

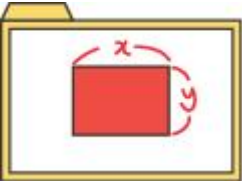
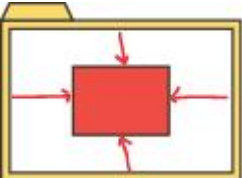
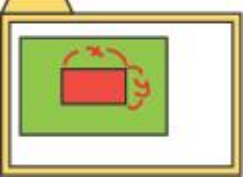
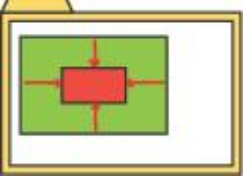


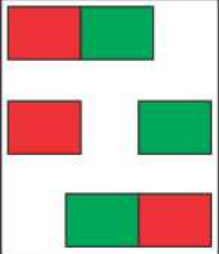

CSS/jQuery ISSUE

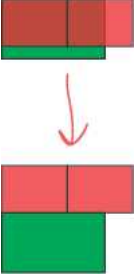
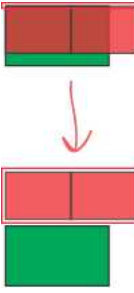
박스 안의 텍스트를 상하 가운데 정렬하는 법			
SOLUTION	HTML	CSS	ISSUE
line-height를 이용하는 법 	<pre><div id= "box" > TEXT </div></pre>	<pre>#box { height: 50px; line-height: 50px; }</pre>	박스의 높이만큼 line-height를 만들어준다. line-height를 쓰면 텍스트가 줄 간격 사이 가운데에 위치하는 현상을 이용한 것. 글이 두 줄 이상 될 경우 두 번째 줄의 글들이 박스를 뚫고 박스의 높이만큼 아래로 떨어진다. 글이 한 줄만 있을 경우 사용 가능.
padding을 이용하는 법 		<pre>#box { padding: 50px 0px; }</pre>	글이 두 줄 이상 생겨도 괜찮지만 박스의 높이를 특정할 수 없다. 박스의 높이가 특별히 정해지지 않아도 되는 경우 사용 가능.
height와 padding을 이용하는 법		<pre>#box { height: 50px; padding-top: 10px; }</pre>	높이를 특정할 수 있고 여러 줄을 쓸 수 있으나 정확한 padding-top값을 찾을 수 없고 글의 내용이 바뀌면 값을 새로 정해주어야 한다.
display: table-cell을 이용하는 법		<pre>#box { display: table-cell; height: 50px; }</pre>	높이를 특정할 수 있고 여러 줄을 쓸 수 있으나 display의 값이 블록이나 인라인이 아닌 테이블 셀의 속성이 되어 버린다.

박스를 화면 전체에 꽉 차도록 만드는 법			
SOLUTION	HTML	CSS	ISSUE
position과 left/right/top/bottom을 이용하는 법 	<pre><div id= "box" > TEXT </div></pre>	<pre>#box { position: absolute; left: 0px; right: 0px; top: 0px; bottom: 0px; }</pre>	자식 박스를 absolute하고 가로와 세로의 크기를 주지 말아야 한다. 부모 중 relative 속성을 가지고 있는 박스가 있을 경우 그 박스 안쪽에 꽉 차게 되며, 그렇지 않은 경우 화면 기준으로 꽉 차게 된다. 스크롤바의 영향을 받지 않게 하려면 absolute 대신 fixed를 사용한다.


박스를 화면의 정 가운데에 배치하는 법 ☆X 6			
SOLUTION	HTML	CSS	ISSUE
position과 transform을 이용하는 법 	<pre><div id= "box" > TEXT </div></pre>	<pre>#box { width: 200px; height: 100px; position: absolute; left: 50%; top: 50%; transform: translate(-50%, -50%); }</pre>	박스의 크기를 특정할 수 있다. 브라우저 화면의 정 중앙에 배치되려면 부모 태그 중 position: relative가 있어야 한다. 스크롤 할 때에도 박스를 화면 중앙에 놓으려면 position의 값을 fixed로 한다.
position과 left/right/top/bottom을 이용하는 법 		<pre>#box { position: absolute; left: 50px; right: 50px; top: 50px; bottom: 50px; }</pre>	박스의 크기를 특정하지 않고 브라우저 화면의 사방 끝으로부터 일정한 여백을 유지하면서 박스를 화면에 가운데에 배치할 수 있다. 위와 마찬가지로 부모 중 relative가 있어야 하며 스크롤의 영향을 받지 않으려면 position: fixed 할 수 있다.

부모 박스 안에 자식 박스를 정 가운데에 배치하는 법			
SOLUTION	HTML	CSS	ISSUE
position과 transform을 이용하는 법 	<pre><div id= "bigbox" > <div id= "box" > TEXT </div> </div></pre>	<pre>#bigbox { width: 400px; height: 200px; position: relative; } #box { width: 200px; height: 100px; position: absolute; left: 50%; top: 50%; transform: translate(-50%, -50%); }</pre>	박스의 크기를 특정할 수 있다. 부모 박스의 정 중앙에 배치하려면 부모 박스 태그에 position: relative가 있어야 한다.
position과 left/right/top/bottom을 이용하는 법 		<pre>#bigbox { width: 400px; height: 200px; position: relative; } #box { position: absolute; left: 50px; right: 50px; top: 50px; bottom: 50px; }</pre>	박스의 크기를 특정하지 않고 부모 박스 사방 끝으로부터 일정한 여백을 유지하면서 박스를 화면에 가운데에 배치할 수 있다. 위와 마찬가지로 부모 박스에 relative가 있어야 한다.

박스를 옆으로 나란히 배치하는 법			
SOLUTION	HTML	CSS	ISSUE
float를 이용하는 법 	<pre><div id= "box1" > TEXT1 </div> <div id= "box2" > TEXT2 </div></pre>	<pre>#box1 { float: left; } #box2 { float: left; }</pre>	box1 태그부터 왼쪽으로 쌓듯이 배치한다. 특별히 margin을 주지 않는 이상 박스 사이에 간격은 0이다.
		<pre>#box1 { float: left; } #box2 { float: right; }</pre>	box1 태그는 화면의 왼쪽 끝, box2는 화면의 오른쪽 끝에 배치한다.
		<pre>#box1 { float: right; } #box2 { float: right; }</pre>	box1 태그를 먼저 화면의 오른쪽 끝에 놓고 box2를 box1의 왼쪽에 배치시킨다. 즉. 박스들이 오른쪽 끝에 몰리지만 박스들의 순서가 뒤집힌다.
		※ float는 일반적인 방법이지만 float의 속성 때문에 이후에 등장하는 요소들이 float된 박스 밑으로 빠지는 경우가 있음. 그러므로 이들을 감싸는 부모 박스에 높이를 주어 이후에 오는 요소가 부모 박스에 막히게 만들든지, 이후에 오는 요소에 clear를 주어 아예 float 밑으로 빠지지 않게 만들어주어야 함.	
inline을 이용하는 법 		<pre>#box1, #box2 { display: inline; }</pre>	박스들이 순서대로 좌우 배치되고 이 둘을 담고 있는 박스에 text-align을 주어 가운데에 나란히 놓을 수도 있으나 사이에 간격이 생긴다. 박스에 크기를 주기 위해서는 inline-block을 사용한다.

float 된 박스 이후에 나타나는 요소들이 float된 박스 밑에 묻힐 때의 해결 방법			
SOLUTION	HTML	CSS	ISSUE
<p>clear를 이용하는 법</p> 	<pre> <div id= "box1" > TEXT1 </div> <div id= "box2" > TEXT2 </div> <div id= "box3" > TEXT2 </div> </pre>	<pre> #box1,#box2 { float: left; } #box3 { clear: both; } </pre>	<p>위에서 float된 요소들 바로 다음에 오는 box3이 clear:both를 가지고 있으므로 float된 요소들 밑에 빠지지 않는다.</p>
<p>부모 박스의 높이를 이용하는 법</p> 	<pre> <div id= "parentBox" > <div id= "box1" > TEXT1 </div> <div id= "box2" > TEXT2 </div> </div> <div id= "box3" > TEXT2 </div> </pre>	<pre> #parentBox { height: 300px; } #box1,#box2 { float: left; height: 300px; } </pre>	<p>float된 박스들을 부모박스가 감싸고 있으며 부모 박스 자체에 높이 값이 있으므로 box3은 부모박스 바로 밑에 위치하게 된다. 따라서 box3은 float된 박스들 밑에 빠지지 않게 된다.</p>
		<pre> #parentBox { overflow: auto; } #box1,#box2 { float: left; height: 300px; } </pre>	<p>float된 박스들을 부모박스가 감싸고 있으며 부모 박스 자체에는 높이 값이 없으나 overflow가 auto이므로 안쪽에 있는 float된 박스들의 높이, 즉 300px를 가지게 된다. 따라서 box3은 float된 박스들 밑에 빠지지 않게 된다.</p>


박스를 옆으로 나란히 배치하고 왼쪽, 중앙 또는 오른쪽에 몰아놓는 법			
SOLUTION	HTML	CSS	ISSUE
<p>왼쪽에 모는 법</p>	<pre> <div id= "parent" > <div id= "box1" > TEXT1 </div> <div id= "box2" > TEXT2 </div> </div> </pre>	<pre> #parent { width: 300px; float: left; } #box1 { width: 150px; float: left; } #box2 { width: 150px; float: left; } </pre>	<p>일반적으로 왼쪽의 예시처럼 부모 박스로 묶지 않아도 왼쪽에 몰수 있으나 다른 박스들과 혼합하여 배치할 경우 부모 박스로 묶어줄 필요가 있다.</p>
<p>중앙에 모는 법</p>		<pre> #parent { width: 300px; margin: auto; } </pre>	<p>부모 박스에 가로길이를 주고 양쪽 margin을 auto로 줌으로써 박스를 가운데에 배치할 수 있다. 가로길이를 주지 않으면 기본값인 100%로 잡히기 때문에 안쪽에 있는 자식 박스들이 왼쪽에 몰려있는 것처럼 보인다.</p>
<p>오른쪽에 모는 법</p>		<pre> #parent { width: 300px; float: right; } </pre>	<p>부모 박스에 가로길이를 주고 오른쪽으로 float 하였다. 가로길이를 주지 않으면 기본값인 100%로 잡히기 때문에 안쪽에 있는 자식 박스들이 왼쪽에 몰려있는 것처럼 보인다.</p>

박스를 겹치는 법			
SOLUTION	HTML	CSS	ISSUE
<p>position을 이용하는 법</p> 	<pre> <div id= "box1" > TEXT1 </div> <div id= "box2" > TEXT2 </div> </pre>	<pre> #box1 { position: absolute; left: 50px; top: 50px; } #box2 { position: absolute; left: 50px; top: 50px; } </pre>	<p>박스들이 겹쳐져야 한다면 position:absolute가 유일한 방법이다. left, right, top, bottom을 이용하여 박스를 움직일 수 있다. 아래쪽에 묻히는 박스는 굳이 absolute가 아닐 수도 있다.</p>

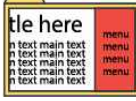
서브메뉴 만드는 법


SOLUTION	HTML	CSS / jQuery
CSS만 이용하는 법	<pre><nav> <li class= "main" > <li class= "main" > </nav></pre>	<pre>.main>ul { display: none; position: absolute; } .main:hover>ul { display: block; }</pre>
CSS와 jQuery를 이용하는 법	<p>메인메뉴를 좌우로 배치하고 서브메뉴를 position:absolute하여 아래쪽으로 띄우면 일반적인 서브메뉴 세트가 되고, 모든 메인메뉴와 서브메뉴를 순수한 position:static으로 두고 상하 배치하면 아코디언 메뉴 세트가 된다.</p>	<pre>.main { float: left; width: 300px; height: 50px; } .main>a { display: none; width: 100%; height: 50px; } .main>ul>li>a { display: none; width: 100%; height: 40px; } .main>ul { width: 300px; position: absolute; display: none; } \$(document).ready(function(){ \$(".main").mouseover(function(){ \$(this).children("ul").slideDown(); }); \$(".main").mouseout(function(){ \$(this).children("ul").slideUp(); }); });</pre>


화면 외부에서 슬라이드 되면서 콘텐츠 위로 겹쳐지며 나타나는 메뉴 만드는 법

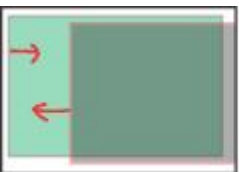
SOLUTION	HTML	CSS / jQuery
CSS와 jQuery를 이용하는 법	<pre>메뉴 <nav> <li class= "main" > <li class= "main" > </nav></pre> 	<pre>nav { position: fixed; top: 0px; bottom: 0px; width: 300px; right: -300px; } \$(document).ready(function(){ var stat = 0; \$("#btn").click(function(){ if(stat == 0){ \$(".nav").animate({right: "0px"}); stat = 1; }else{ \$(".nav").animate({right: "-300px"}); stat = 0; } }); });</pre> <p>※ 토글 버튼 만들기 참조</p>

화면 외부에서 슬라이드 되면서 콘텐츠를 밀며 나타나는 메뉴 만드는 법

SOLUTION	HTML	CSS / jQuery
CSS와 jQuery를 이용하는 법	<pre><div id= "contents" > 메뉴 </div> <nav> <li class= "main" > <li class= "main" > </nav></pre> 	<pre>body { overflow-x: hidden; } nav { position: fixed; top: 0px; bottom: 0px; width: 300px; right: -300px; } \$(document).ready(function(){ var stat = 0; \$("#btn").click(function(){ if(stat == 0){ \$(".nav").animate({right: "0px"}); \$(".contents").animate({marginLeft: "-300px"}); stat = 1; }else{ \$(".nav").animate({right: "-300px"}); \$(".contents").animate({marginLeft: "0px"}); stat = 0; } }); });</pre> <p>※ 토글 버튼 만들기 참조</p>

모달 만드는 법		
HTML	CSS	jQuery
<pre> <div id= "modal" > <div id= "modalBox" > <p></p> </div> <div id= "cl" >close</div> </div> </pre>	<pre> #modal { position: fixed; left: 0px; right: 0px; top: 0px; bottom: 0px; background-color: rgba(0,0,0,0.5); display: none; } #modalBox { width: 400px; height: 200px; background-color: white; position: absolute; left: 50%; right: 50%; transform: translate(-50%, -50%); } #cl { position: absolute; top: 20px; right: 20px; } </pre>	<pre> \$(document).ready(function(){ \$(".con").click(function(){ var aaa = \$(this).children("img") .attr("src"); var bbb = \$(this).children("img") .attr("alt"); \$("#modalBox>img").attr("src",aaa); \$("#modalBox>p").text(bbb); \$("#modal").show(); }); \$("#cl").click(function(){ \$("#modal").hide(); }); }); </pre> 

그림이 페이드 인/아웃 되는 슬라이더 만드는 법		
HTML	CSS	jQuery
<pre> <div id= "slider" > <div id= "left" >&lt;</div> <div id= "right" >&gt;</div> </div> </pre> 	<pre> #slider { width: 300px; height: 200px; overflow: hidden; position: relative; } #slider>img { width: 100%; position: absolute; } #left,#right { position: absolute; width: 50px; top: 0px; bottom: 0px; } #left {left: 0px;} #right {right: 0px;} </pre>	<pre> \$(document).ready(function(){ var len = \$("#slider>img").length; var cur = 0; function sliding(dir){ cur = cur + dir; if(cur >= len){ cur = 0; }else if(cur < 0){ cur = len - 1; } \$("#slider>img").fadeOut(); \$("#slider>img").eq(cur).fadeIn(); } \$("#left").click(function(){ sliding(-1); }); \$("#right").click(function(){ sliding(1); }); }); </pre>

그림들이 화면 오른쪽 밖에 모여 있다가 한 장씩 움직이며 보여주는 슬라이더 만드는 법		
HTML	CSS	jQuery
<pre> <div id= "slider" > <div id= "left" >&lt;</div> <div id= "right" >&gt;</div> </div> </pre> 	<pre> #slider { width: 300px; height: 200px; overflow: hidden; position: relative; } #slider>img { width: 100%; position: absolute; right: -300px; transition: all 0.5s; } #left,#right { position: absolute; width: 50px; top: 0px; bottom: 0px; } #left {left: 0px;} #right {right: 0px;} </pre>	<pre> \$(document).ready(function(){ var len = \$("#slider>img").length; var cur = 0; function sliding(dir){ cur = cur + dir; if(cur >= len){ cur = 0; }else if(cur < 0){ cur = len - 1; } \$("#slider>img").css("right", "-300px"); \$("#slider>img").eq(cur) .css("right", "0px"); } \$("#left").click(function(){ sliding(-1); }); \$("#right").click(function(){ sliding(1); }); }); </pre>

여러 그림이 옆으로 나란히 움직이는 슬라이더 만드는 법

HTML	CSS	jQuery
<pre> <div id= "slider" > <div id= "move" > </div> <div id= "left" >&lt;</div> <div id= "right" >&gt;</div> </div> </pre> 	<pre> #slider { width: 300px; height: 200px; overflow: hidden; position: relative; } #move { height: 200px; } #move>img { display: block; width: 300px; float: left; } #left,#right { position: absolute; width: 50px; top: 0px; bottom: 0px; } #left {left: 0px;} #right {right: 0px;} </pre>	<pre> \$(document).ready(function(){ var len = \$("#slider>img").length; \$("#move").width(len * 300); var cur = 0; function sliding(dir){ cur = cur + dir; if(cur >= len){ cur = 0; }else if(cur < 0){ cur = len - 1; } \$("#move") .animate({marginLeft:cur * -300 + "px"}); } \$("#left").click(function(){ sliding(-1); }); \$("#right").click(function(){ sliding(1); }); }); </pre>

위 슬라이더들을 자동재생 하도록 만드는 법

jQuery에 아래 내용을 추가한다.

```

$(document).ready(function(){
  $("#left").click(function(){
    sliding(-1);
  });
  $("#right").click(function(){
    sliding(1);
  });

  var auto = setInterval(sliding,3000,1); // 3초마다 sliding(1)을 실행
  $("#slider").mouseover(function(){ // 슬라이더에 마우스를 올렸을 땐 interval을 삭제한다.
    clearInterval(auto);
  });
  $("#slider").mouseout(function(){ // 슬라이더에서 마우스를 치웠을 땐 interval을 재개한다.
    auto = setInterval(sliding,3000,1);
  });
});

```

토글 버튼 만드는 법

SOLUTION	HTML	jQuery
toggle()을 이용하는 법	<pre> TEXT <div id= "box" > TEXT </div> </pre>	<pre> \$(document).ready(function(){ \$("#btn").click(function(){ \$("#box").toggle(); }); }); </pre>
stat변수를 이용하는 법		<pre> \$(document).ready(function(){ var stat = 0; \$("#btn").click(function(){ if(stat == 0){ \$("#box").show(); stat = 1; }else{ \$("#box").hide(); stat = 0; } }); }); </pre>