Li 예제

<ul>

<li>first item</li>

<li>second item <!-- Look, the closing </li> tag is not placed here! -->

<ul>

<li>second item first subitem</li>

<li>second item second subitem <!-- Same for the second nested unordered list! -->

<ul>

<li>second item second subitem first sub-subitem</li>

<li>second item second subitem second sub-subitem</li>

<li>second item second subitem third sub-subitem</li>

</ul>

</li> <!-- Closing </li> tag for the li that contains the third unordered list -->

<li>second item third subitem</li>

</ul>

</li> <!-- Here is the closing </li> tag -->

<li>third item</li>

</ul>

Li 예제2

<ul>

<li>first item</li>

<li>second item <!-- Look, the closing </li> tag is not placed here! -->

<ol>

<li>second item first subitem</li>

<li>second item second subitem</li>

<li>second item third subitem</li>

</ol>

</li> <!-- Here is the closing </li> tag -->

<li>third item</li>

</ul>

#### DISPLAY 속성값

* **inline** : 인라인 요소처럼 표현한다.
* **block** : 블럭 요소처럼 표현한다.
* **inline-block** : 인라인의 흐름에 따르되 블록 박스처럼 마진, 테두리, 패딩, 너비, 높이 속성이 적용된다.
* **flex** : CSS3에 새로 생긴 값으로 블록-레벨의 flex 컨테이너처럼 요소를 표현한다.
* **inline-flex** : CSS3에 새로 생긴 값으로 인라인-레벨의 flex 컨테이너처럼 요소를 표현한다.
* **list-item** : 박스를 블록 처리하고 그 안의 아이템들을 인라인 박스 형태로 처리한다.
* **run-in** : 문맥에 따라 블록 또는 인라인 박스로 처리된다.
* **table, inline-table, , table-caption, table-cell, table-column, table-column-group, table-header-group, table-footer-group, table-row, table-row-group** : 이 속성값들은 테이블 스타일의 요소의 동작을 정의한다.
* **none** : 블록이 생성되지 않는다. 이 경우 화면에서 완전히 사라져 어떠한 공간도 차지하지 않는다.
* **inherit** : 부모 요소로부터 값을 상속 받는다.

display 속성값을 inline 또는 inline-block으로 했을 때 여백 조정

<https://www.cmsfactory.net/node/10411>

<!doctype html>

<html lang="ko">

<head>

<meta charset="utf-8">

<title>CSS Tips | Display Inline Margin</title>

<style>

li {

display: inline;

border: 1px solid #bcbcbc;

padding: 10px;

}

</style>

</head>

<body>

<ul>

<li>Lorem</li>

<li>Ipsum</li>

<li>Dolor</li>

</ul>

</body>

</html>

<!doctype html>

<html lang="ko">

<head>

<meta charset="utf-8">

<title>CSS Tips | Display Inline Margin</title>

<style>

ul {

font-size: 0;

}

li {

display: inline;

border: 1px solid #bcbcbc;

padding: 10px;

font-size: 16px;

}

</style>

</head>

<body>

<ul>

<li>Lorem</li>

<li>Ipsum</li>

<li>Dolor</li>

</ul>

</body>

</html>

## CSS 강좌 | Tips > ul, ol 가운데 정렬하는 방법

<https://www.cmsfactory.net/node/30023>

예제

*<!doctype html>*

*<html lang="ko">*

*<head>*

*<meta charset="utf-8">*

*<title>CSS</title>*

*<style>*

*body {*

*font-family: Consolas;*

*}*

*h1 {*

*font-size: 16px;*

*}*

*ul {*

*border: 1px solid #bcbcbc;*

*}*

*ul.a {*

*text-align: center;*

*}*

*ul.b {*

*width: 300px;*

*margin-left: auto;*

*margin-right: auto;*

*}*

*ul.c {*

*display: table;*

*margin-left: auto;*

*margin-right: auto;*

*}*

*</style>*

*</head>*

*<body>*

*<h1><code>text-align: center</code></h1>*

*<ul class="a">*

*<li>Lorem Ipsum</li>*

*<li>Dolor Amet</li>*

*</ul>*

*<h1><code>width: 300px; margin-left: auto; margin-right: auto;</code></h1>*

*<ul class="b">*

*<li>Lorem Ipsum</li>*

*<li>Dolor Amet</li>*

*</ul>*

*<h1><code>display: table; margin-left: auto; margin-right: auto;</code></h1>*

*<ul class="c">*

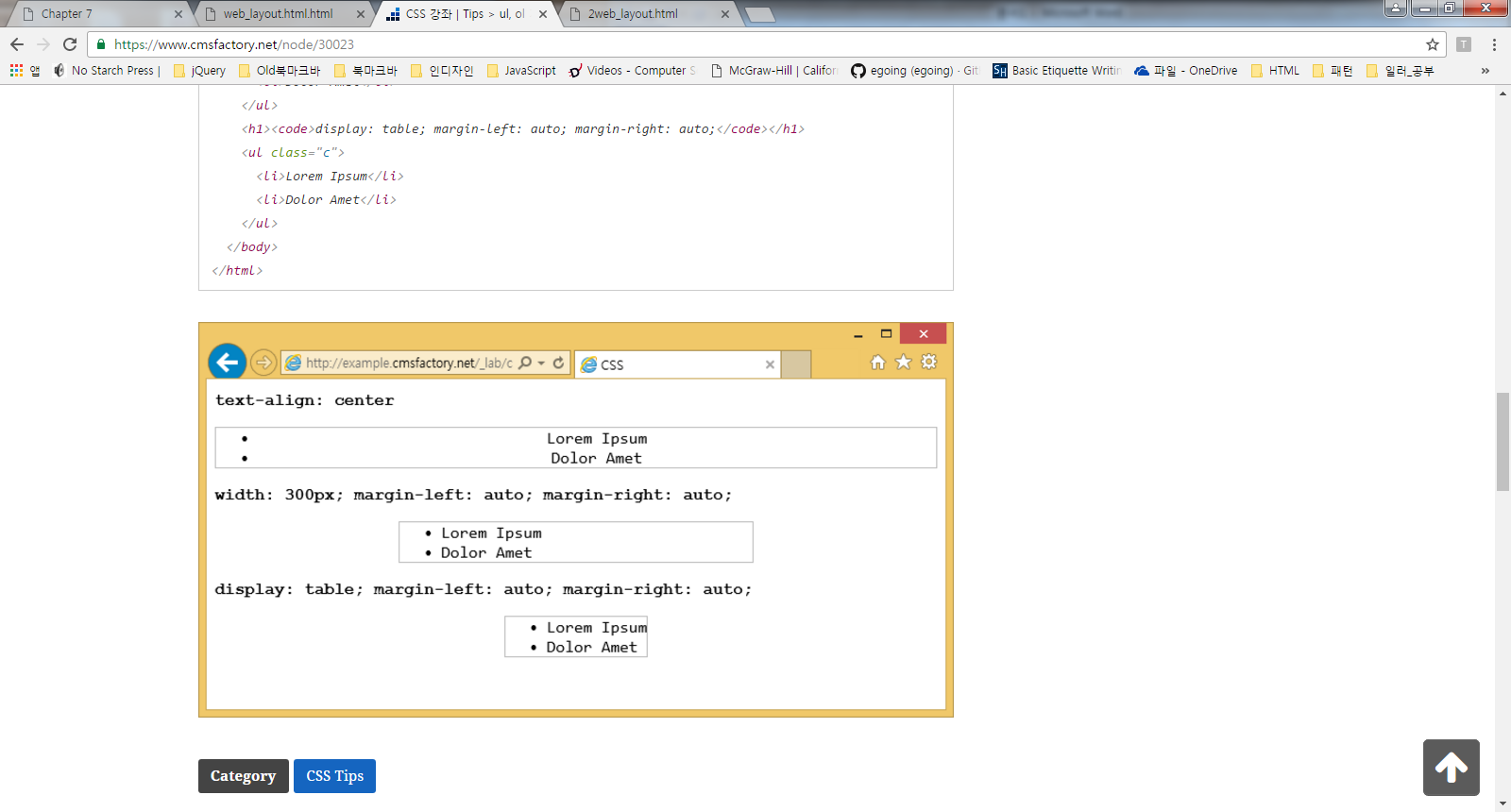
*<li>Lorem Ipsum</li>*

*<li>Dolor Amet</li>*

*</ul>*

*</body>*

*</html>*



## 세로 가운데 정렬하기

1. Padding 속성이용하기

<!doctype html>

<html lang="ko">

<head>

<meta charset="utf-8">

<title>CSS | Coding Factory</title>

<style>

.a {

border: 1px solid #444444;

width: 500px;

padding: 100px 0px;

}

.b {

border: 1px solid #444444;

background-color: #444444;

}

h1 {

text-align: center;

color: #ffffff;

}

</style>

</head>

<body>

<div class="a">

<div class="b">

<h1>Lorem Ipsum Dolor</h1>

</div>

</div>

</body>

</html>

### margin 속성 이용하기

*<!doctype html>*

*<html lang="ko">*

*<head>*

*<meta charset="utf-8">*

*<title>CSS | Coding Factory</title>*

*<style>*

*.a {*

*border: 1px solid #444444;*

*width: 500px;*

*}*

*.b {*

*border: 1px solid #444444;*

*background-color: #444444;*

*margin: 100px 0px;*

*}*

*h1 {*

*text-align: center;*

*color: #ffffff;*

*}*

*</style>*

*</head>*

*<body>*

*<div class="a">*

*<div class="b">*

*<h1>Lorem Ipsum Dolor</h1>*

*</div>*

*</div>*

*</body>*

*</html>*

Css3 Flexbox 이용하기

<http://www.w3schools.com/css/css3_flexbox.asp>

## 일련 번호 붙이기 (counter-reset, counter-increment)

*<!doctype html>*

*<html lang="ko">*

*<head>*

*<meta charset="utf-8">*

*<title>CSS</title>*

*<style>*

*body {*

*counter-reset: chapter 0;*

*}*

*h1:before {*

*counter-increment: chapter;*

*content: counter(chapter) ". ";*

*}*

*</style>*

*</head>*

*<body>*

*<h1>Lorem</h1>*

*<h2>Ipsum</h2>*

*<h2>Dolor</h2>*

*<h1>Consectetur</h1>*

*<h2>Adipiscing</h2>*

*</body>*

*</html>*

만약 번호를 3씩 증가시키고 싶으면 다음과 같이 합니다. > *counter-increment: chapter 3;*

**DISPLAY 속성**

**display** 속성은 요소를 어떻게 보여줄지를 결정한다.

주로 4가지 속성값이 쓰이는데, 태그마다 기본값이 다르다.

**none** : 보이지 않음   
**block** : 블록 박스로 만듬   
**inline** : 인라인 박스로 만듬   
**inline-block** : block 박스로 만들어지나, inline 처럼 배치가 된다.

**display** 속성 중 **block (블록 요소)** 는 요소의 가로 길이가 100%가 되어 (width값을 수동으로 지정해도 보이지 않는 margin 같은 값이 화면을 꽉 채운다) 자동으로 줄 바꿈이 되고 width, height 속성을 지정 할 수 있는데, 자동으로 **display:block** 이 적용되는 대표적인 태그는 **div** 가 있다.

**inline** 요소는 **block** 과 달리 줄 바꿈이 되지 않고, width와 height를 지정 할 수 없다. **display:inline** 이 적용되는 대표적인 태그는 **span** 이 있다.

**inline-block** 요소는 **inline** 요소와 **block** 요소의 특징을 합친 것으로 자동으로 줄 바꿈이 되지 않지만 크기를 지정 할 수 있다. **inline-block**요소와 요소 사이에는 공백이 생기게 된다. (단, ie7 이하는 지원하지 않는데, \*zoom:1; \*display:inline 속성을 이용하면 inline-block 요소 처럼 사용 할 수 있다.)

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <style>  #none{ display:none; border:1px solid blue}  #block{ display:block; height:60px; border:1px solid blue }  #inline{ display:inline; width:100px; height:60px; border:1px solid blue}  #inline-block{ display:inline-block; width:100px; height:60px; border:1px solid blue; \*zoom:1; \*display:inline }  </style>  </head>  <body>  <div id="none">display:none 입니다.</div>  <div id="block">display:block 입니다.</div>  <span id="inline">display:inline 입니다.</span>  <span id="inline-block">display:inline-block 입니다.</span>  </body>  </html> |

**Inline 속성과 block 속성의 분별**

**Inline 속성**

어떤 태그가 인라인 요소인지 일일이 종류를 나열하는 것보다는, CSS를 작성할 때 아래와 같은 특징을 발견해서 파악하는 것이 좋습니다.

* 상, 하단 외부 여백(margin-top, margin-bottom) 속성을 정의해도 적용되지 않습니다. 인라인 요소의 상, 하 여백은 margin이 아니라 line-height 속성에 의해 발생합니다.
* 너비(width)와 높이(height) 속성이 적용되지 않습니다. 인라인 요소의 너비와 높이는 태그가 품고 있는 내부 요소의 부피에 맞춰집니다.
* 인라인 속성을 가진 태그끼리 연속으로 사용되는 경우에는 최소한의 간격을 유지하기 위해서 좌, 우에 약 5px 가량의 외부 여백(margin)이 자동으로 발생합니다.

이와같은 특징은, 인라인 요소를 가진 태그들이 서로 나열되었을 때, 최소한 같은 줄에 같은 높이로 가지런히 위치하면서, 약간의 좌우 간격으로 서로를 분별할 수 있도록 하기 위한 웹 브라우저의 기본 설정 때문입니다.

**block 속성**

Block 속성은 Inline과 달리 **한 줄에 나열되지 않고 그 자체로 한 줄을 완전히 차지**합니다. 대표적인 블록 요소로 <p> 태그가 있습니다.

블록 속성을 가지고 있는 태그는 기본적으로 너비 100%(width: 100%) 속성을 가지고 있습니다. 화면의 가로 폭을 100%로 완전히 차지하기 때문에 다음 요소가 양 옆으로 붙을 공간이 없기 때문에 자연히 줄 넘김이 되는 것이죠.

또한, 인라인 요소와 다르게 margin, width, height 속성을 정의하면 모두 적용됩니다. 모양새를 쉽게 제어할 수 있는 속성 때문에 대부분 블록 속성을 가진 태그를 화면 구성이나 레이아웃에 사용합니다.

특정 태그가 블록 요소인지 아닌지 가장 쉽게 파악하는 방법은, CSS로 배경색을 입혀보는 것입니다. 만일 아래 예문과 같이 배경색이 화면 폭을 모두 차지하는 경우, 그 태그는 블록 요소 태그임을 알 수 있습니다.

**Inline과 Block의 합성 속성**

Inline과 Block, 두 속성의 중간 단계도 있습니다. inline-block이라는 이 속성은, 인라인과 같이 한 줄에 표현하면서도 margin, width, height 속성을 정의하면 표현해 줍니다.

inline-block 속성을 기본값으로 가지고 있는 태그는 없습니다. 이 속성은 작성자의 CSS를 통해 inline-block임을 선언해 주어야만 합니다. 아래는 이전에 알아 본 inline 속성의 특징입니다. inline-block 속성에서는 어떻게 바뀌었는지 살펴보세요.

* 상, 하단 외부 여백(margin-top, margin-bottom) 속성을 정의할 수 있습니다. inline-block요소의 상, 하 여백을 margin과 line-height 두가지 속성 모두를 통해 제어할 수 있습니다.
* 너비(width)와 높이(height) 속성을 정의할 수 있습니다. 기본적으로는 태그가 품고 있는 내부 요소의 부피에 맞춰집니다만, 너비와 높이를 지정하면 이 값 또한 표현됩니다.
* inline-block 속성을 가진 태그끼리 연속으로 사용되는 경우에는 최소한의 간격을 유지하기 위해서 좌, 우에 약 5px 가량의 외부 여백(margin)이 자동으로 발생합니다. 하지만, margin-left나 margin-right를 사용하면 여기에 추가로 여백을 줄 수 있게 됩니다.

## Inline과 Block 속성 태그 표기시 주의할 점

Block은 Inline을 포괄하는 더 큰 개념입니다. 따라서 HTML을 마크업 할 때는 Inline 속성의 태그 안에 Block 속성 태그를 넣으면 문법 오류가 됩니다.

|  |
| --- |
| BAD  <a href="#"><p>문장입니다.</p></a>  GOOD  <p><a href="#">문장입니다.</a></p> |

부득이한 경우에는 Inline 속성의 태그를 CSS에서 a { display: block }과 같이 정의하여 임의로 block 속성으로 변경할 수 있지만, CSS는 표준 문법 규칙을 지키면서 여러분이 원하는 모든 효과를 낼 수 있습니다. 각종 태그와 CSS 속성을 정의하다보면 차차 익힐 수 있으므로, **평소에는 항상 Inline이 Block보다 작은 개념임을 잊지마세요.**

**CSS display 속성**

<!doctype html>

<html lang="ko">

<head>

<meta charset="utf-8">

<title>CSS Tips | Display Inline Margin</title>

<style>

span {

width: 100px;

height: 60px;

margin: 10px;

}

.my-block { display: block; background-color: salmon; }

.my-inline { display: inline; background-color: lightgreen; }

.my-inline-block { display: inline-block; background-color: skyblue; }

</style>

</head>

<body>

<span class="my-block">우리나라의 말이 중국과 </span>

<span class="my-block">우리나라의 말이 </span>

<span class="my-block">우리나라의 말이 </span>

<hr><hr><hr>

<span class="my-inline">우리나라의 말이데</span>

<span class="my-inline">우리나라의 말이 </span>

<span class="my-inline">우리나라의 말이 </span>

<hr><hr><hr>

<span class="my-inline-block">우리나라의 말이 중국과 </span>

<span class="my-inline-block">우리나라의 말이 </span>

<span class="my-inline-block">우리나라의 말이</span>

</body>

</html>