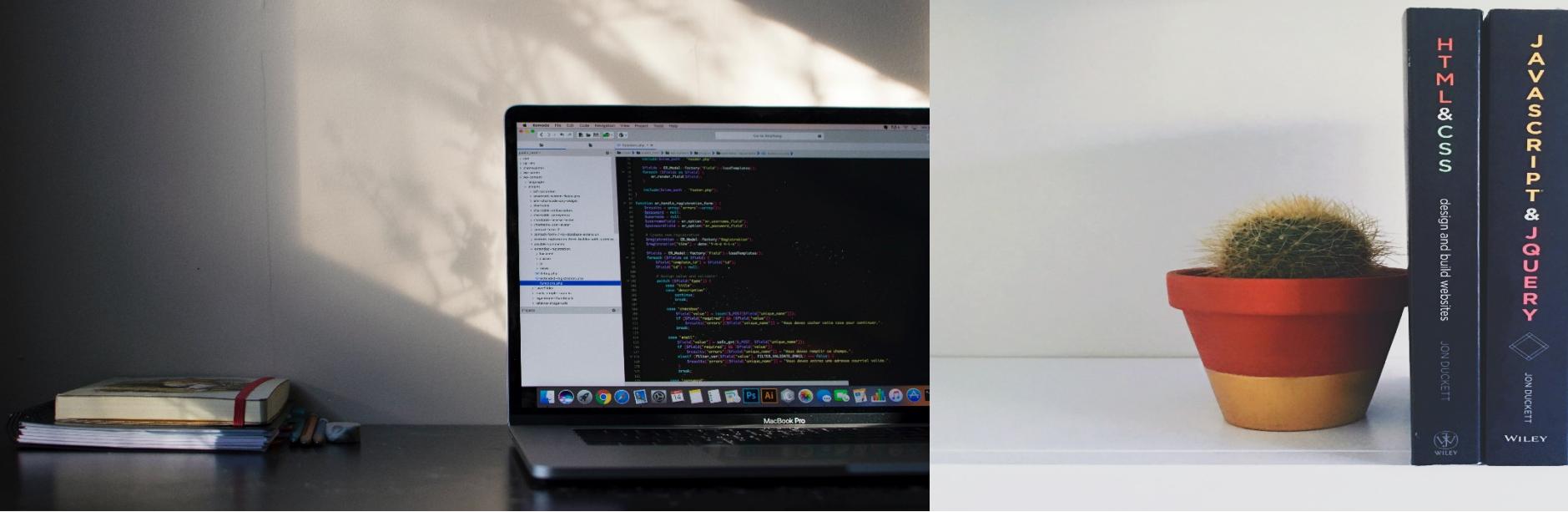


HTML

김기정 (bangry313@gmail.com)

목차 (Table of Contents)

1. WEB / HTML 소개
2. 개발 환경 구축
3. HTML 기초 다지기
4. HTML Markup Elements
5. Semantic Elements



1. WEB / HTML 소개

OM KBE FRS FREng FRSA FBCS
팀 버너스리
Sir Tim Berners-Lee

본명	티머시 존 버너스리 경 [1] Sir Timothy John Berners-Lee
출생	1955년 6월 8일 (66세) 영국 잉글랜드 런던
국적	영국
직업	컴퓨터과학자, 프로그래머, 교수

- 1.1 WEB 개념
- 1.2 Web Server & Web Client 구동 원리
- 1.3 HTML개요
- 1.4 HTML 역사
- 1.5 HTML문서 구조

1.1 WEB 개념

✓ Internet - 1969년

- TCP/IP 통신규약을 기반으로 하는 세계적으로 연결된 컴퓨터 네트워크
- 초기 인터넷에서는 하드웨어와 OS에 따라 다른 명령어를 사용하여 정보 교환

✓ WEB (World Wide Web) - 1989년 Tim Berners-Lee 개발

- 인터넷에 연결된 사용자들이 서로의 정보를 쉽게 공유할 수 있도록 고안된 통신 서비스

✓ 웹은 인터넷 상에서 텍스트, 그림, 소리, 영상 등과 같은 멀티미디어 정보를 하이퍼텍스트 방식으로 연결하여 제공

- 하이퍼텍스트(hypertext) : 문서 내부에 hyperlink를 사용함으로써 웹 상에 존재하는 여러 문서끼리 서로 참조 할 수 있는 기술

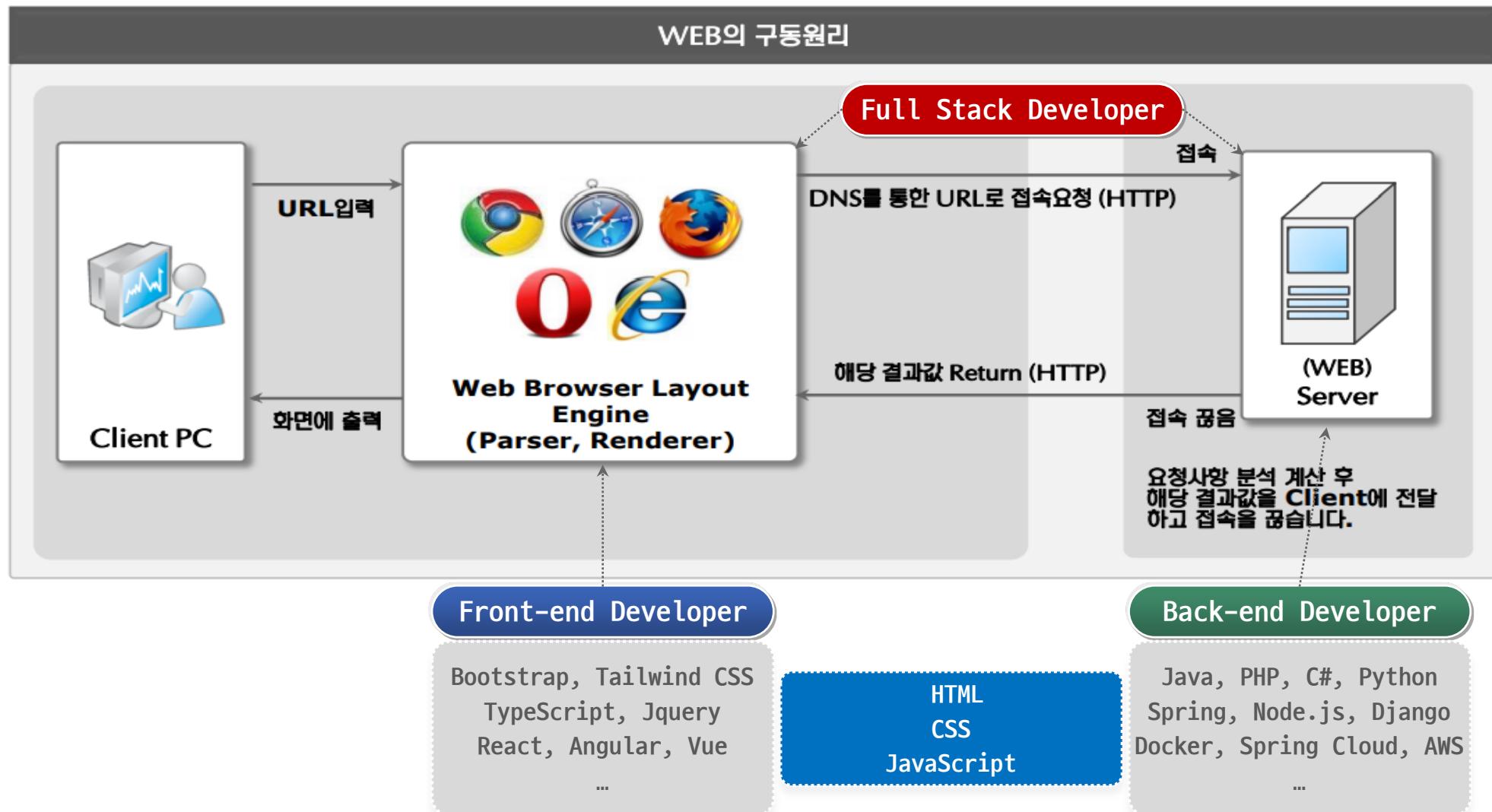
✓ HTML 언어로 작성된 웹 상의 문서는 TCP/IP 기반의 HTTP 프로토콜을 사용하면 누구나 검색하고 접근 가능

✓ W3C (World Wide Web Consortium)

- HTML, CSS, XML 등과 같은 웹 언어나 인터넷 프로토콜 등에 대한 웹 표준을 제정하고 관리하는 국제기구
- 여러 교육기관과 다양한 기업들이 모여 표준화 추진
- <https://www.w3.org>

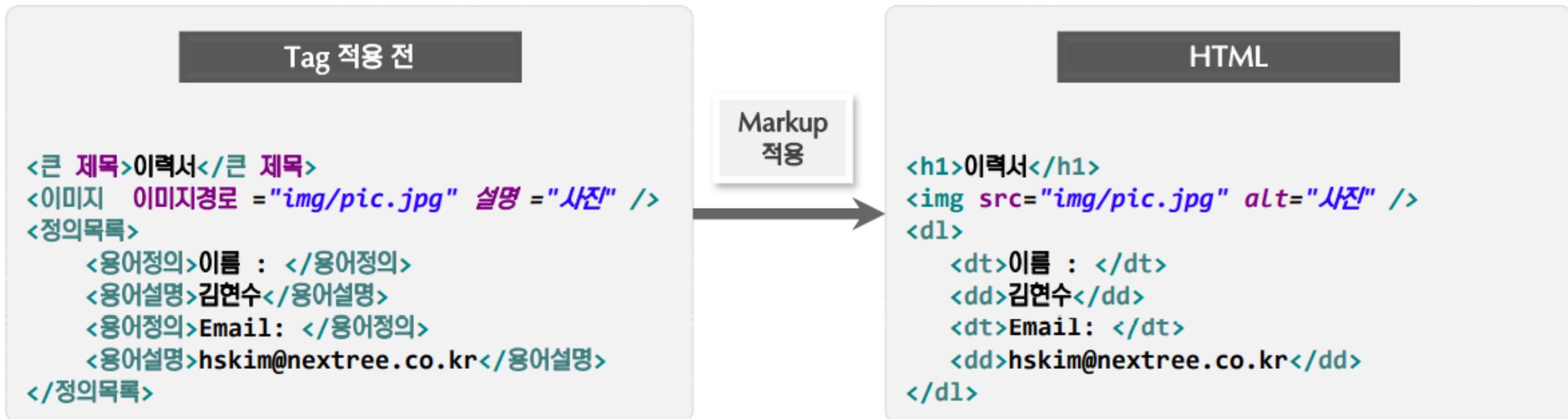
✓ Web Server vs Web Client

1.2 Web Server & Web Client 구동 원리



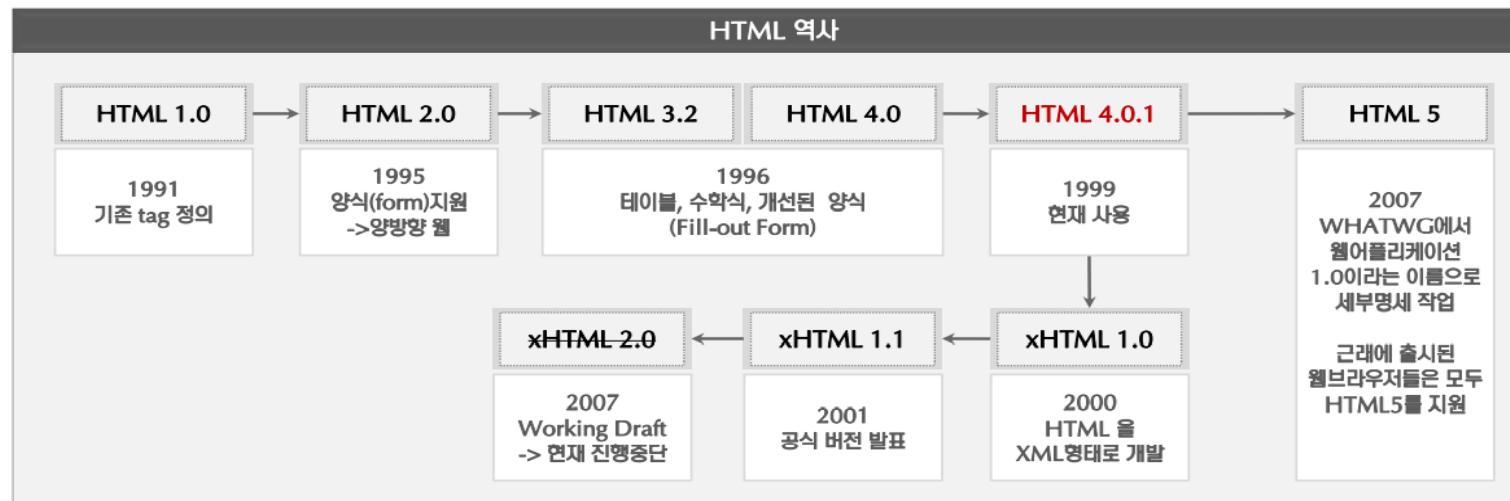
1.3 HTML 개요

- ✓ HTML(HyperText Markup Language)은 웹을 이루는 가장 기본적인 구성 요소로, 웹 문서 작성 시 콘텐츠의 의미와 구조를 정의할 때 사용하는 마크업 언어로 1990년 이후 사용
- ✓ 일반적인 텍스트와 문법적으로 구분하기 위해 문서에 태그를 사용
- ✓ 구조를 위한 태그(Tag)들과 하이퍼텍스트, 표, 목록, 이미지, 비디오 등을 포함할 수 있는 다양한 태그를 제공



1.4 HTML 역사

- ✓ 1991년 웹의 창시자 Tim Berners-Lee가 HTML1.0 처음 도입
- ✓ 1993년 Mosaic 최초 그래픽 웹 브라우저 탄생
 - 미국 일리노이 대학교 대학생 Marc Andressen 개발
- ✓ 1994년 Netscape 회사에서 Netscape navigator 출시
 - Marc Andressen 대학 졸업 후 Netscape 회사 설립
 - 초기 브라우저는 HTML & CSS만을 이용하여 간단한 웹페이지 작성(Hypertext 지원)
- ✓ 1995년 W3C에서 HTML2.0 표준안 발표
- ✓ 그 후, 양방향 웹, 테이블, 수학식 등을 업데이트 하여 1999년도에 HTML4.0.1을 발표
- ✓ HTML을 XML형태로 사용할 수 있도록 xHTML을 별도로 분리하여 2001년도에 xHTML1.1 버전 발표
- ✓ 2018년 HTML5.3 마지막 버전 발표



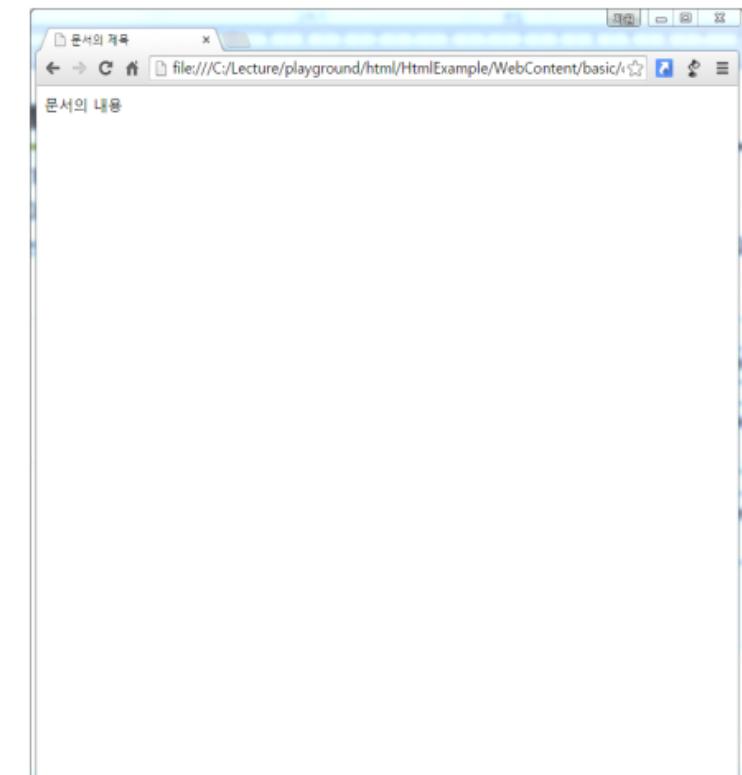
1.5 HTML 문서 구조

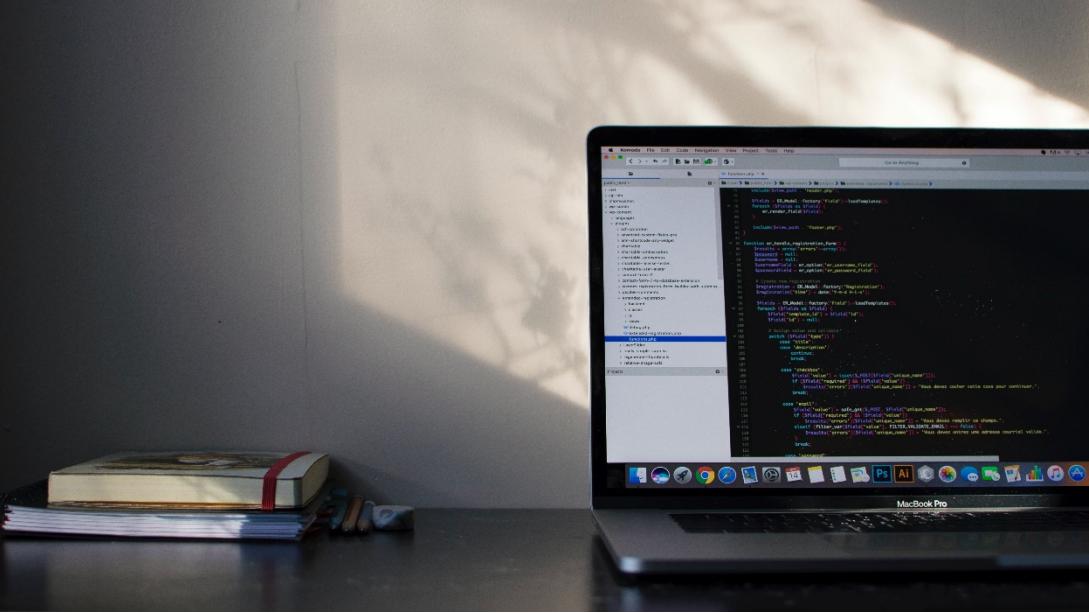
- ✓ <!DOCTYPE html> 태그는 현재 문서가 HTML 문서임을 선언
- ✓ 여는 태그(< >)와 닫는 태그(</ >)가 있으며, 태그 사이에 내용을 정의
- ✓ 여는 태그는 고유의 의미를 가지며, 웹 브라우저는 이 의미에 따라 문서를 화면에 렌더링
- ✓ 여는 태그, 닫는 태그, 내용을 통틀어 요소(Element)라고 한다

소스코드

```
<!DOCTYPE html>                                문서타입정의(DTD) : HTML 문서 정의
<html>
  <head>                                         문서머리글(HEAD) : 제목, 검색엔진에서 사용할 키워드, 기타 정보
    <title>문서의 제목</title>
  </head>

  <body>                                         문서본체(BODY) : 문서의 내용
    문서의 내용
  </body>
</html>
```





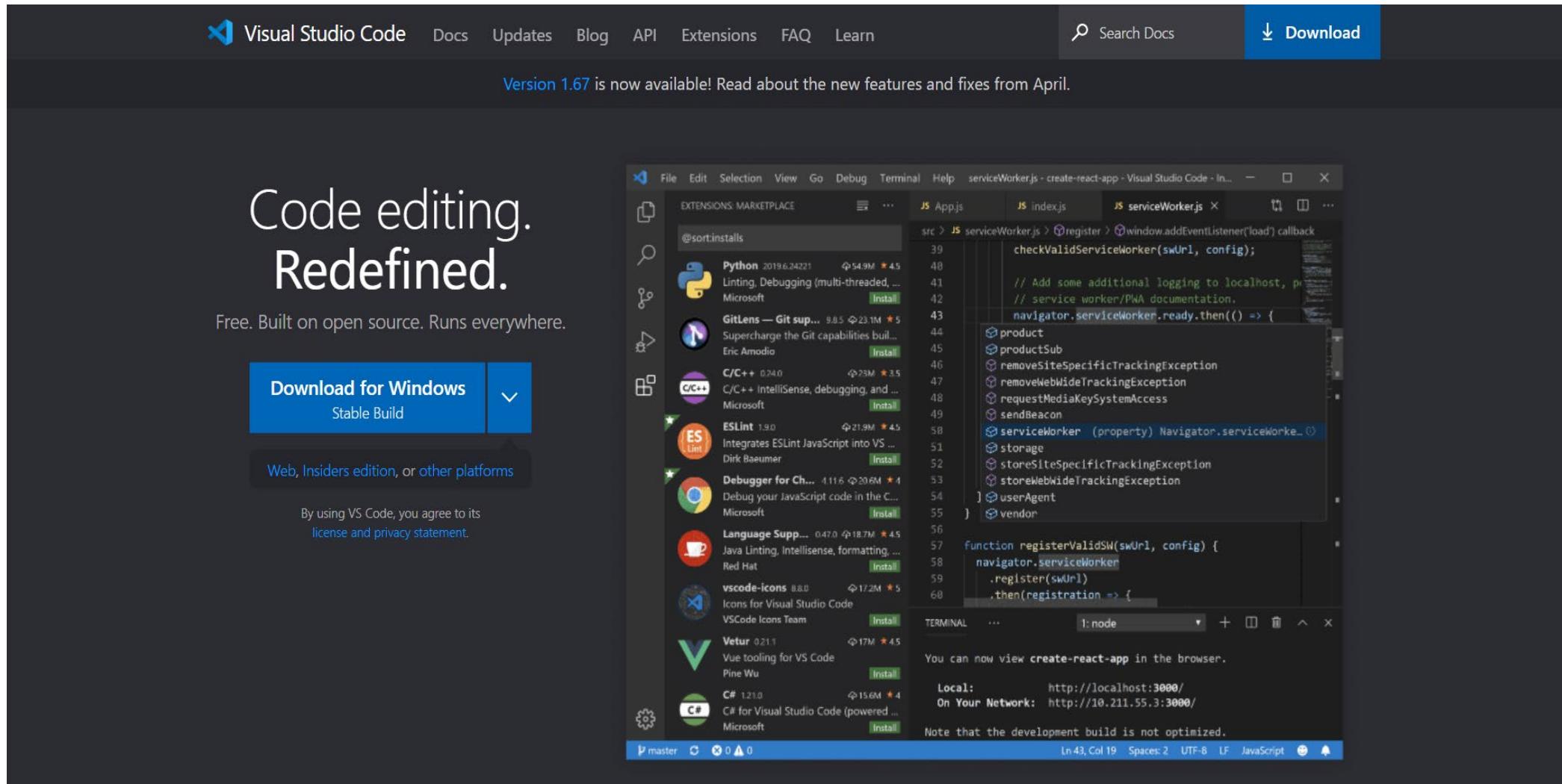
2. 개발 환경 구축

2.1 Visual Studio 설치

2.2 Visual Studio 설정

2.1 Visual Studio 설치

✓ 다운로드 : <https://code.visualstudio.com>

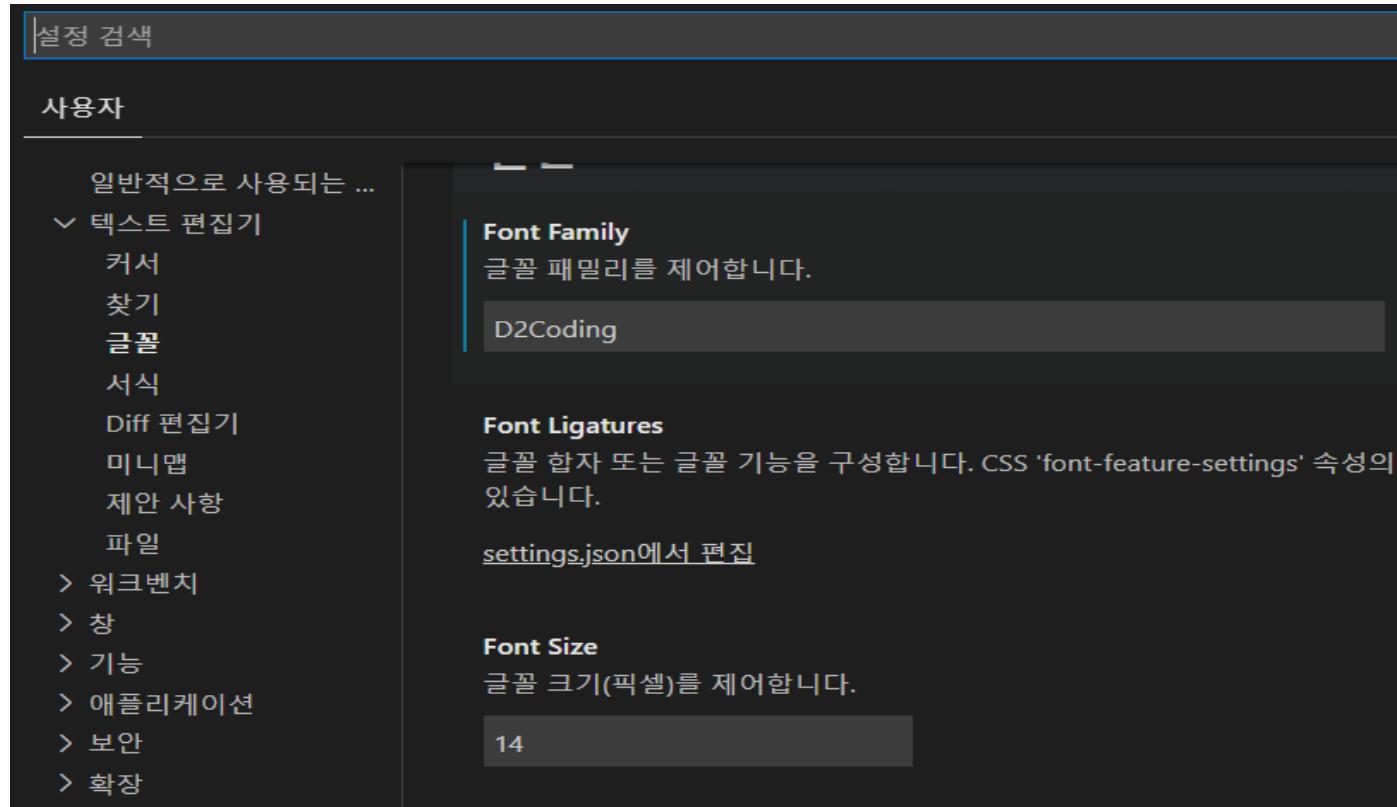


2.2 Visual Studio 설정

- ✓ 기본 언어를 한글로 변경 
 - 확장 기능에서 한글 언어 팩 설치

- ✓ 색 테마 변경 

- ✓ 폰트 변경





3. HTML 기초 다지기



- 3.1 HTML 문서 작성
- 3.2 HTML Tag 작성 규칙
- 3.3 HTML Element
- 3.4 HEAD Element
- 3.5 BODY Element

3.1 HTML 문서 작성

- ✓ <html>/<html> : 최상위 엘리먼트로 HTML 문서의 구조와 의미를 브라우저로 전달한다
- ✓ < head>/<head>
 - 엘리먼트는 UI적인 요소는 전혀 없고, 구글이나 네이버 등에서 검색할 때 사용할 타이틀이나 북마크 제목 등으로 보여질 메타데이터 정의
 - 현재 HTML 문서에 적용할 CSS, JavaScript 파일 연결
- ✓ 각 태그는 고유의 의미를 가지며, 웹 브라우저는 이 의미에 따라 문서를 화면에 렌더링

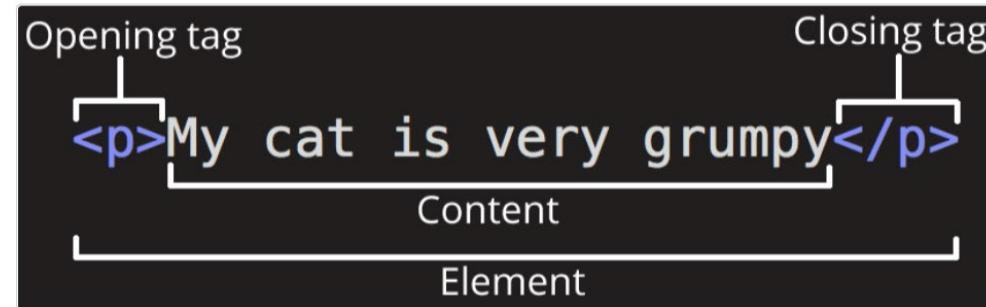
```
1  <!DOCTYPE html>
2  <html lang="ko">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>HTML 기본 구성 요소</title>
8  </head>
9  Lorem ipsum dolor sit amet consectetur, adipisicing elit. Cupiditate harum recusandae,
10 non eaque earum nulla totam, architecto et ab distinctio sit aliquid illo debitis?
11 Numquam voluptatem consequatur vel repudiandae praesentium!
12 <body>
13 </body>
14 </html>
```

3.2 HTML Tag 작성 규칙(1/2)

- ✓ HTML 태그는 대소문자 구별이 없다
- ✓ 시작태그(< >)와 종료태그(</ >)로 쌍을 이루며 중첩되지 않도록 한다

✓ HTML 엘리먼트는 크게 두가지 분류된다

- 블록 레벨 엘리먼트(Block-level elements)
 - 앞뒤 엘리먼트 사이에 새로운 줄(Line)을 만들고 나타난다
 - 일반적으로 페이지의 구조를 표현할 때 사용된다
 - (예) <p>, , <div>, <header> ...
- 인라인 엘리먼트(Inline elements)
 - 항상 블록 레벨 요소내에 포함되어 단락 같은 큰 범위에는 적용할 수 없고 문장, 단어 같은 작은 부분에 대해서만 적용한다
 - 새로운 줄(Line)을 만들지 않는다
 - (예) <a>, , ...



✓ 빈 엘리먼트

- 내용 없이 단일 태그(Single tag)를 사용하는 엘리먼트를 말한다
- (예)

3.2 HTML Tag 작성 규칙(2/2)

✓ 속성(Attribute)

- 속성은 엘리먼트에 추가적인 내용을 담고자 할 때 사용한다
- 하나 이상의 속성들을 추가할 수 있다

```
<p class="editor-note">My cat is very grumpy</p>
```

✓ HTML 공백

- 얼마나 많은 공백을 사용하든(하나 이상의 공백 문자 & 줄 바꿈) 단일 공백으로 나타낸다

✓ 특수문자 포함(Entity reference)

Literal character	Character reference equivalent
<	<
>	>
"	"
'	'
&	&

✓ HTML 주석

- <!-- 주석 내용 -->

3.3 HTML Element

- ✓ <html>태그는 HTML 문서 전체를 정의한다
- ✓ Head와 Body로 구성되어 있으며, 어떠한 카테고리에도 속하지 않는다
- ✓ HTML 문서는 머리부분(<HEAD>,</HEAD>)과 본체부분(<BODY>,</BODY>)으로 구성된다

올바른 예

```
<!DOCTYPE html>

<html>
  <head>
    <title>페이지 제목</title>
  </head>

  <body>
  ...
  </body>
</html>
```

3.4 HEAD Element

- ✓ <head> 태그는 브라우저에게 HTML 문서의 머리 부분임을 알려준다
- ✓ <title>, <meta>, <style>, <script>, <link> 태그를 포함할 수 있다
- ✓ <title> 태그는 문서의 제목을 의미하며, 브라우저의 제목 표시줄에 태그 내용이 출력된다
- ✓ <title> 태그 이외 다른 태그로 표현한 정보는 화면에 출력되지 않는다
- ✓ 문서의 작성자, 날짜, 키워드 등 브라우저의 본문에 나타나지 않는 문서에 대한 일반 정보를 나타낼 때 사용한다
- ✓ name과 content 속성을 이용하여 페이지 설명, 제작자, 키워드 등 다양한 정보를 나타낸다
- ✓ http-equiv 속성을 이용하여 문자 인코딩 설정 및 문서 이동, 새로 고침이 가능하다
- ✓ charset 속성을 이용하여 문서의 문자 인코딩 정보를 설정한다

사용 예

```
<head>
  <meta charset="utf-8">
  <title>강의 첫날</title>
  <meta name="author" content="nextree">
  <meta name="description" content="HTML Guide and Sample page">
  <meta name="keywords" content="HTML GUIDE, WEB">
  <style type="text/css"></style>
  <script type="text/javascript"></script>
  <link rel="stylesheet" href="style.css">
</head>
```

사용 예

```
<meta name="name" content="value">
  - NAME 요소 : description(문서의 요약), keywords(검색어 입력, 콤마로 분리), author(제작자) 등

<meta name="description" content="HTML Sample Page Description">
  - 페이지 설명. 검색엔진 로봇이 수집

<meta name="keywords" content="HTML, meta, tag">
  - 페이지의 키워드를 ,로 구분해서 나열, 검색엔진 로봇이 수집

<meta name="author" content="jiwonpark">
  - 페이지 제작자 정보

<meta http-equiv="refresh" content="30">
  - HTTP-EQUIV 요소 : Refresh(문서를 자동으로 업데이트), content-Type(인코딩 설정) 등

<meta charset="UTF-8">
  - 인코딩 정보를 설정
```

3.5 BODY Element

- ✓ 웹 브라우저에 보여질 문서의 내용을 작성한다
- ✓ id 속성을 이용하여 문서 내에서 태그를 유일하게 식별할 수 있다(id 속성은 중복되지 않아야 한다)
- ✓ class 속성을 이용하여 여러 태그에 공통적인 특성(예:CSS)을 부여할 수 있으며 이 속성은 중복이 가능하다

사용 예

```
<!DOCTYPE html>
<html>
  <head>
    <title>두번째 html 예제</title>
  </head>
  <body>
    <h2>애국가</h2>
    <span class="song">동해 물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라 만세</span>
    <br>
    <span class="song"><q>남산 위에 저 소나무, 철갑을 두른 듯</q>바람서리 불변함은 우리 기상일세</span>
    <br>
    <span class="song">가을 하늘 공활한데 <strong>높고 구름 없이</strong> 밝은 달은 우리 가슴 일편단심일세</span>
    <br>
    <span class="song">이 기상과 이 맘으로 충성을 다하여 괴로우나 즐거우나 <mark>나라 사랑하세</mark></span>
    <p id="repeat">
      - 후렴 -<br>
      무궁화 삼천리 화려강산 대한사람, 대한으로 길이 보전하세
    </p>
  </body>
</html>
```



4. HTML Markup Elements

-
- 4.1 포맷팅 Element
 - 4.2 목록 Element
 - 4.3 테이블 Element
 - 4.4 링크 Element
 - 4.5 이미지 Element
 - 4.6 프레임 Element
 - 4.7 폼 컨트롤 Element

4.1 포맷팅 Element(1/2)

- ✓ HTML은 텍스트에 적용하여 특별한 의미를 갖는 HTML 포맷팅 요소를 제공한다
- ✓ 포맷팅 요소에는 화면에는 동일하게 출력되지만 각 요소가 가진 의미가 태그들이 있다
- ✓ 와 은 모두 텍스트를 굵게 표현하지만, 은 텍스트를 강조한다
- ✓ <cite>는 인용의 출처를 나타내고, 은 텍스트를 강조한다
- ✓ 은 텍스트를 강조하는 의미를 가지며 텍스트를 굵게 출력한다
- ✓ <abbr>은 생략된 약어를 나타낼 때 사용하며 title 속성을 이용하여 원래 의미를 보여줄 수 있다

태그명	설명
<abbr>	생략된 약어 표시(WWW, HTTP)
<address>	연락처 정보 표시
	굵은 글씨로 표시
<blockquote>	긴 인용문구 표시, 좌우로 들여쓰기가 됨
<q>	짧은 인용문구 표시, 좌우로 따옴표가 붙음
	문서 버전관점에서 삭제된 문장을 표시
<ins>	문서 버전관점에서 삽입된 문장을 표시

태그명	설명
<mark>	특정 문자열을 강조. 화면에는 하이라이팅 됨
<pre>	공백, 줄 바꿈 등 입력된 그대로 화면에 표시
	특정 문자열을 강조. 화면에는 굵은 글씨로 표시
<small>	작은 글자로 표시
<sub>	아래 첨자로 표시
<sup>	위 첨자로 표시
<cite>	인용 또는 다른 자원 참조를 나타냄. 이탤릭체로 표시

4.1 포맷팅 Element(2/2)

사용 예

```
As <cite>Harry s.Truman</cite> said,  
<EM>More information</EM> can be found in <cite>[ISO-0000]</cite>. <br/>  
please refer to the following reference number in future  
correspondence:<STRONG>1-234-55</STRONG><br/>  
  
<abbr title="World Wide Web">WWW</abbr>  
<abbr lang="fr" title="Soci&eacute; Nationale des Chemins de Fer">SNCF</abbr>  
<abbr lang="es" title="Do&ntilde;a">Do&ntilde;a</abbr>  
<abbr title="Abbreviation">abbr</abbr>  
<br/><br/>  
  
<code>  
  &lt;SCRIPT type="text/javascript"&gt;<br/>  
  &lt;!--<br/>  
    document.write("test");<br/>  
  //--&gt;<br/>  
  &lt;/SCRIPT&gt;<br/>  
</code>
```

실행결과

As Harry s. Truman said, More information can be found in [ISO-0000].
please refer to the following reference number in future correspondence: 1-234-55
WWW SNCF Doña abbr

```
<SCRIPT type="text/javascript">  
   
</SCRIPT>
```

4.2 목록 Element(1/3)

- ✓ HTML은 목록을 표현하는 여러 기능을 제공한다
- ✓ 목록 태그는 하나 이상의 하위 태그를 포함한다
- ✓ 목록 태그는 각 항목을 들여쓰기로 표시한다
- ✓ 번호 또는 심볼을 이용해서 목록을 표시할 수 있다

태그명	설명
	번호 없는 목록을 표시. 항목 앞에 심볼을 표시
	번호 있는 목록을 표시. 숫자, 알파벳, 로마숫자 등으로 표시 가능
	목록 항목 이나 태그 하위에서 사용되면서 각 항목을 나타냄
<dl>	용어 정의와 설명에 대한 내용을 목록화해서 표시
<dt>	용어 목록의 정의 부분을 나타냄
<dd>	용어 목록의 설명 부분을 나타냄

4.2 목록 Element(2/3)

실행결과

- HTML
 - JavaScript
 - CSS
 - jQuery

1. HTML
2. JavaScript
3. CSS
4. Web

10. HTML
11. JavaScript
12. CSS
13. Web

- A. HTML
- B. JavaScript
- C. CSS
- D. Web

- i. HTML
- ii. JavaScript
- iii. CSS
- iv. Web

소스코드

```
<ul>
  <li>HTML</li>
  <li style="list-style-type:circle;">JavaScript</li>
  <li style="list-style-type:square;">CSS</li>
  <li style="list-style-type:disc;">jQuery</li>
</ul>

<ol>
  <li>HTML </li>
  <li>JavaScript </li>
  <li>CSS </li>
  <li>Web </li>
</ol>

<ol start="10">
  <li>HTML </li>
  <li>JavaScript </li>
  <li>CSS </li>
  <li>Web </li>
</ol>

<ol style="list-style-type:upper-alpha;">
  <li>HTML </li>
  <li>JavaScript </li>
  <li>CSS </li>
  <li>Web </li>
</ol>

<ol style="list-style-type:lower-roman;">
  <li>HTML </li>
  <li>JavaScript </li>
  <li>CSS </li>
  <li>Web </li>
</ol>
```

4.2 목록 Element(3/3)

실행결과

URL

인터넷상의 각종 자원의 위치를 나타내는 주소

HTML

하이퍼텍스트 기능을 가진 문서를 만드는 언어

1. WWW와 HTML

HTTP(Hyper Text Transmission protocol)

HTTP는 WWW 서비스에서 클라이언트와 서버 사이에서 통신할 때 사용하는 통신 프로토콜(protocol)

WWW브라우저

브라우저는 클라이언트에 설치되어 서버로부터 제공받는 정보를 사용자를 위해 재구성하는 역할을 하는 소프트웨어

2. HTML 기본형식

HTML 태그

태그는 HTML문서의 모양과 행동양식을 정해주는 하나의 명령

태그 속성

태그 속성은 태그가 실제 문서를 표현할 때 필요한 여러가지 환경들을 설정함

소스코드

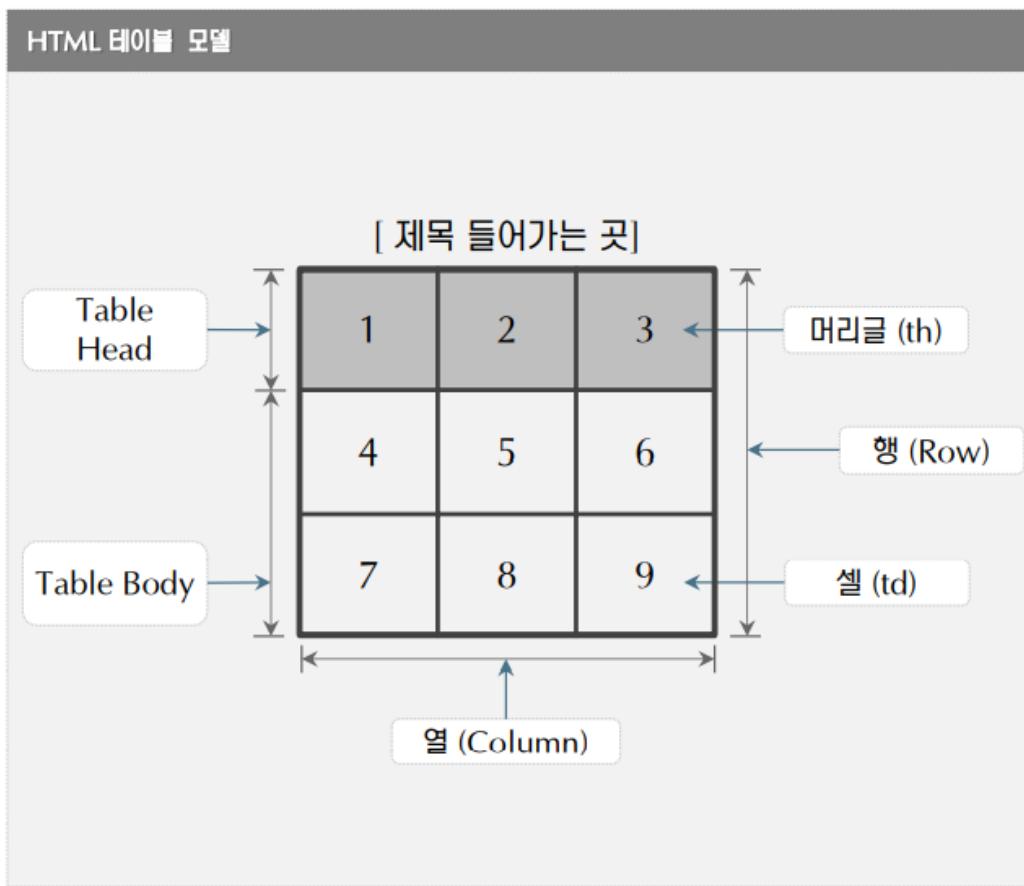
```
<dl>
  <dt>URL</dt>
    <dd>인터넷상의 각종 자원의 위치를 나타내는 주소</dd>
  <dt>HTML</dt>
    <dd>하이퍼텍스트 기능을 가진 문서를 만드는 언어</dd>
</dl>

<ol>
  <li>WWW와 HTML
    <dl>
      <dt>HTTP(Hyper Text Transmission protocol)</dt>
        <dd>HTTP는 WWW 서비스에서 클라이언트와 서버사이에서 통신할 때 사용하는 통신
        프로토콜(protocol)</dd>
      <dt>WWW브라우저</dt>
        <dd>브라우저는 클라이언트에 설치되어 서버로부터 제공받는 정보를 사용자를 위해
        재구성하는 역할을 하는 소프트웨어</dd>
    </dl>
  </li>

  <li>HTML 기본형식
    <dl>
      <dt>HTML 태그</dt>
        <dd>태그는 HTML문서의 모양과 행동양식을 정해주는 하나의 명령</dd>
      <dt>태그 속성</dt>
        <dd>태그 속성은 태그가 실제 문서를 표현할 때 필요한 여러가지 환경들을 설정함
    </dd>
  </li>
</ol>
```

4.3 테이블 Element(1/4)

- ✓ 기본적인 테이블 구성은 <table><tr><td> 요소를 사용하여 구성한다
- ✓ 행 그룹요소인 <thead>, <tfoot>, <tbody> 요소를 사용하여 행들을 그룹화할 수 있다
- ✓ <colgroup>과 <col> 요소는 열들을 그룹화하기 위해 제공된다
- ✓ 테이블의 셀은 머리글(<th>)이나 내용(<td>)을 가질 수 있다



HTML 테이블 소스코드

```
<table>
  <caption>[제목 들어가는 곳]</caption>
  <thead>
    <tr>
      <th>1</th>
      <th>2</th>
      <th>3</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>4</td>
      <td>5</td>
      <td>6</td>
    </tr>
    <tr>
      <td>7</td>
      <td>8</td>
      <td>9</td>
    </tr>
  </tbody>
</table>
```

4.3 테이블 Element(2/4) – 테이블 스타일(style)

- ✓ <table>, <tr>, <td> 태그에는 테이블에 스타일을 적용하는 다양한 속성들이 있다
- ✓ frame 속성은 테이블의 특정 선을 보여줄지를 결정하며, rules 속성은 셀과 셀사이의 줄이 나타날 것인가를 지정한다
- ✓ 이 밖에도, 테이블 정렬을 위한 align 속성, 그리고 background, bgcolor, border 속성 등이 있다
- ✓ 그러나, HTML5 부터는 이러한 속성들을 더 이상 지원하지 않는다. 스타일은 CSS를 사용하여 적용한다

테이블 속성을 사용한 테이블 스타일 적용 [HTML5 부터 미지원]

```
<table cellpadding="3"
       cellspacing="2"
       bordercolorlight="black"
       bordercolordark="lightskyblue"
       bgcolor="steelblue"
       border="1"
       width="140"
       height="50" >

    <tr bgcolor="green">
        <td>A</td>
        <td>B</td>
        <td>C</td>
    </tr>

    <tr bgcolor="lightskyblue">
        <td>1</td>
        <td>2</td>
        <td>3</td>
    </tr>
</table>
```

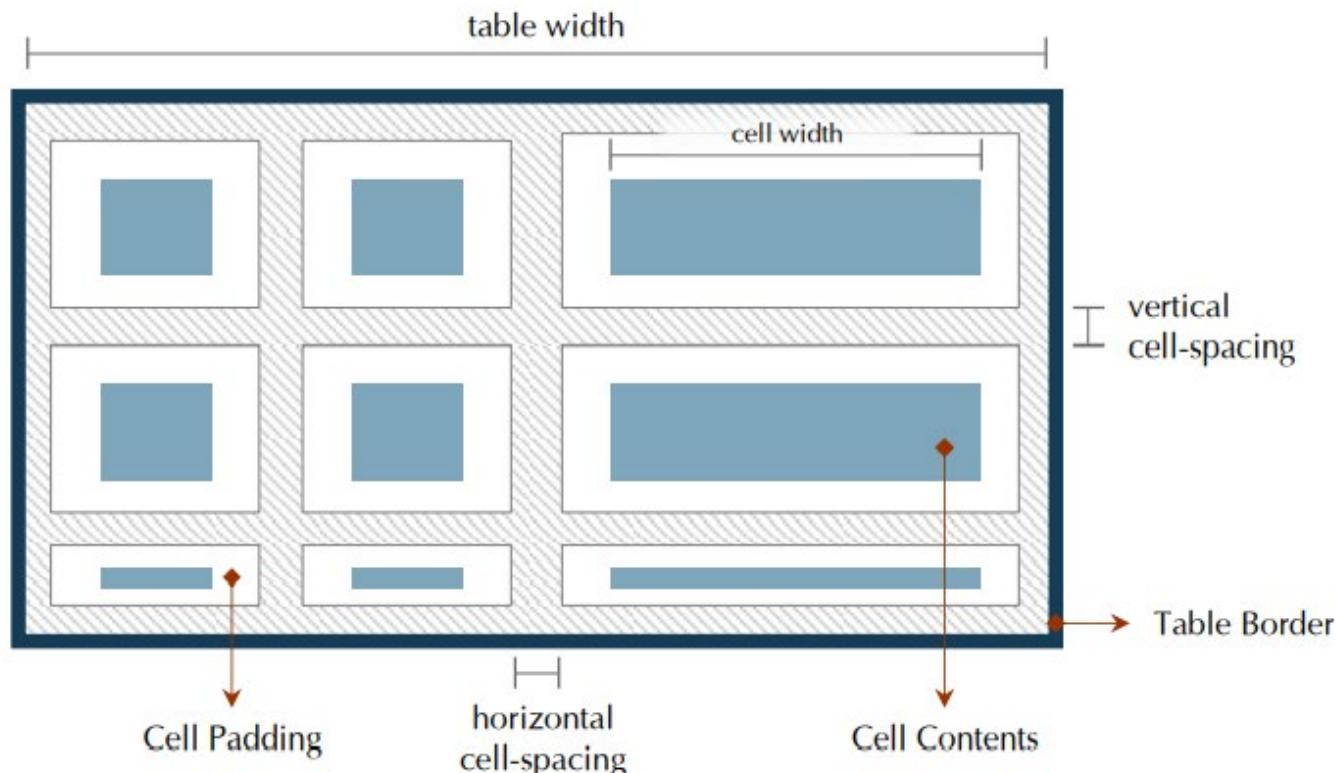


HTML4.01 까지는 테이블 스타일을 위한 속성이 제공되었지만 HTML5 부터는 CSS를 사용하여 스타일을 적용한다

A	B	C
1	2	3

4.3 테이블 Element(3/4) – 테두리(border)

- ✓ 테이블 셀(Cell) 테두리와 관련된 여러 가지 속성이 존재한다
- ✓ Cell-Spacing은 테이블 Cell과 Cell 사이 공간을 말한다
- ✓ Cell-Padding은 Cell 외곽과 Cell 컨텐츠 사이 공간을 말한다
- ✓ HTML5 부터는 테두리 스타일 관련 속성을 지원하지 않고, CSS를 사용한다



4.3 테이블 Element(4/4) – 셀 병합

- ✓ 시간표와 같은 테이블을 작성할 때, 행 또는 열을 병합해야 하는 경우가 있다
- ✓ HTML 테이블의 <td> 요소에는 셀(Cell)을 병합하기 위한 두 개의 속성이 있다
- ✓ colspan 속성은 두 개 이상의 열(Column)을 하나로 합치기 위해 사용한다
- ✓ rowspan 속성은 두 개 이상의 행(Row)을 하나로 합치기 위해 사용한다

1	2
3	4

```
<table>
  <tr>
    <td>1</td>
    <td>2</td>
  </tr>
  <tr>
    <td>3</td>
    <td>4</td>
  </tr>
</table>
```

1	2
3	4

```
<table>
  <tr>
    <td rowspan="2">1</td>
    <td>2</td>
  </tr>
  <tr>
    <td>4</td>
  </tr>
</table>
```

1	1	2
3	4	

```
<table>
  <tr>
    <td colspan="2">1</td>
  </tr>
  <tr>
    <td>3</td>
    <td>4</td>
  </tr>
</table>
```

1	2
3	4

```
<table>
  <tr>
    <td>1</td>
    <td rowspan="2">2</td>
  </tr>
  <tr>
    <td>3</td>
  </tr>
</table>
```

4.4 링크 Element(1/2) – Anchor

- ✓ <a>태그를 사용하여 하나의 문서에서 다른 문서로 연결하기 위해 사용한다
- ✓ href 속성은 하이퍼링크를 클릭했을 때 이동할 문서의 URL이나 문서의 책갈피를 지정하기 위해 사용한다
- ✓ target 속성은 하이퍼링크를 클릭했을 때 현재 윈도우 또는 새로운 윈도우에서 이동할지 지정하기 위해 사용한다

소스코드

```
<body>
  <h2>과목안내</h2>
  <a href="javaFundamental.html">JAVA 프로그래밍 기초</a><br>
  <a href="javaIo.html">JAVA IO</a><br>
  <a href="javaNetworking.html">JAVA Networking</a><br>
  <a href="jdbc.html" target="_self">JDBC</a><br>
  <a href="jspServlet.html"
    target="_self">JSP/ServLet</a><br>
  <h2>참고사이트</h2>
  <a href="http://www.nextree.co.kr"
    target="_blank">넥스트리</a><br>
  <a href="http://www.apache.org"
    target="_blank">Apache Software Foundation</a><br>
</body>
```



target 속성값 중 _self 는 현재 윈도우에서 다른 문서로 이동하고,
_blank는 새로운 윈도우 또는 새로운 탭에서 다른 문서로 이동

실행결과

The screenshot shows the execution result of the provided HTML code. On the left, the '과목안내' page lists several links:

- Java 프로그래밍 기초
- Java IO
- Java Networking
- JDBC
- JSP/Servlet

Below these, there is a '참고사이트' section with links to:

- 넥스트리
- Java 설치
- Apache Software Foundation

An arrow points from the 'Java IO' link to a larger screenshot on the right. This screenshot shows the 'Java 프로그래밍 기초 과목 안내' page, which contains the same list of links as the original page, plus an additional numbered list:

- Java 소개
- Java 문법
- 객체지향 프로그래밍 I
- 객체지향 프로그래밍 II
- 컬렉션 프레임워크

At the bottom of the screenshot, there is a mobile device preview showing a different website for 'NEXTREE'.

4.4 링크 Element(2/2) – link

- ✓ <link> 태그를 사용하여 문서와 외부 자원을 연결하기 위해 사용한다
- ✓ <head> 위치에 정의하며 여러 자원을 연결할 수 있다
- ✓ rel 속성은 현재 문서와 연결된 문서 사이의 연관관계를 지정하기 위해 사용한다
- ✓ href 속성은 연결된 문서의 위치를 지정하기 위해 사용한다

소스코드

```
<head>
  <link rel="stylesheet" href="style/css/basic.css">
  <link rel="stylesheet" href="style/css/spring.css">
</head>
```



주로 스타일시트(CSS)를 연결하기 위한 용도로 사용합니다.
이 부분은 4장 CSS소개 부분에서 살펴봅니다.

4.5 이미지 Element

- ✓ `img` 태그를 사용하여 이미지를 문서에 삽입하기 위해서 사용한다
- ✓ `src` 속성은 이미지 경로를 지정하기 위해 사용한다(상대경로, URL 모두 가능)
- ✓ `height`, `width` 속성은 이미지 사이즈를 지정하기 위해 사용한다
- ✓ `alt` 속성은 이미지를 표시할 수 없을 때 화면에 대신하여 보여질 텍스트를 지정한다

소스코드

```

```

```

```



width와 height 속성값은 픽셀단위로 지정합니다.

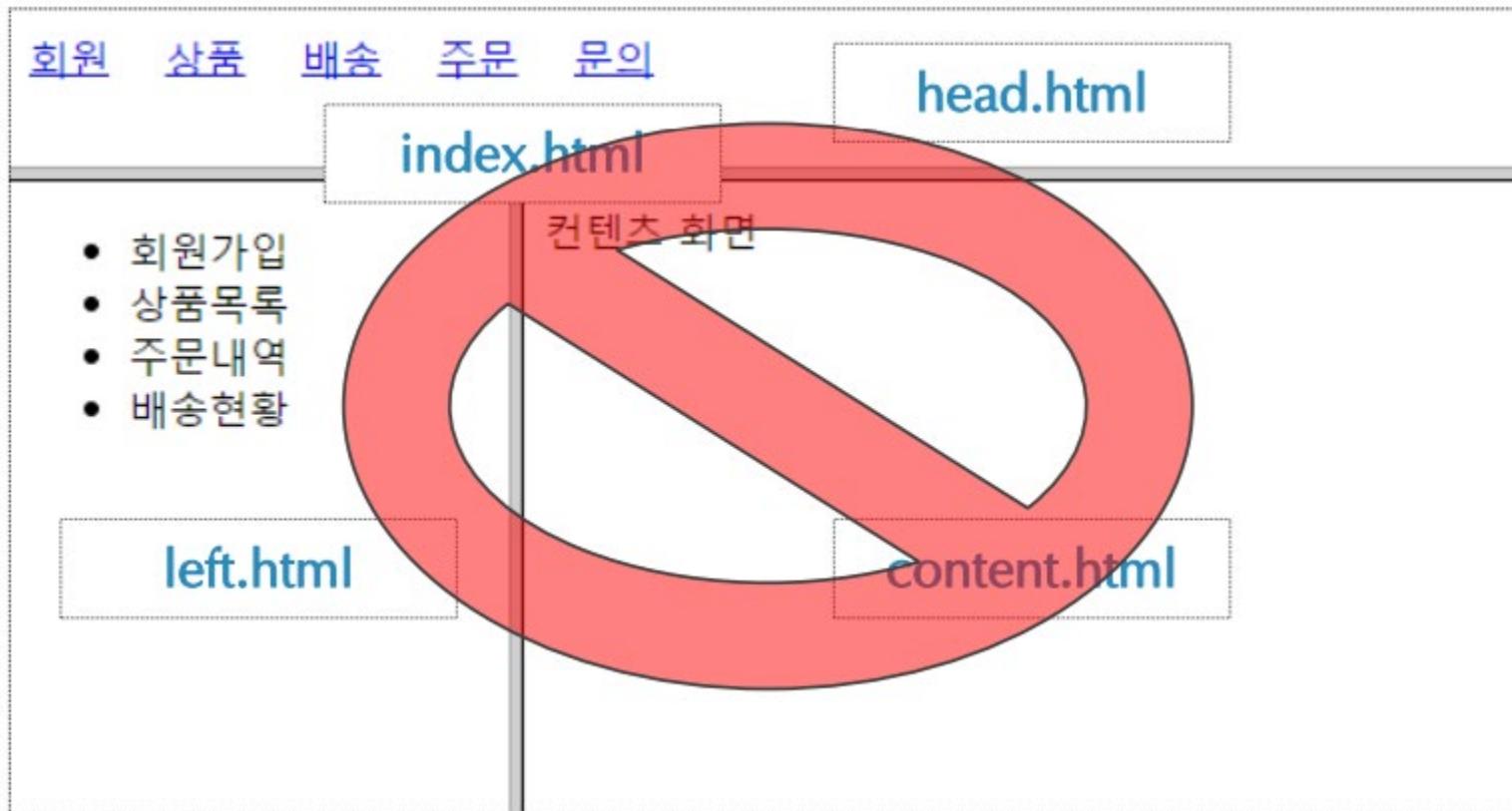
실행결과



주변은 눈으로 하얗게 뒤덮여있고, 나
무들은 하늘로 곧게 뻗어있습니다.

4.6 프레임 Element(1/2)

- ✓ 한 문서에 다른 문서를 포함 시기위해 사용한다
- ✓ 화면을 분할하여 top, left, main과 같은 형식으로 구성하고 각각 별도의 html파일로 관리하고자 할 때 사용한다
- ✓ 프레임 요소는 HTML5에서는 더 이상 지원하지 않는다
- ✓ 인라인 프레임요소인 iframe 요소만 지원한다



4.6 프레임 Element(2/2) – iframe

- ✓ **frameset**처럼 전체 화면을 분할하는 것이 아닌 화면의 일부분에 다른 문서를 포함한다
- ✓ **height, width** 속성은 프레임 사이즈를 지정하기 위해 사용한다
- ✓ **src** 속성은 포함시킬 외부 문서의 경로를 지정한다(상대경로, URL 모두 가능)
- ✓ **name** 속성은 프레임의 이름을 지정하기 위해 사용한다

소스코드

```
<body>
  <h2>iframe 요소 사용</h2>
  <a href="../link/javaFundamental.html"
      target="frameContent">Java 기초</a>
  <a href="../link/javaIo.html"
      target="frameContent">Java IO</a>
  <a href="../link/javaNetworking.html"
      target="frameContent">Java 네트워크</a>
  <a href="../link/jdbc.html"
      target="frameContent">JDBC</a>
  <a href="../link/jspServlet.html"
      target="frameContent">JSP와Servlet</a>
  <br><br>
  <iframe src="../link/javaFundamental.html"
          name="frameContent"
          width="500" height="200">
  </iframe>
</body>
```

실행결과

iframe 요소 사용

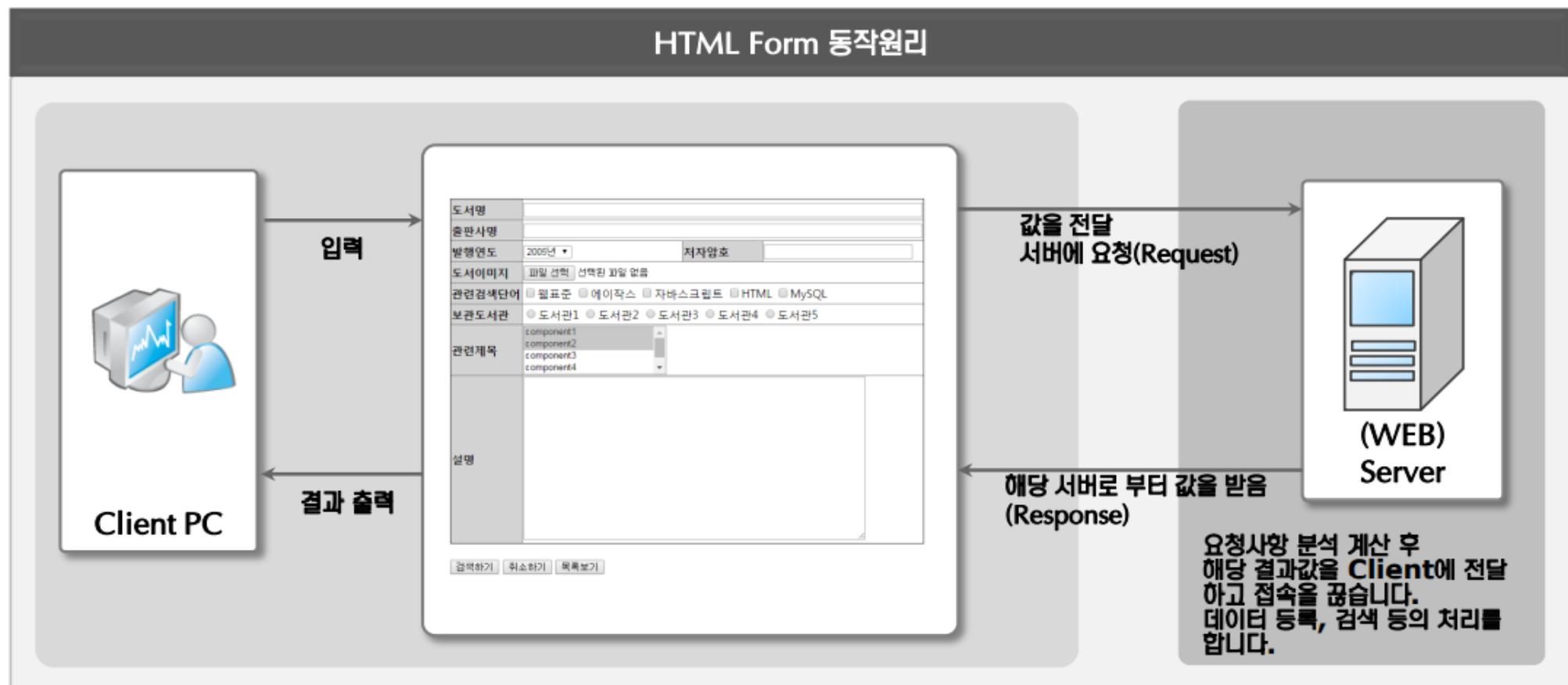
[Java 기초](#) [Java IO](#) [Java 네트워크](#) [JDBC](#) [JSP와Servlet](#)

JSP/Servlet 프로그래밍 과목 안내

1. Web 기본개념
2. Servlet
3. JSP(Java Server Pages)
4. Java 웹 프로그래밍 I
5. Java 웹 프로그래밍 II
6. 프레임워크 기반 웹 개발

4.7 폼 컨트롤 Element[1/7]

- ✓ 사용자로부터 데이터를 입력 받아 서버에서 전송하기 위한 용도로 사용한다
- ✓ 사용자의 요청에 따라 서버에 HTML 폼을 전달합니다(회원가입 양식, 검색 양식 등)
- ✓ 사용자는 HTML 폼에 적절한 데이터를 입력한 후 서버로 전송한다(Submit)
- ✓ 서버는 사용자의 요청을 분석한 후 데이터를 등록하거나, 원하는 데이터를 조회하여 결과를 다시 반환한다



4.7 폼 컨트롤 Element[2/7]

- ✓ 사용자가 입력하기 위한 컨트롤 요소들은 모두 <form> 태그 하위에 위치해야 서버로 전송 된다
- ✓ 각 컨트롤 요소마다 텍스트 입력, 버튼 클릭 등 다양한 형식으로 입력을 받는다
- ✓ <form> 태그의 action 속성은 데이터를 전송할 서버의 URL을 지정하기 위해 사용한다
- ✓ <form> 태그의 method 속성은 데이터의 전송방식을 지정하기 위해 사용한다(GET, POST 등)

태그명	설명
<form>	사용자에게 입력 받을 항목을 정의. form태그 내부에 여러 개의 컨트롤 요소를 포함
<input>	텍스트 박스, 체크박스, 라디오 버튼 등 사용자가 데이터를 입력할 수 있도록 함
<textarea>	여러 줄의 문자를 입력할 수 있도록 함
<button>	버튼을 표시
<select>	셀렉트 박스(드롭다운)를 표시
<optgroup>	셀렉트 박스(드롭다운)의 각 항목들을 그룹화 함
<option>	셀렉트 박스(드롭다운)의 각 항목들을 정의 함
<label>	마우스를 이용하여 <input>항목을 선택 시 편리함을 제공. for 속성을 이용하여 다른 컨트롤 요소와 텍스트를 연결시켜서 더 편리하게 선택할 수 있도록 함
<fieldset>	입력 항목들을 그룹화 함
<legend>	<fieldset>의 제목을 지정 함

4.7 폼 컨트롤 Element[3/7] – input

- ✓ <input> 태그는 type 속성에 따라 여러 가지 형태로 화면에 표시된다
- ✓ size 속성은 InputBox의 크기를 지정하기 위해 사용한다
- ✓ reset 타입은 사용자가 입력한 내용을 모두 초기화 하며 submit 타입은 사용자가 입력한 내용을 서버로 전송한다
- ✓ maxlength 속성은 InputBox에 사용자가 입력할 수 있는 문자열의 최대 길이를 지정하기 위해 사용한다

소스코드

```
<h2>인풋박스</h2>

아이디 : <input type="text" name="LoginId"> <br>
패스워드 : <input type="password" name="password">
<br>

전화번호 : <input type="text" name="job" size="15"
maxLength="11"> <br>

<input type="reset" value="취소">

<input type="submit" value="저장">
```

실행결과

인풋박스

아이디 :

패스워드 :

전화번호 :

4.7 폼 컨트롤 Element[4/7] – select

- ✓ <select> 태그는 셀렉트 박스(드롭다운)를 표시한다
- ✓ <option> 태그는 셀렉트 박스에 포함될 항목들을 정의 하기 위해 사용한다
- ✓ **selected** 속성은 여러 개의 항목 중 선택된 항목을 표시하기 위해 사용한다
- ✓ **value** 속성은 각 항목 값을 지정하기 위해 사용한다.
- ✓ 예를 들어 자바를 선택하면 이 셀렉트의 값은 JAVA가 된다

소스코드

```
<h4>셀렉트 박스(드롭다운)</h4>
<select name="subject" id="selSubject">
    <option value="JAVA">자바</option>
    <option value="HTML">마크업</option>
    <option value="Servlet" selected>서블릿</option>
    <option value="JSP">자바서버페이지</option>
    <option value="ORM">객체-디비 맵핑</option>
</select>
```

실행결과

서블릿

서블릿
자바
마크업
서블릿
자바서버페이지
객체-디비 맵핑

4.7 폼 컨트롤 Element[5/7] – textarea

- ✓ <textarea> 태그는 여러 줄을 입력할 수 있는 박스를 표시한다
- ✓ cols와 rows 속성은 텍스트 박스의 크기를 지정하기 위해 사용한다
- ✓ <textarea></textarea> 태그 사이의 문자열을 텍스트 박스에 표시한다
- ✓ disabled 속성은 화면에 표시는 하되 데이터를 수정할 수 없도록 비활성화 상태로 표시한다

소스코드

```
<h4>TextArea 여러줄 입력</h4>
<textarea cols="30" rows="8">기타 의견을 입력하시면 됩니다.</textarea>

<p>

<textarea cols="30" rows="8" disabled>기타 의견을
입력하시면 됩니다.</textarea>
```

실행결과

TextArea 여러줄 입력

기타 의견을 입력하시면 됩니다.

기타 의견을 입력하시면 됩니다.

4.7 폼 컨트롤 Element[6/7] – checkbox, radio button

- ✓ <input> 태그의 type 속성이 checkbox이면 체크박스를, radio이면 라디오 버튼을 표시한다
- ✓ checked 속성은 여러개의 항목 중 선택된 항목을 표시하기 위해 사용한다
- ✓ 라디오 버튼은 name 속성의 값이 모두 일치해야 한다(name 속성이 같은 항목들 중 한 개만 선택된다)
- ✓ 체크박스는 name 속성의 값과는 상관없이 다중 선택이 가능하다

소스코드

```
<h4>체크박스</h4>
<input type="checkbox" name="JAVA" value="자바">자바
<input type="checkbox" name="JDBC"
       value="DB">데이터베이스
<input type="checkbox" name="SPRING"
       value="스프링" checked>스프링
<br>

<h4>라디오버튼</h4>
<input type="radio" name="fruit" value="사과">사과
<input type="radio" name="fruit"
       value="포도" checked>포도
<input type="radio" name="fruit" value="멜론">멜론
```

실행결과

체크박스

자바 데이터베이스 스프링

라디오버튼

사과 포도 멜론

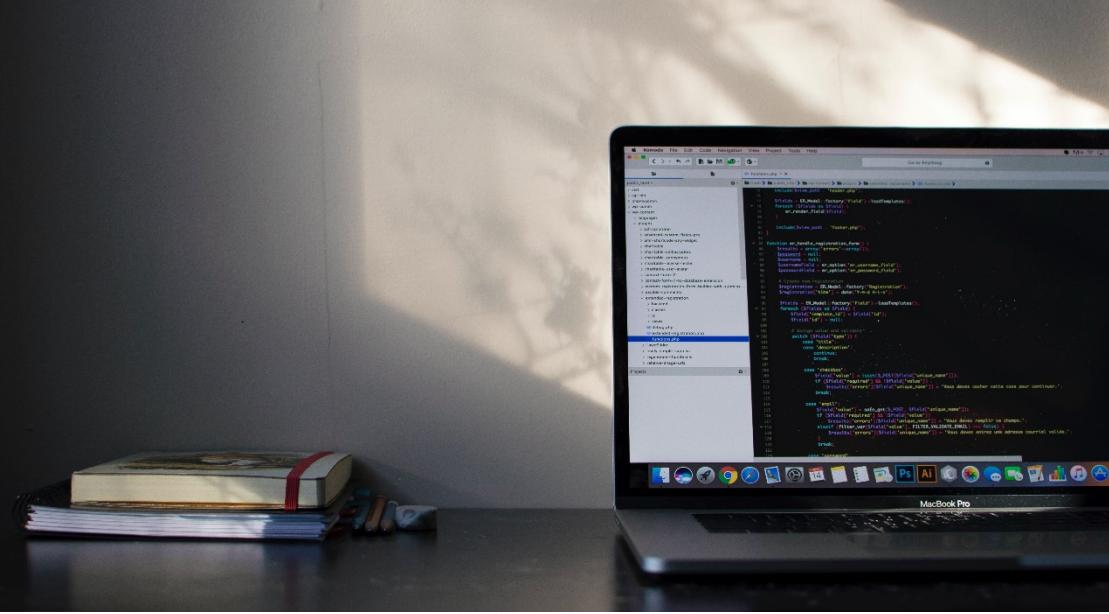
4.7 폼 컨트롤 Element[7/7] – 실습

- ✓ 폼 컨트롤 요소들을 조합하면 다양한 폼을 작성할 수 있다
- ✓ 라디오 버튼 생성시 다중선택이 되지 않도록 name속성의 같은 값으로 지정해야 한다
- ✓ 비밀번호는 사용자가 입력한 데이터가 화면에 *로 표시되도록 type 속성을 password로 지정해야 한다
- ✓ 취소 버튼을 클릭하면 입력 폼에 입력한 내용들이 모두 초기화 된다

실행결과

기본정보 입력

이름		
아이디		중복체크
비밀번호		
비밀번호확인		
이메일	@	선택하세요 ▾
연락처		
메일수신여부	<input checked="" type="radio"/> 받음	<input type="radio"/> 받지않음
관심분야	<input type="checkbox"/> 자바 <input type="checkbox"/> C++ <input type="checkbox"/> 파이썬 <input type="checkbox"/> 안드로이드	



5. Semantic Elements

5.1 HTML 문서 구조

5.2 Semantic 태그가 사용된 HTML5의 문서 구조

5.1 HTML 문서 구조

- ✓ HTML4.0.1 문서에서는 화면을 구성하는 구조는 대부분 <div> 태그를 이용하였고, 수많은 <div> 태그들을 구분하기 위해 id 속성을 이용했다

사용 예

```
<div id="header">  
  Header  
</div>  
<div id="content">  
  Content  
</div>  
<div id="footer">  
  Footer  
</div>
```

<div> 태그로 묶기만 하면 되었고 그 이름에는 제약이 없었기 때문에 홈페이지 제작자 자신이 나중에 보더라도 이해하기 어려웠을 뿐 아니라, 담당자가 바뀌게 되면 홈페이지 구조를 이해하는 데 많은 시간이 걸리곤 했다

5.2 Semantic 태그가 사용된 HTML5의 문서 구조(1/2)

✓ 시멘틱 웹이란?

- "의미론적인 웹"이라는 뜻으로, 기계가 이해할 수 있는 형태로 제작된 웹을 의미한다(사람도 이해)
- 웹의 창시자 '팀 베너스 리(Tim Berners-Lee)'에 의해 제안되었다
- 시멘틱 웹은 기계가 사람을 대신해서 웹 페이지의 정보를 이해하고, 우리에게 필요한 정보만 보여주거나 정보를 가공해서 우리가 필요로 하는 형태로 가공해주는 것을 의미한다

✓ HTML 4.0.1 버전

```
1 <div id="header"></div> cs
```

- 검색 엔진 로봇은 id가 header라고 적혀 있다고 해서 그 부분이 웹 페이지의 헤더에 해당하는지 알 수 없다

✓ HTML 5

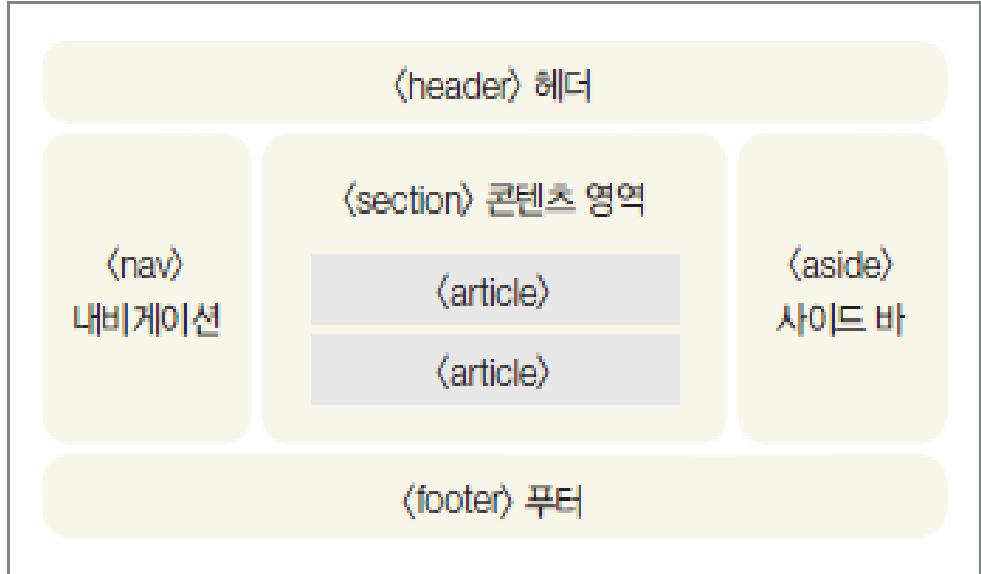
```
1 <header></header> cs
```

- <header>라는 태그를 사용해서 웹 페이지 헤더를 정의해주면 로봇도 이 부분이 웹 페이지의 헤더라는 것을 이해할 수 있다

✓ 다양한 시멘틱 태그들

```
<header>, <section>, <footer>, <article>, <aside>, <nav>, <address>, <item>, <figure>
<figcaption>, <summary>, <time>, <mark>, <main>, <details>
```

5.2 Semantic 태그가 사용된 HTML5의 문서 구조(2/2)



HTML5의 웹 표준 문서 구조

