

# 1 웹 프로그래밍 기초

# Contents

---

- ❖ 웹 프로그래밍 기초
- ❖ 웹 서비스 구성요소
- ❖ URL
- ❖ Servlet & JSP
- ❖ 웹 컨테이너

# 웹 프로그래밍 기초

---

## ❖ 네트워크(Network)

- 네트워크의 사전적 의미는 “그물 처럼 얹혀 있는 통로 등의 망 - 망 형 구조
- 방송사 네트워크, 인맥을 의미하는 휴먼 네트워크(Human Network) 등
- IT에서는 ‘컴퓨터와 컴퓨터가 연결된 망’의 개념으로 컴퓨터 네트워크를 의미
- 1960년대 말 : 인터넷의 태동 시기 -> ARPANET
- 1970 ~ 1990년대
  - 컴퓨터가 네트워크로 연결되지 않음
  - 자료 이동 시 플로피 디스켓 등을 이용함
  - 국내의 경우 1990년대 PC통신(천리안, 하이텔) -> 모뎀을 이용한 인터넷 접속
- 2000년 ~ 2010년
  - 인터넷의 보급으로 전세계 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 기기의 보급으로 개인이 언제 어디서나 네트워크 연결이 가능해짐
  - 클라우드 기반 스토리지 서비스를 통한 개인 및 그룹 자료 공유

# 웹 프로그래밍 기초

---

## ❖ TCP/IP

- TCP/IP(Transmission Control Protocol/Internet Protocol)는 컴퓨터와 컴퓨터 간에 통신하기 위해 필요한 사항을 정의한 통신 규약(프로토콜)
- 인터넷의 기반 프로토콜로 특정 운영체제나 하드웨어에 종속적이지 않고 근거리, 원거리와 관계없이 데이터를 교환 할 수 있는 개방형 구조
- TCP/IP는 OSI(Open System Interconnection) 7 계층을 단순화해서 아래와 같이 4개 계층으로 정의되어 있음

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.

# 웹 프로그래밍 기초

## ❖ IP 주소

- IP 주소(IP Address)는 TCP/IP로 연결된 네트워크에서 각각의 컴퓨터를 구분하기 위해 사용되는 주소
- 4Byte의 숫자로 구성 -> 192.123.123.123과 같이 네 개로 구분된 10진수 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

## IP주소 구분과 범위

# 웹 프로그래밍 기초

## ❖ 인터넷(Internet)

- 인터넷은 전 세계 컴퓨터가 하나로 연결된 거대한 네트워크를 의미하는 고유명사
- 인터넷은 네트워크 인프라이고 www, email 등은 인터넷 기반 서비스를 의미
- 프로토콜(Protocol)
  - 네트워크에 연결된 컴퓨터 간의 통신을 위한 규약
  - 인터넷 서비스들은 TCP/IP 4계층 중 응용계층에 해당함.
- 포트(Port)
  - 네트워크 서비스에 접속하기 위한 통로
  - 하나의 서버가 여러 서비스를 제공할 경우 각 서비스를 구분하기 위한 용도로 사용

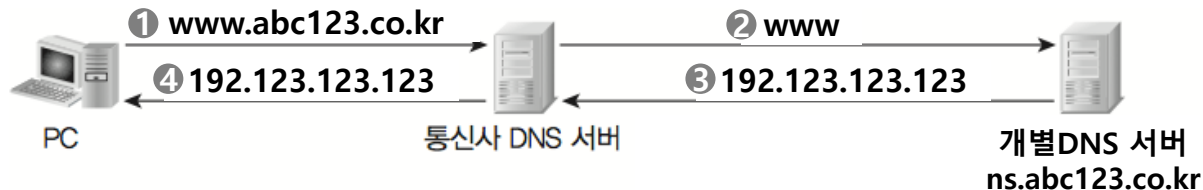
서비스 이름	기능	프로토콜	포트
웹(www)	웹 서비스	HTTP	80
이메일(email)	이메일 서비스	SMTP/POP3/IMAP	25/110/143
FTP	파일 전송 서비스	FTP	21
Telnet	원격 로그인 서비스	TELNET	23
DNS	도메인 이름 변환 서비스	DNS	53
News	인터넷 뉴스 서비스	NNTP	119

대표적인 인터넷 기반 서비스

# 웹 프로그래밍 기초

## ❖ DNS(Domain Name System)

- 인터넷에 연결된 컴퓨터를 구분하기 위해 사용하는 IP주소 대신 사람이 알기 쉬운 이름 형태로 컴퓨터를 구분하기 위한 인터넷 서비스나 시스템
- 서비스 이름(www) + 도메인 이름(naver.com) 형태로 구성됨
  - www.naver.com, www.daum.net 등
- 도메인 관리 기관을 통해 일정 비용을 지불하고 사용 가능



- 1) 인터넷 통신사(KT, SK브로드밴드, LGU+ 등)의 DNS 서버에 도메인 이름 요청
- 2) 통신사 DNS 서버는 ns.abc123.co.kr에 www 서비스를 제공하는 서버의 IP 주소를 요청
- 3) 개별 DNS 서버는 www 서비스를 제공하는 서버의 IP 주소를 통신사 DNS 서버에 알려줌
- 4) 통신사 DNS 서버는 요청한 클라이언트에게 IP 주소를 알려줌

# 웹 프로그래밍 기초

---

## ❖ 웹(Web)

- 월드 와이드 웹(World Wide Web, www)을 의미
- 인터넷에서 정보를 공유하고 검색할 수 있는 서비스로 많은 사람들이 '웹 = 인터넷'이라고 생각할 정도로 대표적인 인터넷 서비스
- 1989년 스위스에 있는 유럽물리입자연구소(CERN)의 팀 버너스리에 의해 개발
- 세계 여러 곳에 흩어져 있는 연구자들이 손쉬운 방법으로 정보를 공유하기 위해 HTML이라는 마크업 언어를 통해 정보를 제공하고 HTTP 프로토콜을 사용해 TCP/IP 네트워크에서 사용하기 시작함
- 인터넷에 연결된 웹 브라우저를 이용해 PC, 스마트폰, 태블릿 등 운영체제와 하드웨어에 상관없이 동일한 서비스 제공이 가능



# 웹 프로그래밍 기초

---

## ❖ 웹 서버와 HTTP

- 서버(Server)
  - 네트워크에서 서비스를 제공하는 컴퓨터 또는 프로그램
  - 웹 서버, FTP 서버(파일 서버), 이메일 서버, 프린트 서버
- 클라이언트(Client)
  - 서비스를 이용하는 컴퓨터 또는 프로그램
  - PC, 스마트폰, 태블릿 등
- HTTP(Hyper Text Transfer Protocol)
  - 웹 서비스에 사용되는 통신 규격
  - 간단한 명령어와 헤더 규격으로 이루어져 있음

# 웹 프로그래밍 기초

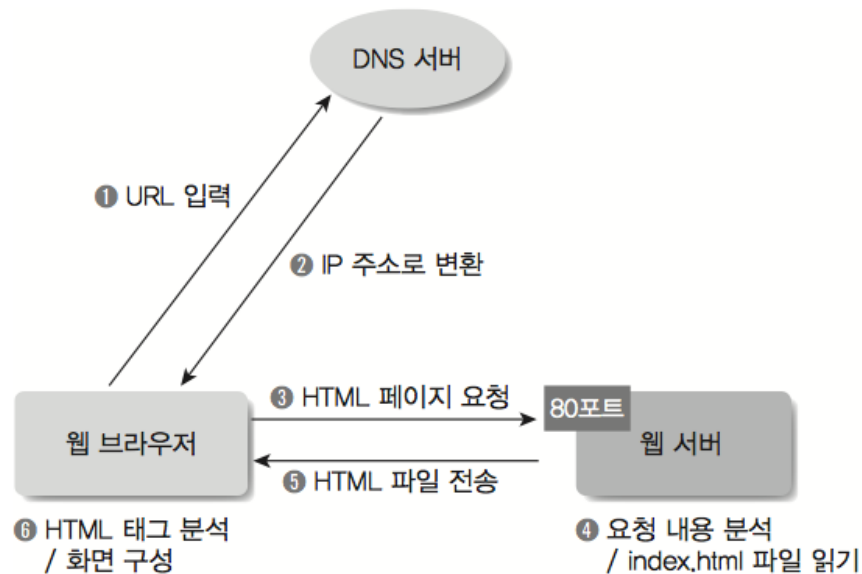
## ❖ 웹 서비스의 동작 과정

### ■ 웹 서버 소프트웨어

- 서버에서 웹 서비스를 제공하는 소프트웨어
- 아파치(Apache), 엔진엑스(Ngnix), MS의 IIS(Internet Information Server)

### ■ 클라이언트 소프트웨어

- 웹 서비스를 이용하기 위한 클라이언트 소프트웨어 -> 웹 브라우저(Web Browser)
- Chrome, Firefox, Safari 등



# 웹 프로그래밍 기초

## ❖ 웹 프로그램의 개요

### ■ 로컬 프로그램

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

### ■ 웹 프로그램

- 별도의 설치 없이 서버에 접속하는 것만으로 필요한 기능이나 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 프로그램 개발을 위해 서버 프로그래밍 기술과 클라이언트 프로그래밍 기술이 필요

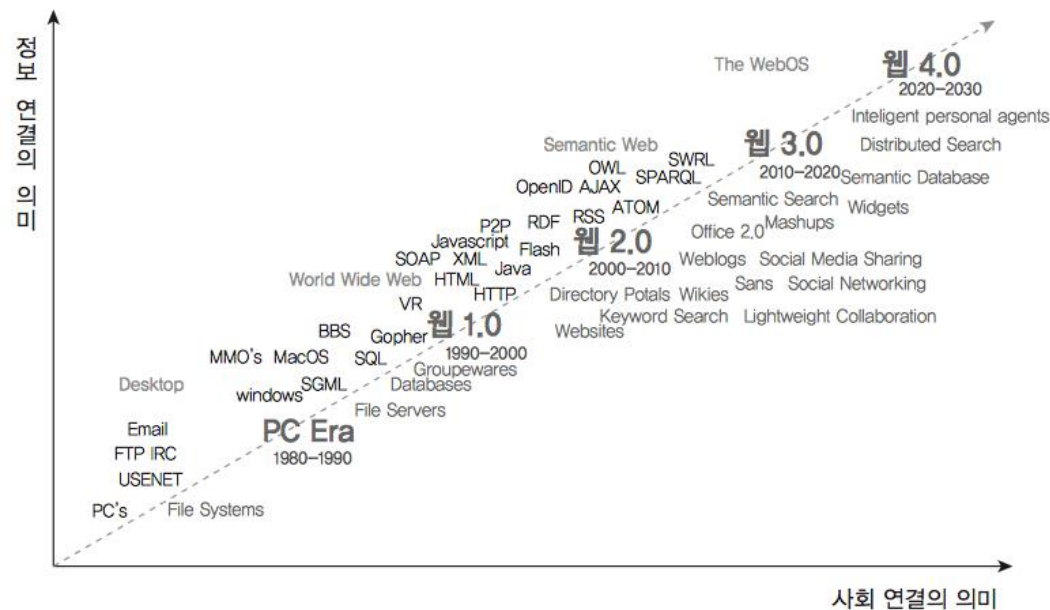
구분	종류
클라이언트 기술	HTML, 자바스크립트, CSS
서버 기술	서블릿, JSP, ASP.Net, PHP
클라이언트/서버 공통 기술	자바, C, C#.Net

### 대표적인 웹 개발 기술

# 웹 프로그래밍 기초

## ❖ 웹 프로그램의 현재와 미래

- 인터넷이 널리 사용되기 시작한 1990년대 후반부터 지금까지 많은 기술변화
- 2000년대 중반 Ajax로 대표되는 웹2.0 기술과 시멘틱 웹 등 차세대 기술 등장
- 2010년 이후 스마트폰과 태블릿 등 개인용 컴퓨팅 환경의 보급 확산으로 HTML5와 같은 새로운 기술이 주목을 받음
- WOA(Web Oriented Architecture)가 확산되고 효과적인 프로그램 개발을 위한 프레임워크 사용이 보편화 됨



# 웹 프로그래밍 기초

---

## ❖ 웹 2.0

- 2004년 이후 -> 플랫폼으로서의 웹
- 가벼운 프로그래밍 모델, 개조와 재조합 가능한 설계 등이 중요한 개념
- Ajax(Asynchronous JavaScript and XML), Google API 등이 대표 기술

## ❖ 웹 3.0

- 2010년 이후
- 컴퓨터가 정보자원의 뜻을 이해하고 논리적 추론 까지 가능한 시멘틱 웹 개념
- 속도와 플랫폼 변화
  - 10Mbps ~ 1Gbps의 초고속 인터넷과 4G LTE 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
  - Chat GPT와 같은 사용자 상황이나 질문의 의도에 따라 지능화된 서비스 제공
- 애플리케이션의 진화
  - open API, SOA(Service Oriented Architecture)등이 새로운 플랫폼 등장으로 더욱 발전
  - 메쉬업은 컴포넌트화된 애플리케이션의 부분을 조합해 개인이나 그룹의 용도에 맞게 여러 장치를 사용하는 사용자들이 손쉽게 자신만의 정보를 구성할 수 있도록 해줌

# 웹 프로그래밍 기초

---

## ❖ WOA(Web Oriented Architecture)

- 기존 PC 중심의 사용자 환경에서 스마트폰, 태블릿, 스마트TV, 스마트 카 등 새로운 기기의 출현으로 한 사람이 여러 기기를 통해 동일한 서비스와 정보로의 접근이 필요해짐
- 여러 기기 간의 끊어짐이 없는(Seamless) 서비스가 요구되면서 One Source Multi Use를 위한 N-Screen 또는 N-Device 서비스가 필요
- 다양한 요구사항을 해결을 위해서 클라우드 서비스가 보편화 됨
- REST(Representational State Transfer) 기반의 경량 웹 서비스 모델로 발전
- Restful 웹 서비스는 JAX-RS(JSR-311)로 자바 규격에 공식적으로 포함됨
- WOA는 웹을 중심으로 전체 시스템 아키텍처를 설계해 나가는 기술임

# 웹 프로그래밍 기초

---

## ❖ 프레임워크(Framework)

- 일반적인 개발의 문제점
  - 프로그램의 규모 확대 -> 높은 생산성, 쉬운 유지 보수, 기능의 쉬운 변경과 확장성이 용이한 개발 기술 필요해 짐
  - 개발방법론, 디자인패턴, 리팩토링, 프레임워크 등 소프트웨어 공학적 기술 등장
- 프레임워크(Framework)는 무언가를 만들기 위한 틀
- 소프트웨어적으로는 목적에 맞게 잘 설계된 구조와 미리 구현된 라이브러리가 포함된 소프트웨어로 응용프로그램을 쉽게 만들기 위한 틀
- 프레임워크를 이용하면 정해진 규격에 따라 프로그램 구조를 만들어야 하며, 개발자가 처리해야 할 많은 일과 이벤트 관리는 프레임워크를 통해 처리
- 유용한 여러 유틸리티 라이브러리를 제공하기 때문에 개발자는 비교적 적은 노력으로 고품질의 소프트웨어 개발이 가능해짐
- 자바 진영의 대표적인 프레임워크로는 스프링프레임워크가 있으며 스프링은 웹 개발을 포함해 대규모 시스템 개발에 적합한 기술을 제공함

# 웹 서비스 구성 요소

웹 브라우저는 웹 서버에 필요한 자료를 요청

웹 브라우저(Client)



요청  
(Request)

응답  
(Response)

Web  
Server

Web  
Application  
Server

웹 서버는 클라이언트 요청을 처리한 결과를  
HTML 형식으로 웹 브라우저에게 응답

Database



# URL

---

## ❖ Uniform Resource Locator

## ❖ URL 구성

[프로토콜]://[호스트][:포트][경로][파일명][.확장자][쿼리문자열]

- 예: <http://www.google.com/search?hl=en&q=jsp&aq=f>
  - 프로토콜: http
  - 호스트: www.google.com
  - 포트: 80 (http 프로토콜의 기본 포트)
  - 경로: /search
  - 쿼리문자열: hl=en&q=jsp&aq=f

## ❖ URL은 웹 애플리케이션에서 요청을 구분하기 위한 용도로 사용됨

# Servlet & JSP

---

## ❖ Servlet과 JSP는 자바를 만든 Sun에서 정한 웹 개발 표준

- 서블릿(Servlet) : 컴파일 방식
- JSP(JavaServerPage) : 스크립트 방식

## ❖ 서블릿(Servlet) 특징

- 자바 클래스로 작성하며 HttpServlet 클래스를 상속해 구현함
- HTTP에 대한 클라이언트의 요청/응답 처리
- 자바 웹 애플리케이션에서 요청을 받아 프로그램의 흐름을 제어하는 용도로 주로 사용  
-> MVC 패턴에서 프로그램의 흐름을 제어하는 컨트롤러로 활용됨

## ❖ JSP

- 자바 기반 서버사이드 스크립트 언어
  - 자바의 장점을 그대로 사용
- HTTP에 대한 클라이언트의 요청/응답 처리
- 자바 웹 애플리케이션에서 요청을 처리한 결과 화면을 생성하는 용도로 주로 사용  
-> MVC 패턴에서 View Page를 작성할 때 활용됨

# 웹 컨테이너

---

- ❖ 웹 애플리케이션을 실행할 수 있는 웹 서버를 포함하는 컨테이너
- ❖ JSP도 웹 컨테이너에 의해서 Servlet으로 변환되어 실행됨
- ❖ Servlet을 관리하고 실행하기 때문에 서블릿 컨테이너라고 함



Servlet 2.5,	JSP 2.1 – Tomcat 6
Servlet 3.0,	JSP 2.2 – Tomcat 7
Servlet 3.1,	JSP 2.3 – Tomcat 8.5
Servlet 4.0,	JSP 2.3 – Tomcat 9
Servlet 5.0,	JSP 3.0 – Tomcat 10

<https://tomcat.apache.org/whichversion.html>

- ❖ 주요 웹 컨테이너
  - 톰캣(Tomcat) : <http://tomcat.apache.org/>
  - 제티(Jetty) : <http://www.mortbay.org/jetty/>
  - 레신(Resin) : <http://www.caucho.com/>

# 웹 컨테이너

---

❖ Java EE 및 Jakarta EE 구현

❖ 주요 상용 WAS

- 제우스(JEUS) : <https://www.tmaxsoft.com/kr/solution>
- 웹로직(WebLogic) : <https://www.oracle.com/kr/java/weblogic/>
- 웹스피어(WebSphere) : <https://www.ibm.com/kr-ko/products/websphere-application-server>