

HTML 5 + CSS 3 완벽 타파하기



@ IT Korea 강사 이정훈

목차

1. 가상 클래스
2. 텍스트 스타일
3. 실습
4. 레이아웃 스타일
5. 색상 스타일

지난시간에 이어서, CSS에 대해 공부를 하게 됩니다.
미리 말씀드리지만 조금 어렵습니다. 그리고 디자인의 영역이므로 집중을 요합니다.

2. 가상 클래스



가상 클래스는 웹 문서에 실제로 존재하지 않지만, 필요에 의해 임의로
가상선택자를 지정하여 사용하는 것을 말합니다.

:link 문서 안에 하이퍼링크 중에 사용자가 아직 방문하지 않은 링크에 스타일을 적용
ex) a:link { color : black; text-decoration : none; }

:visited 문서 링크 중에 한번 이상 방문한 링크에 대한 스타일을 적용
ex) a:visited { color : black; }

:active 해당 요소가 활성화 되었을때의 스타일을 지정합니다.
ex) a:active { color : black; }

:hover 해당 요소 위로 마우스 포인터를 올려놓았을 때의 스타일을 지정합니다.
ex) a:hover { color : red; text-decoration : underline; }

:focus 해당 요소에 포커스가 맞춰졌을 때의 스타일을 지정합니다.
ex) a:focus { background-color : yellow; }

우선적으로 링크에 관련된 가상선택자를 알아보겠습니다.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="utf-8">
5      <title></title>
6      <style>
7          .navi a, .navi a:visited{
8              padding: 10px 5px 5px 35px;
9              display: block;
10             color: #ffff;
11             width: 150px;
12             text-decoration: none;
13         }
14         .navi a:hover, .navi a:focus{
15             text-shadow: 0px 2px 2px #000
16             color: #FC0;
17         }
18         .navi a:active{
19             color: red;
20         }
21     </style>
22 </head>
23 <body>
24     
25     <table class="navi" style="background-color: black">
26         <tr>
27             <td><a href="#">이용 안내</a></td>
28             <td><a href="#">객실 소개</a></td>
29             <td><a href="#">예약방법 및 요금</a></td>
30             <td><a href="#">예약하기</a></td>
31         </tr>
32     </table>
33 </body>
34 </html>
```

:root 문서 안의 루트요소에 스타일을 적용, 뿌리요소인 html 요소에 적용

:nth-child(n), :nth-last-child(n)

문서 중 특정부분에 스타일 적용하려면, id나 class 등을 이용해서 스타일을 정의하는 데 여러 개의 항목이 있을경우에는 몇번째에 있는지를 따져서 스타일을 적용합니다.

:nth-of-type(n), :nth-last-type(n)

:first-child, :last-child

:first-of-type, :last-of-type

형제관계요소에서 첫번째요소, 마지막번째 요소를 뜻합니다.

:only-child, :only-of-type

부모요소 안의 자식요소가 하나일때 적용, 유일하게 남은 요소일때 적용

:target 같은 문서 안에서 다른 위치로 이동할때 앵커(anchor)를 사용.

앵커의 목적지 부분 스타일 지정 가능

:enabled, :disabled, :checked

사용가능한 상태 스타일, 사용불가능한 상태 스타일, 체크박스 체크 되었을때 적용

이번에는 위치를 기준으로 하는 가상선택자를 알아보겠습니다.

```

1  <!DOCTYPE HTML>
2  <html>
3  <head>
4  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5  <title>Gradient Menu</title>
6  <style>
7  ul.navi li {
8      list-style-type: none; /* 불릿 없앰 */
9      display: block; /* 영역 만들기 */
10     width: 8em; /* 너비 */
11     height: 3em; /* 높이 */
12     float: left; /* 왼쪽으로 플로팅 */
13     text-align: center; /* 가운데 정렬 */
14     font-family: "Lucida Grande", sans-serif; /* 글꼴 */
15     border-right: 1px #818181 solid; /* 오른쪽에 테두리 */
16     box-shadow: 1px 1px 1px #bbb; /* 박스 그림자 */
17     background-color: #c9c9c9; /* 배경색 */
18     background: -webkit-gradient(linear, left top, left bottom, from(#c9c9c9), to(#848484), color-stop(0.6, #a1a1a1)); /* 그라데이션 배경 */
19     background: -moz-linear-gradient(top, #c9c9c9, #a1a1a1, #848484);
20 }
21 /* 메뉴 링크 스타일 */
22 ul.navi li a {
23     display: block; /* 영역 만들기 */
24     padding: .8em .5em .5em .5em; /* 안여백 */
25     text-decoration: none; /* 밑줄 없앰 */
26     color: #292929; /* 글자색 */
27     text-shadow: 1px 1px 0px #cccccc; /* 텍스트 그림자 */
28 }
29 /* 메뉴 링크 위로 마우스 오버했을 때의 스타일 */
30 ul.navi li:hover{
31     background-color: #333333; /* 배경색 */
32     background: -webkit-gradient(linear, left top, left bottom, from(#333333), to(#4c4c4c), color-stop(0.6, #474747)); /* 그라데이션 배경 */
33     background: -moz-linear-gradient(top, #333333, #474747, #4c4c4c);
34 }
35 }
36
37 /* 메뉴 링크 위로 마우스 오버한 후 링크 스타일 */
38 ul.navi li:hover > a{
39     color:#FFF; /* 글자색 */
40     text-shadow:0px 1px 0px #fff; /* 텍스트 그림자 */
41 }
42
43 /* 첫번째 항목의 스타일 */
44 ul.navi li:first-child {
45     border-top-left-radius: .5em; /* 왼쪽 상단 코너를 부드럽게 */
46     border-bottom-left-radius: .5em; /* 왼쪽 하단 코너를 부드럽게 */
47 }
48
49 /* 마지막 항목의 스타일 */
50 ul.navi li:last-child {
51     border-top-right-radius: .5em; /* 오른쪽 상단 코너를 부드럽게 */
52     border-bottom-right-radius: .5em; /* 오른쪽 하단 코너를 부드럽게 */
53 }
54
55 </style>
56 </head>
57
58 <body>
59 <div>
60     <ul class="navi">
61         <li class="home"><a href="#">Home</a></li>
62         <li id="html"><a href="#">HTML5</a></li>
63         <li id="css"><a href="#">CSS3</a></li>
64         <li id="jquery"><a href="#">JQuery</a></li>
65     </ul>
66 </div>
67 </body>
68 </html>

```

first child 첫번째 자식요소에 적용할 스타일이며,
last child 는 마지막 자식요소에 적용할 요소입니다.

```

<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>요안도라</title>
<link href="style.css" rel="stylesheet" type="text/css">
<style>
article.intro p:nth-child(2) {
    border:1px solid black;      /* 1px 검은색 실선 */
    margin:10px;                /* 마진(바깥여백) 10px */
    padding:15px;               /* 패딩(안여백) 15px */
    background-color:lightgreen; /* 배경색 연두색 */
}

article.intro p:last-of-type {
    width:80%;                 /* 너비는 부모 요소의 80% */
    padding:15px;
    margin-left:15%;           /* 배경색 노랑 */
    background:yellow;         /* 글자색 파랑 */
}
</style>
</head>

```

```

<body>
<div class="container">
<header>
<h1>Joandora</h1>
    <h2>가장 제주다운 수산리집</h2>
</header>
<nav class="navi">
<ul>
    <li><a href="#">이용 안내</a></li>
    <li><a href="#">객실 소개</a></li>
    <li><a href="#">예약 방법</a></li>
    <li><a href="#">예약하기</a></li>
</ul>
</nav>
<section class="content">
<h2>요안도라 소개</h2>
<article class="intro">
    <p>요안도라는 게스트 하우스(Guest House) 형식의 농어촌 민박입니다</p>
    <p>성산의 날씨는 다음주 내내 높은 구름에 햇살 가득이라고 합니다. 목요일이면 섭씨 27도까지 오른다고 하지만, 늘 부는 바람이 쾌적한 균형을 잡아 마당에 마당에 나가 앉아 있는 시간이 많아질듯 합니다.</p>
    <p>오늘은 사진에 보이는 긴 돌담을 따라 들어오는 요안도레 올레 입구에 특곤색의 대문을 달았습니다.</p>
    <p>내일은 두달 여동안 밖거리에 만든 데스트 하우스에 연백색의 황토 페인트를 칠할 예정입니다. <br>
        그리고 이것저것 사소한 저이를 마치고 나면, 나이 드시고 젊고 한 미지의 새식구들을 설렘으로 만나고 함께하고, 도시의 바쁜 생활로 소원해진 오래된 친구와의 우정을 이 제주에서 새롭게 열어나가기 위해 요안도라를 세상에 알리려고 합니다.</p>
    <p>그럼 이 홈페이지를 공식적으로 체계적으로 갖추면서, 그동안 공사로 어수선해 올려드리지 못한 집 안팎 사진들을 상세하게 올리고 당신을 만날 채비를 마치겠습니다.</p>
</article>
</section>
<footer>
    <p>제주특별자치도 남제주군 성산읍 수산리 000 번지 요안도라 </p>
    <address> <strong>예약문의</strong> 070-###-####, yoan##@naver.com</address>
    <p>Copyright ©. All rights reserved.</p>
</footer>
</div>
</body>
</html>

```

only child 는 부모요소 안의 자식 요소가 유일하게 하나일때 스타일을 적용한다.

only-of-type은 해당요소가 유일한 요소일때 스타일을 적용한다.

::before, ::after

특정 요소 앞이나 뒤에 지정된 스타일을 만들어줍니다.

ex) p::before { content : “내용” ; }

::selection

마우스 선택했을때 커서가 드래그되서 선택되는 영역의 스타일을 지정

ex) ::selection { color : #00ff00 }

:not

뒤에 나오는 모든 부분에 대한 스타일을 부정합니다.

ex) :not(h1) { color : blue; }

기타 가상 클래스 입니다.

```
1 <!doctype html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <title>요안도라</title>
6   <style>
7     ul li{
8       list-style:none;          /* 목록에서 불릿 없앰 */
9       font-size:16px;          /* 글자 크기 16px */
10      margin-top :10px;        /* 위쪽 여백 10px */
11    }
12    ul li label{
13      float:left;             /* 왼쪽으로 배치 */
14      width:80px;             /* 너비 80px */
15      line-height:25px;        /* 줄간격 25px */
16    }
17
18    input:not([type="text"]) {      /* type="text"가 아닌 input 요소 */
19      width:150px;              /* 너비 150px */
20      background-color:lime;     /* 배경색 라임색 */
21      border:1px solid #000;     /* 1px짜리 검은색 실선 */
22    }
23  </style>
24 </head>
25
26 <body>
27   <section id="container">
28
29   <h1>예약하기</h1>
30   <form name="form1" id="form1" method="post">
31     <fieldset>
32       <legend>예약자 정보</legend>
33       <ul>
34         <li>
35           <label for="name">예약자</label>
36           <input type="text" id="name">
37         </li>
38         <li>
39           <label for="contact">연락처</label>
40           <input type="tel" id="contact">
41         </li>
42         <li>
43           <label for="people">예약 인원</label>
44           <input type="number" id="people">
45
46
47         </li>
48         <li>
49           <label for="bank">입금 은행</label>
50           <input type="text" id="bank">
51         </li>
52       </ul>
53     </fieldset>
54   </form>
55 </section>
56 </body>
57 </html>
```

2. 텍스트 스타일

```
1  <!doctype html>
2  <html>
3  <head>
4  <meta charset="utf-8">
5  <title>요안도라</title>
6  <style>
7  body {
8      font-family:Arial;
9      background:#808080;
10 }
11 h1 {
12     font-size:5em;
13     color:white;
14 }
15 .shadow1 {
16     text-shadow:1px 1px 3px #000;
17 }
18 .shadow2 {
19     color:black;
20     text-shadow:3px -3px 5px #fff;
21 }
22 </style>
23 </head>
24
25 <body>
26 <h1>Joandora</h1>
27 <h1 class="shadow1">Joandora</h1>
28 <h1 class="shadow2">Joandora</h1>
29 </body>
30 </html>
31
```

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>요안도라</title>
<style>
body {
    font-family:Arial;
    background:#000;
}
h1 {
    font-size:5em;
    color:white;
}
.shadow {
    color:#000;
    text-shadow:0 0 4px #ccc, 0 -5px 4px #ff3, 2px -10px 6px #fd3, -2px
    -15px 11px #f80, 2px -19px 18px #f20;
}
</style>
</head>

<body>
<h1>Joandora</h1>
<h1 class="shadow">Joandora</h1>
</body>
</html>
```

text-shadow : h-shadow 값 v-shadow 값 blur 값 color 값;
color값은 제외한 나머지 값은 모두 px 단위.

순서대로 그림자의 가로길이, 세로길이, 번지는정도, 그림자 색상을 지정.

```

<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>요안도라</title>
<link href="style.css" rel="stylesheet" type="text/css">
<!--[if lt IE 9]>
<script src="html5shiv.js"></script>
<![endif]-->
<style>
.line1{ text-decoration:underline; }
.line2{ text-decoration:overline; }
.line3{ text-decoration:line-through; }
.line4 { text-decoration:overline underline; }
p {line-height:20px; }
</style>
</head>

<body>
<div class="container">
<header> </header>
<nav class="navi">
<ul>
<li><a href="desk.html">이용 안내</a></li>
<li><a href="room.html">객실 소개</a></li>
<li><a href="intro.html">예약 방법 및 요금</a></li>
<li><a href="reservation.html">예약하기</a></li>
</ul>
</nav>
<section class="content">
<h2>예약 방법 및 요금</h2>
<article>
<h3>▣ 요금</h3>
<p class="line1"><b>1인</b> 40,000원 </p>
<p><b>2인</b> 60,000원</p>
<p class="line2"><b>3인</b> 80,000원</p>
<p><b>4인</b> 100,000원</p>
<p class="line3"><b>5인</b> 120,000원</p>
<p class="line4"><b>6인</b> 140,000원</p>
<p>6인 이상이 머무르게 되면 2층 침대가 아닌 방바닥에 주무시거나<br>
앞뜰에 텐트를 치고 주무실 수 있습니다.<br>
침낭이나 텐트는 직접 가져오셔야 합니다.<br>
이 경우 숙박 요금은 저희와 별도로 상의해 주세요.</p>
<p>&nbsp; </p>
</article>
</section>
</div>
</body>

```

text-decoration 은 텍스트에 줄 표시하는 속성이다.
속성값은, none,underline,overline,line-through 가 있다.

Q1.

notice : 6인이 넘을 경우

6인 이상이 머무르게 되면 2층 침대가 아닌 방바닥에 주무시거나,
앞뜰에 텐트를 치고 주무실수잇습니다.

침낭이나 텐트는 직접 가져오셔야합니다. 이 경우 숙박 요금은 저희와
별도로 상의해 주세요.

Notice : 6인이 넘을 경우

6인 이상이 머무르게 되면 2층 침대가 아닌 방바닥에 주무시거나,
앞뜰에 텐트를 치고 주무실수잇습니다.

침낭이나 텐트는 직접 가져오셔야합니다. 이 경우 숙박 요금은 저희와
별도로 상의해 주세요.

- 조건) 1.시작 글자를 대문자로 변경 2.줄 간격 20px 3. 글 크기 15px

Q2.

조건)

1. **body** : 가로,세로 500px
2. **section, id** 속성 사용
3. 형제 요소를 생각
4. 가상클래스를 이용
5. 이미지 세로 200px

아메리카노

에스프레소의 향은 그대로 느끼면서 뜨거운 물을 부어 1:2 정도의 비율로 연하게 먹는 커피다.

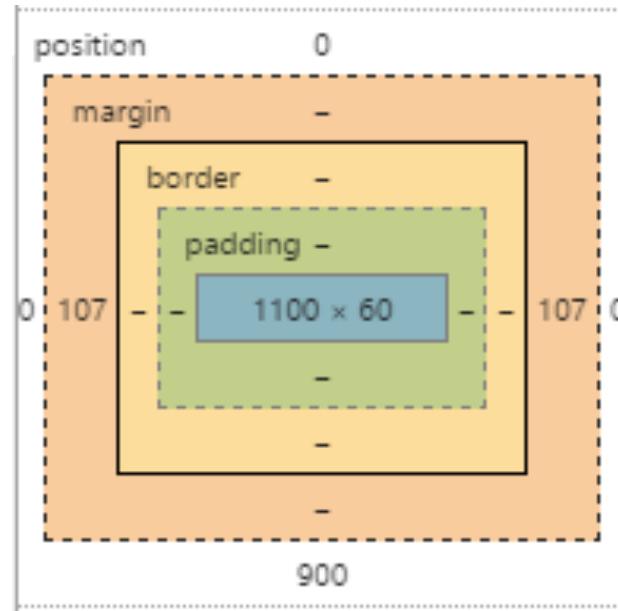
원래 이탈리아 사람들이 마시던 에스프레소를 비싼 값 때문에 희석해서 먹었다는 설이 있다.

미국인들이 에스프레소가 너무 진해 물을 섞어 마셨다는 얘기도 있다.



3. 레이아웃을 위한 스타일

1. 박스 모델



박스 모델 이란, 그림이나 텍스트처럼 사각 영역을 갖는 요소를 통칭해서 부르는 말입니다.

위그림에서 보이는 구조로 이루어져 있으며, position은 없을 수도 있습니다.

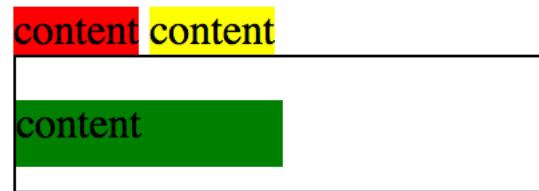
적절한 공간은 두어 읽기 쉽게 만들수도 있으며, 다양한 테두리를 추가할수도 있고,

여백을 둘수도 있습니다.

Box 모델 요소

1. content (width,height 속성)

```
<span style="background: red ">content</span>
<span style="background: yellow; width:auto; height:auto; ">content</span>
<div style=" width: 200px; height: 50px; border: 1px solid black;">
    <p style="background: green; width: 50%; height: 50%; ">content</p>
    ...
</div>
```



태그가 차지하는 공간입니다. 혹은 width,height 가 차지하는 공간입니다.

width 와 height 스타일 속성으로 px, %, auto 가 있습니다.

쉽게 보여드리기 위해 위에서 예시를 보여드립니다.

2. padding (top, right, bottom, left)

```
1  /* padding 값이 4개일때 */
2  #content1{
3      padding: 10px 30px 10px 50px;
4  }
5
6
7  /* padding 값이 2개일때 */
8  #content2{
9      padding: 10px 30px ;
10 }
11
12 /* padding 값이 1개일때 */
13 #content3{
14     padding: 10px ;
15 }
```

Padding 이후 속성 지정이 아닌 숫자만 적는경우가 있는데,

순서대로 top, right, bottom, left 순으로 보시면 됩니다.

가끔 내용이 너무 길어졌을때 스크롤바가 생기는것을 보셨나요?

스크롤바는 padding 공간 안에 생성됩니다.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="utf-8">
5      <title></title>
6      <style>
7          .content1{
8              border:1px solid black;
9              background: lightyellow;
10             padding: 10px 30px 10px 50px;
11         }
12         .content2{
13             border:1px solid black;
14             background: lightyellow;
15             padding: 10px 30px;
16         }
17         .content3{
18             border: 1px solid black;
19             background: lightyellow;
20             padding: 10px;
21         }
22     </style>
23 </head>
24 <body>
25     <div class="content1">
26         <h3> 예약 방법 </h3>
27         <p>요안도라 예약의 가장 빠른 방법은 인터넷 전화로 직접 통화하시거나 문자를 남기시는 것입니다. 인터넷 전화라 시외전화 요금이 부과되지 않습니다.</p>
28     </div>
29     <div class="content2">
30         <h3> 예약 방법 </h3>
31         <p>요안도라 예약의 가장 빠른 방법은 인터넷 전화로 직접 통화하시거나 문자를 남기시는 것입니다. 인터넷 전화라 시외전화 요금이 부과되지 않습니다.</p>
32     </div>
33     <div class="content3">
34         <h3> 예약 방법 </h3>
35         <p>요안도라 예약의 가장 빠른 방법은 인터넷 전화로 직접 통화하시거나 문자를 남기시는 것입니다. 인터넷 전화라 시외전화 요금이 부과되지 않습니다.</p>
36     </div>
37 </body>
38 </html>
```

예약 방법

요안도라 예약의 가장 빠른 방법은 인터넷 전화로 직접 통화하시거나 문자를 남기시는 것입니다. 인터넷 전화라 시외전화 요금이 부과되지 않습니다.

예약 방법

요안도라 예약의 가장 빠른 방법은 인터넷 전화로 직접 통화하시거나 문자를 남기시는 것입니다. 인터넷 전화라 시외전화 요금이 부과되지 않습니다.

예약 방법

요안도라 예약의 가장 빠른 방법은 인터넷 전화로 직접 통화하시거나 문자를 남기시는 것입니다. 인터넷 전화라 시외전화 요금이 부과되지 않습니다.

3. border

border-width : 테두리 두께지정

- 직접 크기값 입력 : 1px, 5px, 10px
- **thin** : 얇은선을 그립니다.
- **medium** : 기본값입니다.
- **thick** : 두꺼운 선을 그립니다.

Border-style : 테두리선 스타일 지정

- **none** : 테두리가 나타나지 않습니다.
- **hidden** : 테두리가 나타나지 않습니다.
- **dotted** : 테두리를 점선으로 표시
- **dashed** : 테두리를 짧은 직선으로 된 점선으로 표시
- **solid** : 테두리를 실선으로 표시
- **double** : 테두리를 이중선으로 표시. 두 선 사이 간격이 border-width값이 됨.
- **groove** : 테두리를 조각된것처럼 표시
- **ridge** : 테두리를 창에서 튀어나온 것처럼 표시

border-color : 테두리 색상지정

- 색상 값이나 16진수 값, RGB 값으로 나타냅니다.
- transparent : 투명하게 나타냅니다. 기본값입니다.

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>테두리</title>
<link href="style.css" rel="stylesheet" type="text/css">
<style>
h3 {
    border-color:blue;
    border-style:solid;
    padding : 10px;
    border-top-width:5px;
    border-right-width:0;
    border-bottom-width:5px;
    border-left-width:0;
}
p {
    border-color:red black red black;
    padding : 10px;
    border-style:dotted;
    border-width:3px;
}
</style>
</head>

<body>
    <h3> 아메리카노</h3>
    <p>에스프레소의 향은 그대로 느끼면서 뜨거운 물을 부어 1:2 정도의 비율로 연하게 먹는 커피다.</p>
</body>
</html>
```

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>테두리</title>
<link href="style.css" rel="stylesheet" type="text/css">
<style>
h3 {
    padding-bottom:5px;
    border-bottom:3px solid blue;
}

p {
    padding:10px;
    border:1px red solid;
}

</style>
</head>

<body>
    <h3> 아메리카노</h3>
    <p>에스프레소의 향은 그대로 느끼면서 뜨거운 물을 부어 1:2 정도의 비율로 연하게 먹는 커피다.</p>
</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title></title>
</head>
<body>
    <div style="background: yellow; width:150px; height: 100px; padding:10px; border: 10px solid black;">
        <p>box-sizing 미사용</p>
    </div>

    <div style="background: yellow; width:150px; height: 100px; padding:10px; border: 10px solid black; box-sizing: border-box;">
        <p>box-sizing 사용</p>
    </div>
</body>
</html>
```

box-sizing 미사용

box-sizing 사용

padding이 없어도 border는 존재할수있기에 굳이 padding을 설정하실 필요는 없습니다.

중요한점은 padding과 border는 width에 포함이 안된다는 것입니다.

만약 width가 50px, padding이 10px, border 10px이라면, 50+20+20 으로,

width 값이 총 90px이 잡히는 셈입니다.

이런게 싫을 때 우리는 box-sizing:border-box 를 주어 해결합니다.

border-radius / border-*radius

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>요안도라</title>
<style>
div{
    float:left;
    width:200px;
    height:100px;
    margin:20px;
    background:skyblue;
    border:1px solid black;
}
.content1 {
    border-radius:10px;
    -moz-border-radius:10px;
}
.content2 {
    border-radius:30px;
    -moz-border-radius:30px;
}

</style>
</head>

<body>
    <div class="content1"></div>
    <div class="content2"></div>
</body>
</html>
```

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>요안도라</title>
<style>
div{
    float:left;
    width:200px;
    height:100px;
    margin:20px;
    background:lightgreen;
    border:1px solid black;
}
.brd1 {
    border-top-left-radius:100px 50px;
    -moz-border-radius-topleft:100px 50px;
}
.brd2 {
    border-bottom-right-radius:50% 30%;
    -moz-border-radius-bottomright:50% 30%;
}
.brd3 {
    border-top-right-radius:50px;
    -moz-border-radius-topright:50px;
}

</style>
</head>

<body>
    <div class="brd1"></div>
    <div class="brd2"></div>
    <div class="brd3"></div>
</body>
</html>
```

4. margin

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>요안도라</title>
<link href="style.css" rel="stylesheet" type="text/css">
<style>
.content1{
    border:1px solid black;
    background:lightyellow;
    padding:10px;           /* 네 방향의 패딩 모두 10px */
}
.content2{
    border:1px solid black;
    background:lightyellow;
    margin-bottom:10px;      /* 아래쪽 마진 10px */
    margin-left:50px;        /* 왼쪽 마진 50px */
    padding:10px 30px;       /* 위아래 패딩 10px, 좌우 패딩 30px */
}
.content3{
    border:1px solid black;
    background:lightyellow;
    margin:20px 10px 0 100px;   /* 위쪽 마진 20px, 오른쪽 마진 10px, 아래쪽
                                마진 없음, 왼쪽 마진 100px */
    padding:10px 30px 10px 50px; /* 위쪽 패딩 10px, 오른쪽 패딩 30px, 아래쪽
                                패딩 10px, 왼쪽 패딩 50px */
}
</style>
</head>
```

margin은 다른 태그와의 거리를 나타냅니다.

사용하는 속성이나 특징은 padding과 거의 동일합니다.

Box-sizing이 border 이든 content 이든 상관없이 width 값에 포함되지 않습니다.

box-shadow : 그림자 효과내기

- 수평거리 : 수평으로 떨어져있는 거리 지정. 양수값은 오른쪽, 음수값은 왼쪽 표시
- 수직거리 : 수직으로 떨어져있는 거리 지정. 양수값은 위쪽, 음수값은 아래쪽 표시
- blur : 그림자의 흐림정도를 지정. 생략시 0으로 기본값을 지님
- 확장거리 : 그림자가 번지는 정도를 표시. 양수값은 전방으로 퍼져나가서 박스보다 그림자가 크게 표시. 음수값은 그림자가 축소되어 보임. 기본값은 0.
- 그림자 색상 : 그림자 색상 지정, 16진수나, 색상이름, RGB값 모두 사용가능. 기본값은 검정
- inset : inset 속성 사용시 박스안쪽에 그림자를 표시, 미사용시 바깥에 그림자 표시.

```
<style>
.bttm{
    width:200px;
    height:100px;
    margin:50px;
    background:blue;
    border:1px solid black;
    border-radius:50px;
    -moz-border-radius:50px;
}
.shd1 {
    box-shadow: 10px 10px 5px 5px #666 ;
}
.shd2 {
    box-shadow: 10px 10px 50px -1px white inset;
}
```

box-shadow : 수평거리값 | 수직거리값 |
blur값 | 확장거리값 | 색상값 | inset;

5. Position

position: static



position: relative



position: absolute



자 포지션은 내용이 많습니다. 말 그대로 위치를 나타냅니다.
무엇의 위치일까요? 바로 태그들의 위치를 나타내주는 CSS 입니다.

position : static

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>요안도라</title>
<style>
div{
width:200px;
height:auto;
margin:10px 5px;
padding:5px;
border:1px solid black;
}
#cont1 {
position:static; /* 다른 요소와 겹치지 않게 배치 */
background:lightyellow;
}
#cont2 {
position:static; /* 다른 요소와 겹치지 않게 배치 */
background:lightgreen;
}
#cont3 {
position:static; /* 다른 요소와 겹치지 않게 배치 */
background:lightblue;
}
</style>
</head>
<body>
<h2>이용 안내</h2>
<div id="cont1">
<p>입실은 오후 3시, 퇴실은 오전 11시입니다.</p>
<p>이곳은 관광지가 아닌, 놀사를 짓는 중산간 마을입니다.</p>
</div>
<div id="cont2">
<p>바베큐, 야외식사 등 바깥채, 외부 앞뜰에서의 활동은 밤 11시 이전까지 마쳐 주셔야 합니다. 바깥채 내부의 소동은 자유롭게 하시되, 자정 이후로 음주나 소란스러운 대화를 자제해 주시면 고맙겠습니다.</p>
</div>
<div id="cont3">
<p>우도, 성산일출봉, 섬지코지, 김영갑 갤러리, 용눈이 오름 등 주변 관광지와 한라산 동쪽 등산로로 오가실 경우 요안도라에서 툴택시를 이용하실 수 있습니다.</p>
</div>
</body>
</html>
```

일단 모든 태그들은 처음에 position : static 상태입니다.
차례대로 왼쪽에서 오른쪽, 위에서 아래로 쌓입니다

position : relative

```
<head>
<meta charset="utf-8">
<title>요안도라</title>
<style>
div{
    width:200px;
    height:auto;
    margin:10px 5px;
    padding:5px;
    border:1px solid black;
}
#cont1 {
    position: relative;           /* 다른 요소와 자연스럽게 어울리도록 */
    background: lightyellow;
}
#cont2 {
    position: relative;           /* 다른 요소와 자연스럽게 어울리도록 */
    left: 20px;                  /* 원래 위치에서 오른쪽으로 20px 떨어져서 배치 */
    top: -20px;                 /* 원래 위치에서 위로 20px 떨어져서 배치 */
    background: lightgreen;
}
#cont3 {
    position: relative;           /* 다른 요소와 자연스럽게 어울리도록 */
    left: -30px;                 /* 원래 위치에서 왼쪽으로 30px 떨어져서 배치 */
    top: 20px;                   /* 원래 위치에서 아래로 20px 떨어져서 배치 */
    background: lightblue;
}
</style>
```

이제 태그의 위치를 변경시킬건데요.

이전에 적은 요소에서 자연스럽게 이어서 위치시키고 싶을때

우리는 position : relative 를 사용합니다.

top, right, bottom, left 속성을 이용하여 표기합니다.

position : absolute

```
<style>
div{
    width:200px;
    height:auto;
    margin:10px 5px;
    padding:5px;
    border:1px solid black;
}
#cont1 {
    position:absolute;
    left:20px;          /* 부모 요소(body) 왼쪽 위로부터 오른쪽으로 20px
    떨어져서 배치 */
    top:50px;           /* 부모 요소(body) 왼쪽 위로부터 아래로 50px 떨어져서
    배치 */
    background:lightyellow;
}
#cont2 {
    position:absolute;
    left:270px;          /* 부모 요소(body) 왼쪽 위로부터 오른쪽으로 27px
    떨어져서 배치 */
    top:100px;           /* 부모 요소(body) 왼쪽 위로부터 아래로 100px 떨어져서
    배치 */
    background:lightgreen;
}
#cont3 {
    position:absolute;
    left:520px;          /* 부모 요소(body) 왼쪽 위로부터 오른쪽으로 520px
    떨어져서 배치 */
    top:150px;           /* 부모 요소(body) 왼쪽 위로부터 아래로 150px 떨어져서
    배치 */
    background:lightblue;
}
</style>
```

Absolute 는 static 속성을 가지고 있지 않은 부모를 기준으로 움직입니다.

만약 부모 중에 포지션이 relative, absolute, fixed인 태그가 없다면, 가장 위 태그인 body 가 기준이 됩니다.

position : fixed

```
<style>
div{
width:200px;
height:auto;
margin:10px 5px;
padding:5px;
border:1px solid black;
}
#cont1 {
position:fixed; /* left와 top 값에 맞춰 위치 고정 */
left:20px;
top:50px;
background:lightyellow;
}
#cont2 {
position:fixed; /* left와 top 값에 맞춰 위치 고정 */
left:270px;
top:200px;
background:lightgreen;
}
#cont3 {
position:fixed; /* left와 top 값에 맞춰 위치 고정 */
left:520px;
top:450px;
background:lightblue;
}
</style>
```

제 블로그를 보시면 아실 겁니다. 스크롤을 내려도 항상 고정되어있는 부분이 있죠.

바로 fixed로 설정되어있기 때문입니다.

이처럼 fixed 속성은 항상 특정 위치에 고정되어 있게 해주는 역할을 합니다.

visibility

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>요안도라</title>
<link href="style.css" rel="stylesheet" type="text/css">
<style>
img {
    border:2px solid black;
    border-radius:10px;
    -moz-border-radius:10px;
    -webkit-border-radius:10px;
    box-shadow:5px 5px 3px #ccc;
}
.visible{
    visibility:visible;
}
.hidden{
    visibility:hidden;
}
</style>
<!--[if lt IE 9]>
<script src="html5shiv.js"></script>
<![endif]-->
</head>

<body>
<section class="content">
    
    
    
</section>
</body>
</html>
```

- **visible** : 요소가 화면에 표시됨. 기본값입니다.
- **hidden** : 요소가 화면에서 감춰짐. 크기는 유지되기 때문에 배치에 영향을 줍니다.
- **collapse** : 표의 행이나 열, 행그룹, 열그룹 등에서 서로 겹쳐지도록 지정합니다.
- **inherit** : 부모요소의 visibility 속성을 상속합니다.

float

```
<style>
body {
    font: 14px 맑은 고딕;
}
.container {
    width: 90%;
    margin: 0 auto;
    background-color:#fff;
    border:1px solid #E7E7E7;
}
header {
    height: 280px;
    margin-top: 0;
    background-image: url(bg.png);
    background-repeat: no-repeat;
    background-position: left top;
    background-size:cover;
}
.content {
    float: left;
    background-color: #FFFFFF;
    padding-top:5px;
    padding-left:10px;
    width: 60%;
}
h4 {
    font-size:12px;
    color:#FF3300;
}

aside{
    float: right;
    width: 30%;
    height:auto;
    background-color:#DFD;
    padding-top:20px;
    padding-left:10px;
    padding-right:10px;
}
aside img{
    display:block;
    margin-bottom:5px;
    margin-left:5px;
}
footer {
    clear: both;
    padding: 10px 0;
    background-color:#030;
    color:#fff;
    text-align:center;
}
header, section, footer, aside, article, figure {
    display: block;
}
</style>
<body>
<div class="container">
<section class="content">
<h2>예약 방법 및 요금</h2>
<article>
<h3>• 예약 방법</h3>
<p>요안도라 예약의 가장 빠른 방법은<br>
인터넷 전화 070.4529.8743으로 직접 통화하시거나 문자를 남기시는 것입니다.<br>
인터넷 전화라 시외전화 요금이 부과되지 않습니다.</p>
</article></section>
<aside>


</aside>
<footer>
<p>제주특별자치도 남제주군 성산읍 수산리 000 번지 요안도라 </p>
<p>Copyright ©. All rights reserved.</p>
</footer>
</div>
</body>
```

float 속성은 문서를 정렬하는 방법중 하나로, 웹 요소를 문서위에 떠있게 만들어줍니다.

정확한 위치값을 가지고 배치하는것과 다르게, 떠있다 혹은 플로팅된다 라고 합니다.

요소가 떠있으려면, 위치가 고정되서는 안되기 때문에 position 속성이 absolute 가 되게 해
서는 안됩니다.

z-index

```
<style>
h1 {
    font-family:Arial;
    font-size:60px;
    font-weight:bold;
    text-align:center;
}
div{
    width:200px;
    height:150px;
    margin:10px 5px;
    padding:20px;
    border:1px solid black;
    color:yellow;
}
#cont1 {
    position:absolute;
    left:10px;
    top:10px;
    background:#ff6600;
    z-index:1;
}
#cont2 {
    position:absolute;
    left:180px;
    top:20px;
    background:#00b000;
    z-index:3;
}
#cont3 {
    position:absolute;
    left:25px;
    top:160px;
    background:#0000ff;
    z-index:2;
}
</style>

<body>
<div id="cont1">
    <h1>A</h1>
</div>
<div id="cont2">
    <h1>B</h1>
</div>
<div id="cont3">
    <h1>C</h1>
</div>
</body>
</html>
```

보통 나중에 온 태그가 먼저 나온 태그 위에 위치합니다. 먼저 나온 태그를 위에 배치하고 싶을 때는 어떻게 해야 할까요? 이 때 우리는 position과 z-index라는 녀석을 사용합니다.

먼저 z-index를 사용하는 규칙입니다.

1. position 속성이 없는 태그들은 나오는 순서대로 쌓입니다.
2. position 속성이 있는 태그들은 없는 태그들보다 위에 나오는 순서대로 쌓입니다.
3. position 속성에 z-index 까지 있다면, z-index가 가장 큰 순서대로 쌓입니다.
4. 하지만 z-index가 아무리 크더라도 부모 z-index가 우선입니다.

z-index는 position이 relative거나 absolute 여야 동작합니다.

먼저 position이 static인 태그들이 나오고 순서대로 쌓이고, 그위에 relative나 absolute인 태그들이 나오는 순서대로 쌓입니다.

```
* {
  margin: 0;
}
html {
  padding: 20px;
  font: 12px/20px Arial, sans-serif;
}
div {
  opacity: 0.7;
  position: relative;
}
h1 {
  font: inherit;
  font-weight: bold;
}

#div1 {
  z-index: 5;
  margin-bottom: 190px;
}
#div2 {
  z-index: 2;
  bottom: -10px;
}
#div3 {
  z-index: 4;
  opacity: 1;
  position: absolute;
  top: 40px;
  left: 180px;
  width: 330px;
  border: 1px dashed #900;
  background-color: #fd0;
  padding: 40px 20px 20px;
}
#div4, #div5 {
  border: 1px dashed #996;
  background-color: #ffc;
}
#div4 {
  z-index: 6;
  margin-bottom: 15px;
  padding: 25px 10px 5px;
}
#div5 {
  z-index: -1;
  margin-top: 15px;
  padding: 5px 10px;
}
#div6 {
  position: absolute;
  top: 20px;
  left: 180px;
  width: 150px;
  height: 125px;
  border: 1px dashed #009;
  padding-top: 125px;
  background-color: #ddf;
  text-align: center;
}

<body>
  <div id="div1">
    <h1>Division Element #1</h1>
    <code>position: relative;<br/>
    z-index: 5;</code>
  </div>

  <div id="div2">
    <h1>Division Element #2</h1>
    <code>position: relative;<br/>
    z-index: 2;</code>
  </div>

  <div id="div3">

    <div id="div4">
      <h1>Division Element #4</h1>
      <code>position: relative;<br/>
      z-index: 6;</code>
    </div>

    <h1>Division Element #3</h1>
    <code>position: absolute;<br/>
    z-index: 4;</code>

    <div id="div5">
      <h1>Division Element #5</h1>
      <code>position: relative;<br/>
      z-index: -1;</code>
    </div>

    <div id="div6">
      <h1>Division Element #6</h1>
      <code>position: absolute;</code>
    </div>
  </div>

  <div id="div7">
    <h1>Division Element #7</h1>
    <code>position: static;</code>
  </div>
</body>
```

Division Element #1

```
position: relative;  
z-index: 5;
```

Division Element #4

```
position: relative;  
z-index: 6;
```

Division Element #3

```
position: absolute;  
z-index: 4;
```

Division Element #6

```
position: absolute;
```

Division Element #2

```
position: relative;  
z-index: 2;
```

Division Element #5

```
position: relative;  
z-index: -1;
```

Division Element #7

```
position: static
```