로 있다.

❖ position

- 정적위치

position_static.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Position Static</title>
  <style>
    body { font-weight: bold; font-size: 12pt; }
    .sp1 { position: static; top: 100px; /* 적용되지 않음 */
          background-color: cyan; width: 400px; height: 50px; }
    .sp2 { position: static; left: 30px; /* 적용되지 않음 */
          background-color: orange; width: 400px; height: 50px; }
    .sp3 { background-color: lightgreen;
          width: 400px; height: 50px; }
  </style>
</head>
<body>
  <h1>positioning style1</h1>
  <p class="sp1">정적 위치 설정 적용1</p>
  <div class="sp2">정적 위치 설정 적용2</div>
  <p class="sp3">기본 위치 설정</p>
</body>
</html>
```

❖ position - 상대위치

position_relative.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Position Relative</title>
</head>
<style>
  body { font-weight: bold; font-size: 12pt; }
  .sp { position: static; left: 30px; /* 적용되지 않음 */
        background-color: cyan; width: 400px; height: 50px; }
  .rp1 { position: relative; left: 30px; top: -10px;
        background-color: orange; width: 400px; height: 50px; }
  .rp2 { position: relative; left: 60px; top: 20px;
        background-color: lightgreen; width: 400px;
        height: 50px; }
</style>
</head>
<body>
  <h1>positioning style2</h1>
  <p class="sp">정적 위치 설정 적용</p>
  <div class="rp1">상대 위치 설정 적용 - left 30px, top -10px</div>
  <p class="rp2">상대 위치 설정 적용 - left 60px, top 20px</p>
</body>
</html>
```

❖ position - 절대위치

position_absolute.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Position Absolute</title>
</head>
<style>
  body { font-weight: bold; font-size: 12pt; }
  .ap1 { position : absolute; left: 30px; top: 70px;
        background-color: yellow; width: 400px; height: 50px; }
  .ap2 { position: absolute; left: 40px; top: 90px;
        background-color: lightgreen; width: 400px;
        height: 50px; }
  .rp { position: relative; left: 50px; top: 80px;
        background-color: cyan; width: 400px; height: 50px; }
</style>
</head>
<body>
  <h1>positioning style3</h1>
  <div class="ap1">절대 위치 설정 적용 - left 30px, top 70px</div>
  <div class="ap2">절대 위치 설정 적용 - left 40px, top 90px</div>
  <div class="rp">상대 위치 설정 적용 - left 50px, top 80px</div>
</body>
</html>
```

❖ position - 고정위치

position_fixed.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Position Fixed</title>
</head>
<style>
  body { font-weight: bold; font-size: 12pt; }
  .p { background-color: yellow; width: 300px; height: 50px; }
  .fp { position: fixed; right: 5px; top: 5px;
    background-color: lightgreen; width: 300px; height: 50px; }
</style>
</head>
<body>
  <h1>positioning style4</h1>
  <p class="p">기본 위치 설정 박스1</p>
  <p class="p">기본 위치 설정 박스2</p>
  <p class="p">기본 위치 설정 박스3</p>
  <p class="p">기본 위치 설정 박스4</p>
  <p class="p">기본 위치 설정 박스5</p>
  <p class="fp">고정 위치 설정 박스 : 오른쪽 스크롤
    위아래로 이동해보기</p>
</body>
```

❖ float

- 화면을 구성하는 요소 간의 관계를 고려하여 각 요소를 배치하는 방법

속성값	설명
inherit	요소를 감싸는 부모 요소의 float 속성을 상속받는다.
left	요소를 왼쪽으로 떠 있는 상태로 만든다.
right	요소를 오른쪽으로 떠 있는 상태로 만든다.
none	float 속성을 적용하지 않는다(요소를 떠 있지 않게 한다).

❖ float

float_position.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Float Position</title>
</head>
  <style>
    img { float: right; margin: 0 0 10px 10px; }
  </style>
</head>
<body>
  <p>float 속성은 웹 문서의 레이아웃을 설계하는 과정에서 많이 사용하는 속성입니다.</p>
  <p>
  float 속성은 특정 요소를 떠 있게 해줍니다. 여기서 '떠 있다'라는 말의 의미는 특정 요소가
  기본 레이아웃에서 벗어나 웹 문서의 왼쪽이나 오른쪽에 이동하는 것을 말합니다. float 속성
  은 복잡한 형태의 레이아웃을 구성하는 데 필요한 핵심 속성으로, 특정 요소가 주변 요소와 자
  연스럽게 어울리도록 해줍니다. 주의할 점은 float 속성을 사용할 때 요소의 위치가 고정되면
  안 되기 때문에 position 속성의 absolute를 사용하면 안 됩니다.</p>
</body>
</html>
```

❖ float - clear

float_clear.html

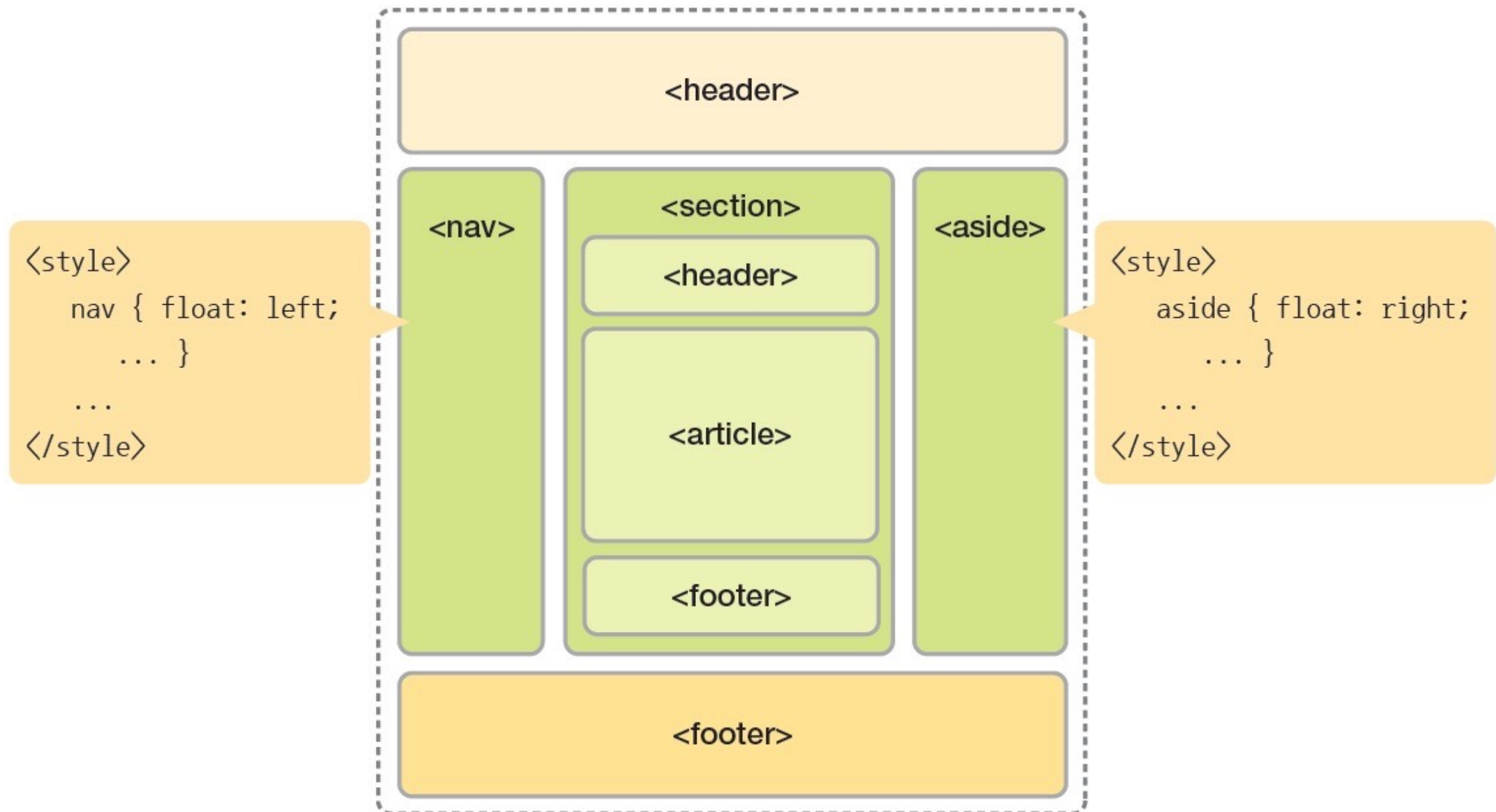
```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Float Clear</title>
</head>
<style>
  .div1 { float: left; width: 100px; height: 50px;
    margin: 10px; border: 3px solid #73AD21; }
  .div2 { border: 1px solid red; }
  .div3 { float: left; width: 100px; height: 50px;
    margin: 10px; border: 3px solid #73AD21; }
  .div4 { border: 1px solid red; clear: left; }
</style>
</head>
<body>
  <h2>float 속성 사용</h2>
  <div class="div1">div1</div>
  <div class="div2">div2 - float 속성을 사용하여 대상 요소를 웹 문서에 배치하면 그 다음 요소에도 똑같은 속성이 적용됩니다. 하지만 float 속성이 사용되는 것을 원하지 않을 때도 있습니다. 이때 clear 속성을 사용합니다. 다양한 레이아웃 설계할 때에는 float 속성과 clear 속성을 적절히 잘 사용해야 합니다.</div>
  <h2>clear 속성 사용</h2>
  <div class="div3">div3</div>
  <div class="div4">div4 - clear 속성은 float 속성이 적용되는 것을 원하지 않는 요소에 사용하여 float 속성을 초기화시킵니다. float: left;를 사용했다면 clear: left;로, float: right;를 사용했다면 clear: right;로 초기화합니다. float 속성 값이 left 인지 right 인지 상관없이 무조건 초기화하고 싶다면 clear: both;를 사용합니다. 보통 clear: both;를 많이 사용합니다.</div>
</body>
</html>
```

❖ float - overflow

float_overflow.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Float Overflow</title>
</head>
<style>
  div {
    border: 3px solid #73AD21;
  }
  .img1 {
    float: right;
  }
  .fix {
    overflow: auto;
  }
  .img2 {
    float: right;
  }
</style>
</head>
<body>
  <p>이미지가 박스 영역을 벗어남</p>
  <div>
  이미지가 오른쪽 정렬로 되어 있는데, 박스 영역을 벗어났습니다.</div>
  <p style="clear:right">overflow: auto; 속성을 사용하여 해결</p>
  <div class="fix">
  이미지가 박스 영역을 벗어날 경우에는 overflow 속성을 auto로 설정하여 해결합니다.</div>
</body>
</html>
```


❖ 시맨틱 문서 구조에서 float 속성의 사용

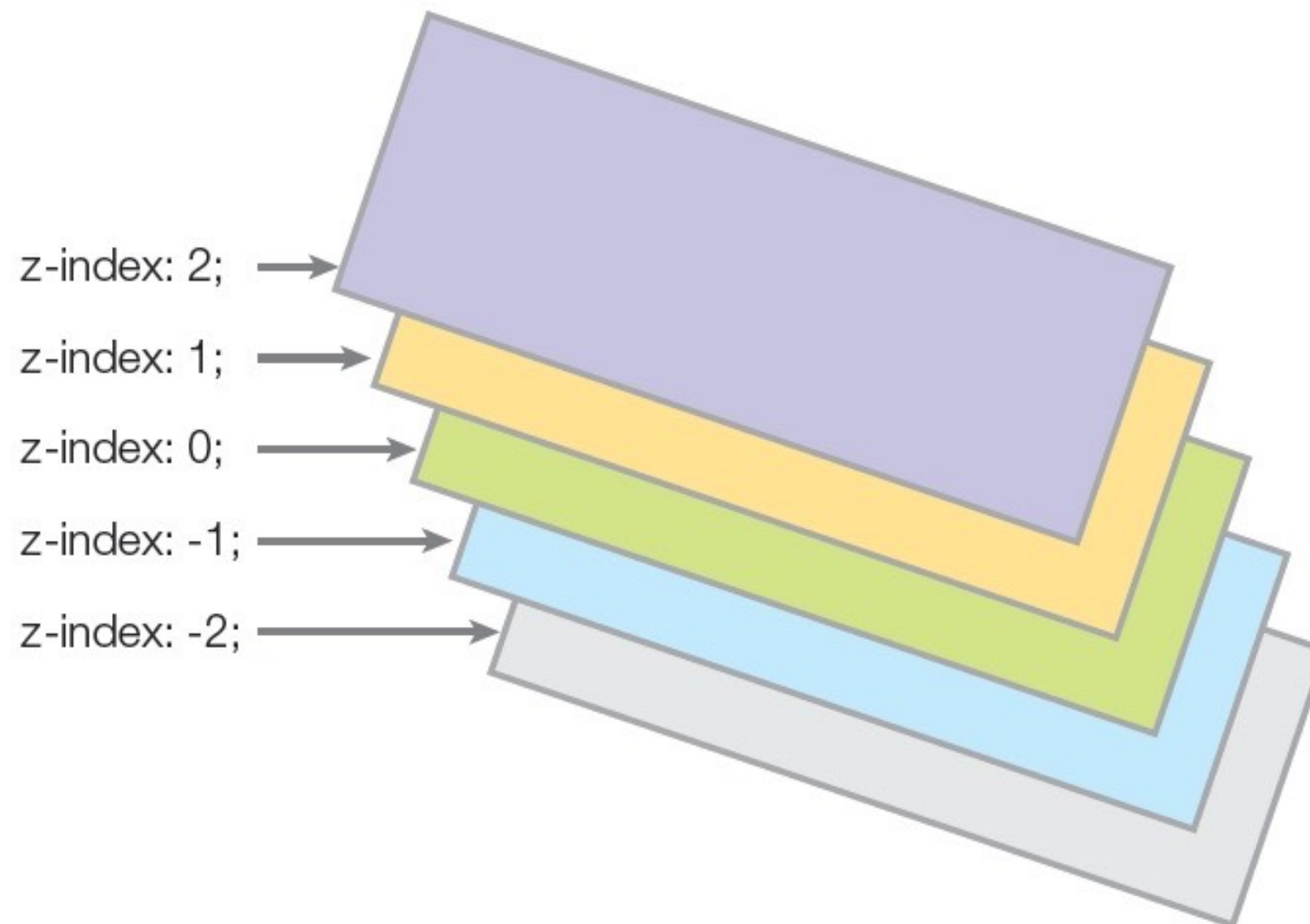


```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Float Semantic</title>
</head>
<style>
  div { border: 3px solid blue; }
  .fix { overflow: auto; }
  nav { float: left; width: 200px; border: 3px solid #73AD21; }
  section { margin-left: 206px; border: 3px solid red; }
</style>
</head>
<body>
  <div class="fix">
    <nav>
      <span>목차</span>
      <ul>
        <li><a target="_blank" href="http://www.google.com">Google</a></li>
        <li><a target="_blank" href="http://www.apple.com">Apple</a></li>
        <li><a target="_blank" href="http://www.w3.org">W3C</a></li>
        <li><a target="_blank" href="http://www.oracle.com">Oracle</a></li>
        <li><a target="_blank" href="http://www.adobe.com">Adobe</a></li>
        <li><a target="_blank" href="http://www.amazon.com">Amazon</a></li>
        <li><a target="_blank" href="http://www.Mysql.com">Mysql</a></li>
      </ul>
    </nav>
    <section>
      <span>section 1</span>
      <p>float 속성은 시맨틱 문서 구조에 유용하게 사용할 수 있습니다. 예를 들면 nav나 aside에 float 속성을 추가하면 떠다니는 내비게이션 또는 사이드 바를 만들 수 있습니다.</p>
    </section>
    <section>
      <span>section 2</span>
      <p>시맨틱 문서 구조에 float 속성을 적용할 때는 footer 부분에 적용되지 않도록 해야 합니다.</p>
    </section>
  </div>
```

float_semantic.html

❖ z-index

- 한 요소 위에 다른 요소를 쌓을 때 사용
- z-index 속성값이 작을수록 아래에 쌓임



❖ z-index

z_index.html

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Z-Index</title>
</head>
<style>
  #box1 { position: absolute; top: 0px; left: 0px;
    width: 100px; height: 100px;
    background: blue; z-index: 3;
  }
  #box2 { position: absolute; top: 30px; left: 30px;
    width: 100px; height: 100px;
    background: yellow; z-index: 2;
  }
  #box3 { position: absolute; top: 60px; left: 60px;
    width: 100px; height: 100px;
    background: green; z-index: 1;
  }
</style>
</head>
<body>
  <div id="box1">box #1</div>
  <div id="box2">box #2</div>
  <div id="box3">box #3</div>
</body>
```