

# Web Server Programming

- 담당교수 : 김재현
- 연구실 : 북악관 501호
- 전화 : 940-7740                      휴대폰 : 010-4741-6252
- Email : statsr@skuniv.ac.kr
- 교재 : **프로젝트로 배우는 자바 웹 프로그래밍**  
          **-황희정, 2014, 한빛아카데미**  
부교재 : JSP 웹프로그래밍  
          - 이성욱, 장종준, 2019, 21세기사
- 선수과목 : Java Programming



# Web Server Programming

- 강의목표 : 개발자의 입장에서 본 웹에 대한 기본개념과 HTML 기초를 배우고 Servlet의 원리와 JSP의 개발방법 및 Java Beans에 대해 다루고 JDBC와 데이터베이스 사용법과 함께 JSP에서 DB를 활용하는 방법에 대해 살펴본다. 주어진 문제에 대해 JSP실습을 통해 문제를 해결하는 능력을 배양하는 것을 목표로 한다.
- 강의 방법 :
  - (1)JSP의 개념과 문법에 대해 설명한다(40%)
    - 가.수업은 실시간 온라인으로 진행한다.
    - 나.매주 수업과 관련된 강의자료나 참고자료를 강의자료실에서 다운받아 준비하도록 한다.
    - 다.수업중 질의응답, 문제풀이, 응용 및 실습을 통해 수업에 적극 참여토록 유도한다.
  - (2) Eclipse를 이용한 실습을 통해 문제해결능력을 키우고 실무능력을 배양한다.(60%)

# Web Server Programming

- 평가 방법 :
- 중간고사:30%
- 기말고사:30%
- 과제:20%
- 출석:20%(매시간 호명하여 체크)
- 교과목별 해당 수업시간수의 1/4이상 결석시 낙제처리됨.(F)
- 수업과 관련된 의문사항은 이메일, 또는 전화를 통해 해결한다.

# 수업 내용 요약

## ① Web Programming에 대한 이해(1주)

웹 프로그래밍, 자바 웹 프로그래밍 개발 환경 구축

## ② JSP 프로그래밍(2~4주)

JSP의 기본구조, Servlet, Servlet과 JSP의 관계, JSP 기본 문법, JSP 내장 객체

## ③ JSP와 Database(5~7주)

JSP Java Beans class, JDBC(Java Database Connectivity)

<중간 고사> (8주. 4월 19일 예정)

## ④ Web Application Architecture와 JSTL(9~10주)

Tomcat Server 관리 기법, 표현 언어, Custom Tag, JSTL(Java Standard Tag Library)

## ⑤ Connection Pool(11주)

Connection Pool, Transaction

## ⑥ Listener와 Filter(12주)

## ⑦ Log와 Error(13~14주)

<기말 고사>

# Google Classroom

## 3tha6mf

1. skuniv.ac.kr 계정으로 Google에 로그인
2. Classroom.google.com으로 이동
3. 상단에서 추가 + > 수업 참여 클릭
4. 제공한 수업 코드 (3tha6mf) 입력하고 참여 클릭
5. 상단에서 수업을 클릭
6. 해당 과제 클릭
7. 과제 작성 후 파일(pdf 형식) 업로드



# 프로젝트로 배우는 자바 웹 프로그래밍

Servlet, JSP, JDBC

황희정 지음

## Chapter 01. 웹 프로그래밍에 대한 이해

# 목차

1. 네트워크, 인터넷, 웹
2. 웹 프로그래밍 언어와 주요 기술
3. 스마트 시대의 웹 프로그래밍

# 학습목표

- 웹 기반이 되는 인터넷의 기술적 배경을 이해한다.
- 대표적인 웹 프로그래밍 언어와 기술을 배운다.
- 새로운 인터넷 환경의 변화와 웹 프로그램 개발의 미래를 알아본다.



## 1. 네트워크

- 네트워크(Network)의 사전적 의미는 “전선이나 혈관, 통로 등으로 이루어진 망형 조직”.
- 방송사 네트워크, 인맥을 의미하는 휴먼 네트워크(Human Network) 등
- IT분야에서는 ‘컴퓨터와 컴퓨터를 연결해 주는 망’의 개념으로 컴퓨터 네트워크의 의미로 사용.
  
- **1990년대**
  - 컴퓨터와 컴퓨터가 네트워크로 연결되지 않음.
  - 자료 이동 시 플로피 디스켓 등을 이용함.
  
- **2000년 ~ 2010년**
  - 인터넷 보급으로 컴퓨터와 컴퓨터가 네트워크에 연결됨.
  - 이메일, 웹하드 등 인터넷 서비스를 활용한 자료 이동.
  
- **2010년 이후**
  - 4G LTE 등 고속 무선 이동 네트워크 보급, 스마트폰, 태블릿 등 모바일 기기 보급으로 개인이 언제 어디서나 네트워크에 연결되는 기기의 사용이 가능해 짐.
  - 클라우드 기반 스토리지 서비스를 통한 개인 및 그룹 자료 공유.

# 01. 네트워크, 인터넷, 웹

## TCP/IP

- TCP/IP(Transmission Control Protocol/Internet Protocol)는 컴퓨터 간에 통신할 수 있도록 만든 프로토콜.
- 인터넷의 기반 네트워크 시스템으로 하드웨어, 운영체제, 접속 매체에 관계없이 동작할 수 있는 개방형 구조.

응용 계층 — WWW, FTP, Telnet, SMTP와 같은 네트워크 프로그램으로 구성된다.

전송 계층 — 각 시스템을 연결하고 TCP 프로토콜을 이용하여 데이터를 전송한다.

인터넷 계층 — IP 프로토콜을 이용하여 데이터를 정의하고 경로를 배정한다.

물리 계층 — 실제 네트워크를 접근할 수 있도록 하는 물리적인 부분이다.

[그림 1-1] TCP/IP의 계층

## IP 주소

- IP 주소(IP Address)는 TCP/IP로 연결된 네트워크에서 각각의 컴퓨터를 구분하려고 사용하는 주소.
- 숫자로 구성되어 있고 "123.123.123.123" 과 같이 네 개로 구분된 10진수를 사용.

구분	범위	네트워크 수	사용 목적 / 네트워크당 주소
클래스 A	1.0.0.0 ~ 127.0.0.0	128	대형 통신망 / 16,777,214개
클래스 B	128.0.0.0 ~ 191.255.0.0	16,384	중형 통신망 / 65,534개
클래스 C	192.0.0.0 ~ 223.255.255.0	2,097,152	소형 통신망 / 256개
구분	범위	네트워크 수	사용 목적 / 네트워크당 주소
클래스 D	224.0.0.0 ~ 239.255.255.255	—	멀티 캐스트용
클래스 E	240.0.0.0 ~ 255.255.255.255	—	실험 목적 / 배포 중지

[표 1-1] IP주소 구분과 범위

# 01. 네트워크, 인터넷, 웹

## 2. 인터넷(Internet)

- 인터넷은 전 세계가 하나로 연결된 네트워크를 의미하는 고유명사로, Internet으로 표기.
- internet은 내부 네트워크를 의미.
- 인터넷은 네트워크 인프라이고 www, email 등은 인터넷 기반 서비스 임.
- 인터넷 ≠ 웹
- 대표적인 인터넷 기반 서비스

서비스 이름	기능	프로토콜	포트
웹(WWW)	웹 서비스	HTTP	80
이메일(Email)	이메일 서비스	SMTP/POP3/IMAP	25/110/143
FTP	파일 전송 서비스	FTP	21
Telnet	원격 로그인 서비스	TELNET	23
DNS	도메인 이름 변환 서비스	DNS	53
News	인터넷 뉴스 서비스	NNTP	119

[표1-2] 대표적인 인터넷 기반 서비스

### ■ 프로토콜

- 네트워크에 연결된 컴퓨터들 간의 통신 규약
- 인터넷 서비스들은 TCP/IP 4계층 중 응용계층에 해당함.

### ■ 포트

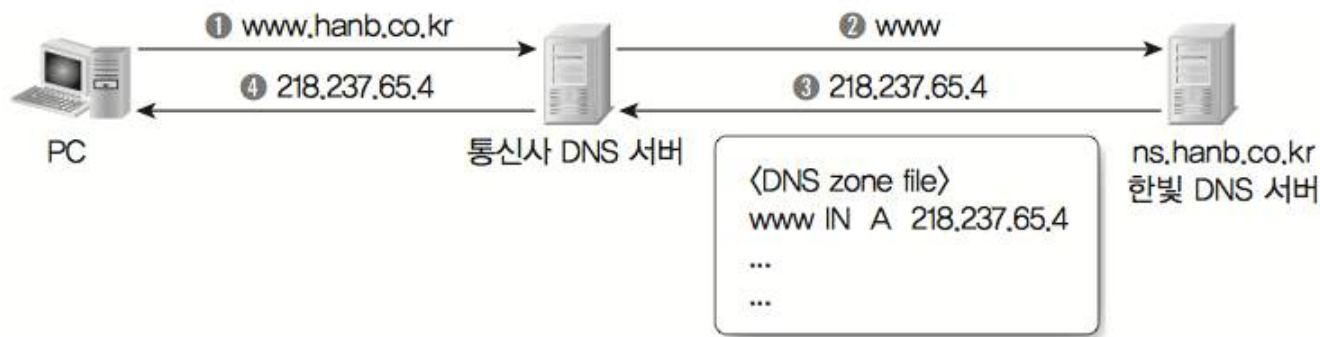
- 네트워크 서비스를 접속하기 위한 접점.
- 컴퓨터에서 여러 네트워크 서비스를 제공하는 경우 이를 구분하기 위한 용도로 사용.
- 은행의 구분된 업무 창구와 유사한 개념.

# 01. 네트워크, 인터넷, 웹

## ■ 도메인 네임 시스템

### ■ DNS(Domain Name System)

- 인터넷에 연결된 컴퓨터 구분을 위해 사용하는 IP주소 대신 알기 쉬운 이름 형태로 컴퓨터를 구분하기 위한 인터넷 서비스/시스템 중 하나.
- 호스트(컴퓨터) 이름(www) + 도메인 이름(hanb.co.kr) 형태로 구성됨.
- 예) [www.naver.com](http://www.naver.com), [www.daum.net](http://www.daum.net), [www.hanb.co.kr](http://www.hanb.co.kr) 등.
- 도메인 관리 기관을 통해 일정 비용을 지불하고 사용 가능.



[그림 1-2] DNS 처리 과정

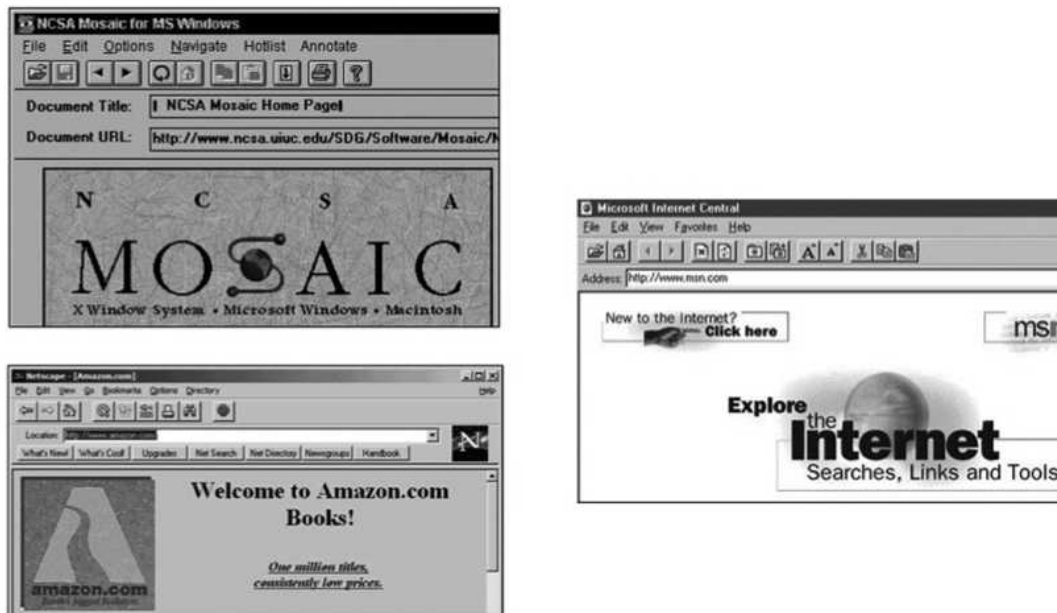
### ■ DNS 처리 과정

- ① 네트워크 서비스 제공 회사(KT, SK브로드밴드, LGU+ 등)의 DNS 서버에 도메인 이름을 요청한다.
- ② DNS 서버는 한빛미디어 DNS 서버인 ns.hanb.co.kr에 www라는 컴퓨터의 IP 주소를 요청한다.
- ③ 한빛미디어 DNS 서버는 DNS zone file에서 www 컴퓨터의 IP 주소를 찾아서 알려준다.
- ④ DNS 서버는 다시 요청한 클라이언트에 IP 주소를 알려준다.

# 01. 네트워크, 인터넷, 웹

## 3. 웹

- 월드 와이드 웹(World Wide Web, www)
- 인터넷에서 운영되는 서비스 중 하나로 많은 사람들이 '웹 = 인터넷'이라고 생각할 정도로 대표적인 인터넷 서비스임.
- 1989년 스위스에 있는 유럽물리입자연구소(CERN)의 팀 버너스리에 의해 개발됨.
- 흩어져 있는 연구자들이 손쉬운 방법으로 정보를 공유하기 위해 HTML이라는 간단한 마크업 언어를 통해 콘텐츠를 제공하고 HTTP라는 프로토콜을 사용해 TCP/IP 네트워크에서 사용하기 시작함.
- 웹 브라우저 소프트웨어와 인터넷에 연결만 되어 있으면 PC, 스마트폰, 태블릿 등 운영체제와 하드웨어 상관 없이 동일한 서비스 제공이 가능.



[그림 1-4] 초기 웹 화면

# 01. 네트워크, 인터넷, 웹

## ■ 웹 서버와 HTTP

### ■ 서버(Server)

- 네트워크에서 서비스를 제공하는 컴퓨터
- 웹 서버, FTP 서버, 파일 서버, 프린트 서버

### ■ 클라이언트(Client)

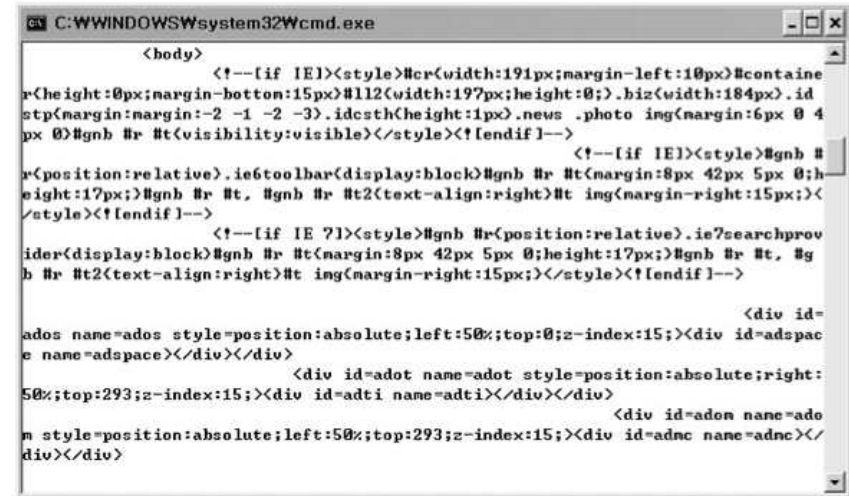
- 서비스를 이용하는 컴퓨터
- PC, 스마트폰, 태블릿 등

### ■ HTTP(Hyper Text Transfer Protocol)

- 웹 서비스에 사용되는 통신 규격
- 간단한 명령어와 헤더 규격으로 되어 있음

### ■ HTTP 프로토콜 체험

- 명령창 에서 telnet [www.naver.com](http://www.naver.com) 80 입력
- GET /index.html HTTP/1.0 입력 후 엔터
- 네이버 서버의 index.html 파일을 보내 달라는 요청임.
- 실제 index.html은 서버 설정에 따라 실제 파일이 아닐 수도 있음.



```
<body>
<!--[if IE]><style>#c<div id=
r<div id=adot name=adot style=position:absolute;right:
50%;top:293;z-index:15;><div id=adti name=adti></div></div>
<div id=adon name=ado
m style=position:absolute;left:50%;top:293;z-index:15;><div id=adnc name=adnc></
div></div>
```

[그림 1-5] HTTP GET 명령 결과



[그림 1-6] 웹 브라우저로 해석된 HTTP GET 명령 결과

# 01. 네트워크, 인터넷, 웹

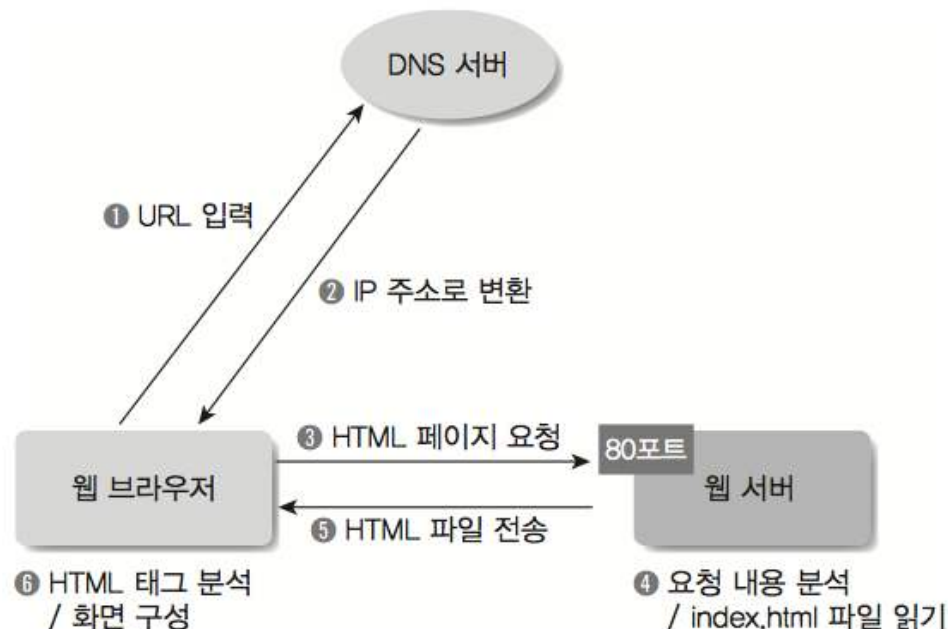
## ■ 웹 서비스의 동작 과정

### ■ 웹 서버 소프트웨어

- 서버에서 웹 서비스를 제공하는 소프트웨어
- 아파치(Apache), 마이크로소프트 IIS(Internet Information Server) 가 대표적임.

### ■ 클라이언트 소프트웨어

- 웹 서비스를 이용하기 위한 클라이언트 소프트웨어 → 웹 브라우저(Web Browser)
- 인터넷 익스플로러(Internet Explorer), 크롬(Chrome), 파이어폭스(Firefox), 애플 사파리(Safari) 등



- 1 웹 브라우저에서 `http://www.xxx.com/index.html`을 입력
- 2 `www.xxx.com` 도메인의 IP 주소를 DNS 서버로부터 받음
- 3 IP 주소의 해당 서버 80번 포트로 접속을 시도
- 4 웹 서버는 요청 내용을 분석하고 요청된 `index.html` 파일을 디스크에서 읽음.
- 5 웹 서버는 파일 내용을 텍스트 그대로 요청한 클라이언트에 전송.
- 6 웹 브라우저는 웹 서버에서 보내는 텍스트 내용 중 HTML 태그를 분석해 적절히 변환하여 화면을 구성.

[그림 1-7] 클라이언트와 서버간 동작 과정



## 02. 웹 프로그래밍 언어와 주요 기술

### 1. 웹 프로그램의 개요

#### ■ 일반적인 프로그램

- 컴퓨터에 설치 후 사용(스마트폰, 태블릿도 동일함)
- 기본적으로 해당 디바이스에서 프로그램이 실행되고 경우에 따라 필요한 데이터는 서버로부터 수신함.

#### ■ 웹 프로그램

- 별도의 설치 없이 서버에 접속하는 것만으로 필요한 기능/서비스를 이용할 수 있음.
- 프로그램은 서버에서 실행되고 실행 결과만 컴퓨터의 브라우저를 통해 보여짐.
- 데이터 입력, 메뉴선택, 버튼 클릭 등 사용자와의 상호작용 처리를 위해 클라이언트에서 처리해야 되는 프로그램적인 요소도 있음.
- 웹 프로그램은 서버와 클라이언트의 협력에 의해 구현됨.
- 웹 프로그램 개발을 위해서는 서버 프로그래밍 기술과 클라이언트 프로그래밍 기술을 모두 알아야 함.

구분	종류
클라이언트 기술	HTML, 자바스크립트, CSS
서버 기술	서블릿, JSP, ASP.Net, PHP
클라이언트/서버 공통 기술	자바, C, C#.Net

[표 1-3] 대표적인 웹 개발 기술



## 02. 웹 프로그래밍 언어와 주요 기술

### ■ 클라이언트 기술

- 기본적으로 브라우저에 의해 처리되는 요소
- 화면 구성, 스타일, 동적 이벤트처리 등 사용자 상호작용 담당
- **HTML(Hyper Text Markup Language)**
  - 웹 서비스를 표현하기 위해 사용하는 언어.
  - <HTML></HTML> 과 같은 마크업 구조.
  - 최근 HTML5 가 널리 사용되고 있음.
- **자바스크립트(JavaScript)**
  - 자바와 유사한 문법구조를 제공하는 웹 클라이언트 개발 언어. 실제 자바와는 무관함.
  - 웹 브라우저 에서 해석함. 웹 브라우저 성능 평가에서 매우 중요한 요소가 됨.
  - jQuery 와 같은 공개 라이브러리가 유명함.
  - JSON(JavaScript Object Notation)은 클라이언트 서버간 정보 교환에 널리 사용.
- **CSS(Cascading Style Sheet)**
  - HTML에서 레이아웃이나 디자인 요소를 분리.
  - 최근 웹 클라이언트 개발은 HTML5+CSS3+JavaScript 임.

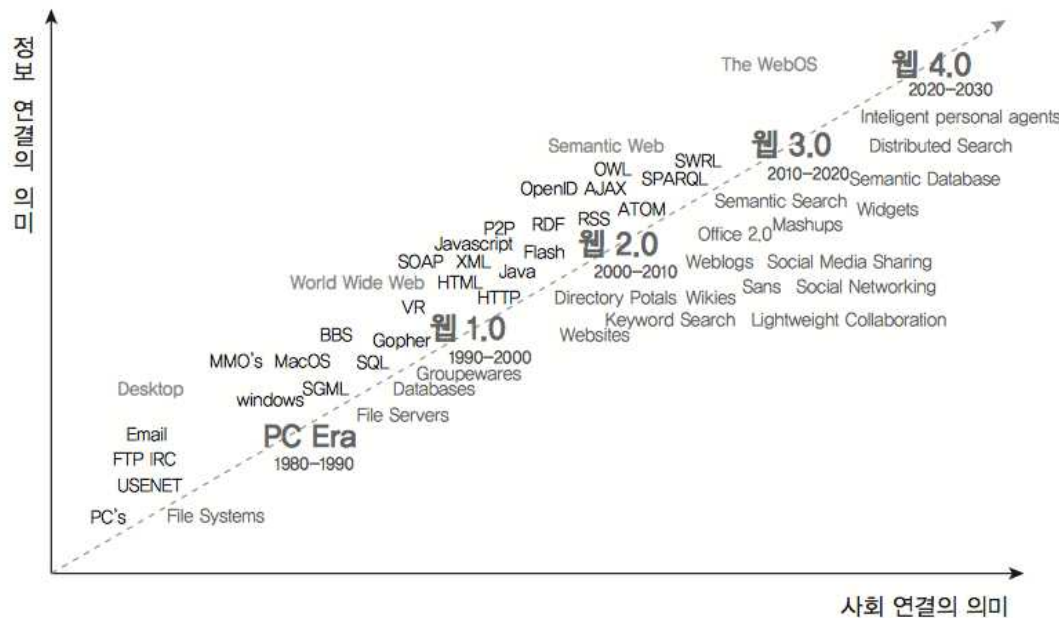
### ■ 서버 기술

- HTML 파일은 파일 내용을 수정하기 전까지는 내용이 변하지 않는 정적인 구조.
- 사용자 요청에 따라 다른 정보를 제공하거나 데이터베이스를 통한 서비스를 위해서는 별도의 프로그램 기술이 필요.
- **ASP.Net**
  - MS 윈도우 기반의 서버 프로그램 기술
  - .Net 기반의 컴포넌트 사용 가능.
  - MS 윈도우 종속과 상용 라이선스 정책으로 대규모 공개 웹 서비스 개발에는 거의 사용되지 않음.
- **PHP(Professional Hypertext Preprocessor)**
  - 처리속도가 빠르고 다양한 운영체제와 웹 서버 환경에서 실행 가능.
  - APM = Apache + PHP + MySQL
  - 오픈소스 게시판, 위키 프로그램, 연구용으로 널리 사용됨.
- **JSP(Java Server Page)**
  - 자바 서블릿 기반의 웹 프로그래밍 기술
  - 자바의 모든 기능을 사용할 수 있으며 안정성과 확장성이 뛰어나 가장 많이 사용되고 있는 웹 프로그래밍 기술임.

# 03. 스마트 시대의 웹 프로그래밍

## 1. 웹 프로그램의 현재와 미래

- 인터넷이 널리 사용되기 시작한 1990년대 후반부터 지금까지 많은 기술변화
- 2000년대 중반 Ajax 로 대표되는 웹2.0 기술과 시멘틱 웹 등 차세대 웹 프로그램 기술 등장
- 2014년 현재 스마트폰과 태블릿 등 개인용 컴퓨팅 환경의 보급 확산으로 새로운 기술 주목
- WOA(Web Oriented Architecture)가 확산되고 효과적인 프로그램 개발을 위한 프레임워크 보편화



[그림 1-9] 웹 발전 지도

## ■ 2.0과 웹 3.0

- 웹 2.0
  - 2004년 처음 등장
  - 플랫폼으로서의 웹
  - 가벼운 프로그래밍 모델, 신디케이션, 개조와 재조합 가능한 설계 등이 중요 개념
  - Ajax(Asynchronous JavaScript and XML), Google API 등이 대표 기술

## 03. 스마트 시대의 웹 프로그래밍

### ■ 웹 2.0과 웹 3.0

#### ■ 웹 3.0

- 2010년 등장
- 컴퓨터가 정보자원의 뜻을 이해하고 논리적 추론 까지 가능한 시멘틱 웹 개념 등장.
- 속도와 플랫폼 변화
  - 10Mbps ~ 1Gbps 의 초고속 인터넷 환경과 5G 등 초고속 무선 인터넷 서비스 보급
  - 인터넷 사용장치가 PC → 스마트폰, 태블릿, 스마트TV 등으로 급격히 변화
- 똑똑한 데이터와 인공지능의 향상
  - 컴퓨터가 정보자원의 뜻을 이해하고 논리적 추론 까지 가능한 차세대 지능형 웹 기술
  - 구글 나우(Google Now), 애플 시리(Siri) 등 사용자의 상황이나 질문의 의도에 따라 지능화된 서비스 제공
- 애플리케이션의 진화
  - 웹 2.0에서 시도되었던 open API, SOA 등이 새로운 플랫폼 등장으로 더욱 발전
  - 메쉬업은 컴포넌트화된 애플리케이션의 부분을 조합해 개인이나 그룹의 용도에 맞게 여러 서비스 장치를 사용하는 사용자들이 손쉽게 자신만의 콘텐츠나 정보를 구성할 수 있도록 해줌.



[그림 1-10] 웹 3.0 기술 태그 클라우드

## 03. 스마트 시대의 웹 프로그래밍

### ■ WOA(Web Oriented Architecture)

- 기존 PC 중심의 사용자 환경에서 스마트폰, 태블릿, 스마트TV, 스마트카 등 새로운 기기들이 출현
- 한 사람이 여러 기기를 통해 동일한 서비스와 정보로의 접근이 필요해짐.
- 즉 여러 기기 간의 끊어짐이 없는(Seamless) 서비스가 요구되면서 One Source Multi Use 를 위한 N-Screen 혹은 N-Device 서비스가 요구 증대.
- 이러한 요구사항 해결을 위해 다음과 같은 기술들이 급성장 하고 있음.
  - 하드웨어 인프라적인 측면 : 클라우드라고 불리는 대규모의 공용 컴퓨팅 서비스, 가상화 SW 포함.
  - 소프트웨어적인 측면 : WOA
- 기존 SOAP(Simple Object Access Protocol) 기반의 SOA(Service Oriented Architecture) 에서 REST(Representational State Transfer) 기반의 경량 웹 서비스 모델 발전.
- Restful 웹 서비스는 JAX-RS(JSR-311)로 자바 규격에 공식적으로 포함됨.
- WOA는 웹을 중심으로 전체 시스템 아키텍처를 설계해 나가는 기술임.

## 03. 스마트 시대의 웹 프로그래밍

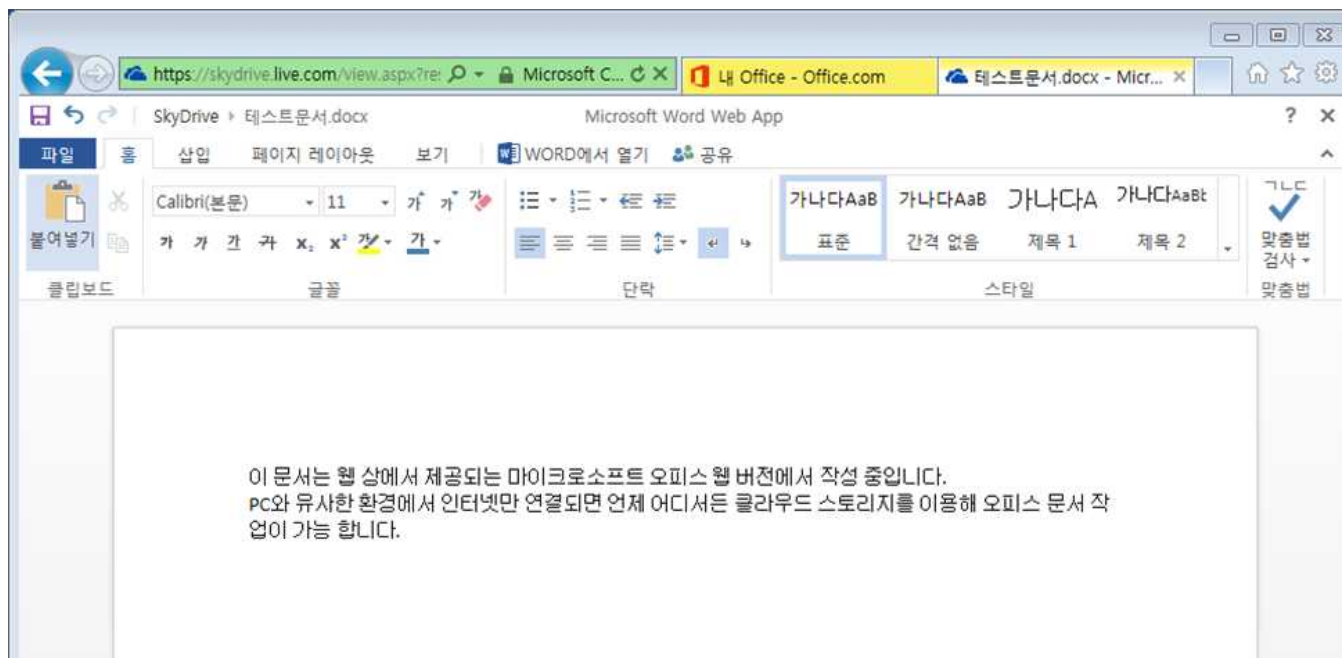
### ■ 프레임워크(Framework)

- 일반적인 개발의 문제점
  - 프로그램의 규모 확대 -> 높은 생산성, 쉬운 유지 보수, 기능의 변경이 확장이 용이한 개발 기술 필요
  - 개발방법론, 소프트웨어 디자인 패턴, 리팩토링, 프레임워크 등 소프트웨어 공학적 기술 등장
- 프레임워크(Framework)는 무언가를 만들기 위한 틀.
- 소프트웨어적으로는 목적에 맞게 잘 설계된 구조와 미리 구현된 라이브러리가 포함된 소프트웨어 형태.
- 프레임워크를 사용하면 정해진 규격에 따라 프로그램 구조를 만들어야 하며, 개발자가 신경쓰거나 처리해야 할 많은 일과 이벤트 관리는 프레임워크를 통해 처리.
- 여러 유틸리티 라이브러리도 제공하기 때문에 개발자는 비교적 적은 노력으로도 고품질의 소프트웨어 개발이 가능해짐.
- 대표적인 프레임워크는 스프링프레임워크로 웹 개발을 포함해 대규모 시스템 개발에 적합한 기술 구조를 제공함.
- **스프링 프레임워크 참고 서적과 사이트**
  - 예제로 쉽게 배우는 스프링 프레임워크 3.0(사카타 코이치 지음, 황선유 옮김), 한빛미디어
  - 토비의 스프링 3.1(이일민 지음), 에이콘
  - 프로 스프링 3(클라렌스 호, 롭 해롭 지음, 유윤선 옮김), 위키북스
  - 스프링 인 액션 3판(크레이그 월즈 지음, 홍영표 옮김), 제이펍
  - [www.springsource.org](http://www.springsource.org)(스프링 프레임워크 메인 사이트)
  - [www.ksug.org](http://www.ksug.org)(한국 스프링 사용자 모임 사이트)

# 03. 스마트 시대의 웹 프로그래밍

## ■ 웹 프로그램의 미래

- 현재는 웹의 전성기임.
- 애플의 아이폰에서 시작된 스마트폰 열풍은 기존 컴퓨터 사용 패턴을 변화시킴.
- 전통적인 PC 시장의 변화 -> 대규모 PC 제조업체인 DELL 상장 폐지 -> MS 시장 지배력 약화.
- 상당수의 PC소프트웨어들이 설치 형에서 웹 형태로 전환됨.
- 더 많은 스마트 기기, 안드로이드, iOS, 웹 OS 등 새로운 모바일 운영체제의 성장과 함께 고속의 무선 인터넷을 기반으로 한 소프트웨어 발전이 가속화될 전망.



[그림 1-11] 마이크로소프트 오피스 웹

# 03. 스마트 시대의 웹 프로그래밍

## 2. 모바일 앱 개발과 웹 프로그래밍

### ■ 모바일 기기의 특징

- 화면크기
  - 스마트폰의 경우 3.5 ~ 6인치
  - 태블릿의 경우 7 ~ 12인치
- 하드웨어 아키텍처
  - PC의 x86 과 다른 ARM 계열 CPI사용
  - 멀티코어, 고성능 GPU, 64bit 지원
- 운영체제
  - 애플의 iOS와 구글 안드로이드가 대표적
  - 그 외 구글 크롬, WebOS, MS 윈도우8 등
- 무선통신
  - LTE, WIFI 등 고속 무선 인터넷 연결
  - 블루투스, NFC 등 근거리 통신 지원
- 입력장치
  - 터치 스크린, 터치 펜
  - 가속센서, 자이로센서 등
- 배터리
  - 대용량 배터리
  - 배터리 절약을 위한 각종 저전력 기술

### ■ 모바일 앱 개발 유형

- 네이티브 앱
  - 각 운영체제별 별도 개발환경과 실행 파일
  - 빠른 속도, 기기별 특화 기능 사용의 편리함
- 여러 기기와 운영체제 지원의 어려움으로 웹 기술을 사용한 모바일 웹 기술 주목.
- 모바일 웹 제약 사항
  - 항상 인터넷에 연결되어 있어야 함.
  - 하드웨어나 운영체제 기능 접근 제한(GPS, 센서, 카메라, 주소록 등)
- 하이브리드 앱
  - 메인 프로그램은 네이티브로, 프로그램 구성요소는 웹 기반으로 구현
  - 장치 기능 접근을 위해 센차터치, 폰갭 등의 툴킷 사용

