HTML*Article*요소 (**<article>**) 는 문서,페이지,앱,사이트 등이 신디케이션 같은것에 의해 독립적으로 배포되거나 재사용될수 있는 독립적인 구성을 나타냅니다. 이것은 포럼의 포스트나, 뉴스/신문 기사나, 블로그 대문 등과 같은 다른 독립적인 항목의 컨텐츠가 될수 있습니다. 각각의 <article>은 식별가능해야하며, 일반적으로 제목 요소 ([h1-h6](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/Heading_Elements) 요소)를 자식으로 포함하고 있습니다.

|  |
| --- |
| <style>  header { background-color:black; color:white; text-align:center; padding:5px; }  nav { line-height:30px; background-color:#eeeeee; height:300p width:10 float:left; padding:5px; }  section { width:350px; float:left; padding:10px; }  footer { background-color:black; color:white; clear:both; text-align:center; padding:5px; }  </style>  </head>  <body>  <header>  <h1>City Gallery</h1>  </header>  <nav>  London<br>  Paris<br>  Tokyo  </nav>  <section>  <h1>London</h1>  <p>London is the capital city of England. It is the most populous city in the United Kingdom,  with a metropolitan area of over 13 million inhabitants.</p>  <p>Standing on the River Thames, London has been a major settlement for two millennia,  its history going back to its founding by the Romans, who named it Londinium.</p>  </section>  <footer>  Copyright W3Schools.com  </footer>  </body> |

Block-level Elements

* <div>
* <h1> - <h6>
* <p>
* <form>
* <header>
* <footer>
* <section>

## Inline Elements

* <span>
* <a>
* <img>

# **CSS Box Model**

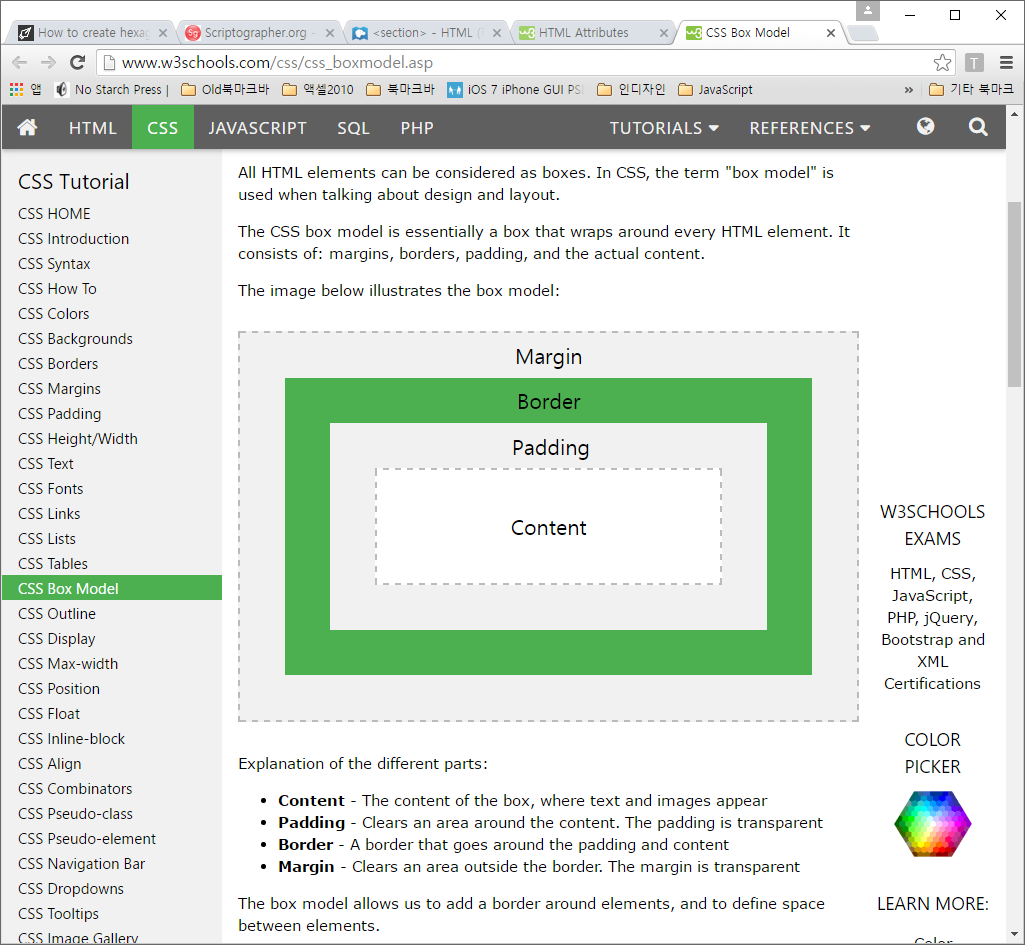
margin 위 , 오른쪽, 아래, 왼쪽

Table { margin : 5px 7px 3px 0px ; }

->위, 오른쪽, 아래, 왼쪽

Table { margin : 5px 10px 10px ; }

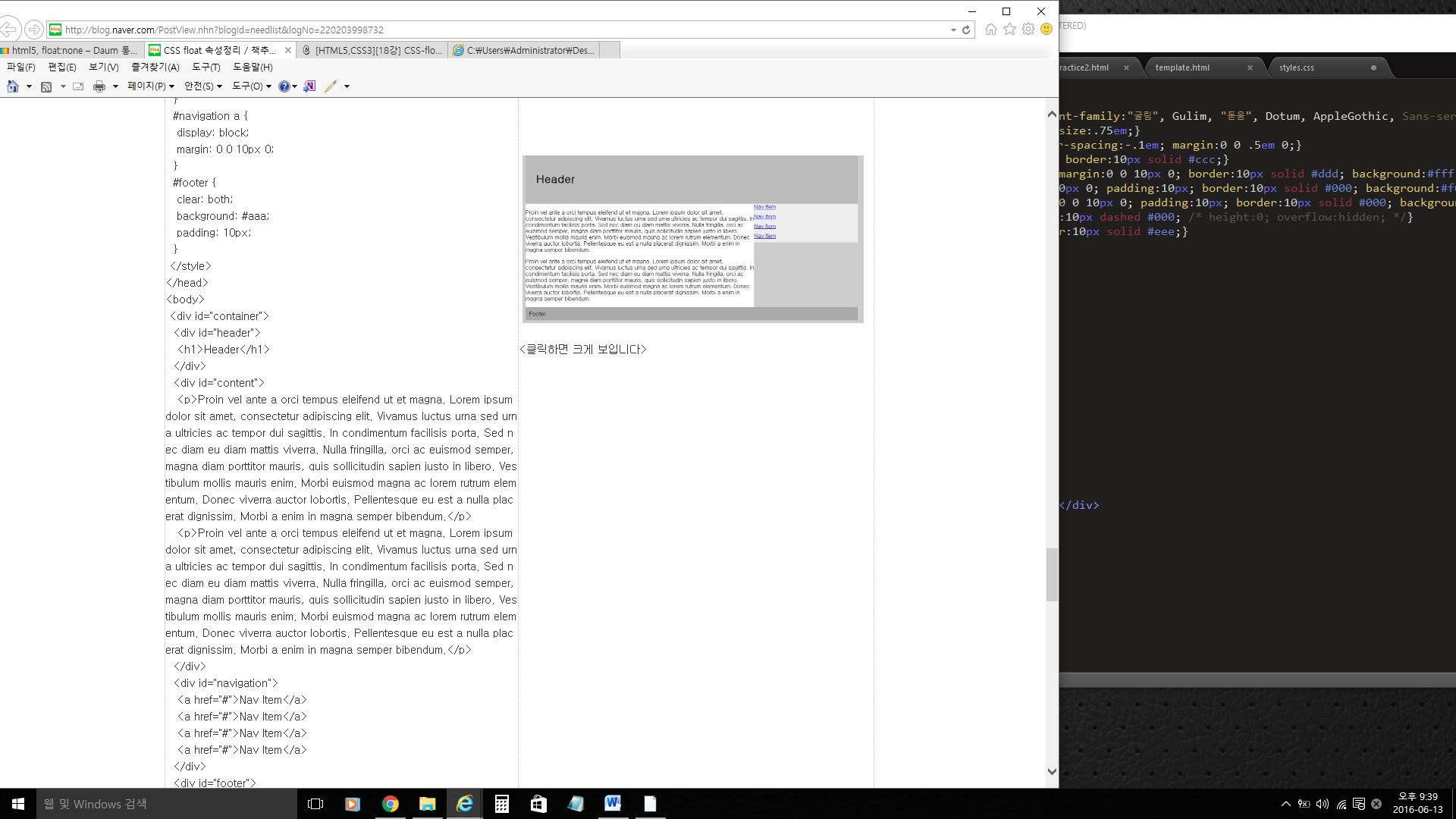
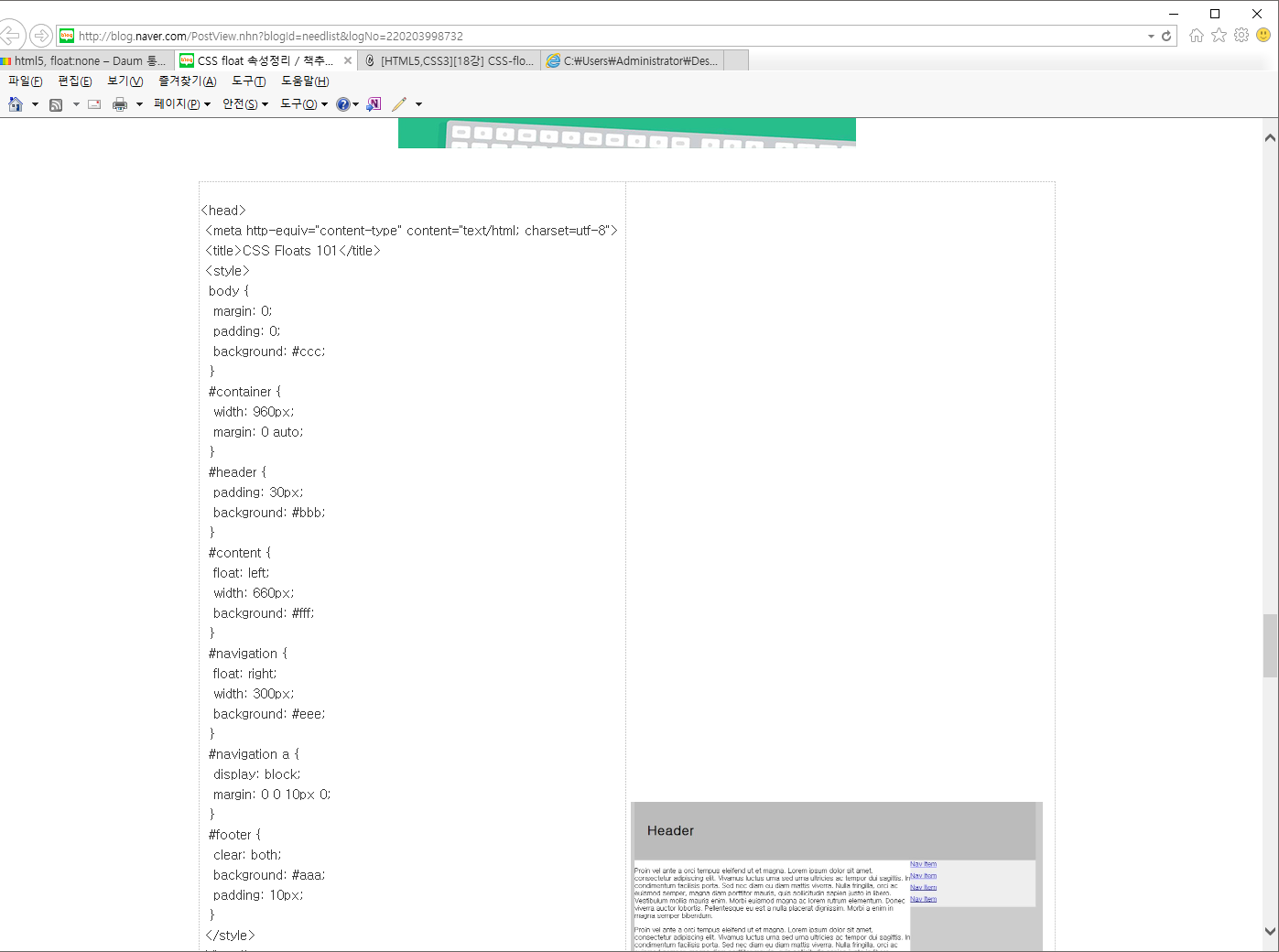
-> 위, 오른쪽 & 왼쪽, 아래쪽

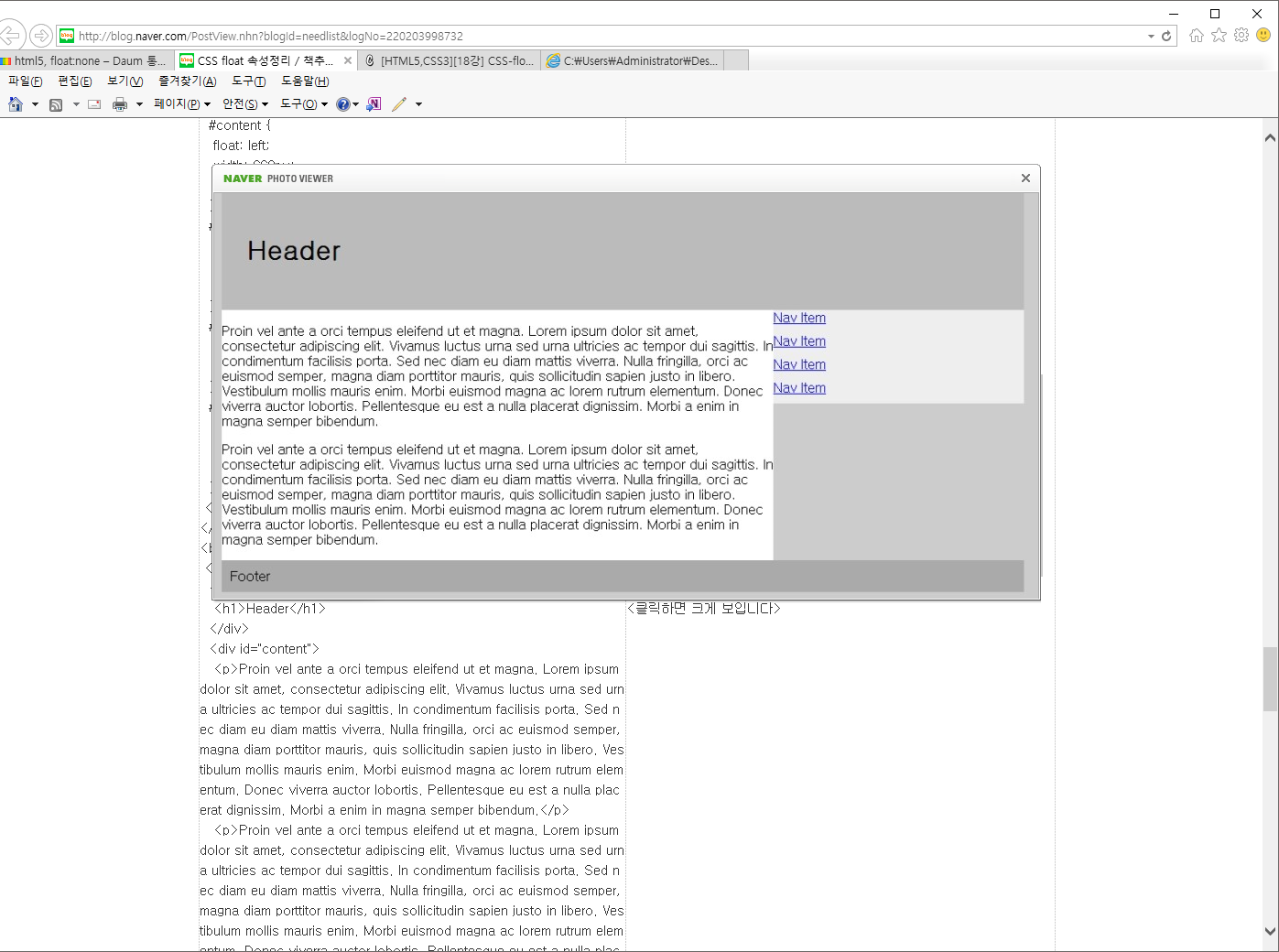
Table { margin : 5px 10px ; }

-> 위&아래, 오른쪽 & 왼쪽

|  |
| --- |
| <style type="text/css">    nav { width: 200px; border: 1px solid #585858; float: left; }  section { border : 3px solid red; margin-left: 210px; }  </style>  </head>  <body>  <nav >  <ul>  <span> Navigation </span>  <li> home </li>  <li> css</li>  <li>html</li>  <li>JavaScript</li>  </ul>  </nav>  <section>  <span> Section Title</span>  <p> Notice we have put a clearfix on the div container. It is not needed in this example, but it would be if the nav element was longer than the non-floated section content</p>  </section>  <section>  <span> Section Another Title</span>  <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet.</p>  </section>  </body> |
| Css 추가 ( li 의 속성을 변환해서 목록처럼 보이지 않게)  nav { width: 200px; border: 1px solid #585858; float: left; }  nav ul li { display : block;}  section { border : 3px solid red; margin-left: 210px; } |
| Css 추가 ( 메뉴를 가로로 쭉 )  nav { width: 600px; border: 1px solid #585858; }  nav ul li { display : inline;}  section { border : 3px solid red; } |
| Float 와 이미지  <style>  img {  float: right;  border: 1px dotted black;  margin: 0px 0px 15px 20px;  }  </style>  </head>  <body>  <p>In the paragraph below, the image will float to the right. A dotted black border is added to the image.  We have also added margins to the image to push the text away from the image:  0 px margin on the top and right side, 15 px margin on the bottom, and 20 px margin on the left side of the image.  </p>  <p><img src="w3css.gif" width="100" height="140">  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  </p>  </body> |
| CSS 속성 가운데 [float](http://trio.co.kr/webrefer/css2/visuren.html#floats) 속성은 자기 자신의 위치를 주변의 콘텐츠로부터 상대적으로 배치하는 속성입니다.  float은 높이가 가변적인 다단 컬럼 형태의 CSS 레이아웃을 위하여 반드시 요구되는 속성이다. clear 속성은 float이 더이상 주변 엘리먼트의 배치에 영향을 미치지 않도록 해제시키는 속성입니다.  <style>  span {  float: left;  width: 0.7em;  font-size: 400%;  font-family: algerian, courier;  line-height: 80%;  }  </style>  </head>  <body>  <p>  <span>T</span>his is some text.  This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  This is some text. This is some text. This is some text.  </p>  <p>  In the paragraph above, the first letter of the text is embedded in a span element.  The span element has a width that is 0.7 times the size of the current font.  The font-size of the span element is 400% (quite large) and the line-height is 80%.  The font of the letter in the span will be in "Algerian".  </p> |
| Float 속성 연구하기  <style type="text/css">  div.container { width: 100%; margin: 0px; border: 1px solid gray;  line-height: 150%; }  div.header, div.footer { padding: 0.5em; color: white; background-color: gray; clear: left; }  h1.header { padding: 0; margin: 0; }  div.left { float: left; width: 160px; margin: 0; padding: 1em; }  div.content { margin-left: 190px; border-left: 1px solid gray; padding: 1em; overflow: hidden;}  </style>  </head>  <body>  <div class="container">  <div class="header">  <h1 class="header">W3Schools.com</h1>  </div>  <div class="left">  <p>"Never increase, beyond what is necessary, the number of entities required to explain anything." William of Ockham (1285-1349)</p>  </div>  <div class="content">  <h2>Free Web Building Tutorials</h2>  <p>At W3Schools you will find all the Web-building tutorials you need.</p>  <p>W3Schools.com - The largest Web Developers' Site on the internet!</p>  </div>  <div class="footer">© Copyright by Refsnes Data.</div>  </div>  </body> |
| <style>  ul {  list-style-type: none; margin: 0; padding: 0; overflow: hidden; background-color: #333; }  li { float: left; }  li a {  display: inline-block; color: white; text-align: center; padding: 14px 16px; text-decoration: none;}  li a:hover { background-color: #111;}  </style>  </head>  <body>  <ul>  <li><a class="active" href="#home">Home</a></li>  <li><a href="#news">News</a></li>  <li><a href="#contact">Contact</a></li>  <li><a href="#about">About</a></li>  </ul> |
| <style type="text/css">  **1**  .\_1 { width: 300px; height: 200px; background-color:grey; float:left; }  .\_2 { width: 200px; height: 200px; background-color: red; float:left;}  .\_3 { width: 400px; height: 200px; background-color: pink; border: 5px solid pink; 나중에 추가해서 clear : left 왼쪽에서 채우면서 내려오던 속성을 해제}  </style>  </head>  <body>  <div class="\_1"> 1 </div>  <div class="\_2"> 2 </div>  <div class="\_3"> 3 </div>  </body> |
| 본문 : **2**  <body>  <p id="container"> container </p>  <div id="lnb"> <p> lnb</p> </div>  <div id="content"> <p> content</p> </div>  <div id="footer" > footer</div>  </body>  -----------------  <title>**float에 아무런 대응도 하지 않은 상태의 예제</**title>  <style type="text/css"> (css 첫번째 )  \* { margin:0; padding: 0; font-size : 1em; font-family: Arial; }  body {background : #ccc; margin-left: 10px; font-size: 1.2em;}  h1{font: 26px; letter-spacing: 0.5em; margin: 0 0 .5em 0; }  #header { padding: 10px; margin: 0 0 10px 0 border: 1px solid #ddd; }  #lnb { width: 25%; margin : 0 0 10px 0; padding: 10px; border: 10px solid #000;}  #container { padding: 10px; margin: 0 0 10px 0; border : 10px solid #000; background:#f00;}  #footer { padding: 10px; border: 10px solid #eee; }  </style>  ----------  <meta charset="utf-8"> css 바꾼거  <style type="text/css">  \* { margin:0; padding: 0; font-size : 1em; font-family: Arial; }  body {background : #ccc; margin-left: 10px; font-size: 1.2em;}  h1{font: 26px; letter-spacing: 0.5em; margin: 0 0 .5em 0; }  #header { padding: 10px; margin: 0 0 10px 0 border: 1px solid #ddd; clear: both; }  #lnb { float: left; width: 25%; margin : 0 0 10px 0; padding: 10px; border: 10px solid #000;}  #content { width: 55%; margin: 0 0 10px 0; padding:10px; border: 10px solid #000; background:#f00;}  #container { float: right; padding: 10px; margin: 0 0 10px 0; border : 10px solid #000; backgr  ound:#f00;}  #footer { clear:both; padding: 10px; border: 10px solid #eee; }  </style>  </head>  </head>  <body>  <h1>**float에 아무런 대응도 하지 않은 상태의 예제**</h1>  <div id="header">  <p>#header {clear:both;}</p>  </div>  <div id="container">  <p>#container {float:none;}</p>  <div id="lnb">  <p><strong>#lnb {float:left;}</strong></p>  </div>  <div id="content">  <p><strong>#content {float:right;}</strong></p>  </div>  </div>  <div id="footer">  <p>#footer {clear:both;}</p>  </div>  </body> |
| 부모가 자식 엘리먼트를 포함하면서 자식의 너비만큼 줄어든다  <style type="text/css">  \*{ margin:0; padding:0; font-size:1em; font-family:"굴림", Sans-serif;}  body{ background:#fff; margin:10px; font-size:.75em;}  h1{ font:26px "바탕", Batang, Serif; letter-spacing:-.1em; margin:0 0 .5em 0;}  #header{ padding:10px; margin:0 0 10px 0; border:10px solid #ccc;}  #container{ float:left; padding:10px; margin:0 0 10px 0; border:10px solid #ddd; background:#fff;}  #lnb{ float:left; width:25%; margin:0 0 10px 0; padding:10px; border:10px solid #000; background:#f00;}  #content{ float:right; width:55%; margin:0 0 10px 0; padding:10px; border:10px solid #000; background:#f00;}  #footer{ clear:both; padding:10px; border:10px solid #eee;}  </style>  </head>  <body>  <h1>**float에 float으로 대응하는 방법 예제 ( 권장하지 않음)** </h1>  <div id="header">  <p>#header {clear:both;}</p>  </div>  <div id="container">  <p><strong>#container {float:left;}</strong></p>  <div id="lnb">  <p><strong>#lnb {float:left;}</strong></p>  </div>  <div id="content">  <p><strong>#content {float:right;}</strong></p>  </div>  </div>  <div id="footer">  <p>#footer {clear:both;}</p>  </div>  </body>` |
| 부모 엘리먼트에 overflow:auto 또는 overflow:hidden 속성을 부여하는 방법  overflow:auto 속성은 자식의 너비가 가변적이고 부모의 너비보다 커지는 상황이 발생할 때 가로 스크롤바를 유발하기 때문에 일반적으로 권장하는 방식이 아닙니다. overflow:hidden 속성은 그러한 상황에서 가로 스크롤바를 유발하지는 않지만 자식의 너비가 넘치는 경우 넘치는 부분이 잘리기 때문에 이 역시 완전하게 안전한 방법은 아닙니다.  **3**  <style type="text/css">  \*{ margin:0; padding:0; font-size:1em;}  body{ background:#fff; margin:10px; font-size:.75em;}  h1{ font:26px margin:0 0 .5em 0;}  #header{ padding:10px; margin:0 0 10px 0; border:10px solid #ccc;}  #container{ overflow:hidden ( auto) ; padding:10px; margin:0 0 10px 0; border:10px solid #ddd; }  #lnb{ float:left; width:25%; margin:0 0 10px 0; padding:10px; border:10px solid #000; background:#f00;}  #content{ float:right; width:55%; margin:0 0 10px 0; padding:10px; border:10px solid #000; background:#f00;}  #footer{ clear:both; padding:10px; border:10px solid #eee;}  </style>  </head>  <body>  <h1>float에 overflow 속성으로 대응하는 방법</h1>  <div id="header">  <p>header</p>  </div>  <div id="container">  <p>container</p>  <div id="lnb">  <p>lnb </p>  </div>  <div id="content">  <p>content </p>  </div>  </div>  <div id="footer">  <p>footer</p>  </div> |
| 이 방법은 #container 영역이 끝나기 직전 빈 엘리먼트를 넣고 빈 엘리먼트에 clear:both 속성을 부여하여 부모가 자식의 높이를 인식하도록 하는 방법입니다. 하지만 의미 없는 빈 엘리먼트를 사용하기 때문에 이 역시 권장되는 방법은 아닙니다. [float을 빈 엘리먼트로 clear 하는 방법 예제](http://naradesign.net/open_content/reference/float/floatEmptyClear.html). 예제에서는 .clear 라는 빈 엘리먼트를 가시적으로 보이도록 하였지만 실무에서는 보통 .clear {clear:both; height:0; overflow:hidden;} 처리하여 .clear 라는 빈 엘리먼트가 스스로 높이를 갖지 않도록 하고 보이지 않도록 처리 합니다.  **4**  <title>**float을 빈 엘리먼트로 clear 하는 방법 예제**</title>  <style type="text/css">  \*{ margin:0; padding:0; font-size:1em; font-family:"굴림", Gulim, "돋움", Dotum, AppleGothic, Sans-serif;}  body{ background:#fff; margin:10px; font-size:.75em;}  h1{ font:26px "바탕", Batang, Serif; letter-spacing:-.1em; margin:0 0 .5em 0;}  #header{ padding:10px; margin:0 0 10px 0; border:10px solid #ccc;}  #container{ float: none (float 속성를 주지않으면 자동으로 none 으로 알아듣는다. None 으로 지정한 것은 그냥 아래로 차례대로 정렬하라는 의미 ); padding:10px; margin:0 0 10px 0; border:10px solid #ddd; background:#fff;}  #lnb{ float:left; width:25%; margin:0 0 10px 0; padding:10px; border:10px solid #000; background:#f00;}  #content{ float:right; width:55%; margin:0 0 10px 0; padding:10px; border:10px solid #000; background:#f00;}  .clear{ clear: both; padding:10px; border:10px dashed #000; /\* height:0; overflow:hidden; \*/}  #footer{ clear: both; padding:10px; border:10px solid #eee;}  </style></head>  <body>  <h1>float을 빈 엘리먼트로 clear 하는 방법 예제</h1>  <div id="header">  <p>#header {clear:both;}</p>  </div>  <div id="container">  <p>#container {float:none;}</p>  <div id="lnb">  <p><strong>#lnb {float:left;}</strong></p>  </div>  <div id="content">  <p><strong>#content {float:right;}</strong></p>  </div>  <div class="clear">  <p><strong>.clear {clear:both;}</strong></p>  </div>  </div>  <div id="footer">  <p>#footer {clear:both;}</p>  </div>  </body> |
| 가장 탁월하다고 생각하는 방법 입니다. 우선 ‘**가상 선택자**‘라는 개념을 이해하셔야 하기 때문에 약간 상세히 설명드리겠습니다. 여러분들이 익히 알고 계시는 [:link, :visited](http://trio.co.kr/webrefer/css2/selector.html#link-pseudo-classes), [:hover, :active, :focus](http://trio.co.kr/webrefer/css2/selector.html#dynamic-pseudo-classes)는 모두 가상 선택자 입니다. ‘가상 선택자’는 다시 ‘[가상 클래스](http://trio.co.kr/webrefer/css2/selector.html#q15)‘와 ‘[가상 엘리먼트](http://trio.co.kr/webrefer/css2/selector.html#q20)‘로 구분할 수 있는데요. ‘**가상 클래스**‘는 특정 엘리먼트에 대하여 아무런 class를 부여하지 않았지만 마치 역동적으로 class를 변경한것과 같은 효과를 낼 수 있는 것들로서 이미 존재하는 엘리먼트에 조합해서 사용할 수 있습니다.  [:link, :visited](http://trio.co.kr/webrefer/css2/selector.html#link-pseudo-classes), [:hover, :active, :focus](http://trio.co.kr/webrefer/css2/selector.html#dynamic-pseudo-classes), [:first-child](http://trio.co.kr/webrefer/css2/selector.html#first-child)가 가상 클래스에 해당됩니다. 한편 ‘**가상 엘리먼트**‘란, 존재하지 않는 엘리먼트를 가상으로 생성해내는 선택자로서 [:first-line](http://trio.co.kr/webrefer/css2/selector.html#first-line-pseudo), [:first-letter](http://trio.co.kr/webrefer/css2/selector.html#first-letter), [:before, :after](http://trio.co.kr/webrefer/css2/selector.html#before-and-after)가 있습니다. 심지어 :before와 :after는 HTML문서상에 존재하지 않는 콘텐츠를 출력시키기도 합니다  **( css1강 워드파일의 17번 before 참조 6번 ) 7번**  <title>**float을 가상선택자 :after로 clear 하는 방법 예제**</title>  <style type="text/css">  \*{ margin:0; padding:0; font-size:1em; font-family:"굴림", Gulim, "돋움", Dotum, AppleGothic, Sans-serif;}  body{ background:#fff; margin:10px; font-size:.75em;}  h1{ font:26px "바탕", Batang, Serif; letter-spacing:-.1em; margin:0 0 .5em 0;}  #header{ padding:10px; margin:0 0 10px 0; border:10px solid #ccc;}  #container{ padding:10px; margin:0 0 10px 0; border:10px solid #ddd; background:#fff; \*height:1%;}  #container:after{ content:"이 상자는 가상 엘리먼트 선택자 #container:after에 의하여 생성되었습니다"; display:block; clear:both; padding:10px; font-weight:bold; border:10px dashed #000; /\* height:0; visibility:hidden; \*/}  #lnb{ float: left; width:25%; margin:0 0 10px 0; padding:10px; border:10px solid #000; background:#f00;}  #content{ float: right; width:55%; margin:0 0 10px 0; padding:10px; border:10px solid #000; background:#f00;}  #footer{ clear : both; padding:10px; border:10px solid #eee;}</style>  </head>  <body>  <h1>**float을 가상선택자 :after로 clear 하는 방법 예제**</h1>  <h1>float을 가상선택자 :after로 clear 하는 방법 예제</h1>  <div id="header">  <p>header</p>  </div>  <div id="container">  <p>container </p>  <div id="lnb">  <p>lnb </p>  </div>  <div id="content">  <p>contddddent</p>  </div>  </div>  <div id="footer">  <p>footer</p>  </div>  </body>  #container:after {content:""; display:block; clear:both;}  하지만 Internet Explorer는 :before, :after 가상 엘리먼트 선택자를 지원하지 않기 때문에 다음과 같은 Hack이 필요합니다.  #container {\*zoom:1;} /\* IE5.5~7 브라우저 대응 Hack \*/ #container:after {content:" "; display:block; clear:both;} /\* 표준계열 브라우저에 대응하는 float 해제용 가상 엘리먼트의 생성 \*/  IE 5~7 브라우저는 [hasLayout](http://msdn.microsoft.com/en-us/library/bb250481(VS.85).aspx)이라는 고유한 성질을 갖게 되면 float을 해제하는 트리거로 작용하는 성질이 있고 zoom:1 속성이 hayLayout 이라는 성질을 갖도록 하기 때문에 IE 5~7 브라우저 고유의 특징을 이용한 해결방법 입니다 |
|  |

Float 예제 **8**





<style type="text/css">

body { margin:0; padding:0; background:#ccc;}

#container { width:960px; margin: 0 auto; }

#header { padding:30px; background: #bbb; }

#content { float:left; width:660px; background: #fff;}

#navigation { float: right; width: 300px; background: #eee; }

#navigation a { display: block; maring:0 0 10px 0; }

#footer { clear:both; background: #aaa; padding: 10px; }

</style>

</head>

<body>

<div id="container">

<div><h1> Header </h1></div>

<div id="content">

<P> Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions</P>

</div>

</div>

<div id="navigation">

<a href="#"> item1 </a>

<a href="#"> item2 </a>

<a href="#"> item3 </a>

<a href="#"> item4 </a>

</div>

<div id="footer"> <p>footer</p> </div>

Origin.html 바꿀 내용

Article { overflow : hidden; }

Figure { float : right; overflow: hidden; }

.clear:after{ content: "."; display: block; height: 0; clear: both; visibility: hidden;}

**9 하기**

<style type="text/css">

header, nav, article, aside, footer { font-family: Arial; padding: 20px;}

header { height: 100px; width: 840px; background-color: #afeeee; }

nav { height: 50p; width: 840px; background-color: #7fffd4; }

article { float: left; height: 500px; width: 700px; background-color: #87cefa; }

aside { float: left; height: 500px; width: 100px; background-color: #add8e6; }

footer { clear:both; width: 840px; background-color: #7fffd4; }

</style>

</head>

<body>

<header> StdCobbli's </header>

<nav>

<ul style="display:inline">

<li style="display:inline"> About </li>

<li style="display:inline"> contact</li>

</ul>

</nav>

<article>

<p> stddfl;adfadfa</p>

<p> Blog : <a href="http://www.daum.net" > jeawondf1233</a></p>

</article>

<aside>

<ul>

<li> main</li>

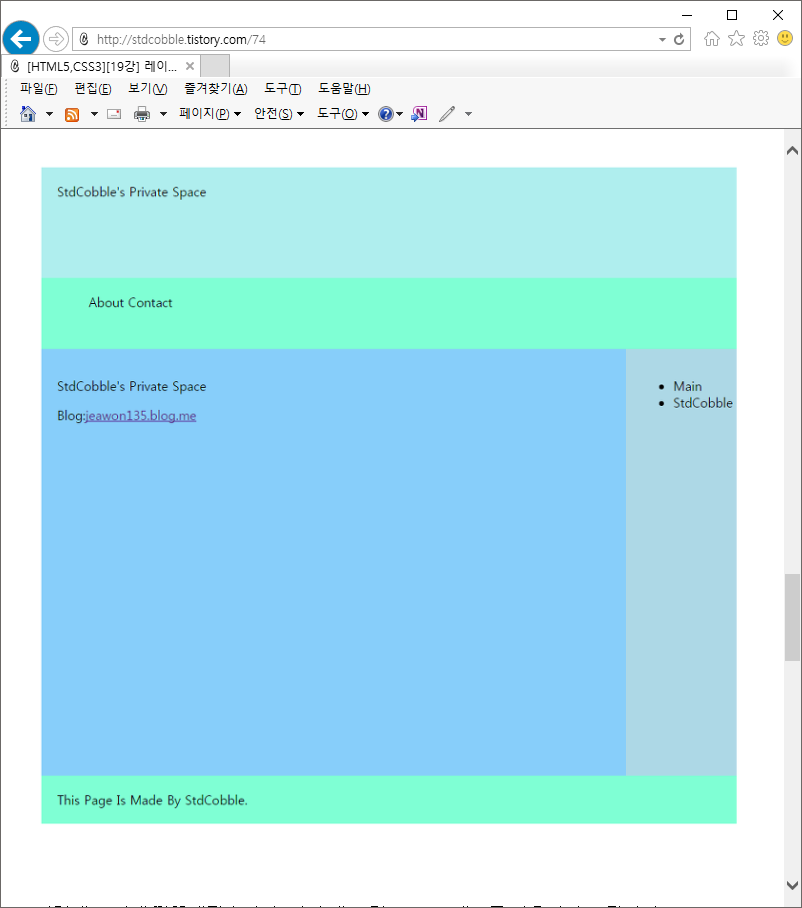
<li> another</li>

</ul>

</aside>

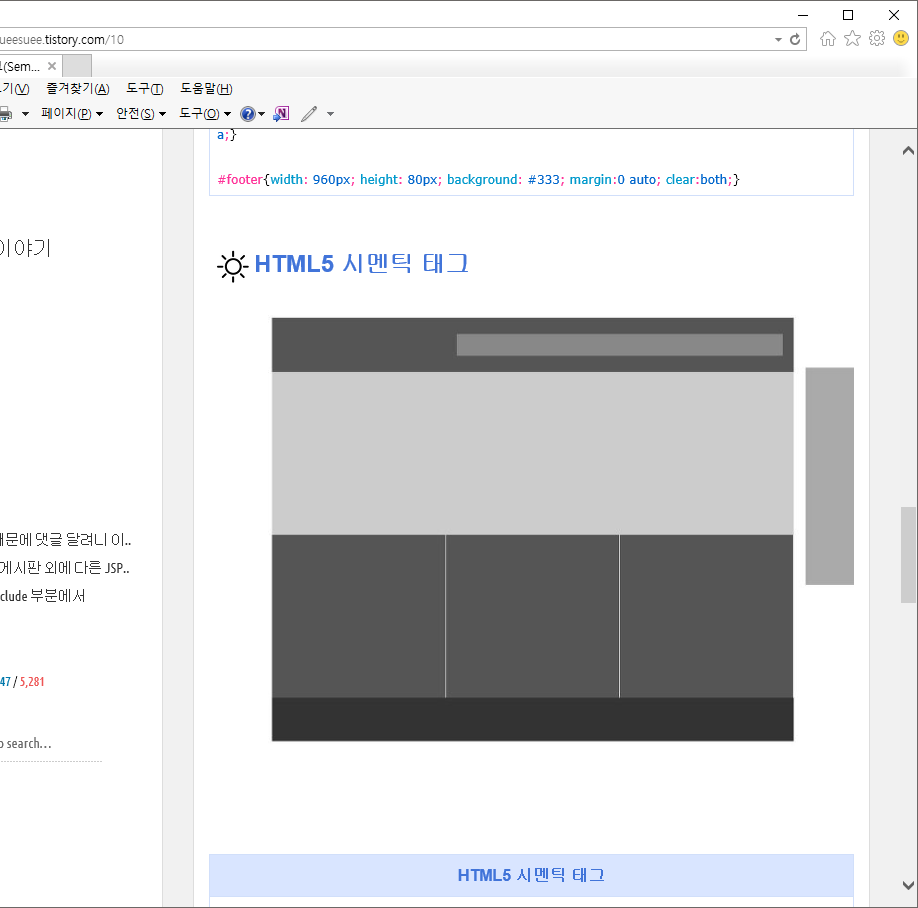
<footer> this page </footer>

</body>



잼있는 요소 <details> : <details> <summary> what is the meaning of Std</summary> <p> studiod dfk;dfkadjfa</p> </details> 라고 적으면 작은 삼각형이 나타납니다. 클릭하면 p 태그안의 내용이 나타나납니다.

또한 삼각형이 눌려진 상태로 하고 싶으면 < details open >



<style type="text/css">

header { width:960px; height:100px; background:#555; margin:0 auto;}

header nav { width:600px; height:40px; float:right; margin-top:30px; margin-right:20px; background:#888;}

figure { width:960px; height:300px; background: #ccc; margin:0 auto;}

section {width:960px; margin:0 auto;}

section article {width:319px; height: 300px; background: #555; margin-right: 1px; float: left;}

aside {position: absolute; top: 100px; right: 0px; width: 100px; height: 400px; background: #aaa;}

footer {width: 960px; height: 80px; background: #333; margin:0 auto; clear:both;}

</style>

</head>

<body>

<header>

<link rel="stylesheet" type="text/css" href="css.cs">

<nav></nav>

</header>

<figure></figure>

<section>

<article></article>

<article></article>

<article></article>

</section>

<aside></aside>

<footer></footer>

</body>

* + - 다른 속성들 설명은 http://stdcobble.tistory.com/67