

# Java Web Programming 입문

(보강 HTML, etc...)

32일차

# 오늘의 키워드

- ▶ Servlet/JSP 외 Web 기술 **겉핥기**
  - HTML (Hypertext Markup Language)
  - CSS (Cascading Style Sheet)
  - Javascript
  - jQuery
  - XML (eXtensible Markup Language)
  - AJAX (Asynchronous Javascript + XML)
  - JSON

# HTML

## ▶ Hypertext Markup Language

### ◦ Markup

- 문서의 활자, 조판 지정 표시
- 출력 시 문서에 추가되는 정보 (Tag)

### ◦ Hypertext

- Hyper(초월) + Text(텍스트) = 기존 텍스트를 초월하는 텍스트
- 문장 중 일부의 어구나 단어가 다른 문서의 열람을 지정하는 포인터(Pointer)로 사용 가능
- 문장을 읽다가 Hypertext로 만들어진 텍스트를 클릭하면
  - 그 텍스트에 지정된 다른 정보에 접근이 가능
- Hypertext와 정보간의 연결을 Hyperlink
- 기존 텍스트가 가지는 선형성, 고정성, 유한성 등의 제약을 극복
  - 사용자의 정보이동능력 향상



# HTML

## ▶ Tag의 기본형

```
<태그> 문장 <태그>  
<태그 속성 = "값"> 문장</태그>  
<태그>, <태그/>
```

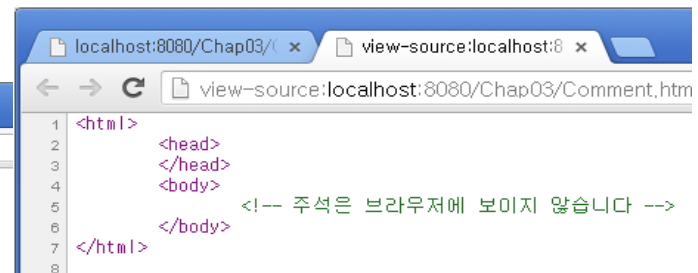
## ▶ HTML 문서의 구조

<html>	: HTML 시작
<head>	: HTML 헤더 시작
...	: HTML 헤더 정보
</head>	: HTML 헤더 끝
<body>	: HTML 바디 시작
...	: HTML 바디 내용
</body>	: HTML 바디 끝
</html>	: HTML 끝

## ▶ HTML의 주석

```
<!-- 주석처리 -->
```

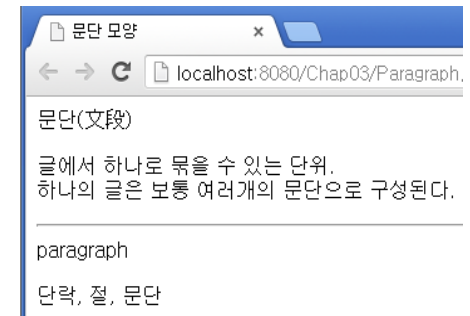
```
<html>  
  <head>  
  </head>  
  <body>  
    <!-- 주석은 브라우저에서 보이지 않습니다 -->  
  </body>  
</html>
```



# HTML

## ▶ 문단 및 선 태그

```
<html>
  <head>
    <title>문단 모양</title>
  </head>
  <body>
    문단(文段) <p> 글에서 하나로 묶을 수 있는 단위. <br/>
    하나의 글은 보통 여러개의 문단으로 구성된다. <hr>
    paragraph <p> 단락, 절, 문단
  </body>
</html>
```



태그	속성	범위	설명
<p>	align	left, right, center	문단의 정렬 방식을 지정
<hr>	align	left, right, center	수평선의 정렬 방식을 지정
	size	숫자	수평선의 두께를 지정 (픽셀 단위)
	width	숫자	수평선의 폭을 지정 (픽셀 단위, %)
	color	색상코드	수평선의 색을 지정
	noshade		수평선의 음영 효과를 없앨 때 사용

# HTML

## ▶ 문자 모양 태그

```
<html>
  <head>
    <title>문자 모양</title>
  </head>
  <body>
    <dfn>      Definition Tag      </dfn>
    <em>      Emphasis Tag        </em>
    <cite>     Citation Tag        </cite>
    <code>     Code Tag            </code>
    <kbd>     Keyboard Tag        </kbd>
    <samp>     Sample Tag          </samp>
    <strong>   Strong Tag          </strong>
    <var>     Variable Tag        </var>
    <b>       Boldface Tag        </b>
    <i>       Italics Tag         </i>
    <u>       Underlined Tag      </u>
    <tt>     Typewriter Tag      </tt>
    <blink>   Blinking Tag       </blink>
    <sub>     Subscript Tag       </sub>
    <sup>     Superscript Tag     </sup>
  </body>
</html>
```

```
<html>
  <head>
    <title>문자 크기</title>
  </head>
  <body>
    <h1>h1 태그를 이용한 문자</h1>
    <h2>h2 태그를 이용한 문자</h2>
    <h3>h3 태그를 이용한 문자</h3>
    <h4>h4 태그를 이용한 문자</h4>
    <h5>h5 태그를 이용한 문자</h5>
    <h6>h6 태그를 이용한 문자</h6>
  </body>
</html>
```

문자 크기

localhost:8080/Chap03/CharSize.hr

h1 태그를 이용한 문자

h2 태그를 이용한 문자

h3 태그를 이용한 문자

h4 태그를 이용한 문자

h5 태그를 이용한 문자

h6 태그를 이용한 문자

문자 모양

localhost

Definition Tag

Emphasis Tag

Citation Tag

Code Tag

Keyboard Tag

Sample Tag

Strong Tag

Variable Tag

Boldface Tag

Italics Tag

Underlined Tag

Typewriter Tag

Blinking Tag

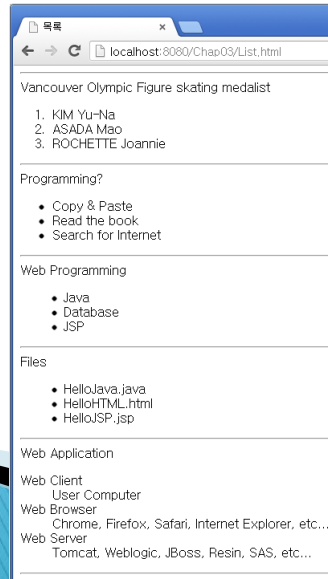
Subscript Tag

Superscript Tag

# HTML

## ▶ 목록 태그

태그	태그의미	설명
<ol>	Ordered List	순서 있는 목록
<ul>	Unordered List	순서 없는 목록
<menu>	Menu	짧은 문장용 목록 (순서 없는 목록과 유사)
<dir>	Directory	디렉토리, 파일명에 사용 되는 목록
<dl>	Definition List	용어 설명 목록
<dt>	Definition Term	설명할 용어
<dd>	Definition Description	용어 설명



```
<html>
<head>
  <title>목록</title>
</head>
<body>
  <hr>
  Vancouver Olympic Figure skating medalist <br/>
  <ol>
    <li>KIM Yu-Na</li>
    <li>ASADA Mao</li>
    <li>ROCHETTE Joannie</li>
  </ol>
  <hr>
  Programming?
  <ul>
    <li>Copy & Paste</li>
    <li>Read the book</li>
    <li>Search for Internet</li>
  </ul>
  <hr>
  Web Programming
  <menu>
    <li>Java</li>
    <li>Database</li>
    <li>JSP</li>
  </menu>
  <hr>
  Files
  <dir>
    <li>HelloJava.java</li>
    <li>HelloHTML.html</li>
    <li>HelloJSP.jsp</li>
  </dir>
  <hr>
  Web Application
  <dl>
    <dt>Web Client</dt>
    <dd>User Computer</dd>
    <dt>Web Browser</dt>
    <dd>Chrome, Firefox, Safari, Internet Explorer, etc...</dd>
    <dt>Web Server</dt>
    <dd>Tomcat, Weblogic, JBoss, Resin, SAS, etc...</dd>
  </dl>
  <hr>
</body>
</html>
```

# HTML

## ▶ <table>

속성	속성 설명
align="정렬 방식"	표의 정렬 방식 지정. Default 값은 left (left, right, center)
bgcolor="표의 배경 색"	표의 배경 색을 색상이나 색상 코드로 지정
background="표의 배경이 될 파일"	표의 배경 이미지 지정
border="표의 테두리 두께"	표의 테두리 두께 지정. Default 값은 1
bordercolor="표의 테두리 색"	표의 테두리 색을 색상이나 색상 코드로 지정
cellspacing="셀 간의 간격"	표의 셀과 셀간의 간격 지정
cellpadding="셀 내부 여백"	표 셀 내부의 여백 지정
width="표의 너비"	표의 너비 지정
height="표의 높이"	표의 높이 지정
frame="표의 바깥쪽 테두리"	표의 바깥쪽 테두리 지정
rules="표의 안쪽 테두리"	표의 안쪽 테두리 지정

### <tr> 태그의 속성

속성	의미
align="정렬 방식"	행의 정렬 방식 지정. Default 값은 left (left, right, center)
bgcolor="행의 배경색"	행의 배경색을 색상이나 색상 코드로 지정
valign="세로 정렬 방식"	행의 세로 정렬 방식 지정 (top, middle, bottom)
height="높이"	행의 높이를 지정
rowspan="행의 병합개수"	행의 병합개수를 지정

## ▶ <table> 태그 내부 태그

태그명	의미
<caption>	표에 대한 설명 혹은 제목을 표시
<th>	표의 헤더를 지정
<tr>	표의 행을 관리
<td>	표의 열을 관리
<col>	해당 열의 속성을 지정
<colgroup>	다수의 열 속성을 지정
<thead>	표 내부 그룹의 헤더 지정
<tfoot>	표 내부 그룹의 꼬릿말 지정
<tbody>	표 내부 그룹의 본문 지정

### <td> 태그의 속성

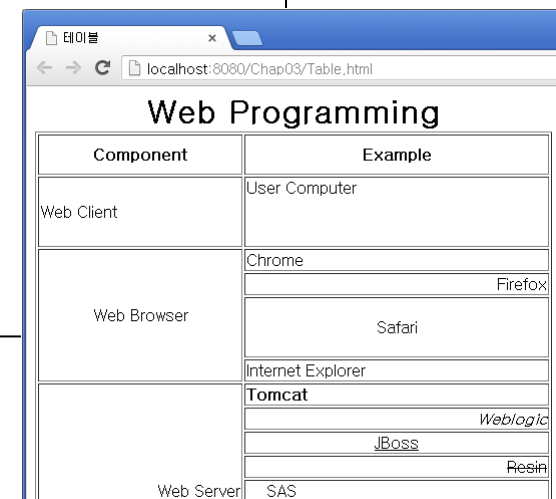
속성	의미
align="정렬 방식"	셀의 정렬 방식 지정. Default 값은 left (left, right, center)
valign="세로 정렬 방식"	셀의 세로 정렬 방식 지정 (top, middle, bottom)
background="배경이 될 파일"	셀의 배경 이미지 지정
bgcolor="배경색"	셀의 배경 색을 색상이나 색상 코드로 지정
colspan="열의 병합개수"	열의 병합개수를 지정
nowrap	셀 줄바꿈 금지 지정
width="셀의 너비"	셀의 너비 지정



# HTML

## ▶ <table>

```
<html>
  <head>
    <title>테이블</title>
  </head>
  <body>
    <table border="1">
      <caption><h1>Web Programming</h1></caption>
      <tr>
        <th width="200" height="40">Component</th><th width="300">Example</th>
      </tr>
      <tr>
        <td align="left">Web Client</td><td height="70" valign="top">User Computer</td>
      </tr>
      <tr>
        <td rowspan="4" align="center">Web Browser</td>
        <td>Chrome</td>
      </tr>
      <tr><td align="right">Firefox</td></tr>
      <tr><td height="60" align="center" valign="middle">Safari</td></tr>
      <tr><td>Internet Explorer</td></tr>
      <tr>
        <td rowspan="5" align="right" valign="bottom">Web Server</td>
        <td><b>Tomcat</b></td>
      </tr>
      <tr><td align="right"><i>Weblogic</i></td></tr>
      <tr><td align="center"><u>JBoss</u></td></tr>
      <tr><td align="right"><s>Resin</s></td></tr>
      <tr><td>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&SAS</td></tr>
    </table>
  </body>
</html>
```



The screenshot shows a web browser window with the title '테이블' and the address 'localhost:8080/Chap03/Table.html'. The rendered table is titled 'Web Programming' and contains the following data:

Component	Example
Web Client	User Computer
Web Browser	Chrome
	Firefox
	Safari
	Internet Explorer
Web Server	Tomcat
	Weblogic
	JBoss
	Resin
	SAS

# HTML

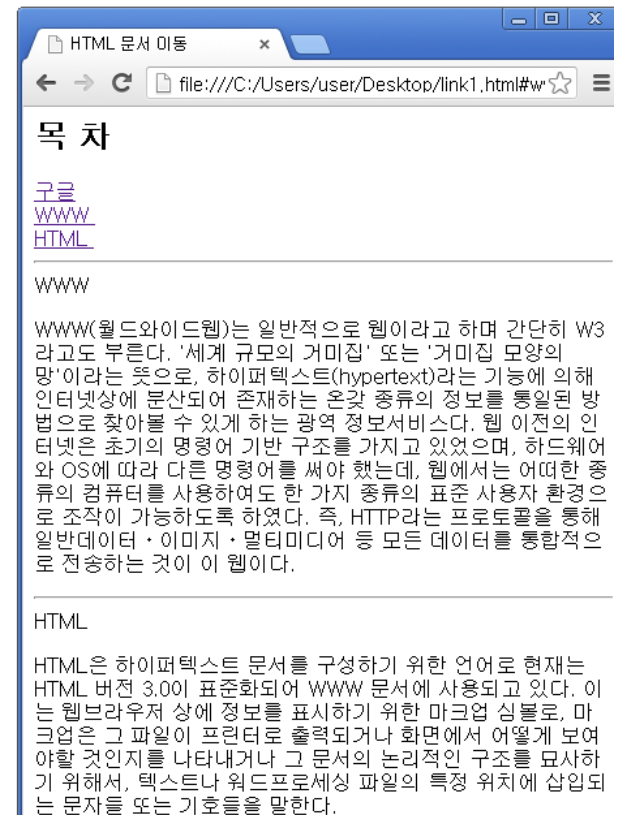
## ▶ <a> Anchor(닻)

```
<a name="이동할 정보 대상의 이름">XXX</a>
```

```
<a href="#이동할 정보 대상의 이름">XXX</a>
```

속성	의미
name="동일문서내위치이름"	"하이퍼텍스트로 동일 HTML 문서 내 이동한 특정 위치를 지정"
href="연결위치"	연결한 HTML 문서의 위치를 지정 (URL 사용)
target="출력대상"	연결된 HTML 문서가 출력될 대상지정 (_blank, _parent, _self, _top)

```
<html>
<head>
  <TITLE> HTML 문서 이동 </TITLE>
</head>
<body>
  <h2> 목 차 </h2>
  <a href="http://www.google.com">구글</a> <br/>
  <a href="#www"> WWW </a> <br/>
  <a href="#html"> HTML </a> <br/>
  <hr>
  <a name="www"> WWW </a>
  <p> WWW(월드와이드웹)는 일반적으로 웹이라고 하며 간단히 W3
  라고도 부른다. '세계 규모의 거미집' 또는 '거미집 모양의
  망'이라는 뜻으로, 하이퍼텍스트(hypertext)라는 기능에 의해
  인터넷상에 분산되어 존재하는 온갖 종류의 정보를 통일된 방
  법으로 찾아볼 수 있게 하는 광역 정보서비스다. 웹 이전의 인
  터넷은 초기의 명령어 기반 구조를 가지고 있었으며, 하드웨
  어와 OS에 따라 다른 명령어를 써야 했는데, 웹에서는 어떠
  한 종류의 컴퓨터를 사용하여도 한 가지 종류의 표준 사용자
  환경으로 조작이 가능하도록 하였다. 즉, HTTP라는 프로토콜을
  통해 일반데이터·이미지·멀티미디어 등 모든 데이터를 통합
  적으로 전송하는 것이 이 웹이다.
  </p>
  <hr>
  <a name="html"> HTML</a>
  <p> HTML은 하이퍼텍스트 문서를 구성하기 위한 언어로 현재는
  HTML 버전 3.0이 표준화되어 WWW 문서에 사용되고 있다. 이는
  웹브라우저 상에 정보를 표시하기 위한 마크업 심볼로, 마크
  업은 그 파일이 프린터로 출력되거나 화면에서 어떻게 보여야
  할 것인지를 나타내거나 그 문서의 논리적인 구조를 묘사하기
  위해서, 텍스트나 워드프로세싱 파일의 특정 위치에 삽입되는
  문자들 또는 기호들을 말한다.
  </p>
</body>
</html>
```



# HTML

## ▶ <img>

속성	의미
src="이미지 경로"	이미지 파일의 위치를 가리키는 URL 지정
align="정렬 방식"	이미지의 정렬 방식 지정 (top, bottom, middle, left, right)
alt="이미지 말풍선"	이미지의 말풍선 (마우스를 갖다 댔을 때 나타나는 텍스트) 지정
border="표의 테두리 두께"	이미지를 둘러싸는 테두리 지정
width="이미지의 너비"	이미지 너비를 픽셀 단위로 지정
height="이미지의 높이"	이미지 높이를 픽셀 단위로 지정
hspace="이미지 좌우여백"	이미지 좌우 여백 크기를 픽셀 단위로 지정
vspace="이미지 상하여백"	이미지 상하 여백 크기를 픽셀 단위로 지정



```
<html>
<head>
<title> HTML 문서 내 이미지 사용 </title>
</head>
<body>
SBS아카데미<br/>
SBS아카데미<br/>
SBS아카데미<br/>
<a href="http://www.sbsart.com/">

</a>
</body>
</html>
```



- ▶ 입력양식
  - <form>

- `<input>`

## <input> 태그의 속성 및 적용타입

<b>입 사 지 원 서</b>											
<b>성명</b>	(한글)	<b>해당분야</b>			<b>해당직무</b>	(1순위)	<b>사 진</b>				
	(한자)	<b>수련현</b>				(2순위)					
	(영문)	<b>해당근무처</b>				(3순위)					
<b>학 력</b>	년 월 일	부터				고등학교					
	년 월 일	부터				(대학교)	대학	과	부전공/ 복수전공		
	년 월 일	부터				(대학교)	대학원	과	부전공/ 복수전공		
	년 월 일	부터									
<b>성 적</b>		<b>1학년</b>	<b>2학년</b>	<b>3학년</b>	<b>4학년</b>	<b>대학 졸업</b>			<b>1학년</b>	<b>2학년</b>	<b>대학입결금</b>
	1학기							1학기			
	2학기							2학기			
<b>영 역</b>	<b>국 립</b>					<b>대 외</b>	입영예정일자 :				<b>해당요청</b>
	<input type="checkbox"/> 군영/ <input type="checkbox"/> 동계				제급	<input type="checkbox"/>	입영구분 <input type="checkbox"/> 경역 <input type="checkbox"/> 보통역 <input type="checkbox"/> 일반형/일반사				<input type="checkbox"/>
<b>가 속 관 계</b>	<b>관계</b>	<b>성명</b>	<b>연령</b>	<b>학 력</b>	<b>근무처 및 직위</b>			<b>동거여부(Y/N)</b>			
<b>외 국 예</b>	<b>외국어명</b>	<b>시 형 명</b>	<b>출생년월</b>	<b>점 수</b>	<b>종류기사용능력 (사용가능 연애 및 TOOL)</b>						
<b>경력 사항</b>	<b>기간</b>	<b>직장명</b>	<b>부서/직위</b>		<b>담당업무</b>			<b>의뢰사유</b>			
<b>지 적 증</b>	<b>자격 및 면허</b>					<b>등 급</b>		<b>취 득 일</b>			
<b>보통대상예우</b>		<input type="checkbox"/> 비대상 <input type="checkbox"/> 대상 (보통번호 : )									
<b>취득</b>						<b>특기</b>					

# HTML

## ▶ 입력양식

### ◦ <select>, <option>

태그	속성	의미
<select>	name="이름"	<select> 태그를 식별할 수 있는 이름 지정
	size="리스트개수"	리스트 박스에서 보여질 항목의 개수 지정
	multiple	리스트 박스에서 여러 항목 선택을 가능하게 지정
<option>	value="전송 데이터값"	선택된 항목이 실제 전송할 데이터 값 지정
	selected	리스트 박스에서 기본 선택이 되어 있는 항목을 지정

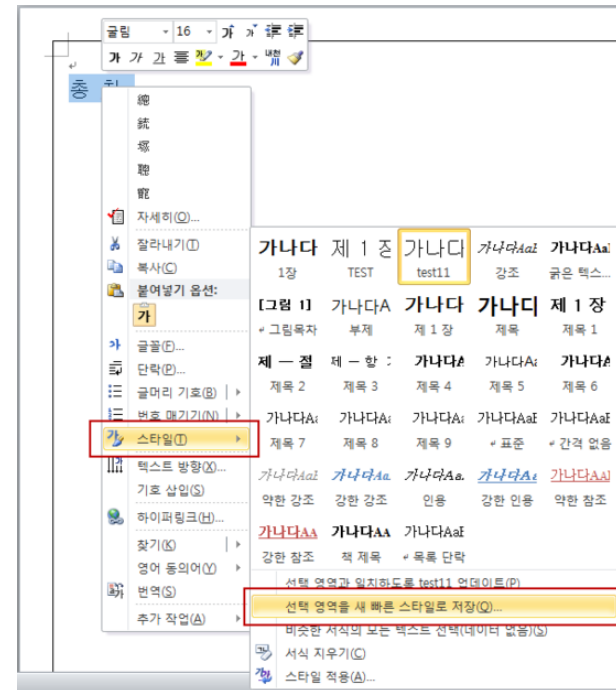
### ◦ <textarea>

속성	의미
name	<textarea> 태그를 식별할 수 있는 이름 지정
cols	글상자의 너비를 지정
rows	글상자의 높이를 지정
readonly	글상자에 클릭은 가능하지만, 텍스트 입력 수정은 불가하도록 설정
disabled	글상자에 클릭 및 텍스트 입력 수정이 모두 불가하도록 설정



# CSS

- ▶ CSS (Cascading Style Sheets)
  - Style?



# CSS

## ▶ CSS 사용방법

- HTML 태그 내부 style 속성

```
<tag style="about style;"> 내용 </tag>
```

- HTML 페이지 내부 <style>태그 사용

```
<style type="text/css">
스타일 정의
</style>
```

```
<style type="text/css">
선택자 { 속성:속성값; 속성:속성값; ... 속성:속성값; }
</style>
```

```
<parent_tag> <child_tag> 내용 </child_tag> </parent_tag>
```

요소	의미
선택자(selector)	CSS 를 이용해 스타일을 정의해줄 HTML 특정 요소(태그, 클래스, 아이디)를 지정
속성(property)	선택자를 통해 지정된 요소에 정의할 스타일 종류를 지정
속성값(value)	정의할 스타일의 값을 지정

방식	사용법
HTML 태그	HTML 태그 { 속성:속성값; 속성:속성값; ... 속성:속성값; }
클래스	.스타일클래스 { 속성:속성값; 속성:속성값; ... 속성:속성값; }
아이디	#아이디 { 속성:속성값; 속성:속성값; ... 속성:속성값; }



```
tag1 tag2 { 속성:속성값; 속성:속성값; ... 속성:속성값; }
```

```
tag.className { 속성:속성값; 속성:속성값; ... 속성:속성값; }
```

```
<tag class="className"> 내용 </tag>
```

```
.className { 속성:속성값; 속성:속성값; ... 속성:속성값; }
```

```
<tag class="className"> 내용 </tag>
```



# CSS

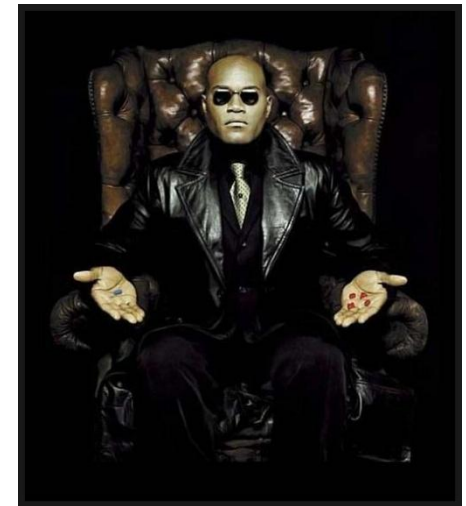
## ▶ CSS 사용방법

### ◦ 외부 css 사용

```
<parent_tag> <child_tag> 내용 </child_tag> </parent_tag>
```

### ◦ 우선순위

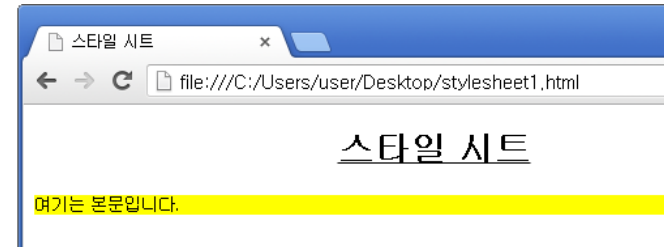
방법	우선 순위
HTML 태그에 <b>style</b> 속성을 이용하여 적용	높음
<b>&lt;style&gt;</b> 태그에 스타일을 정의하여 적용	
외부의 CSS 파일을 <b>&lt;link&gt;</b> 태그를 이용하여 적용	낮음



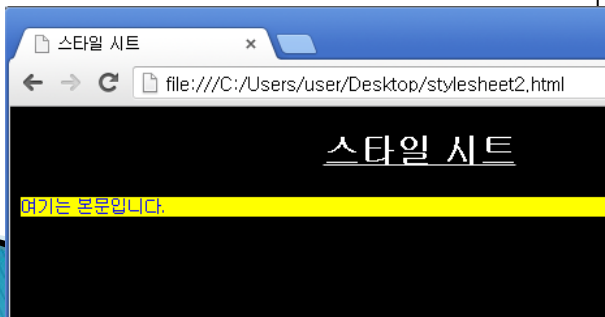


# CSS

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
    <title>스타일 시트</title>
    <style type="text/css">
      h1 {
        font-family : Arial;
        font-size : 20pt;
        text-align : center;
        text-decoration : underline;
      }
      p {
        font-size : 10pt;
        background-color : yellow;
      }
    </style>
  </head>
  <body>
    <h1>스타일 시트</h1>
    <p>여기는 본문입니다.</p>
  </body>
</html>
```



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html;
    charset=euc-kr">
    <title>스타일 시트</title>
    <style type="text/css">
      body {
        background-color : black;
        color : blue
      }
      h1 {
        color : white;
        font-family : Helvetica;
        font-size : 20pt;
        text-align : center;
        text-decoration : underline;
      }
      p {
        font-size : 10pt;
        background-color : yellow;
      }
    </style>
  </head>
  <body>
    <h1>스타일 시트</h1>
    <p>여기는 본문입니다.</p>
  </body>
</html>
```

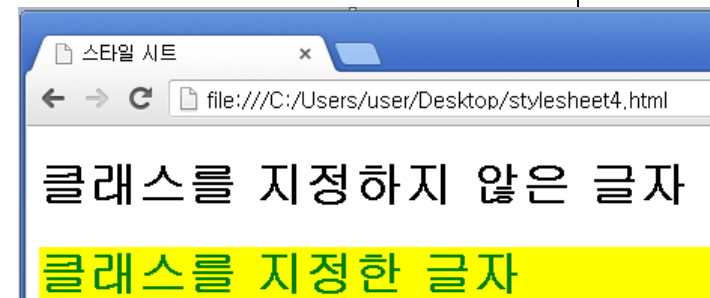


# CSS

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
    <title>스타일 시트</title>
    <style type="text/css">
      p em {
        text-decoration : underline;
      }
    </style>
  </head>
  <body>
    <h1><em>스타일 시트</em></h1>
    <p>p태그의 <em>em 태그 안의 글자</em>에 밑줄이 나타나게 된다.</p>
  </body>
</html>
```



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
    <title>스타일 시트</title>
    <style type="text/css">
      .mystyle {
        background-color : yellow;
        color : green;
      }
    </style>
  </head>
  <body>
    <h1>클래스를 지정하지 않은 글자</h1>
    <h1 class="mystyle">클래스를 지정한 글자</h1>
  </body>
</html>
```



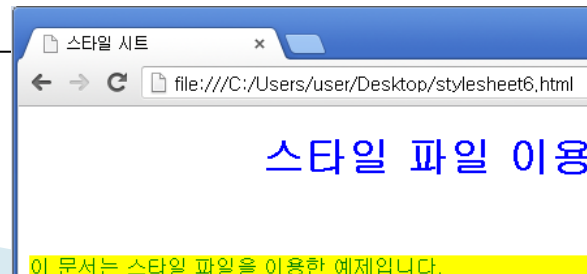
# CSS

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
    <title>스타일 시트</title>
    <style type="text/css">
      #myid {
        color : blue;
        text-align : center;
      }
    </style>
  </head>
  <body>
    <h1>스타일 없는 글자</h1>
    <h1 id="myid">ID에 스타일을 지정 한 글자</h1>
  </body>
</html>
```



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
    <title>스타일 시트</title>
    <link rel="stylesheet" type="text/css" href="mystyle.css" title="MyStyle">
  </head>
  <body>
    <h1 class="mystyle">스타일 파일 이용</h1><br/>
    <p class="mystyle">이 문서는 스타일 파일을 이용한 예제입니다.</p>
  </body>
</html>
```

```
h1.mystyle {
  color : blue;
  text-align : center;
}
p.mystyle {
  background-color : yellow;
  color : green;
}
```



# CSS

## ▶ font

속성	의미	값
font-size	폰트의 크기 지정	고정 사이즈 : x-small, small, medium, large, x-large, xx-large  상대 사이즈 : larger, smaller 숫자 : px 값 백분율 : %값
font-family	폰트의 종류 지정	글꼴 종류 : Arial, cursive, Helvetica, 굴림체
font-style	폰트의 기울임체 지정	normal, italic, oblique
font-variant	작은 대문자 지정	normal, small-caps
font-stretch	폰트의 폭 지정	ultra-condensed, extra-condensed, condensed, semi-condensed, normal, semi-expanded, expanded, extra-expanded, ultra-expanded, wider, narrower
font-weight	폰트의 두께 지정	normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900
color	폰트의 색상 지정	16진수 컬러코드, 색상명
font	폰트의 여러 속성을 한번에 지정	

EDDY'S BONES PLANK Cracked  
 CREEPY Blur CHOCOLATE  
 COFFEE SACK FLOWER POWER  
 Cathedral GAUSS RUBBER STAMP  
 celtic SPIKE Beau Amaze  
 Borghs Cracked Southwestern  
 ELIZABETH Wright Bart BIG NIB  
 FANZEE Oak Ridge NEW YORK  
 Lauren DECO Parisian Crane  
 FATSO DEVON NATIVE  
 Catchup Expose Heidleberg  
 IRONWORK Asia TEMPLETT  
 Cocoa Billboard DAVIS  
 GREMLIN Brando EXPUL

# CSS

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
  <title>스타일 시트</title>
  <style type="text/css">
    h3.family1 { font-family : Helvetica; font-size : 15pt; }
    h3.family2 { font-family : Arial; font-size : 15pt; }
    h3.family3 { font-family : serif; font-size : 15pt; }
    h3.family4 { font-family : sans-serif; font-size : 15pt; }

    h3.size1 { font-size : x-small; }
    h3.size2 { font-size : small; }
    h3.size3 { font-size : medium; }
    h3.size4 { font-size : large; }
    h3.size5 { font-size : x-large; }
    h3.size6 { font-size : xx-large; }

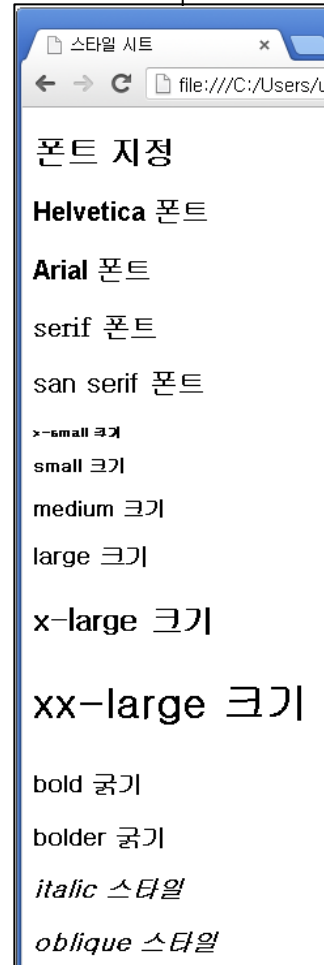
    h3.weight1 { font-weight : bold; }
    h3.weight2 { font-weight : bolder; }

    h3.style1 { font-style : italic; }
    h3.style2 { font-style : oblique; }
  </style>
</head>
<body>
  <h2> 폰트 지정 </h2>
  <h3 class="family1"> Helvetica 폰트 </h3>
  <h3 class="family2"> Arial 폰트 </h3>
  <h3 class="family3"> serif 폰트 </h3>
  <h3 class="family4"> san serif 폰트 </h3>

  <h3 class="size1"> x-small 크기 </h3>
  <h3 class="size2"> small 크기 </h3>
  <h3 class="size3"> medium 크기 </h3>
  <h3 class="size4"> large 크기 </h3>
  <h3 class="size5"> x-large 크기 </h3>
  <h3 class="size6"> xx-large 크기 </h3>

  <h3 class="weight1"> bold 굵기 </h3>
  <h3 class="weight2"> bolder 굵기 </h3>

  <h3 class="style1"> italic 스타일 </h3>
  <h3 class="style2"> oblique 스타일 </h3>
</body>
</html>
```



# CSS

## ▶ text, box

속성	의미	값
text-align	텍스트의 가로 정렬 방식 지정	left, right, center, justify
vertical-align	텍스트의 세로 정렬 방식 지정	baseline, super, sub, top, bottom, text-top, text-bottom, middle, %
text-decoration	실선을 사용하여 링크와 같은 텍스트 꾸밈 방식을 지정	none, underline, overline, line-through
text-indent	텍스트의 들여쓰기 지정	숫자 : px 백분율 : %
text-shadow	텍스트의 그림자 효과 지정	none, color, length
text-transform	텍스트의 대소문자 지정	none, capitalize, uppercase, lowercase
word-spacing	단어 사이의 간격을 지정	normal 숫자 : px, pt, cm
letter-spacing	글자 사이 간격 지정	normal 숫자 : px, pt, cm
white-spacing	공백 지정	normal : 문자열이 너비를 넘어가면 줄바꿈 nowrap : 줄바꿈하지 않음 pre : 모든 공백 인정
line-height	문장과 줄의 간격 지정	normal 숫자 : 숫자, px 백분율 : %

속성	의미	값
margin	박스 모델 외부 여백 지정	한번에!
margin-left	박스 모델 외부 좌측 여백 지정	auto : 자동조절 숫자 : 절대 길이 백분율 : %
margin-right	박스 모델 외부 우측 여백 지정	
margin-top	박스 모델 외부 상단 여백 지정	
margin-bottom	박스 모델 외부 하단 여백 지정	
padding	박스 모델 내부 전체 여백 지정	
padding-left	박스 모델 내부 좌측 여백 지정	
padding-right	박스 모델 내부 우측 여백 지정	
padding-top	박스 모델 내부 상단 여백 지정	
padding-bottom	박스 모델 내부 하단 여백 지정	

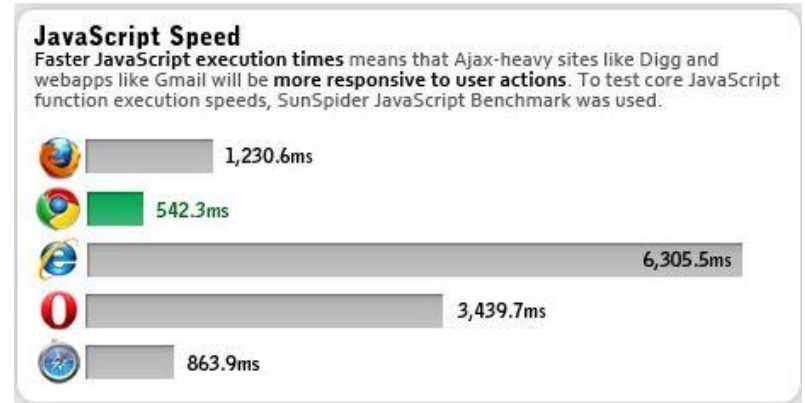
# CSS

- ▶ 그 외에도...
  - 배경이미지 (background)
  - 테두리 (border)
  - 이미지 테두리 (border-image)
  - 치수
  - 위치 제어 (position)
  - 윤곽 (outline)
  - 음성 (speech)
  - 투명색
  - 그라데이션
  - 움직임
  - 콘텐츠 생성
  - 다단
  - 가상 요소
  - 가상 클래스
  - 기기 타입별 속성
  - 미디어 쿼리
  - 브라우저, 벤더별 프리픽스
  - Etc...



# Javascript

- ▶ Javascript
  - Netscape + Sun Microsystems
  - 다 망함. 기술력과 생존과의 관계는?
- ▶ 문법
  - C 계열, Java 계열과 비슷
- ▶ Interpreter 방식
  - No Compile
  - No Link
- ▶ Client Side Script
  - Web Server (X)
  - Web Browser (O)
    - Web Browser 에서 직접 처리
    - 사용자의 입력에 대한 응답 속도 향상



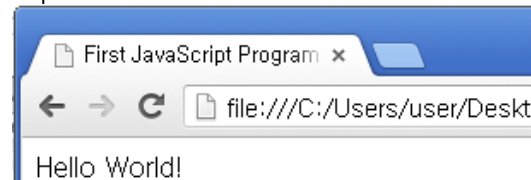
애플리케이션	자바스크립트의 종류
모질라 파이어폭스	자바스크립트
구글 크롬	자바스크립트
인터넷 익스플로러	Jscript
오페라	ECMAScript
애플 사파리	자바스크립트
마이크로소프트 닷넷 프레임워크	JScript.NET
어도비 플래시와 플렉스	액션스크립트
어도비 아크로벳	자바스크립트



# Javascript

## ▶ Hello World!

```
<html>
  <head>
    <title>First JavaScript Program</title>
    <script type="text/javascript">
      document.write('Hello World!');
    </script>
  </head>
  <body>
  </body>
</html>
```



- Javascript 코드는 보통 <head> 태그에 작성
  - <body> 태그를 읽으면서 실행 되버릴 수도 있다
  - 가능하면 <head> 태그에 넣을것



## ▶ 외부 Javascript 파일 불러오기

```
<html>
  <head>
    <title>First JavaScript Program</title>
    <script type="text/javascript" src="first2.js">
    </script>
  </head>
  <body>
  </body>
</html>
```

```
document.write("Hello World!");
```

# Javascript

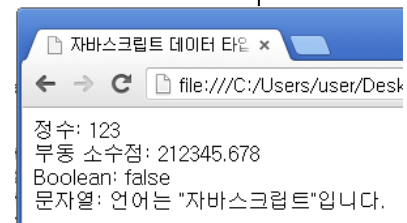
## ▶ Data Type

- 정수
- 부동 소수점
- Boolean (true/false)
- 문자열 ( " ~ " , ' ~ ' )

특수 문자	의 미
\n	다음 줄로 이동(newline)
\t	탭 문자 삽입(tab)
\b	백스페이스(backspace)
\r	리턴(return)
\\	역슬래시(\)      사용 예) '\\'
\'	작은 따옴표(')      사용 예) \"
\"	큰 따옴표(")      사용 예) \"
\0	NULL 문자
\v	세로 탭
\f	폼 피드
\u XXXX	유니코드 문자(16진수)
\x XX	Latin-1 문자(16진수)



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
    <title>자바스크립트 데이터 타입</title>
  </head>
  <body>
    <script type="text/javascript">
      <!--
        /* 정수 */
        document.write("정수: " + 123 + "<br>");
        /* 부동 소수점 */
        document.write("부동 소수점: " + 2.12345678e5 + "<br>");
        /* Boolean, 1과 2가 같은 여부를 리턴 */
        document.write("Boolean: " + (1 == 2) + "<br>");
        /* 문자열 */
        document.write("문자열: " + "언어는 \"자바스크립트\"입니다." + "<br>");
      // -->
    </script>
  </body>
</html>
```



# Javascript

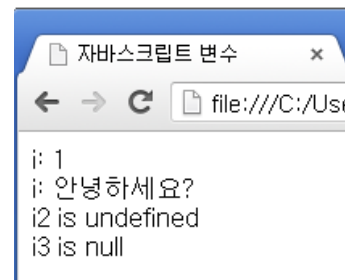
## ▶ 변수 (Global, Local)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
    <title>자바스크립트 변수</title>
  </head>
  <body>
    <script type="text/javascript">
      <!--
        /* 정수형 변수 선언 */
        var i = 1;
        document.write("i: " + i + "<br>");

        /* 문자열 변수로 변환 */
        i = "안녕하세요?";
        document.write("i: " + i + "<br>");

        /* 선언만 하고 값을 할당하지 않은 변수 출력 */
        var i2;
        if (i2 == undefined)
          document.write("i2 is undefined" + "<br>");
        if (i2)
          document.write("if(i2)가 true일까?" + "<br>");

        /* 선언만 하고 null을 할당한 변수 출력 */
        var i3 = null;
        if (i3 == null)
          document.write("i3 is null" + "<br>");
        if (i3)
          document.write("if(i3)가 true일까?" + "<br>");
        // -->
      </script>
    </body>
  </html>
```



# Javascript

## ▶ 연산자

- 수치 연산자
- 오칙 연산자 (+, -, \*, /, %)
- 증감 연산자 (++ , --)
- 단항 연산자
- 비교 연산자 (>, <, >=, <=, ==, !=, ===, !==)
- 조건 연산자 (AND, OR, !, ?)
- 비트 연산자 (ㄱ ㄱ ㄱ)
- 대입 연산자
- typeof 연산자
- 연산자의 우선순위

## ▶ 제어문

- if
- switch
- for
- for ~ in
- while
- do ~ while
- break
- continue
- throw
- return
- try ~ catch ~ finally



# Javascript

## ▶ Function

```
<html>
  <head>
    <title>First JavaScript Program</title>
    <script type="text/javascript">
      function 함수이름 (매개변수1, 매개변수2, ...){
        함수문장
      }
      function 매개 변수가 없는 함수 (){
        함수문장
      }
      function 매개 변수가 있는 함수 (매개변수){
        함수문장
      }
      function 리턴값이 있는 함수 (매개변수){
        함수문장
        return 리턴값;
      }
    </script>
  </head>
  <body>
</body>
</html>
```



```
<body>
  <script type="text/javascript">
    function1();
  </script>
  <script type="text/javascript">
    function2(parameter);
  </script>
  <script type="text/javascript">
    var returnVar
    returnVar = function3(parameter);
  </script>
</body>
```

- Function Literal
- Inner Function
- Recursive Function

# Javascript

## ▶ 그 외... (라고 하기엔 너무 많다)

- 내장 함수
  - alert, confirm, eval, parseInt, parseFloat, escape, unescape, isNaN (Not A Number), isFinite (finite number)
- Javascript의 Object
  - new / delete
  - Method
- Exception
- 내장 객체
  - String
  - Date
  - Math
  - Array
  - Function
  - Screen
  - Number
  - Boolean



- Event + Event handler
  - abort, blur, click, change, dblclick, error, focus, keydown, keypress, keyup, load, mousedown, mousemove, mouseout, mouseover, mouseup, reset, resize, select, submit, unload
- Web Browser 내장 객체
  - window
  - document
  - cookie
  - history
  - location
  - link
  - anchor
  - navigator
  - image
  - frame
- Javascript로 Form 태그 제어
- Javascript로 CSS 제어

# jQuery

## ▶ 사용

- <http://jquery.com>
- CDN Host 사용

```
<!DOCTYPE html>
<html>
  <head>
    <script src="http://code.jquery.com/jquery-1.7.js"></script>
    <script>
    </script>
  </head>
  <body>
  </body>
</html>
```



## ▶ 공부

- <http://jquery.com>
  - 영어 공부
  - 어떤 책도 여기보다 정확할 수 없다
- jQuery 만 알아도 상관없지만...
  - HTML, CSS, Javascript 를 알면 많은 도움이 된다

# jQuery

## ▶ ready()

- 문서가 준비되면 jQuery를 시작

```
<script>
    $(document).ready(function() {

    });
</script>
```

```
<script>
    window.onload = function() {

    };
</script>
```

```
<script>
    $(function () {

    });
</script>
```



## ▶ jQuery Selector

- jQuery 최고의 무기
- CSS의 선택자와 유사
- jQuery(\$)(Selector).Method();

```
<script>

    $(selector)

    $('h1').css('color', 'red');

    jQuery(selector)

    jQuery('h').css('color', 'red');

</script>
```



# jQuery

- Selector

- All

```
<script>
  $(document).ready(function () {
    $('*').css('color', 'Red');
  });
</script>
```

- Tag

```
<script>
  $(document).ready(function () {
    $('h1').css('color', 'Orange');
  });
</script>
```

```
<script>
  $(document).ready(function () {
    $('h1, p').css('color', 'Orange');
  });
</script>
```

- ID

```
<script>
  $(document).ready(function () {
    $('#target').css('color', 'Orange');
  });
</script>
```

```
<script>
  $(document).ready(function () {
    $('h1#target').css('color', 'Orange');
  });
</script>
```

- Class

```
<script>
  $(document).ready(function () {
    $('.item').css('color', 'Orange');
    $('h1.item').css('background', 'Red');
  });
</script>
```

# jQuery

## ▶ Selector

### ◦ 자식

```
<script>
    $(document).ready(function () {
        $('body > *').css('color', 'Red');
    });
</script>
```

### ◦ 후손

```
<script>
    $(document).ready(function () {
        $('body *').css('color', 'Red');
    });
</script>
```

### ◦ 속성

요소[속성]	특정 속성을 가지고 있는 문서 객체를 선택
요소[속성=값]	속성 안의 값이 특정 값과 같은 문서 객체를 선택
요소[속성~=값]	속성 안의 값이 특정 값을 단어로써 포함하는 문서 객체를 선택
요소[속성^=값]	속성 안의 값이 특정 값으로 시작하는 문서 객체를 선택
요소[속성\$=값]	속성 안의 값이 특정 값으로 끝나는 문서 객체를 선택
요소[속성*=값]	속성 안의 값이 특정 값을 포함하는 문서 객체를 선택

```
<body>
    <input type="text" />
    <input type="password" />
    <input type="radio" />
    <input type="checkbox" />
    <input type="file" />
</body>
```

```
<script>
    $(document).ready(function () {
        $('input[type=text]').val('Hello, jQuery');
    });
</script>
```

# jQuery

## ▶ Selector

### ◦ <input> tag filter

요소:button	input 태그 중 type 속성이 button인 문서 객체와 button 태그를 선택
요소:checkbox	input 태그 중 type 속성이 checkbox인 문서 객체를 선택
요소:file	input 태그 중 type 속성이 file인 문서 객체를 선택
요소:image	input 태그 중 type 속성이 image인 문서 객체를 선택
요소:password	input 태그 중 type 속성이 password인 문서 객체를 선택
요소:radio	input 태그 중 type 속성이 radio인 문서 객체를 선택
요소:reset	input 태그 중 type 속성이 reset인 문서 객체를 선택
요소:submit	input 태그 중 type 속성이 submit인 문서 객체를 선택
요소:text	input 태그 중 type 속성이 text인 문서 객체를 선택

요소:checked	체크된 입력 양식을 선택
요소:disabled	비활성화된 입력 양식을 선택
요소:enabled	활성화된 입력 양식을 선택
요소:focus	초점이 맞춰져 있는 입력 양식을 선택
요소:input	모든 입력 양식을 선택(input, textarea, select, button 태그).
요소:selected	option 객체 중 선택된 태그를 선택

# jQuery

## ▶ Selector

### ◦ Location Filter

요소:odd	홀수 번째에 위치한 문서 객체를 선택
요소:even	짝수 번째에 위치한 문서 객체를 선택
요소:first	첫 번째 위치한 문서 객체를 선택
요소:last	마지막에 위치한 문서 객체를 선택

```
<!DOCTYPE html>
<html>
  <head>
    <script src="http://code.jquery.com/jquery-1.7.js"></script>
    <script>
      $(document).ready(function () {
        $('tr:odd').css('background', '#F9F9F9');
        $('tr:even').css('background', '#9F9F9F');

        $('tr:first').css('background', '#000000').css('color', '#FFFFFF');
      });
    </script>
  </head>
  <body>
    <table>
      <tr><th>이름</th><th>혈액형</th><th>지역</th></tr>
      <tr><td>홍길동</td><td>AB형</td><td>서울</td></tr>
      <tr><td>박문수</td><td>B형</td><td>대전</td></tr>
      <tr><td>심청이</td><td>AB형</td><td>대구</td></tr>
      <tr><td>김두환</td><td>O형</td><td>부산</td></tr>
      <tr><td>마당쇠</td><td>A형</td><td>수원</td></tr>
      <tr><td>누렁이</td><td>B형</td><td>인천</td></tr>
    </table>
  </body>
</html>
```

# jQuery

## ▶ 그 외...

- 배열 관리
  - `each()`, `addClass()`, `removeClass()`
- 객체 확장
  - `extend()`
- 문서 객체 탐색
  - `filter()`, `add()`, `is()`, `find()`
- 문서 객체 조작
  - `addClass()`, `removeClass()`, `attr()`, `css()`, `html()`, `text()`, `remove()`, `empty()`, `$()`, `appendTo()`, `prependTo()`, `insertAfter()`, `insertBefore()`, `append()`, `prepend()`, `after()`, `before()`, `clone()`
- 이벤트
  - `bind()`, `toggle()`, `hover()`, `unbind()`, `one()`, `pageX`, `pageY`, `preventDefault()`, `stopPropagation()`, `trigger()`, `delegate()`, `live()`, `one()`, `die()`, `on()`, `off()`, `click`, `dblclick`, `mousedown`, `mouseup`, `mouseenter`, `mouseleave`, `mousemove`, `mouseout`, `mouseover`, `keydown`, `keypress`, `keyup`, `ready`, `load`, `unload`, `resize`, `scroll`, `error`, `change`, `focus`, `focusin`, `focusout`, `blur`, `select`, `submit`, `reset`
- 효과
  - `show()`, `hide()`, `toggle()`, `slideDown()`, `slideUp()`, `slideToggle()`, `fadeIn()`, `fadeOut()`, `fadeToggle()`, `animate()`, `clearQueue()`, `stop()`, `delay()`

# XML

## ▶ eXtensible Markup Language

- 확장 가능한 Markup 언어
- 기존에 정의되지 않은 Tag를 새로 만들어서 씀
- Database와 같은 구조화된 Data 저장에 부적합한 HTML의 대체 수단으로 등장

## ▶ 공부해야할 부분

- DTD
- XML Name Space
- XML Schema
- XPath
- XSL
- DOM
- SAX

# AJAX

- ▶ Asynchronous Javascript and XML
- ▶ Web Page 로딩된 후 Javascript를 이용하여 Web Server에 정보를 요청
  - 응답 처리 결과가 ‘새로고침’ 되지 않고 화면에 업데이트
- ▶ 화면 갱신에 필요한 최소한의 정보만을 ‘비동기적’으로 주고 받음
  - 서버의 효율을 높임
- ▶ Background HTTP 요청 처리
  - 사용자는 Web Page의 깜빡임과, 새 화면 응답을 위한 대기가 화연히 줄어듦
- ▶ XMLHTTP
- ▶ jQuery 에서의 AJAX 사용

# JSON

- ▶ Javascript Object Notation
  - Data 전송방식
  - XML 보다 빠르다





# 앞으로 숙제

## ▶ Theory

- Java
- Data Structure
- Design Pattern

## ▶ Framework

- iBatis -> MyBatis
- Hibernate
- Struts2
- Spring
  - 전자정부 표준프레임워크 (eGovFrame)

## ▶ Source Configuration Management

- CVS -> SVN

## ▶ View

- JavaScript -> jQuery
- HTML 4 -> HTML 5
- Web Standard

## ▶ Data Binding

- XML
  - DOM (Document Object Model)
  - SAX (Simple API for XML)
- JSON (JavaScript Object Notation)

## ▶ OS

- Unix 계열 명령어
  - Solaris, Linux, Unix, etc...
- Batch (Crontab 등등)

## ▶ Database

- SQL Tuning
- Backup & Recovery