

9. CSS 레이아웃

<https://www.youtube.com/watch?v=qvcApvNprec&list=PLG7te9eYUi7sxAaXX74J6lqiV8vtStuLr&index=13>

- Do it! 웹코딩의 정석 개정 2판 -

1. CSS 포지셔닝이란?

웹 문서에 들어있는 여러 요소들을 **적절히 배치**하는 것을 뜻합니다.

CSS 포지셔닝을 이용해 왼쪽 html 마크업을 오른쪽처럼 보기 좋게 꾸며줄 수 있습니다.

HTML 마크업으로 작성한 웹 문서



**CSS
포지셔닝**



우리가 보는 웹 사이트



box-sizing 속성 - 박스 너비 기준 정하기

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

<속성 값>

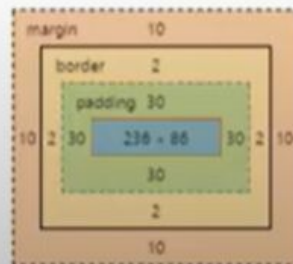
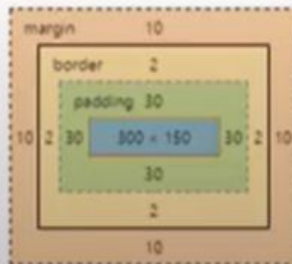
content-box : width 속성값을 **콘텐츠 영역 너비값**으로 사용합니다.기본 값입니다.

border-box : width 속성값을 콘텐츠 영역에 테두리까지 포함한 **박스 전체 너비값**으로 사용합니다.

CSS의 width 속성은 콘텐츠 영역의 너비를 나타내기 때문에 해당 요소에 적용한 패딩이나 테두리 크기는 따로 계산해서 배치해야하는데 box-sizing 속성을 사용하면 콘텐츠 영역의 너비에 패딩과 테두리 크기까지 합쳐서 width속성을 지정할 수 있습니다.

box-sizing 속성 예제

```
<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>
```

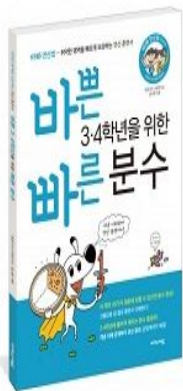


1번 박스는 좌우 패딩 각 30px과 좌우 테두리 2px, 콘텐츠 영역너비 300px이므로 실제화면에서 차지하는 너비는 364px입니다.

float 속성 - 왼쪽이나 오른쪽에 배치하기

기본형 float: left | right | none

float속성은 웹 요소를 문서에 떠있게 만듭니다. 여기서 ‘떠 있다’라는 의미는 **왼쪽 구석이나 오른쪽 구석에 배치**된다는 뜻입니다.



바쁜 3, 4학년을 위한 빠른 분수

3, 4학년이 꼭 알아야 할 분수를 한 권에 모았어요!

한국 교육과정평가원이 최근 발표한 보고서에 따르면 '수포자'는 초등 3학년 때 분수를 배우면서 시작된다고 합니다. 분수를 어려워하는 이유는 분모와 분자, 2개의 수가 나와 낯설기 때문입니다. 이렇게 낯설고 어려운 분수, 어떻게 공부해야 할까요?

```
<style>
    .left-img {
        float: left;
        margin-right: 25px;
    }
</style>
```

왼쪽에 배치돼
있으므로 left !

```

<h1>바쁜 3, 4학년을 위한 빠른 분수</h1>
<h2>3, 4학년이 꼭 알아야 할 분수를 한 권에 모았어요!</h2>
<p>한국 교육과정평가원이 최근 발...
```

position 속성 - 배치 방법 지정하기

기본형 position: static | relative | absolute | fixed

<속성 값>

static : 요소를 문서의 흐름에 맞추어 배치합니다. (기본값)

relative : 원래 있던 위치를 기준으로 **top right left bottom** 을 이용해 움직입니다.

absolute : 부모로부터 자유로워지고 문서의 흐름에서 벗어납니다. **position**이 **relative**인 가까운 부모나 조상을 기준으로 움직입니다.

fixed : **absolute**와 마찬가지로 부모로부터 자유로워지고 문서의 흐름에서 벗어나며 브라우저 기준으로 화면에 고정됩니다.

position: static;

static은 **position** 속성의 기본값으로 요소를 나열한 순서대로 배치하며 **top**이나 **right**, **bottom**, **left**같은 속성을 사용할 수 없습니다.

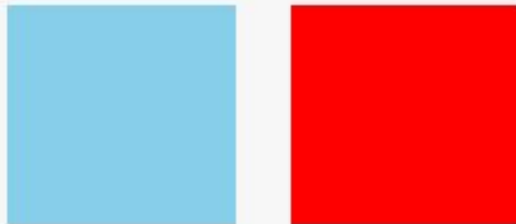
단지 **float** 속성을 이용해 좌우로 배치할 수 있습니다.

CSS ▾

```
.a {  
  width:100px;  
  height:100px;  
  background-color:skyblue;  
  float: left;  
}  
  
.b {  
  width:100px;  
  height:100px;  
  background-color:red;  
  float: left;  
}
```

Output

Run



position: relative;

relative도 **static**에서처럼 나열한 순서대로 배치되지만 **top**이나 **right**, **left**, **bottom** 속성을 사용해 좌표값을 지정할 수 있습니다.

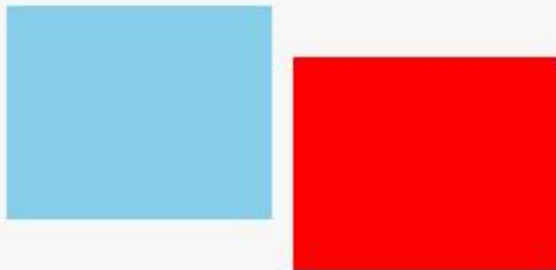
아래 사진을 보면 아래쪽으로 24px 왼쪽으로 16px 이동했습니다.

CSS ▾

```
.a {  
  width:100px;  
  height:100px;  
  background-color:skyblue;  
  float: left;  
}
```

```
.b {  
  width:100px;  
  height:100px;  
  background-color:red;  
  float: left;  
  position: relative;  
}
```

Output



position: absolute;

absolute는 문서의 흐름과 상관없이 **left**와 **right**, **top**, **bottom** 속성을 이용할 수 있습니다. 기준이 되는 위치는 가장 가까운 부모 요소나 조상 요소 중 **position: relative;** 인 요소입니다. 그래서 **<div>**로 감싸고 **position**을 **relative**로 지정하고 사용해야 합니다.

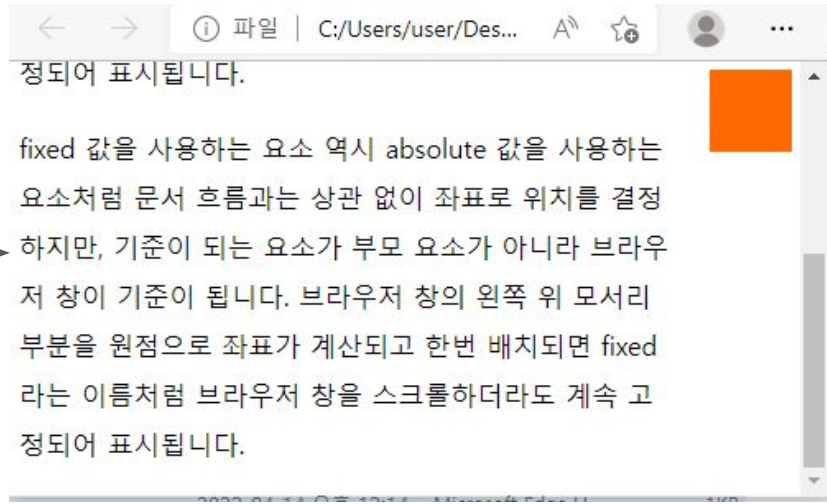
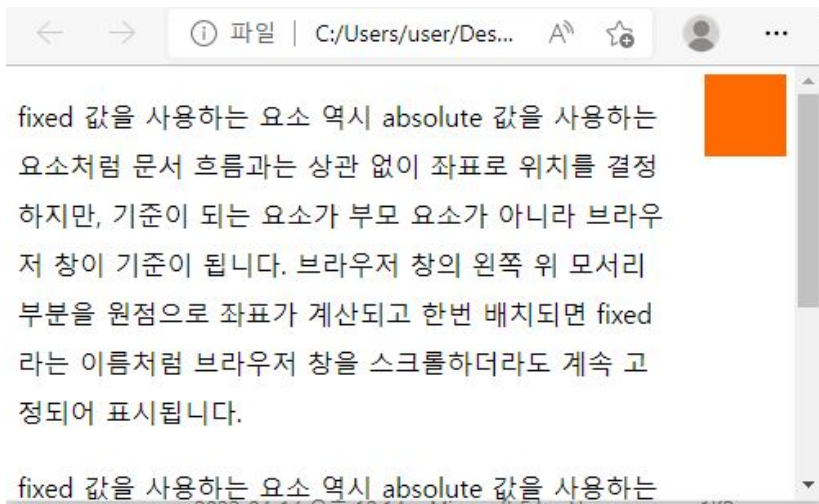
```
<style>
#wrap{
  position: relative;
  width: 300px;
  height: 300px;
  border: 1px solid #ccc;
}
.box{
  position: absolute;
  width: 50px;
  height: 50px;
  background-color: #0000ff;
}
```



position: fixed;

fixed속성값도 **absolute** 속성값처럼 문서의 흐름과 상관없이 위치를 좌표로 결정하지만 부모요소가 아닌 **브라우저창이 기준**이 됩니다.

브라우저창을 스크롤해도 **화면에 고정**되어 표시됩니다.



visibility 속성 - 요소를 보이게 하거나 보이지 않게 하기

기본형 visibility: visible | hidden | collapse

<속성 값>

visible : 화면에 표시합니다. 기본값입니다.

hidden : 화면에서 **요소를 감춥니다**. 하지만 크기는 그대로 유지하기 때문에 배치에 영향을 미칩니다.

collapse : 표의 행, 열, 행그룹, 열그룹등에서 지정하면 **서로 겹치도록 조절합니다**.

visibility를 hidden으로 지정하면 화면에서 감춘 요소들은 눈에 보이지않지만 **실제로는 공간을 차지**하기 때문에 보이지 않는 요소들까지도 고려해야 합니다.

visibility 속성 예제

다음 예제는 두번째 이미지를 `visibility:hidden;` 으로 설정해 화면에서 감췄지만 웹 문서에는 실제로 존재하기 때문에 세번째 이미지가 아래의 형식처럼 배치됩니다.



```
.invisible {  
  visibility: hidden;  
}
```

z-index 속성 - 요소 쌓는 순서 정하기

기본형 z-index: 숫자

CSS에서 한 요소위에 다른 요소를 쌓을 수도 있습니다. 이 때 순서를 지정하는 것이 **z-index** 속성입니다. 값이 작을수록 아래에 쌓입니다.

z-index값을 지정하지 않을경우 맨 먼저 삽입되는 요소가 **z-index:1**값을 가집니다.

즉, 맨 밑에 깔립니다.

노란박스가 두번째에
배치되어야하지만
z-index:3 이므로 맨
위로 올라옴.



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

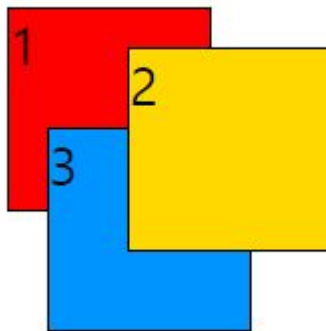


표 스타일 **caption-side** 속성 - 제목위치 정하기

기본형 **caption-side**: top | bottom

<속성 값>

top : 캡션을 표 위에 표시합니다. 기본값입니다.

bottom: 캡션을 표 아래에 표시합니다.

표 제목은 <caption>태그를 이용해 표시하며 기본적으로 표 위에 표시되지만 **caption-side** 속성을 이용하여 아래쪽으로 옮길 수 있습니다.

border-collapse 속성 - 테두리 통합, 분리하기

기본형 border-collapse: collapse | separate

<table>태그와 <td>태그에 border속성을 사용하면 두줄로 표시되는데 border-collapse속성을 이용하면 표의 바깥 테두리와 셀의 각 테두리가 떨어져 있는것을 합칠건지 그대로 둘건지 정할 수 있습니다.

CSS ▾ Saved

```
table, td {  
  border: 1px red solid;  
}  
  
td {  
  border: 1px black dashed;  
}
```

Output

캥거루	고양이	강아지
갈냥이	흰검냥	흰냥이
식당냥	주인냥	얼룩냥



CSS ▾

```
table, td {  
  border: 1px red solid;  
  border-collapse: collapse;  
}  
  
td {  
  border: 1px black dashed;  
}
```

Output

캥거루	고양이	강아지
갈냥이	흰검냥	흰냥이
식당냥	주인냥	얼룩냥

border-spacing 속성 - 인접한 셀 테두리 사이거리 지정

기본형 border-spacing: 크기

border-collapse: separate;로 셀들을 분리했을 경우, 인접한 셀 테두리 사이의 거리를 지정합니다.

크기는 **em**이나 **px**를 사용합니다.

border-spacing 속성값은 1~2개를 지정 할 수 있는데 하나라면 수평거리와 수직거리를 한번에 지정한것이고 두개라며 첫번째 값은 수평거리 두번째 값은 수직거리이다.

```
CSS +
table, td {
  border: 1px red solid;
  border-collapse: separate;
  border-spacing: 20px 10px;
}
```

Output

캥거루	고양이	강아지
갈냥이	흰검냥	흰냥이
사슴냥	주머니	여우냥

table-layout 속성 - 콘텐츠에 맞게 셀 너비 지정하기

기본형 table-layout: fixed | auto

셀 안의 내용 양에 따라 셀 너비를 변하게 할지, 고정시킬지 결정

CSS ▾

```
table, td {  
  border: 1px solid black;  
}  
  
table {  
  width: 300px;  
}  
  
td {  
  width: 150px;  
}
```

Output

한글로 띄어쓰기 없 이 길게 붙여쓰기	long description
-------------------------	------------------

table 너비는 300px
td 너비는 150px 이다.
하지만 영문을
여백없이 길게 쓰면
너비는 변한다.
table-layout: fixed를
넣으면 너비는
고정되지만 셀 밖으로
내용이 밀려나간다.
word-break: break-all
넣으면
자동줄바꿈된다.
한글이나 여백있는
영문은 너비가