

09. CSS 포지셔닝

09-1 CSS 포지셔닝과 주요 속성들

09-2 다단으로 편집하기

09-3 표 스타일



CSS 포지셔닝이란

CSS를 웹 문서 요소를 적절히 배치하는 것

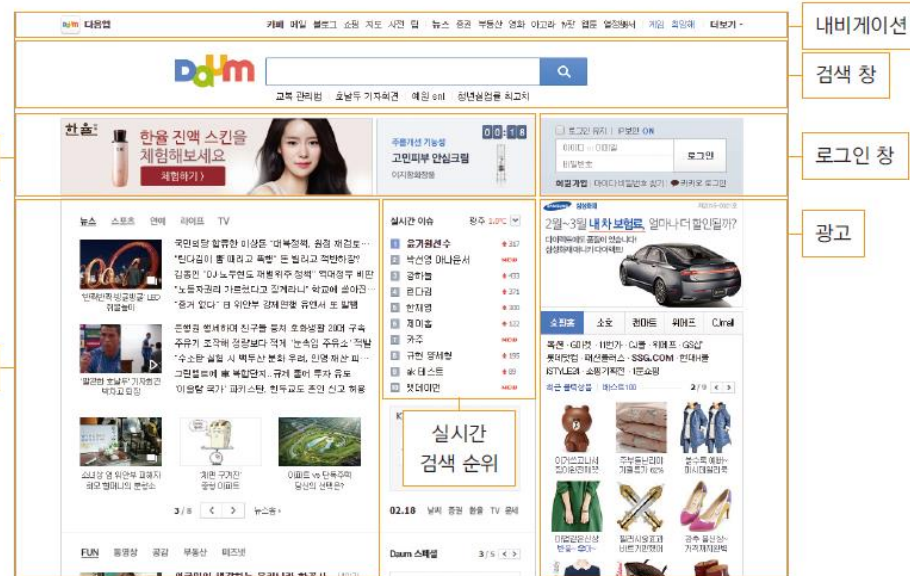
HTML 마크업으로 작성한 웹 문서



**CSS
포지셔닝**



우리가 보는 웹 사이트

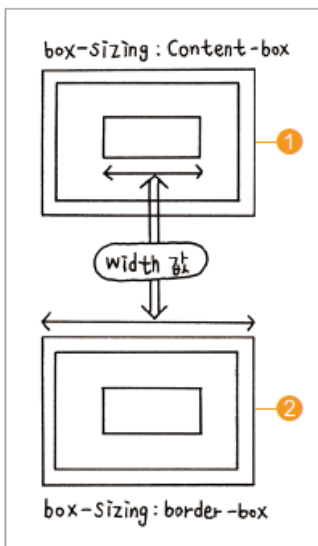


box-sizing 속성

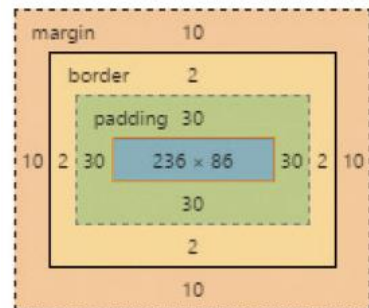
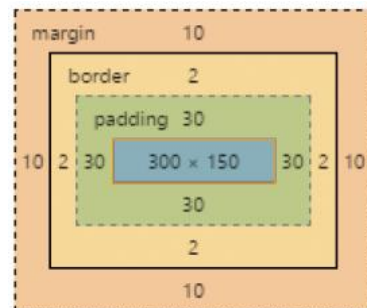
박스 모델의 너비 값 기준 지정

기본형 box-sizing: content-box | border-box

속성 값	설명
content-box	width 속성 값을 콘텐츠 영역 너비 값으로 사용합니다. 기본 값입니다.★
border-box	width 속성 값을 콘텐츠 영역에 테두리까지 포함한 박스 모델 전체 너비 값으로 사용합니다.



```
<style>
.box1 {
  box-sizing: content-box; /* 콘텐츠 영역 기준 */
  width: 300px; /* 콘텐츠 영역 너비 300px */
  height: 150px; /* 높이 */
  margin: 10px; /* 마진 */
  padding: 30px; /* 패딩 */
  border: 2px solid red; /* 테두리 */
}
.box2 {
  box-sizing: border-box; /* 테두리까지(박스 전체) 기준 */
  width: 300px; /* 박스 모델 전체 너비 */
  height: 150px; /* 박스 높이 */
  margin: 10px; /* 마진 */
  padding: 30px; /* 패딩 */
  border: 2px solid red; /* 테두리 */
}
</style>
```



CSS 포지셔닝과 주요 속성들

```
<style>
  .box {
    box-sizing: content-box; /* 콘텐츠 영역 기준 */
    width: 300px; /* 콘텐츠 영역 너비 300px */
    height: 150px; /* 높이 */
    margin: 10px; /* 마진 */
    padding: 30px; /* 패딩 */
    border: 2px solid red; /* 테두리 */
  }
</style>
</head>
<body>
  <div class="box">box-size</div>
</body>
```

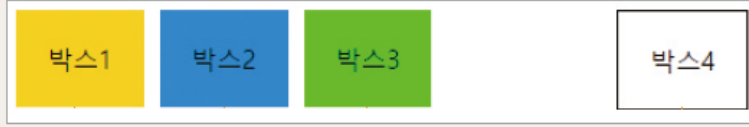
float 속성

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.

```
<style>
  .box1 { float:left; }
  .box2 { float:left; }
  .box3 { float:left; }
  .box4 { float:right; }
</style>
```



clear 속성

float 속성을 무효화 시키는 속성

기본형 clear: none | left | right | both

```
<style>
  .box1 { float:left; }
  .box2 { float:left; }
  .box3 { }
  .box4 { clear:both; }
</style>
```



CSS 포지셔닝과 주요 속성들



토마토

토마토는 비타민A, C가 풍부하여 칼륨과 같은 각종 미네랄은 혈액의 산성도를 낮추는 역할을 해주며 혈압을 내리고 혈관을 튼튼하게 해준다. 토마토에 들어있는 피코펜이라는 성분은 뛰어난 항암 작용을 보이며 잘 알려져 있듯이 블루베리와 함께 대표적인 항산화 음식이기도 하다.

```
<style>
  img {
    float:left;
    margin-right:10px;
  }
  p {
    line-height:30px;
  }
</style>
</head>
<body>
  
  <h1>토마토</h1>
  <p>토마토는 비타민A, c가 풍부하여 칼륨과
</body>
```

CSS 포지셔닝과 주요 속성들

박스1

박스2

박스3

박스4

```
<style>
  .box1{
    padding:20px;
    margin-right:10px;
    background:yellow;
    float:left;
  }
  .box2 {
    padding:20px;
    margin-right:10px;
    background: skyblue;
    float:left;
  }
  .box3 {
    padding:20px;
    background: lightgreen;
    float:left;
  }
}
```

```
.box4 {
  background:white;
  padding:20px;
  float:right;
  border:1px solid black;
}
</style>
</head>
<body>
  <div class="box1">박스1</div>
  <div class="box2">박스2</div>
  <div class="box3">박스3</div>
  <div class="box4">박스4</div>
</body>
```

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.

1) static

- 문서의 흐름대로 배치한다
- left 속성이나 top 속성을 지정할 수 없다.

```
<style>
.box1{
  float:left;
  width:100px;
  top:300px;
  background:#ffd800;
  margin-right:10px;
  padding:20px;
}
.box2 {
  width:300px;
  background:#0094ff;
  float:left;
  padding:20px;
}
</style>
```

박스1

박스2

CSS 포지셔닝과 주요 속성들

박스1

박스2

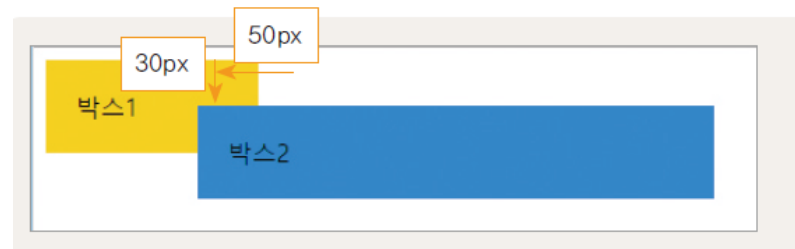
```
<style>
  .box1{
    float:left;
    width:100px;
    background:orange;
    margin-right:10px;
    padding:20px;
  }
  .box2 {
    width:300px;
    background:skyblue;
    float:left;
    padding:20px;
  }
</style>
</head>
<body>
  <div class="box1">박스1</div>
  <div class="box2">박스2</div>
</body>
```

position 속성

2) relative

- 자연스럽게 배치
- 고정되어 있지 않고 다른 요소에 의해 바뀔 수 있다.
- 상대적인 위치를 사용하기 때문에 다른 요소와 조화를 이룬다.
- left나 top 속성을 이용해 요소의 위치를 옮길 수도 있다.

```
<style>
.box1{
  float:left;
  width:100px;
  background:#ffd800;
  margin-right:10px;
  padding:20px;
}
.box2 {
  position:relative;
  left:-50px;
  top:30px;
  width:300px;
  background:#0094ff;
  float:left;
  padding:20px;
}
</style>
```



CSS 포지셔닝과 주요 속성들

```
<style>
  .box1{
    float:left;
    width:100px;
    background:orange;
    margin-right:10px;
    padding:20px;
  }
  .box2 {
    position:relative;
    left:-50px;
    top:30px;
    width:300px;
    background:skyblue;
    float:left;
    padding:20px;
  }
</style>
```

```
<body>
  <div class="box1">박스1</div>
  <div class="box2">박스2</div>
</body>
```

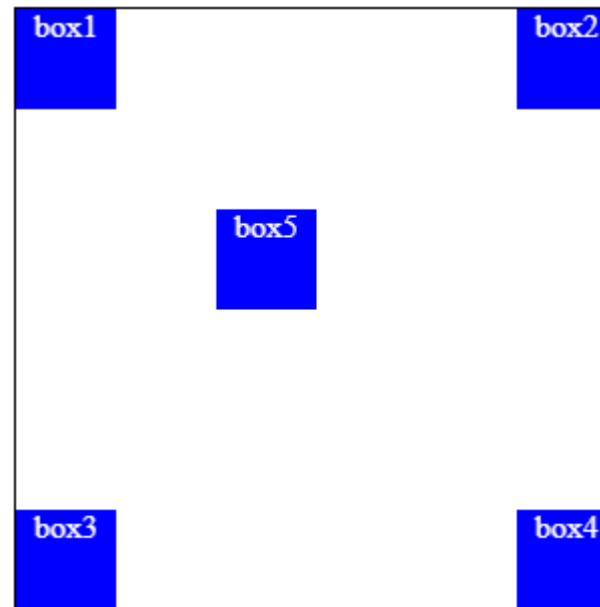


CSS 포지셔닝과 주요 속성들

position 속성

3) absolute

- 문서의 흐름과는 상관없이 원하는 위치에 요소를 배치
- 요소의 위치는 가장 가까운 부모 요소나 조상 요소 중 position:relative인 요소
- left, top, right, bottom 속성을 사용해 네 모서리에서 얼마나 떨어져 있는지 지정



CSS 포지셔닝과 주요 속성들

```
<style>
  .nemo{
    position:relative;
    width:300px;
    height:300px;
    border:1px solid black;
  }
  .box{
    position:absolute;
    width:50px;
    height:50px;
    background:blue;
    color:white;
    text-align:center;
  }
  #b1{top:0; left:0; }
  #b2{top:0; right:0; }
  #b3{bottom:0; left:0; }
  #b4{bottom:0; right:0; }
  #b5{top:100px; left:100px; }
</style>
```

```
<body>
  <div class="nemo">
    <div class="box" id="b1">box1</div>
    <div class="box" id="b2">box2</div>
    <div class="box" id="b3">box3</div>
    <div class="box" id="b4">box4</div>
    <div class="box" id="b5">box5</div>
  </div>
</body>
```

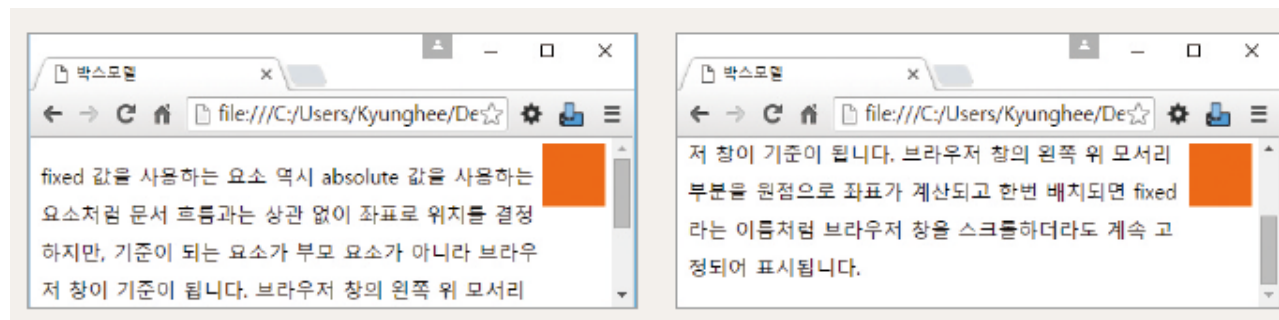
CSS 포지셔닝과 주요 속성들

position 속성

4) fixed

- 문서의 흐름과는 상관없이 원하는 위치에 요소를 배치
- 부모 요소가 아닌 브라우저 창 기준
→ 브라우저 창 왼쪽 위 꼭지점(0,0) 기준으로 좌표 계산
- 브라우저 창 화면을 스크롤하더라도 계속 같은 위치에 고정

```
<style>
#fx{
  position:fixed;
  top:5px;
  right:5px;
  width:50px;
  height:50px;
  background:#ff6a00;
}
</style>
```



CSS 포지셔닝과 주요 속성들

```
<style>
  #nemo{
    position:fixed;
    top:5px;
    right:5px;
    width:50px;
    height:50px;
    background:red;
  }
  #content { width:400px; }
  p { line-height:30px;}
</style>
```

```
<body>
  <div id="nemo"></div>
  <div id="content">
    <p>Lorem ipsum dolor sit amet, consectetur
    <p>Lorem ipsum dolor sit amet, consectetur
  </div>
</body>
```

visibility 속성

특정 요소를 화면에 보이거나 보이지 않게 설정하는 속성

기본형 `visibility: visible | hidden | collapse`

속성 값	설명
visible	화면에 요소를 표시합니다. 기본 값입니다.★
hidden	화면에서 요소를 감춥니다. 하지만 크기는 그대로 유지하기 때문에 배치에 영향을 미칩니다.
collapse	표의 행, 열, 행 그룹, 열 그룹 등에서 지정하면 서로 겹치도록 조절합니다. 그 외의 영역에서 사용하면 'hidden' 처럼 처리합니다.

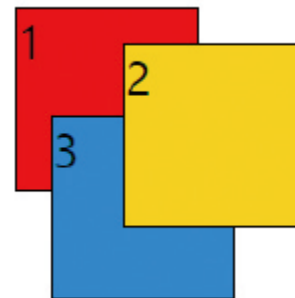
```
<style>
img {
  margin:10px;
  padding:5px;
  border:1px solid black;
}
.invisible {
  visibility:hidden;
}
</style>
```



z-index 속성

- 요소 쌓는 순서 정하기
- z-index 값이 크면 값이 작은 요소보다 위에 쌓인다.
- z-index 값을 명시하지 않으면 1부터 시작해서 1씩 커진다

```
<style>
div#wrapper {
  position: relative;
}
.b1 { z-index:1; }
.b2 { z-index:3; }
.b3 { z-index:1; }
</style>
```



CSS 포지셔닝과 주요 속성들

```
<style>
  #wrapper {
    position: relative;
  }
  #box {
    position: absolute;
    width: 100px;
    height: 100px;
    border: 1px solid black;
    font-size: 26px;
  }
```

```
.b1 {
  left: 50px;
  top: 50px;
  background: red;
  z-index: 1;
}
.b2 {
  left: 110px;
  top: 70px;
  background: yellow;
  z-index: 3;
}
.b3 {
  left: 70px;
  top: 110px;
  background: skyblue;
  z-index: 1;
}
</style>
```

```
<body>
  <div id="wrapper">
    <div id="box" class="b1">1</div>
    <div id="box" class="b2">2</div>
    <div id="box" class="b3">3</div>
  </div>
</body>
```

column-width

- 단의 너비를 고정해 놓고 화면 분할
- 화면이 커지면 단의 개수가 많아진다.

기본형 `column-width: <크기> | auto`

```
<style>
.multi {
  -webkit-column-count: 3;
  -moz-column-count: 3;
  column-count: 3;
}
</style>
<div class="multi"> ..... </div>
```



column-count

- 단의 개수를 먼저 정해 놓고 화면 분할
- 화면이 커질수록 단의 너비가 넓어진다.

기본형 `column-count: <숫자> | auto`

```
<style>
div#wrapper {
  position: relative;
}
.b1 { z-index:1; }
.b2 { z-index:3; }
.b3 { z-index:1; }
</style>
```

```
<style>
  body {
    font-family: "맑은 고딕", "고딕", "굴림";
  }

  .multi {
    -webkit-column-width: 120px;
    -moz-column-width: 120px;
    column-width: 120px;
  }
</style>
```

```
<body>
  <header>
    <h2>건강에 좋은 식품 - Super Food</h2>
  </header>
  <div class="multi">
    <p><b>코코넛 오일 </b>: 남미 파나마에서는 “코코넛오일 한 컵</p>
    <p><b>블루베리 </b>: 블루베리는 우리 몸이 산화하는 현상을 막</p>
    <p><b>아몬드 </b>: 혈관질환에 좋은 오메가3, 섬유소, 비타민</p>
  </div>
</body>
```

```
<style>
  body {
    font-family: "맑은 고딕", "고딕", "굴림";
  }
  .multi {
    -webkit-column-count: 3;
    -moz-column-count: 3;
    column-count: 3;
  }
</style>
```

```
<body>
  <header>
    <h2>건강에 좋은 식품 - Super Food</h2>
  </header>
  <div class="multi">
    <p><b>코코넛 오일 </b>: 남미 파나마에서는 “코코넛오일 한 컵</p>
    <p><b>블루베리 </b>: 블루베리는 우리 몸이 산화하는 현상을 막</p>
    <p><b>아몬드 </b>: 혈관질환에 좋은 오메가3, 섬유소, 비타민</p>
  </div>
</body>
```

caption-side

- 캡션(설명글)은 기본으로 표 위쪽에 표시됨.
- 이 속성을 이용해 아래쪽에 표시 가능

기본형 caption-side: top | bottom

border

표의 바깥 테두리와 셀 테두리 모두 지정해야 함.

```
<style>
.table1 {
  border:1px solid black;
}
.table1 td {
  border:1px dotted black;
}
</style>
```

울산	울산 vs 인천
부산	부산 vs 대전
서울	서울 vs 강원

border-collapse

표 테두리와 셀 테두리를 합칠 것인지 설정

기본형 border-collapse: collapse | separate

속성 값	설명
collapse	테두리를 하나로 합쳐 표시합니다.
separate	테두리를 따로 표시합니다. 기본 값입니다.★

```
<style>
.table1 {
  border:1px solid black;
  border-collapse:collapse;
}
.table1 td {
  border:1px dashed black;
}
</style>
```

울산	울산 vs 인천
부산	부산 vs 대전
서울	서울 vs 강원

```
<style>
  body{
    font-family:"맑은 고딕", "고딕", "굴림";
  }
  .table1 {
    border:1px solid black;
  }
  .table1 td {
    border:1px dotted black;
    padding:10px;
    text-align:center;
  }
</style>
```

```
<body>
  <table class="table1">
    <caption>프로축구 경기 일정</caption>
    <tr>
      <td>울산</td>
      <td>울산 vs 인천</td>
    </tr>
    <tr>
      <td>부산</td>
      <td>부산 vs 대전</td>
    </tr>
    <tr>
      <td>서울</td>
      <td>서울 vs 강원</td>
    </tr>
  </table>
</body>
```

```
<style>
  body{
    font-family:"맑은 고딕", "고딕", "굴림";
  }
  .table1 {
    border:1px solid black;
    border-collapse:collapse;
  }
  .table1 td {
    border:1px dashed black;
    padding:10px;
    text-align:center;
  }
</style>
```

border-spacing

- border-collapse:separate를 사용해 셀들을 분리했을 경우, 인접한 셀 테두리 사이의 거리를 지정
- 값이 1개 : 수평 거리 & 수직 거리를 같게
- 값이 2개 : 첫번째 값은 수평 거리, 두번째 값은 수직 거리

기본형 border-spacing: <크기>

```
<style>
.table1 {
  border:1px solid black;
  border-collapse:separate;
  border-spacing:20px 10px;
}
.table1 td {
  border:1px solid black;
}
</style>
```

울산	울산 vs 인천
부산	부산 vs 대전
서울	서울 vs 강원

```
<style>
body{
  font-family:"맑은 고딕", "고딕", "굴림";
}
.table1 {
  border:1px solid black;
  border-collapse:separate;
  border-spacing:20px 10px;
}
.table1 td {
  border:1px solid black;
  padding:10px;
  text-align:center;
}
</style>
```


empty-cell

border-collapse:separate를 사용해 셀들을 분리했을 경우, 내용이 없는 빈 셀들의 표시 여부를 지정

기본형 empty-cells: show | hide

```
<style>
.schedule { border-collapse:separate; }
td { border:1px solid black; }
#tb1 td{ empty-cells:show; }
#tb2 td { empty-cells:hide; }
</style>
```

```
<table class="schedule" id="tb1"> ..... </table>
<table class="schedule" id="tb2"> ..... </table>
```

프로축구 경기 일정

울산	울산 vs 인천	TV 중계
부산	부산 vs 대전	
서울	서울 vs 강원	

프로축구 경기 일정

울산	울산 vs 인천	TV 중계
부산	부산 vs 대전	
서울	서울 vs 강원	

```
<style>
body{
  font-family:"맑은 고딕", "고딕", "굴림";
}
.schedule {
  border-collapse:separate;
  margin:20px;
}
td {
  border:1px solid black;
  padding:10px;
  text-align:center;
  empty-cells:show;
}
</style>
```

width, height

- 너비나 높이를 지정하지 않으면 셀 안의 내용이 표시될 만큼만 표시된다.
- width 값을 지정할 경우 padding 속성을 이용해 여백을 넣어주면 보기 좋게 꾸밀 수 있다.

```
<style>
table {
border-collapse: collapse;
width: 300px;
}
td {
padding: 10px;
}
</style>
```

울산	울산 vs 인천
부산	부산 vs 대전
서울	서울 vs 강원

```
<style>
body{
font-family:"맑은 고딕", "고딕", "굴림";
}
table {
border-collapse: collapse;
width: 300px;
border: 1px solid black;
text-align: center;
}
td {
padding: 10px;
border: 1px dashed black;
}
</style>
```

table-layout

셀 안의 내용 양에 따라 셀 너비를 변하게 할지, 고정시킬지 결정

기본형 table-layout: fixed | auto

속성 값	설명
fixed	셀 너비를 고정합니다. 즉, 셀 내용에 따라 셀의 너비가 달라지지 않습니다.
auto	셀 내용에 따라 셀의 너비가 달라집니다. 기본 값입니다.★

```
<style>
.table1 {
  border-collapse: collapse;
  width: 300px;
  table-layout: fixed;
  word-break: break-all;
  height: auto;
}
.table1 td {
  width: 150px;
  border: 1px solid black;
  padding: 5px;
}
</style>
```

Table Layout

한글로 띄어쓰기 없기 길게 붙여쓰기	long_description_wit hout_space
------------------------	------------------------------------

```
<style>
  body{
    font-family:"맑은 고딕", "고딕", "굴림";
  }
  table {
    width:300px;
    table-layout:fixed;
  }
  td {
    width:150px;
  }
</style>
```

```
<body>
  <h1>Table Layout</h1>
  <table border="1">
    <tr>
      <td>
        한글로 띄어쓰기 없기 길게 붙여쓰기
      </td>
      <td>
        long_description_without_space
      </td>
    </tr>
  </table>
</body>
```

text-align

셀 안에서의 수평 정렬 방법

기본형 `text-align : left | right | center`

```
<style>
.table1 td {
  text-align:center;
}
</style>
```

셀 가운데 정렬			
내용1	내용2	내용3	내용4

vertical-align

셀 안에서의 수직 정렬 방법

기본형 `vertical-align: top | bottom | middle`

속성 값	설명
top	위쪽 패딩 가장자리에 내용의 윗부분을 맞춥니다.★
bottom	아래쪽 패딩 가장자리에 내용의 아랫부분을 맞춥니다.
middle	패딩의 중앙에 내용의 중앙을 맞춥니다.

```
<style>
.va1 { vertical-align:top; }
.va2 { vertical-align:bottom; }
.va3 { vertical-align:middle; }
</style>
```

vertical-alignment		
alignment		alignment
	alignment	

```
<style>
  td {
    height:100px;
    padding:20px;
  }
  .va1 { vertical-align:top; }
  .va2 { vertical-align:bottom; }
  .va3 { vertical-align:middle; }
</style>
```

```
<body>
  <table border="1">
    <caption>vertical-alignment</caption>
    <tr>
      <td class="va1">alignment</td>
      <td class="va2">alignment</td>
      <td class="va3">alignment</td>
    </tr>
  </table>
</body>
```