

웹 표준 기초

발생배경

HTML5가 인터넷 업계에서 알려지게 된 계기는 바로 2009년 구글의 웹 개발자 콘퍼런스인 ‘구글 I/O’에서 자사의 서비스가 아닌 **HTML5**를 데모로 시연하면서 차세대 웹 기술로 지원하겠다는 천명을 하면서부터이다. 특히, 스티브 잡스가 애플 아이폰에 플래시 탑재를 거부하면서 대응 기술로 **HTML5**를 홍보하기 시작하였다.

HTML5는 2004년 당시 시장 점유율 **5%** 미만이었던 마이너 웹 브라우저들이 웹 개발자들과의 토론을 통해 만들어 온 개방형 웹 표준으로서 웹 개발자의 생산성과 가치를 높이기 위한 다양한 기술들을 담았다. 이러한 점을 간략하게 살펴 보고자 한다.

발생배경



발생배경

- 2000년대 중반부터 혁신의 단초를 제공하게 된 것은 바로 웹 표준과 웹2.0이다.
- 우선 구글 같은 검색 엔진과 검색 광고의 성장과 특히, 블로그와 같은 사용자 생산 콘텐츠를 잘 검색하기 위하여 **HTML**과 **CSS** 레이아웃을 통한 웹 표준 기법이 각광 받기 시작했다. 이른바 구조(Structure)와 표현(Presentation) 그리고 동작(Behavior)를 분리하여 검색 크롤러(기계)가 콘텐츠를 읽고 쓸 수 있도록 하는 것은 매우 중요한 시작점이 되었다.

발생배경

- 웹 2.0의 주 개념인 ‘플랫폼으로서 웹’은 웹 그 자체를 소프트웨어로 보는 웹 애플리케이션 시대를 열었다. 대표적인 기술이 바로 Ajax(Asynchronous JavaScript and XML)로서 지메일과 구글맵이 그 시초를 이루었다. 기존의 문서 형식의 정보의 제공이라는 틀에서 벗어나 데스크톱 소프트웨어와 같은 사용자 경험을 제공하는 것이다.
- 또한, 오픈**API**라는 데이터 기반 서비스는 전문 개발자뿐만 아니라 전문사용자까지 웹 플랫폼에 끌어들었다. 오픈 **API**를 이용하면 자신의 블로그나 홈페이지에 네이버나 다음의 검색 결과나 구글 맵의 위성 지도, 이베이의 중고 상품 목록 같은 것을 쉽게 추가할 수 있다.

발생배경

- 웹 2.0의 주 개념인 ‘플랫폼으로서 웹’은 웹 그 자체를 소프트웨어로 보는 웹 애플리케이션 시대를 열었다. 대표적인 기술이 바로 Ajax(Asynchronous JavaScript and XML)로서 지메일과 구글맵이 그 시초를 이루었다. 기존의 문서 형식의 정보의 제공이라는 틀에서 벗어나 데스크톱 소프트웨어와 같은 사용자 경험을 제공하는 것이다.
- 또한, 오픈**API**라는 데이터 기반 서비스는 전문 개발자뿐만 아니라 전문사용자까지 웹 플랫폼에 끌어들었다. 오픈 **API**를 이용하면 자신의 블로그나 홈페이지에 네이버나 다음의 검색 결과나 구글 맵의 위성 지도, 이베이의 중고 상품 목록 같은 것을 쉽게 추가할 수 있다.

발생배경

- XML과 (X)HTML, CSS, 자바 스크립트 같은 웹 표준 기술들을 리치 인터넷 애플리케이션(Rich Internet Application)을 만드는 데 사용하기 시작
- 예) 모질라의 파이어폭스 확장 기능, 야후! 위젯, 마이크로소프트의 실버라이트(Silverlight), 어도비의 플렉스(Flex) 및 AIR 등

* 실버라이트 기반의 리치 인터넷 애플리케이션 사이트

<http://photosynth.net/create.aspx>

발생배경

- 웹의 기술적 혁신은 웹 브라우저 업계에도 역시 시작 되었다. 오픈 소스 프로젝트인 모질라(Mozilla) 커뮤니티에서 개발한 파이어폭스와 애플의 사파리, 오페라 그리고 구글 크롬에 이르기 까지 **2004**년부터 다양한 웹 브라우저들이 시장에 쏟아져 나오기 시작했다.
- **2004**년 비IE 브라우저 세계 점유율이 **5%** 안팎이던 것이 **2010**년 현재 거의 **40%**에 육박하고 있으며 유럽의 경우는 이미 **50%**를 넘었다. 웹 기술의 변화에는 이러한 마이너 웹 브라우저 업체의 혁신과 사용자들의 선택에 힘입은 바 컸으며 마이크로소프트가 **2007**년 IE 개발팀을 다시 만들 정도였다.

발생배경

- 웹 표준화 기구인 W3C는 이러한 변화를 수용할 준비를 하고 있지 못했다. 2004년 W3C의 한 워크샵에서 생긴 의견 차이 때문에 모질라,실전 HTML5 가이드10애플, 오페라 등은 W3C 밖에서 새로운 버전의 HTML 표준을 준비하기 시작했다. W3C의 다른 표준화 기구 보다는 상대적으로 개방되어 있었지만,다양한 웹 브라우저 환경에서의 웹 개발자의 고충과 웹 애플리케이션이라는 현실적인 변화를 받아들이지 못했다.이들은 2006년 6월 웹 하이퍼텍스트 워킹그룹(WHATWG)이라는 공개그룹을 형성하여 자신들이 만드는 새로운 표준안에 누구나 참여할 수 있도록 개방 하였다. W3C의 회원사 중심 표준안이 아닌 웹 개발자가 진정 원하는 표준을 만들기 위해서였다. 누구나 표준안 논의에 참여할 수 있었으며 이들은 오랜 공개 토론을 거쳐 Web Form 2.0 과 Web Applications 1.0이라는 표준안을 만들어 냈다.

발생배경

- W3C HTML5 W/G에서는 기존 표준 문서 외에도 웹 개발자의 이해를 돕기 위한 다양한 문서를 함께 만들고 있다. 그 중 **HTML5** 디자인 원칙이라는 문서에는 웹 표준을 만드는 데 있어, 의사 결정의 기본 원칙이 되는 사항을 제시하고 있다.

발생배경

- 첫째, 기존의 **HTML** 문법이랑 사용법을 최대한 지원하고 단계적 기능축소(**Graceful degradation**)이 가능하도록 한다. ****, **<i>** 같은 기존의 비표준 태그의 사용도 용법을 정해 가능하게 했으며 **<embed>** 같은 이미 사용하던 표준도 재사용하도록 하여 웹 개발자들이 너무 문법에 얽매이지 않도록 하는 ‘호환성(**Compatibility**)’을 제공한다.

발생배경

- 둘째, 실제 웹 개발자들이 겪고 있는 가장 중요한 문제를 순위에 따라 나누되 문제점을 분리해서 독립적으로 해결 하는 유용성(Utility)의 원칙이다. 예를 들어, 웹폼(Web Form)에 **email**, **number**, **date** 같은 새로운 속성을 추가함으로써 사용자 입력 값의 유효성 확인에 드는 (항상 하는) 삽질을 줄일 수 있도록 하였다. **<input>** 태그에 **datetime** 속성을 넣어주면 웹 브라우저가 자동으로 달력을 표시해 준다. 또한 IE에서만 사용 가능 했던 **contenteditable** 속성이 표준화 되어 모든 **HTML** 요소를 사용자가 직접 편집할 수 있게 함으로서 워지웍 에디터의 호환성 문제도 사라질 것이다. 특히, 이미 웹 콘텐츠의 일부가 되어 버린 비디오와 오디오 콘텐츠 재생을 웹 브라우저에서 내부적으로 구현하여 보편적 접근이 가능하고 캔버스(Canvas)와 벡터 그래픽(SVG)를 통해 2차원 도표와 같은 콘텐츠도 마크업으로 표현 할 수 있도록 멀티미디어의 보편적 접근성을 높였다.

발생배경

- 셋째, 상호 호환성(Interoperability)으로 웹 브라우저가 상호 호환을 위해 최대한 자세하게 기술하되 오류 처리 방법을 명시하도록 하였다.
- HTML5의 기본 표준 문서 첫 부분은 웹 브라우저 간 HTML 문법 오류에 대한 자세한 사례와 이에 대한 브라우저의 처리 방법을 명시해 두었다. 따라서 웹 브라우저간 이러한 문법적 오류로 인해 웹 개발자들이 실질적인 어려움을 겪는 문제를 해결하였다.

문법 기초

<!DOCTYPE html> 중요

모든 태그는 열었으면 닫음

<p>hello</p>

모든 태그는 소문자로 (내용 말고 태그들..)

html, css, js를 반드시 분리

html은 구조와 내용만, css는 속성, 표현만, js는 기능만 기술

영역 겹치지 않도록 주의

<p>hello<a>link</p> ← 절대 금지

<meta charset="utf-8" /> 한글이 깨질 경우 utf-8을 euc-kr로 변경

색상 코드는 반드시 6자리로

#fff -> #ffffff

크로스 브라우징은 조건부 태그로 (단, 조건부 태그는 웹표준이 아님)

<!--[if lt IE 9]>

<link rel="stylesheet" href="main.css" type="text/css" media="screen" />

<![endif]-->

문법 기초

빈칸은 한칸만 인식함

특수문자

< <

> >

& &

 빈칸

<p>태그는 문단을 나타내는 태그임. 즉, 문단이 바뀔때로 인해 줄바꿈이 일어난다면
태그를 쓰는 것이 아니라 <p>태그로 내용을 분리시키는 것이 옳음

그와 비슷한 경우로 빈칸을 일부러 발생시키고 싶을 때는 위에 있는 빈칸 특수문자를 이용하는 것이 아니라, css스타일링을 통해 간격 수치를 변경해야 함

기초 / 미리보기

```
<!DOCTYPE html>
<html lang="ko">
  <head>
    <meta charset="utf-8" />
    <title>html5sample</title>

    <!--[if lt IE 9]>
      <script
src="http://html5shiv.googlecode.com/svn/trunk/html5.js"> </script>
      <![endif]-->

    <!--[if lt IE 9]>
      <script src="http://ie7-
js.googlecode.com/svn/version/2.1(beta4)/IE9.js"> </script>
      <![endif]-->

    <link rel="stylesheet" href="main.css" type="text/css" media="screen" />
    <link rel="stylesheet" href="print.css" type="text/css" media="print" />
  </head>
  <body>

  </body>
</html>
```


기초 / 미리보기

```
@charset "utf-8";

*{
    position: relative;
    margin: 0;
    padding: 0;
}
header, footer, section, article, aside, nav, hgroup, details, menu, figure, figcaption {
    display: block;
}
a{
    color: #6600cc;
    text-decoration: none;
}
a:hover, a:active, a:focus{
    color: #00cc33;
}
body{
    width: 1000px;
    font-family: "나눔고딕", "맑은 고딕", "돋움", Dotum, Verdana, Tahoma;
    font-size: 12px;
    line-height: 1.6em;
    margin: 0 auto;
}
p{
    text-align: justify;
}
```

기초 / 미리보기



main.js



index.html

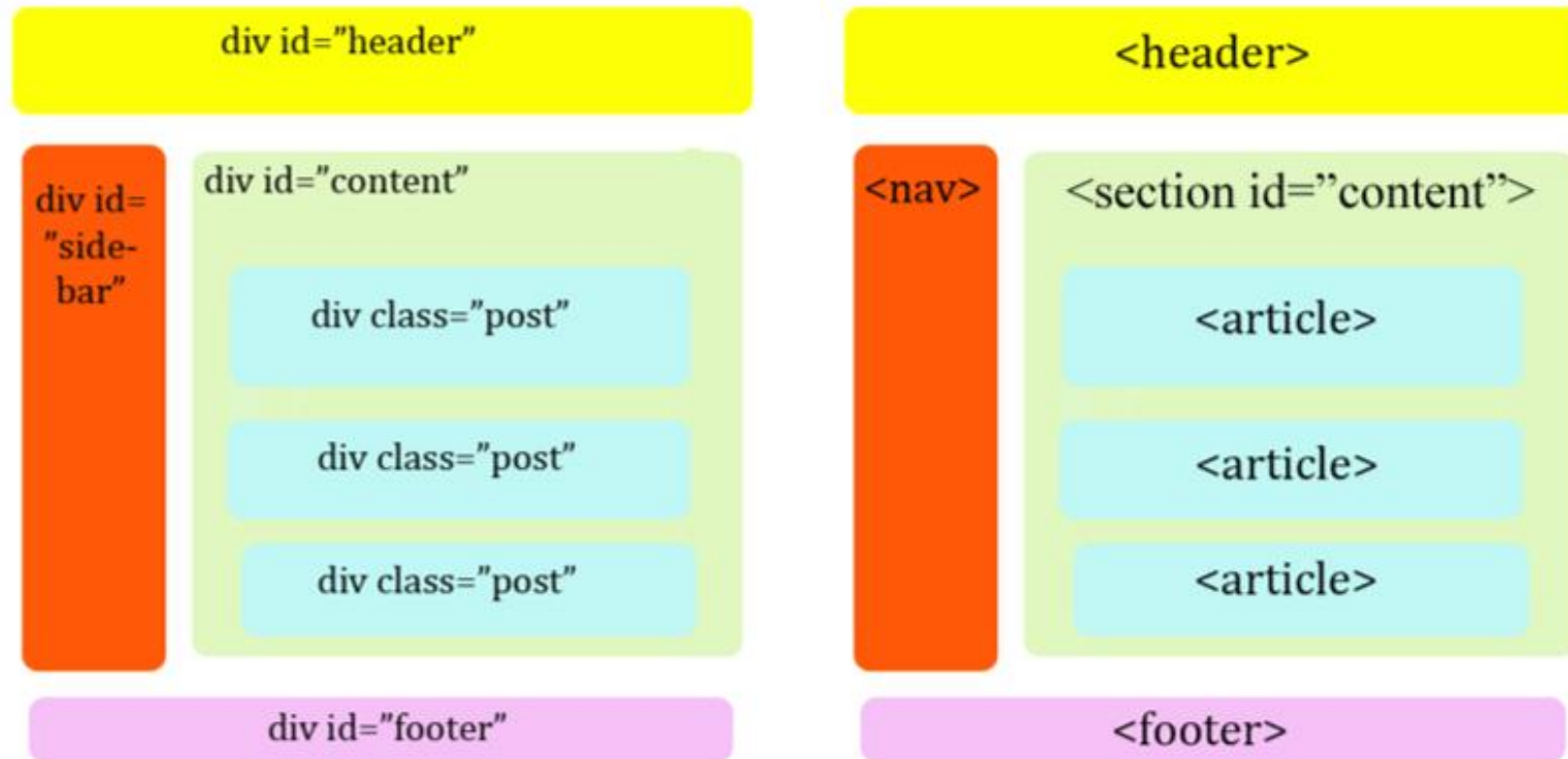


main.css

3,35KB

내 컴퓨터

기초 / 미리보기



HTML5에서는 기존의 HTML4 보다 확장된 태그들을 지원한다.

기초 / 미리보기

```
<body>
  <div id="header"> </div>
  <div id="side-bar"> </div>
  <div id="content">
    <div class="post"> </div>
    <div class="post"> </div>
    <div class="post"> </div>
  </div>
  <div id="footer"> </div>
</body>
```

```
<body>
  <header> </header>
  <nav> </nav>
  <section id="content">
    <article> </article>
    <article> </article>
    <article> </article>
  </section>
  <footer> </footer>
</body>
```

새로운 구조적 태그와 속성

기존 div 증후군을 탈피

- ul이라는 요소로 묶어 놓고도 그 외부에 div요소로 다시 묶음

*참고 : 블록 요소 (자체 줄바꿈이 발생), 인라인 요소는 그 반대

<header>

<footer>

<nav>

<section>

<article>

<aside>

기초 / 미리보기

회사들의 이해 관계

- 주도권 싸움

HTML5와 CSS3는 아직 진행 중인 명세임

- 기존에 사용한 명세가 바뀌거나 폐기될 수 있음

시맨틱 웹 ?

컴퓨터가 정보자원의 뜻을 이해하고, 논리적 추론까지 할 수 있는 차세대 지능형 웹.

시맨틱 웹이 실현되면 컴퓨터가 자동으로 정보를 처리할 수 있어 정보 시스템의 생산성과 효율성이 극대화된다. 컴퓨터 혼자 전자상거래를 할 수 있고, 기업의 시스템 통합(SI), 지능형 로봇 시스템, 의료 정보화 등 다양한 분야에 응용할 수 있다.

CSS

주요 CSS 사용 방법

<http://hooney.net/docs/css/documentsindex>

- CSS의 요소의 의미를 정확히 알아야 제대로 표현 가능
- **position:relative** 속성을 명시 해야 제대로 작동
(단, 명시할경우 ie6에서 오동작이 나는 경우가 있음)

Index.html

```
<!DOCTYPE html>
<html lang="ko">
  <head>
    <meta charset="utf-8" />
    <title>html5sample</title>

    <!--[if lt IE 9]>
    <script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
    <![endif]-->

    <!--[if lt IE 9]>
    <script src="http://ie7-js.googlecode.com/svn/version/2.1(beta4)/IE9.js"></script>
    <![endif]-->

    <link rel="stylesheet" href="main.css" type="text/css" media="screen" />
    <link rel="stylesheet" href="print.css" type="text/css" media="print" />
  </head>
  <body>
    <section class="a1">
      <article>
        <p>hello html</p>
      </article>
    </section>
  </body>
</html>
```


main.css

```
.a1{  
  width: 900px;  
  padding: 10px 5px 10px 5px;  
  background: #ffff00 url(img/mainbg.jpg) left top no-repeat;  
}  
  
.a1>article{  
  border: 2px solid #eeeeee;  
}  
  
#id{  
  color: rgb(255,0,0);  
}
```

선택자

[illegible]

선택자

CR 5 SF 5 FF 3.6 OP 10.5 IE 8 IE 7 IE 6

CSS2 *	yes	yes	yes	yes	yes	yes	yes	모든 요소를 선택
E[attr]	yes	yes	yes	yes	yes	yes	no	'attr' 속성이 포함된 요소 E를 선택
E[attr=val]	yes	yes	yes	yes	yes	yes	no	'attr' 속성의 값이 정확하게 'val'과 일치하는 요소 E를 선택
E[attr~val]	yes	yes	yes	yes	yes	yes	no	'attr' 속성의 값에 'val'이 포함되는 요소를 선택(공백으로 분리된 값이 일치해야 한다)
E[attr =val]	yes	yes	yes	yes	yes	yes	no	'attr' 속성의 값이 'val' 또는 'val-' 으로 시작되는 요소 E를 선택
E:first-child	yes	yes	yes	yes	yes	yes	no	첫 번째 등장하는 요소가 E 라면 선택(E 아닌 요소의 순서가 계산에 포함됨)
E:lang(en)	yes	yes	yes	yes	yes	no	no	HTML lang 속성의 값이 'en'으로 지정된 요소를 선택
E:before	yes	yes	yes	yes	yes	no	no	E 요소의 시작 지점에 생성된 요소를 선택
E:after	yes	yes	yes	yes	yes	no	no	E 요소의 끝 지점에 생성된 요소를 선택
E>F	yes	yes	yes	yes	yes	yes	no	E 요소의 자식인 F 요소를 선택
E+F	yes	yes	yes	yes	yes	yes	no	E 요소를 뒤따르는 F 요소를 선택(E와 F 사이에 다른 요소가 존재하면 선택 안함)

선택자

	Selectors	CR 5	SF 5	FF 3.6	OP 10.5	IE 8	IE 7	IE 6	Meaning
CSS3	E[attr^=val]	yes	yes	yes	yes	yes	yes	no	'attr' 속성의 값이 'val'으로 시작하는 요소를 선택(공백으로 분리된 값이 일치해야 한다)
	E[attr\$=val]	yes	yes	yes	yes	yes	yes	no	'attr' 속성의 값이 'val'으로 끝나는 요소를 선택(공백으로 분리된 값이 일치해야 한다)
	E[attr*=val]	yes	yes	yes	yes	yes	yes	no	'attr' 속성의 값에 'val'이 포함되는 요소를 선택(공백으로 분리된 값과 정확하게 일치하지 않아도 선택)
	E:root	yes	yes	yes	yes	no	no	no	문서의 최상위 요소(html)를 선택
	E:nth-child(n)	yes	yes	yes	yes	no	no	no	앞으로부터 지정된 순서와 일치하는 요소가 E 라면 선택(E 아닌 요소의 순서가 계산에 포함됨)
	E:nth-last-child(n)	yes	yes	yes	yes	no	no	no	뒤로부터 지정된 순서와 일치하는 요소가 E 라면 선택(E 아닌 요소의 순서가 계산에 포함됨)
	E:nth-of-type(n)	yes	yes	yes	yes	no	no	no	E 요소 중 앞으로부터 순서가 일치하는 E 요소를 선택(E 요소의 순서만 계산에 포함됨)

선택자

Selectors	CR 5	SF 5	FF 3.6	OP 10.5	IE 8	IE 7	IE 6	Meaning
E:nth-last-of-type(n)	yes	yes	yes	yes	no	no	no	E 요소 중 끝으로부터 순서가 일치하는 E 요소를 선택(E 요소의 순서만 계산에 포함됨)
E:last-child	yes	yes	yes	yes	no	no	no	마지막에 등장하는 요소가 E 라면 선택(E 아닌 요소의 순서가 계산에 포함됨)
E:first-of-type	yes	yes	yes	yes	no	no	no	E 요소 중 첫 번째 E를 선택(E 요소의 순서만 계산에 포함됨)
E:last-of-type	yes	yes	yes	yes	no	no	no	E 요소 중 마지막 E를 선택(E 요소의 순서만 계산에 포함됨)
E:only-child	yes	yes	yes	yes	no	no	no	E 요소가 유일한 자식이면 선택(E 아닌 요소가 하나라도 포함되면 선택 안함)
E:only-of-type	yes	yes	yes	yes	no	no	no	E 요소가 유일한 타입이면 선택(E 아닌 요소가 포함되어도 E 타입이 유일하면 선택)
E:empty	yes	yes	yes	yes	no	no	no	텍스트 및 공백을 포함하여 자식 요소가 없는 E를 선택
E:target	yes	yes	yes	yes	no	no	no	E의 URI가 요청되면 선택(따라서 E는 ID가 지정되어 있어야 한다)

선택자

Selectors	CR 5	SF 5	FF 3.6	OP 10.5	IE 8	IE 7	IE 6	Meaning
E:enabled	yes	yes	yes	yes	no	no	no	사용 가능한 폼 컨트롤 (input, textarea, select, button) E 를 선택
E:disabled	yes	yes	yes	yes	no	no	no	사용 불가능한 폼 컨트롤 (input, textarea, select, button) E 를 선택
E:checked	yes	yes	yes	yes	no	no	no	선택된 폼 컨트롤 (input checked="checked")을 선택
E:not(s)	yes	yes	yes	yes	no	no	no	S가 아닌 E 요소를 선택
E~F	yes	yes	yes	yes	yes	yes	no	E 요소가 앞에 존재하면 F를 선택 (E가 F보다 먼저 등장하지 않으면 선택 안함)

css3 선택자 사용예제

```
<body>
  <table >
    <tr>
      <th>품목</th><th>단가</th><th>수량 </th><th>금액 </th>
    </tr>
    <tr>
      <td> 머그컵 </td><td>$10.00</td><td>5</td><td>$50.00</td>
    </tr>
    <tr>
      <td> 티셔츠 </td><td>$20.00</td><td>5</td><td>$100.00</td>
    </tr>
    <tr>
      <td>호치키스</td><td>$9.00</td><td>4</td><td>$36.00</td>
    </tr>
    <tr>
      <td colspan="3">소계</td><td>$186.00</td>
    </tr>
    <tr>
      <td colspan="3">운송비 </td><td>$12.00</td>
    </tr>
    <tr>
      <td colspan="3">총계 </td><td>$198.00</td>
    </tr>
  </table>
</body>
```

css3 선택자 사용예제

기본 테두리 없애기

```
table{  
    width: 600px;  
    border-collapse: collapse;  
}  
  
th, td{  
    border: none;  
}
```

헤더 부분 변경

```
th{  
    background-color: #000000;  
    color: #ffffff;  
}
```


css3 선택자 사용예제

:nth-of-type으로 스트라이프 만들기

```
tr:nth-of-type(even){  
    background-color: #f3f3f3;  
}  
tr:nth-of-type(odd) {  
    background-color:#dddddd;  
}
```

css3 선택자 사용예제

:nth-child 로 열에 따라 정렬하기

```
td:nth-child(n+2){
    text-align: right;
}
```

**** 참고 ****

- 전체 행 선택

table tr:nth-child(n)

- 첫째 행만 빼고 모두 선택

table tr:nth-child(n+2)

- 짝수행 모두 선택

table tr:nth-child(2n)

- 네번째 행에서 시작하여 한줄씩 건너뛰기

table tr:nth-child(2n+4)

css3 선택자 사용예제

:last-child 로 마지막 행을 굵게 하기

```
tr:last-child{  
    font-weight: bolder;  
}
```

마지막 열을 굵게 하기

```
td:last-child{  
    font-weight: bolder;  
}
```

last-child를 두번 써서 구석 선택 가능

```
tr:last-child td:last-child{  
    font-size:24px;  
}
```

**** 참고 ****

일반적으로 마지막 문단의 마진 값을 없애는 등의 용도로 사용

```
#sidebar p:last-child{margin-bottom:0;}
```

(id:sidebar 밑의 p태그 중 제일 마지막 태그의 margin-bottom값을 0으로)

css3 선택자 사용예제

거꾸로 찾기 :nth-last-child

```
tr:nth-last-child(2){  
    color: green;  
}
```

마지막 세줄의 첫 열도 오른쪽 정렬로

```
tr:nth-last-child(-n+3) td{  
    text-align: right;  
}
```

TEXT

<basefont>, <big>, <center>, , <s>, <strike>, <tt>, <u>
사용 금지를 권장

, <i>, <small>, <hr>, , <menu>,
HTML5에서 의미 변경 (모양은 동일)

TEXT

기존 : 굵은 글꼴을 표현할 때 사용합니다.

변경 : 특별한 중요성을 갖지는 않으나, 굵은 글꼴로 표현하고자 할 경우에 사용합니다.

예) 문서 초록의 키워드, 제품 소개 내 상품명

코드 예제

<p>차가운 창 밖으로 하나둘 하얗게 새어가는 별들에 입김을 불어본다

해가 뜨고 석양이 잠겨도 시간이 흘러도 난 영원히 이대로 일 것 같아</p>

<p>이대로 일 것 같아</p>

과 동일하게 표현되긴 하지만은, 중요한 의미를 나타내고자 할 때는 이 더 적합합니다.

명세에 따르면, 굵게 표현하길 원하나 다른 엘리먼트로 표현하기 어려운 경우에 사용할 것을 권장하고 있습니다.

가령, 제목을 나타내고자 할때는 <h1>~<h6>를, 강조를 하고자 할 때는 을, 중요한 의미를 표현하고자 할 때는 을, 하이라이트를 주고싶을 때는 <mark>로 마크업하는 것이 바람직 하며, 이 외의 경우에서 굵은 글꼴로 표현하고자 할때 를 사용하는 것이 바람직 합니다.

TEXT

<hr />

기존 : 수평선을 표현합니다.

변경 : 주제의 구분을 위해 사용합니다.

예) 이야기 내에서 장면의 변화, 한 섹션에서 다른 주제를 논하고 싶을 때.

코드 예제

<p>Dudley was ninety-two, in his second life, and fast approaching time for another rejuvenation. Despite his body having the physical age of a standard fifty-year-old, the prospect of a long degrading campaign within academia was one he regarded with dread. For a supposedly advanced civilization, the Intersolar Commonwealth could be appallingly backward at times, not to mention cruel.</p>

<p><i>Maybe it won't be that bad</i>, he told himself. The lie was comforting enough to get him through the rest of the night's shift.</p>

<hr />

<p>The Carlton AllLander drove Dudley home just after dawn. Like the astronomer, the vehicle was old and worn, but perfectly capable of doing its job. It had a cheap diesel engine, common enough on a semi-frontier world like Gralmond, although its drive array was a thoroughly modern photoneural processor. With its high suspension and deep-tread tyres it could plough along the dirt track to the observatory in all weather and seasons, including the metre-deep snow of Gralmond's winters.</p>

TEXT

<i>

기존 : 기울어진 글꼴을 표현합니다.

변경 : 어조를 바꿔 말할 때, 기본 산문체에서 다른 방법으로 표현하고자 할 때 사용합니다.

예) 분류학 명칭, 기술용어, 생각 등

코드 예제

<p>The Japanese Red Pine (<i>Pinus densiflora</i>) has a home range that includes Japan, Korea, northeastern China (Heilongjiang, Jilin, Liaoning, Shandong) and the extreme southeast of Russia (southern Primorsky Krai). This pine has become a popular ornamental and has several cultivars, but in the winter it becomes yellowish. The height of this tree is 20-35 m. The Japanese red pine prefers full sun on well-drained, slightly acidic soil.</p>

과 동일하게 표현되지만, <i>는 강조의 기능이 없다는 것에서 구분될 수 있습니다.

가령, 위의 예제에서 이탤릭체로 표현된 부분은 강조의 의미를 표현하고자 하는 것이 아니라, 단순히 학명을 나타내고자 하는 것이므로 보다는 <i>로 마크업되는 것이 맞는 예라 할 수 있겠지요.

TEXT

<menu>

기준 : 과 동일한 구조를 갖는 단일 컬럼 메뉴 목록을 표현합니다. 이 엘리먼트보다는 엘리먼트의 사용을 권장하고 있습니다.

변경 : 문서 내의 메뉴를 의미합니다. **type** 애트리뷰트를 통해서 팝업메뉴나 툴바 형태로 표현할 수 있습니다.

코드 예제

```
<menu type="toolbar">
  <li>
    <menu label="File">
      <button type="button" onclick="fnew()">New...</button>
      <button type="button" onclick="fopen()">Open...</button>
      <button type="button" onclick="fsave()">Save</button>
      <button type="button" onclick="fsaveas()">Save as...</button>
    </menu>
  </li>
</menu>
```

TEXT

<small>

기존 : 작은 사이즈의 글꼴을 표현합니다.

변경 : 의미적으로 작게 출력되어야 하는 구문에 사용합니다. 예) 법적 책임, 경고

코드 예제

<footer>

<address>

문의하실 내용은 펄님에게 메일을 보내주시면 됩니다.

</address>

<p><small>© copyright 2010 PearlNim.com</small></p>

</footer>

TEXT

기본 : 강한 강조(**em**보다 강한)의 의미를 나타내고자 할 때 사용합니다.

변경 : 매우 중요한 의미를 나타내고자 할 때 사용합니다.

코드 예제

```
<p><strong>경고.</strong><br />
```

```
이 던전은 매우 위험합니다. <strong>오리를 피하세요.</strong><br />
```

```
발견한 골드를 모두 수집하세요.<br />
```

```
폭발해서 반경 10미터 이내의 어떤 사물이든지 파괴해버리는 <strong>다이아몬드는 절대 수집하지  
마세요.</strong><p>
```

TEXT

"강조"와 "중요"에 대한 차이는 뭘까요?

아래와 같은 문장이 있습니다.

"갓 태어난 강아지는 귀엽습니다"

여 기서 화자가 강조하고자 하는 것이 "갓 태어난" 이라면, 아래와 같이 표현될 수 있고, 의미적으로 볼 때, 보다는 이 적합합니다.

"갓 *태어난* 강아지는 귀엽습니다."

"갓 태어난" 이 강조되면서 문장의 의미가 기존과는 약간?! 바뀌었다고 볼 수 있습니다.

이와는 다르게, 은 중요한 의미를 갖는 구문에 사용할 수 있으며, 의 사용 유무에 따라 전체적인 문장의 의미가 변하지는 않습니다.

아래 예제를 참고해주세요.

경고.

이 던전은 매우 위험합니다. **오리를 피하세요.**

발견한 골드를 모두 수집하세요.

폭발해서 반경 10미터 이내의 어떤 사물이든지 파괴해버리는 다이아몬드는 절대 수집하지 마세요.

참고로, HTML4.01에서 보다 강한 의미를 지니는 것이 이었다면,

HTML5에서

강조의 강조는 ...으로,

중요한 것 중에서 더 중요한 것을 나타내고자 할 때는 ...으로 같이 마크업 해야 합니다.

TEXT

참고 URL

HTML: The Markup Language

<http://dev.w3.org/html5/markup/>

HTML 5 differences from HTML 4

<http://www.w3.org/TR/html5-diff/>