

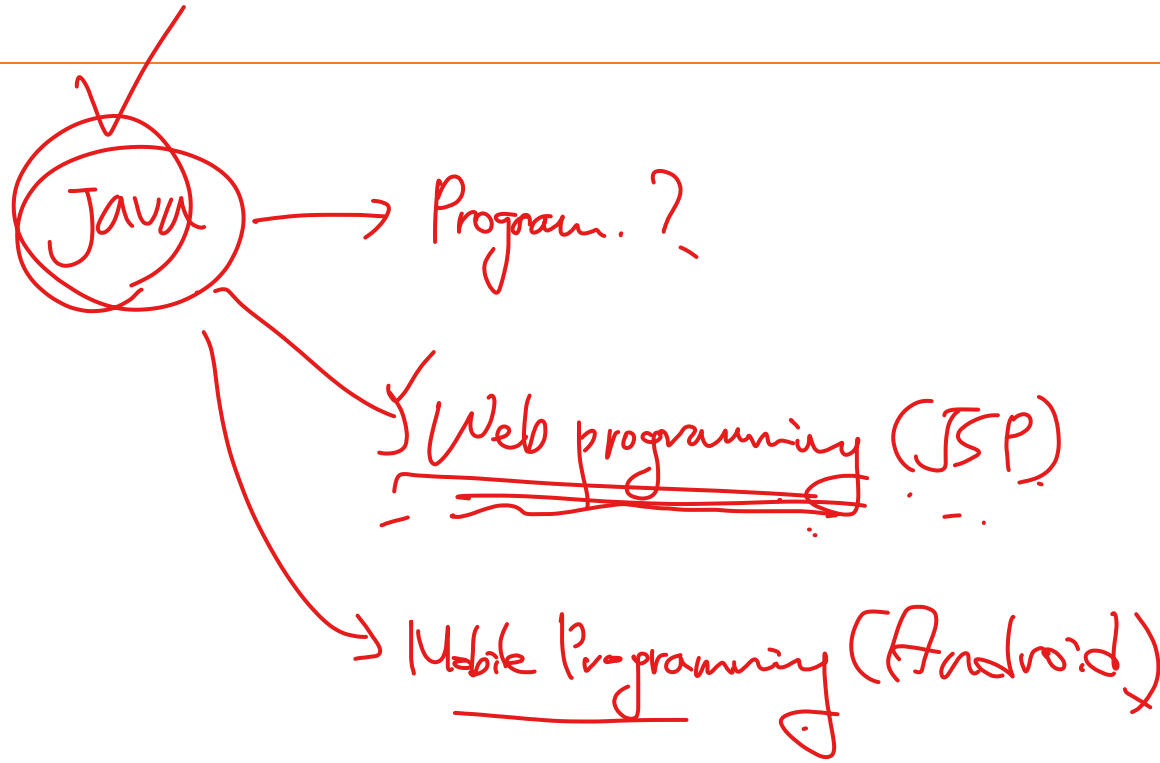
# 제1장 JSP 개요와 실행환경 구축



# 목차

---

- ✓ 1. 네트워크 프로토콜
- ✓ 2. 인터넷과 웹
- ✓ 3. 클라이언트와 서버
- ✓ 4. HTTP
- ✓ 5. 웹 프로그래밍
- ✓ 6. JSP 개요
- ✓ 7. JSP 학습 순서
- ✓ 8. 개발환경 구축



# 1. 네트워크 프로토콜

- 네트워크는 정보 교환을 위해 여러 컴퓨터를 통신 회선으로 연결해서 구성한 시스템
- 프로토콜 Protocol 은 네트워크를 위해 컴퓨터 간의 연결 규약
- TCP/IP는 네트워크 프로토콜은 HTTP 인터넷 기반 프로토콜

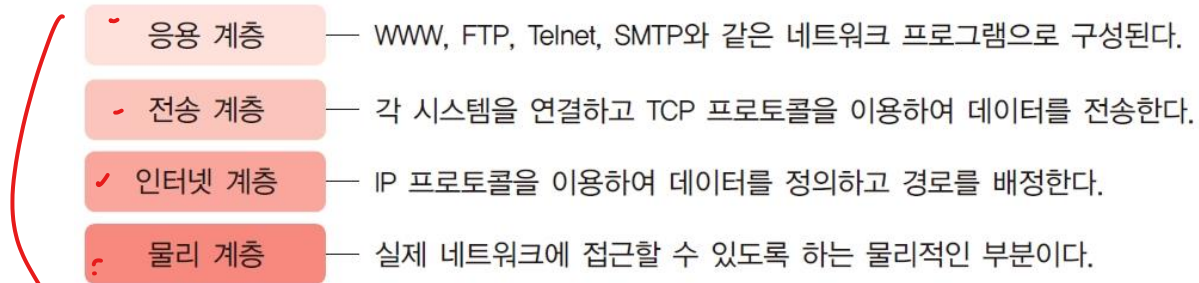


그림 1-3 TCP/IP의 4계층 구조

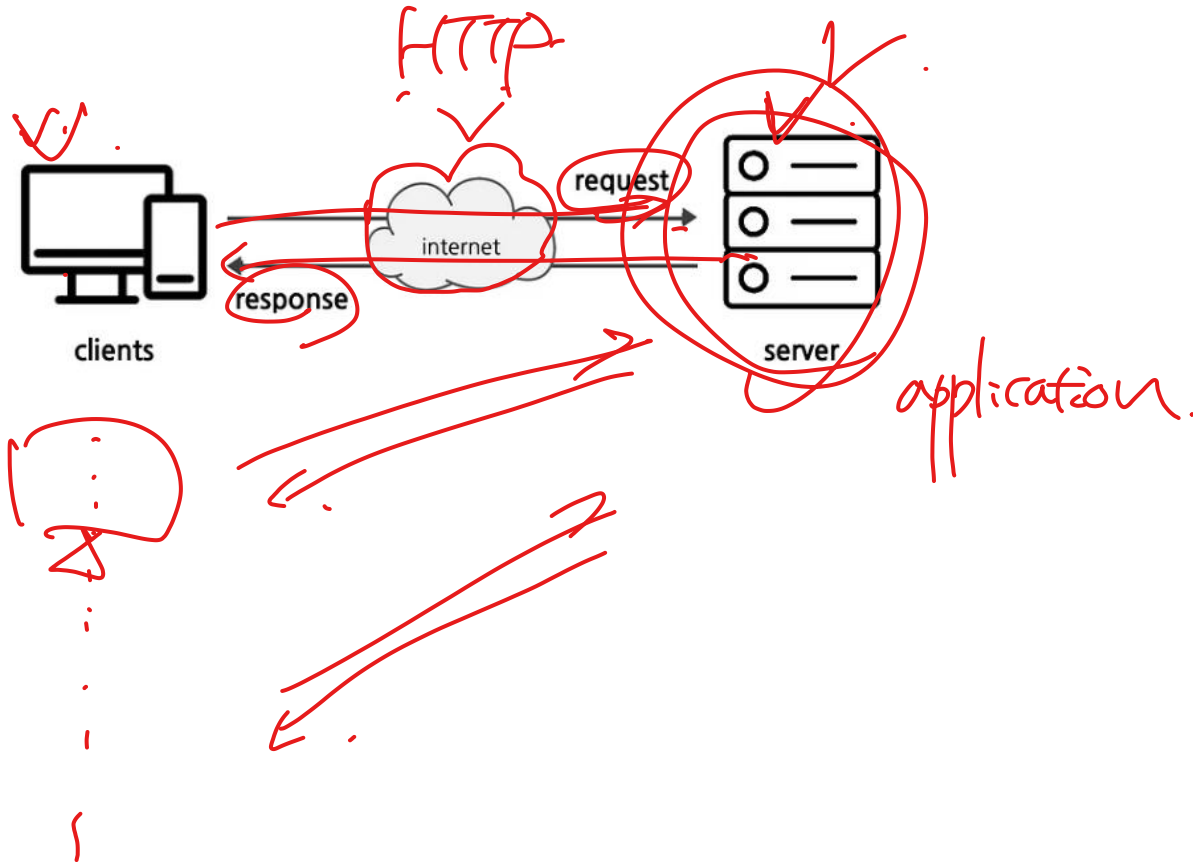
## 2. 인터넷과 웹

- 인터넷은 전 세계를 네트워크를 연결한 통신망으로, 다양한 디지털기기로 연결되어 정보를 공유할 수 있는 공간
- 웹(Web, WWW)은 인터넷에 연결된 컴퓨터를 이용해 사람들과 정보를 공유 하는 서비스
- 포트(Port)는 인터넷 서비스를 구분하기 위한 번호

서비스	기능	프로토콜	포트
✓ 웹 WWW	웹 서비스	HTTP / HTTPS	80 / 443
DBMS	데이터베이스 관리	TCP/IP	3306
이메일	이메일 전송	SMTP / POP3 / IMAP	25 / 110 / 143
FTP	파일 전송	FTP	21
• Telnet / SSH	원격 로그인	TELNET / SSH	23 / 22
DNS	도메인 IP 변환	DNS	53

### 3. 클라이언트와 서버

- 클라이언트(Client)는 네트워크를 통해 서버에 요청을 보내고, 서버로부터 응답을 받는 역할을 수행  
*레이터. HTML, JSON, XML...*
- 서버(Server)는 네트워크를 통해 클라이언트의 요청을 수신하고, 요청을 처리하여 응답을 보내는 역할을 수행
- 클라이언트와 서버는 일반적으로 HTTP, HTTPS, FTP, TCP/IP 등과 같은 통신 프로토콜을 사용하여 상호작용



## 4. HTTP

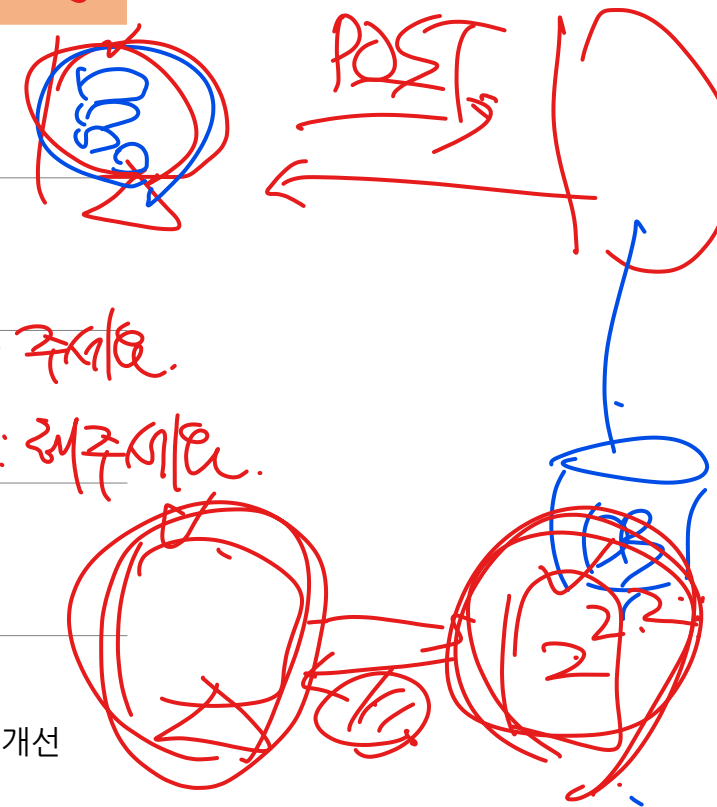


- HTTP Hyper Text Transfer Protocol은 웹에서 데이터를 주고받기 위한 프로토콜로 클라이언트와 서버 간의 요청과 응답을 정의
- HTTP는 TCP/IP 프로토콜 기반으로 동작, 인터넷을 통해 하이퍼텍스트 문서와 기타 리소스를 전송하는 데 사용

DB

HTML, CSS, JS, Image, MP4, ...

구분	내용
✓ 주요 특징	<ul style="list-style-type: none"><li>• HTTP은 무상태성Stateless 과 비연결성Connectionless 프로토콜</li><li>• HTTP(80), HTTPS(443)</li></ul>
✓ HTTP 메시지	<ul style="list-style-type: none"><li>• HTTP는 요청-응답 모델로 클라이언트가 서버간 요청Request 과 응답Response 메시지로 구성</li><li>• 요청-응답 메시지는 시작 라인Start Line, 헤더Headers, 빈줄Blank, 본문Body으로 구성</li></ul>
✓ HTTP 요청 메서드	<ul style="list-style-type: none"><li>• HTTP 요청 메서드는 클라이언트가 서버에 요청을 할 때 사용되는 다양한 종류의 명령어</li><li>• GET, POST, PUT, DELETE, PATCH, HEAD, OPTIONS 등</li></ul>
✓ HTTP 상태 코드	<ul style="list-style-type: none"><li>• HTTP 상태 코드는 클라이언트가 요청을 보낸 후, 서버가 응답으로 보내는 3자리 숫자 코드</li><li>• 정보 응답(1xx), 성공(2xx), 리다이렉션(3xx), 요청 오류(4xx), 응답 오류(5xx)</li></ul>
✓ HTTP 버전	<ul style="list-style-type: none"><li>• HTTP/1.0 - 최초의 HTTP 버전, 기본적인 요청/응답 모델을 지원</li><li>• HTTP/2.0 - 멀티플렉싱Multiplexing, 헤더 압축Header Compression, 서버 푸시Server Push 등 효율성과 성능 개선</li><li>• HTTP/3.0 - QUIC 프로토콜을 기반으로 하여, 연결 설정 속도와 보안성 향상</li></ul>



## 5. 웹 프로그래밍

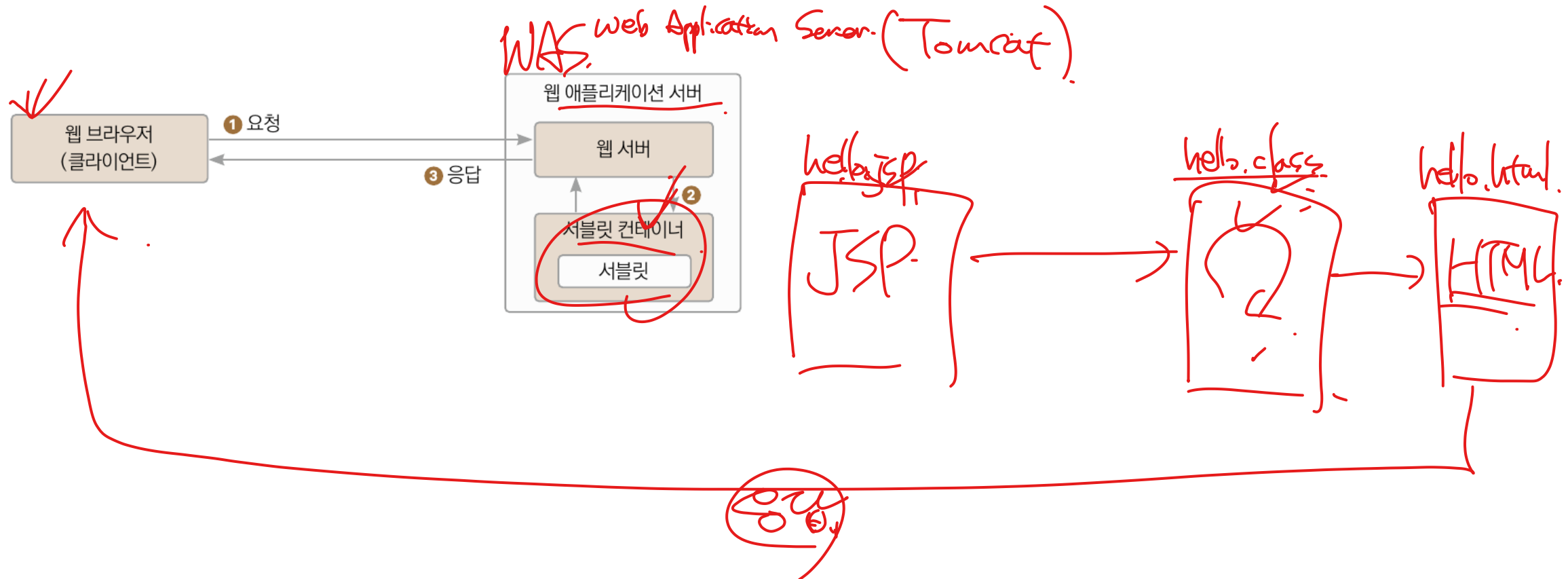
- 웹 프로그래밍은 클라이언트와 서버 모두를 포함, 웹 애플리케이션의 전체적인 기능을 구현하는 데 필요한 기술과 도구.
- 프론트엔드 front-end <sup>프론트</sup>는 사용자와 상호작용하는 사용자 인터페이스 UI, User Interface <sup>백엔드</sup>와 사용자 경험 UX, User Experience을 설계하고 개발
- 백엔드 back-end는 서버에서 실행되는 프로그램으로 데이터 처리, 비즈니스 로직 등 클라이언트의 요청을 처리하고 응답 생성

구분	기술	설명
프론트엔드	React	Facebook에서 개발한 Javascript 라이브러리, 컴포넌트 기반 아키텍처
	Angular	Google에서 개발한 프론트엔드 프레임워크, MVVM 개발 패턴
	Vue.js	Evan You가 개발한 컴포넌트 기반 프론트엔드 프레임워크
백엔드	JSP <sup># Spring</sup>	Java 언어 기반 웹 프로그래밍으로 강력한 엔터프라이즈 개발에 많이 사용
	ASP.NET	C# .NET 기반 웹 프로그래밍으로 윈도우 기반 기업용 애플리케이션 개발
	PHP	오픈소스 기반 웹 프로그래밍으로 쉬운 학습과 웹 프레임워크 지원, 중소규모 웹 서비스
	Node.js	Javascript 기반 웹 프로그래밍으로 가볍고 빠른 웹 애플리케이션 개발
	Python	범용 프로그래밍 언어 기반 웹 프로그래밍으로 웹 애플리케이션 및 데이터 분석 개발

Full-Stack

## 6. JSP 개요

- JSP Java Server Page ~~는~~ Java 서블릿 Servlet 기반 서버 사이드 동적 웹 프로그래밍 기술
- HTML 문서 내에 Java 프로그래밍으로 서버에서 동작하는 웹 애플리케이션 개발
- JSP 파일은 웹 애플리케이션 서버로 실행되면서 서블릿으로 변환되어 클라이언트의 요청을 처리하여 최종 HTML 생성





## 7. JSP 실습 순서

---

- ch01. JSP 개요와 실습환경 구축
- ch02. 기본 스크립트 실습
- ch03. JSP 내장객체
- ch04. JSP 액션태그
- ch05. 쿠키와 세션
- ch06. JSP 데이터베이스
- ch07. 파일 업로드/다운로드와 이메일 전송
- ch08. 표현언어와 JSTL
- ch09. 서블릿과 필터, 리스너
- ch10. MVC 아키텍처
- ch11. 로그와 배포

! 기판 (Model 1)

! 기판 (Model 2)

~~기판~~