

Spring Framework 맛보기

EC 23기 양희찬

CONTENTS

01 Server란?

Server와 Client
Spring Framework 구조
Spring Boot란?

02 실습 (1)

프로젝트 구조
프로젝트 생성
간단한 API 개발
Controller 나누기

03 배운 내용 정리

오늘 배운 내용
다음 실습 내용

01 Server란?

#Frontend #Server #BackEnd #Spring

01. Server란?

Server

Servèng

고객(Client)에게 무언가를 제공하는 역할을 하는 무언가

영어 ▾

⇒

한국어 ▾

높임말

serve

×

제공하다

서브

5 / 5000

번역 수정

🔊

📄

번역하기

🔊

📄

☆

🔗

자동완성

serve [sɜːrv]

동사

1. (식당 등에서 음식을) 제공하다; (음식을 상에) 차려 주다
[VN] Breakfast is served between 7 and 10 a.m.
아침 식사는 7시부터 10시 사이에 제공됩니다.

제공 (제공하다) [提供]

1. offer, supply, provide, furnish (sb with sth)
일 자리를 제공하다
offer sb a job

출처: 능률교육

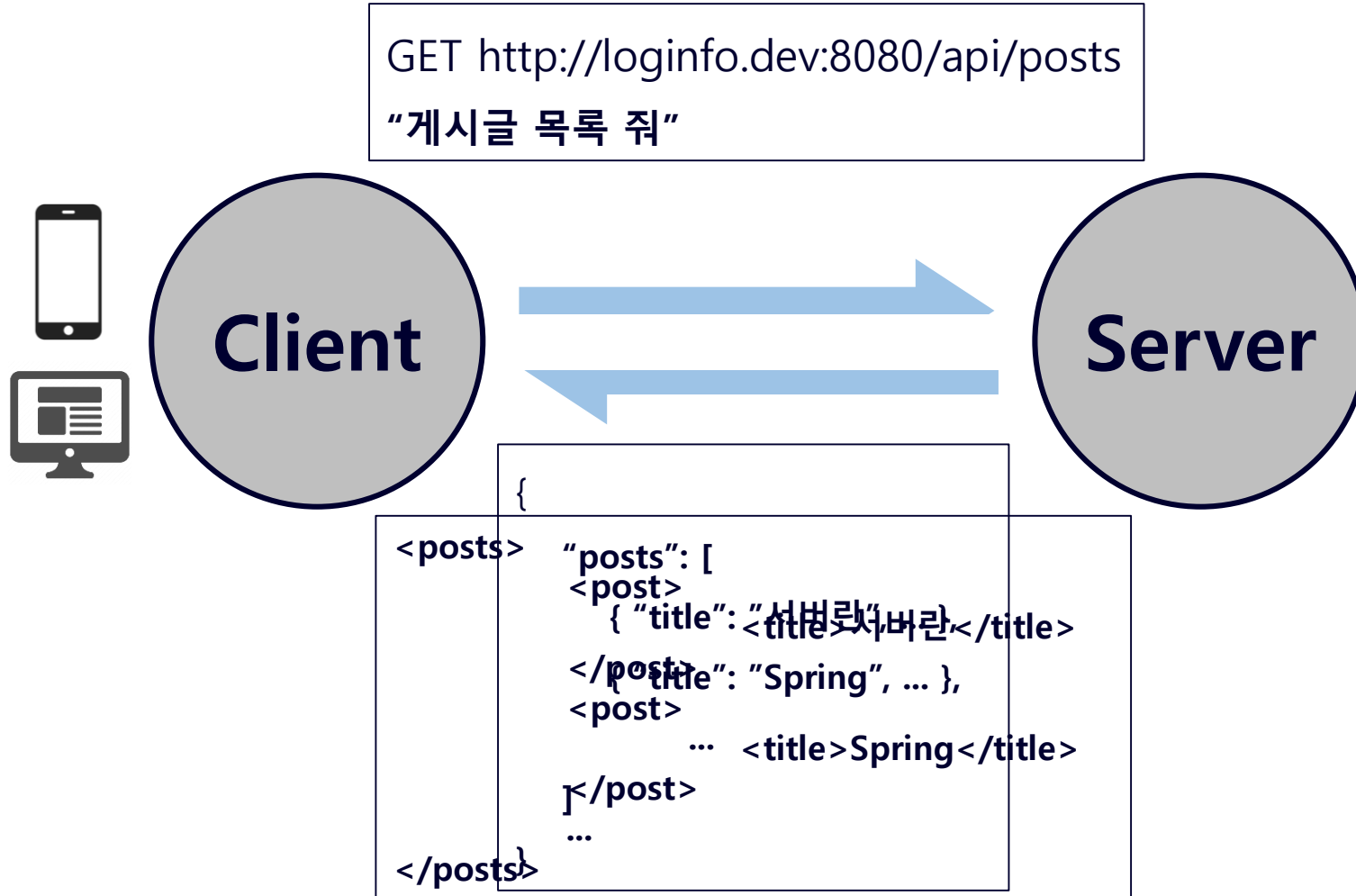
01. Server란?

Server



01. Server란?

Server와 Client



01. Server란?

Server와 Client

Client

표현계층
Front-End
요청

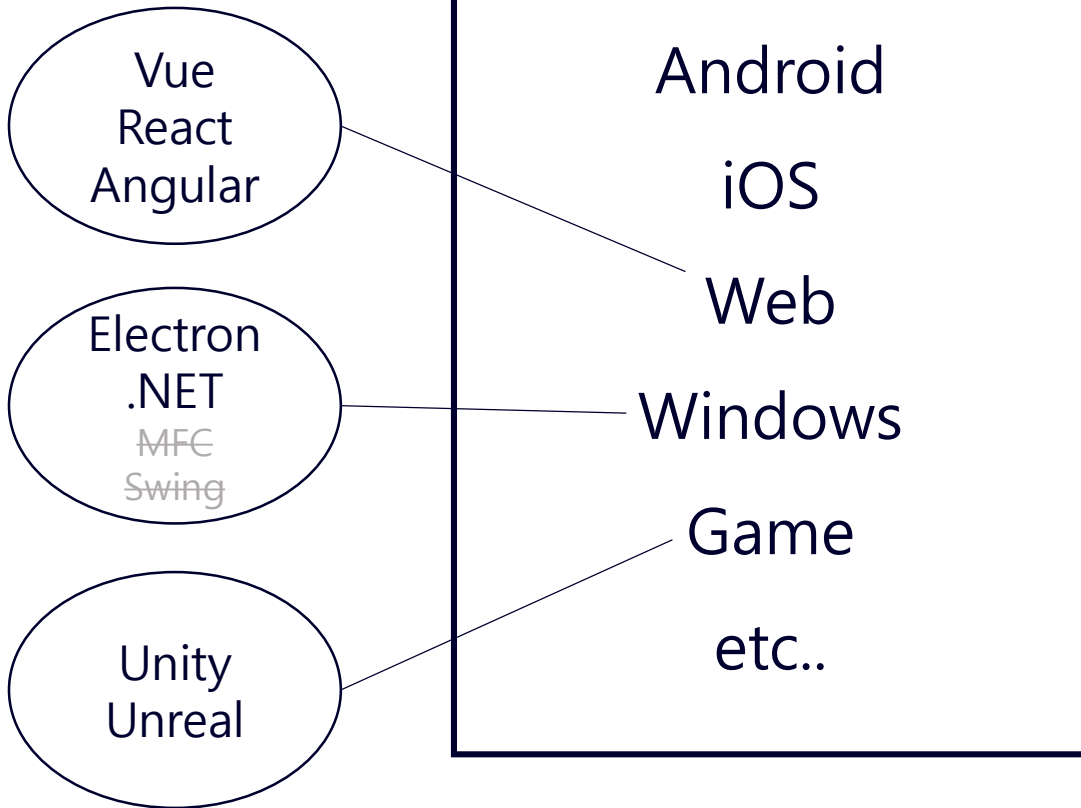
Server

데이터 접근 계층
Back-End
응답

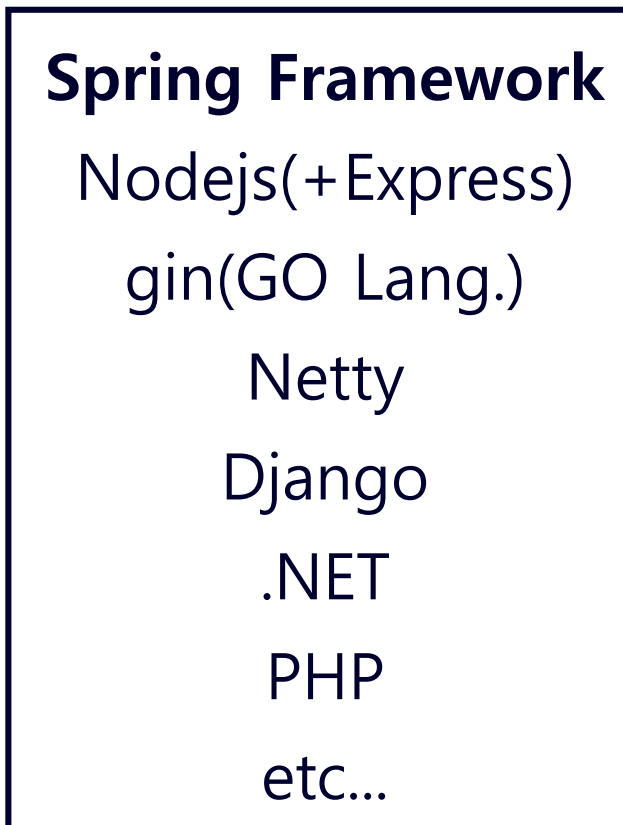
01. Server란?

Server와 Client - 개발

Front End



Back End



01. Server란?

Server의 종류



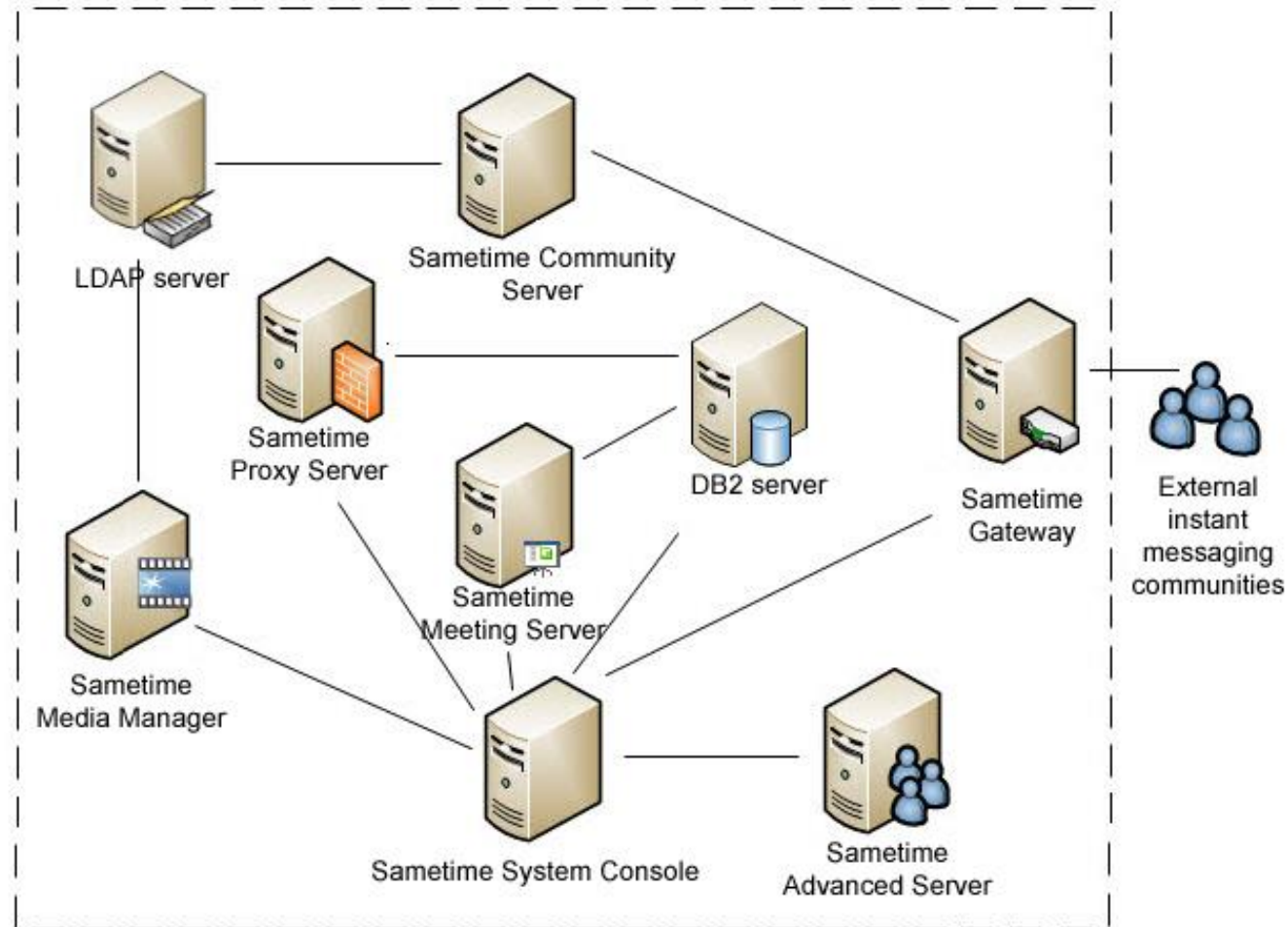
Sametime
Mobile Clients



Browser-based
clients

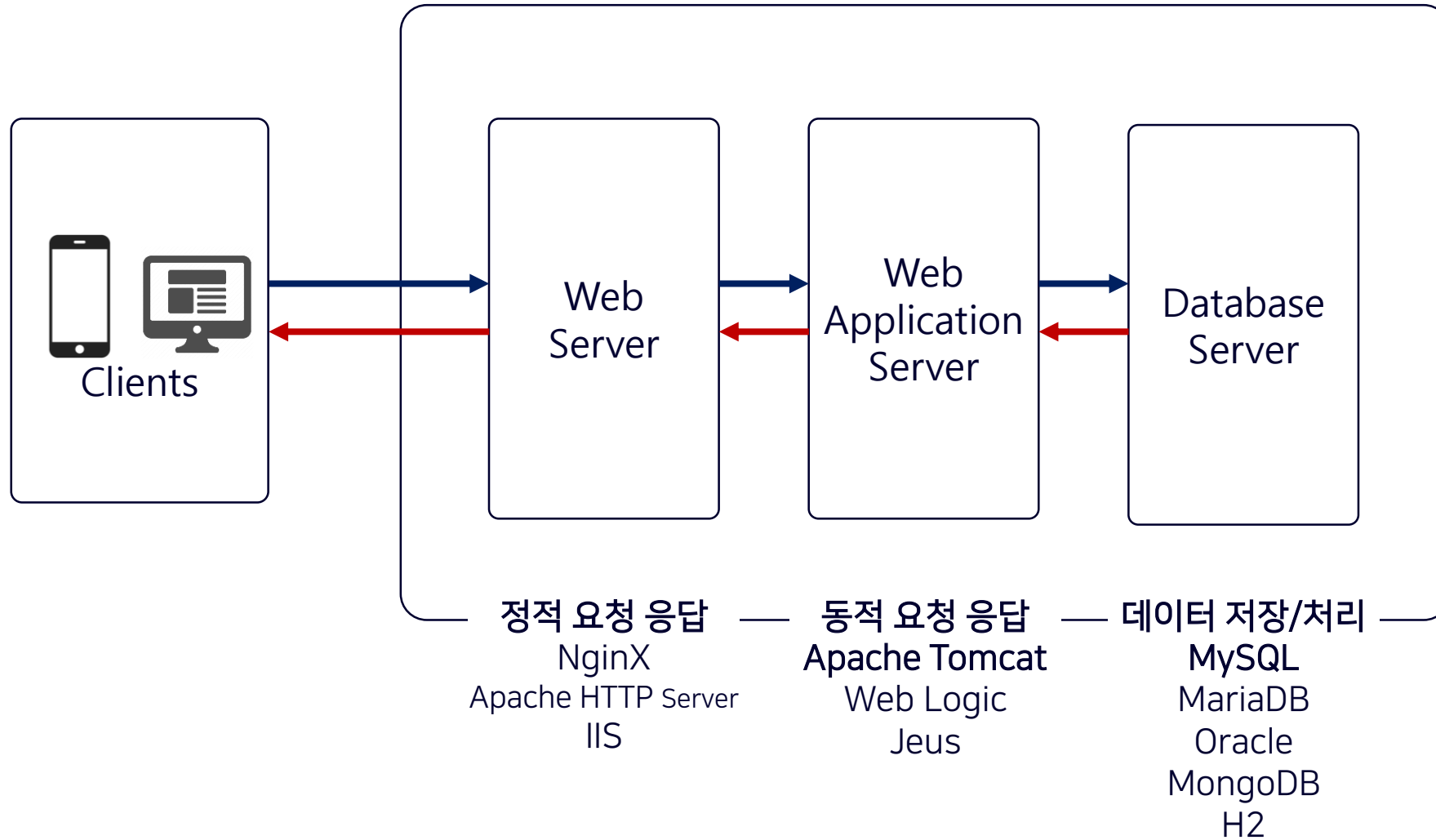


Sametime
Connect
and
Embedded
Clients



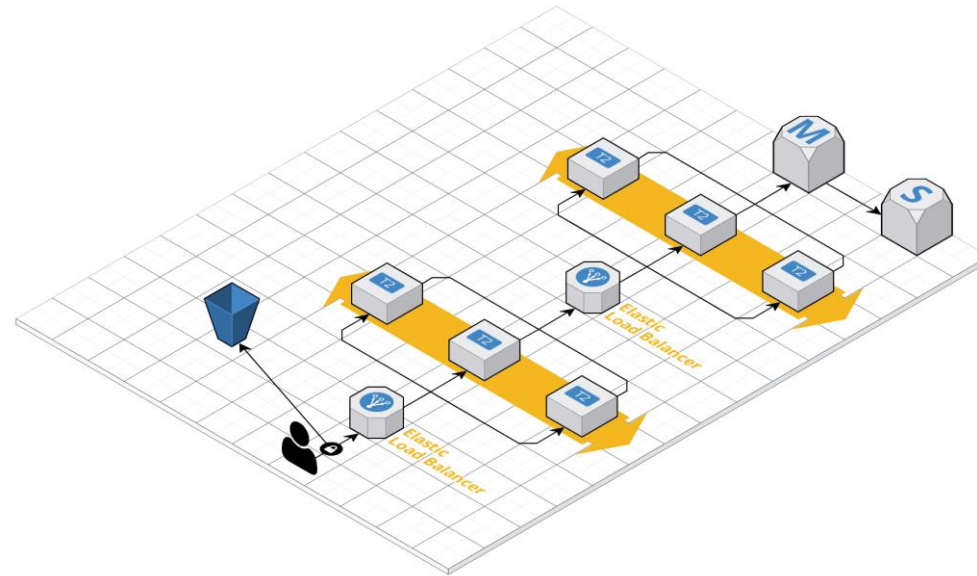
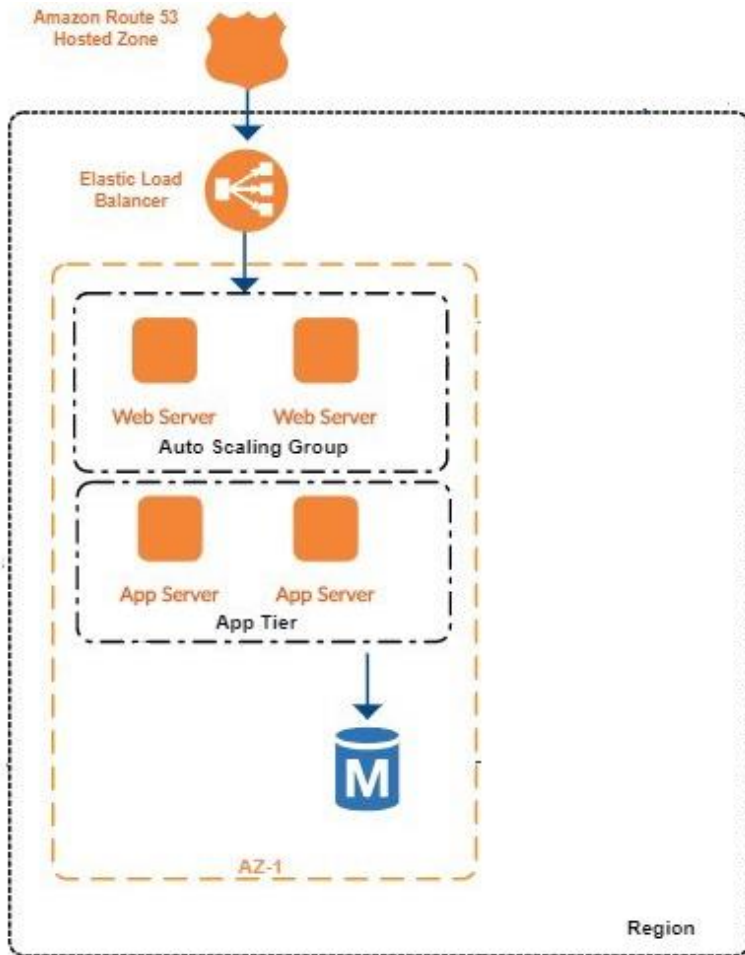
01. Server란?

Server와 Client - 아키텍처

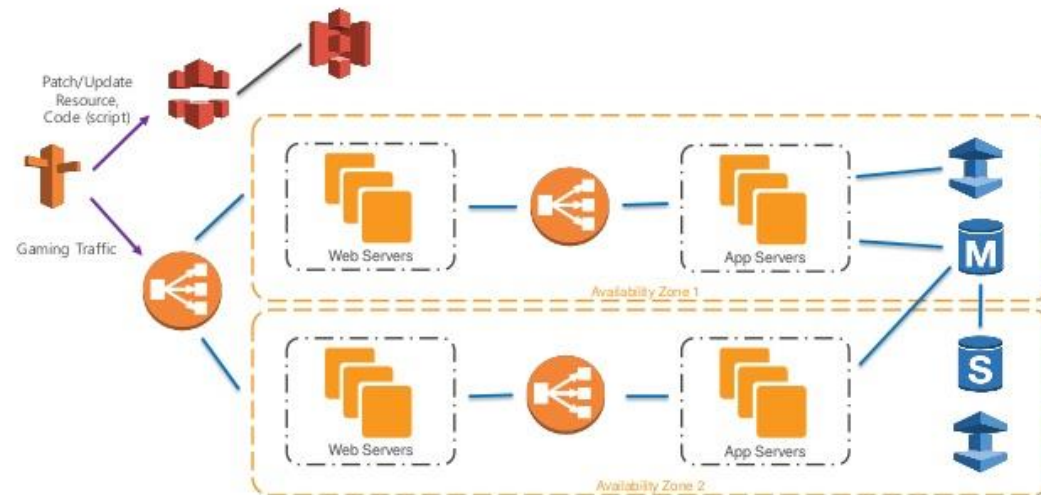


01. Server란?

Server와 Client - 아키텍처 예시

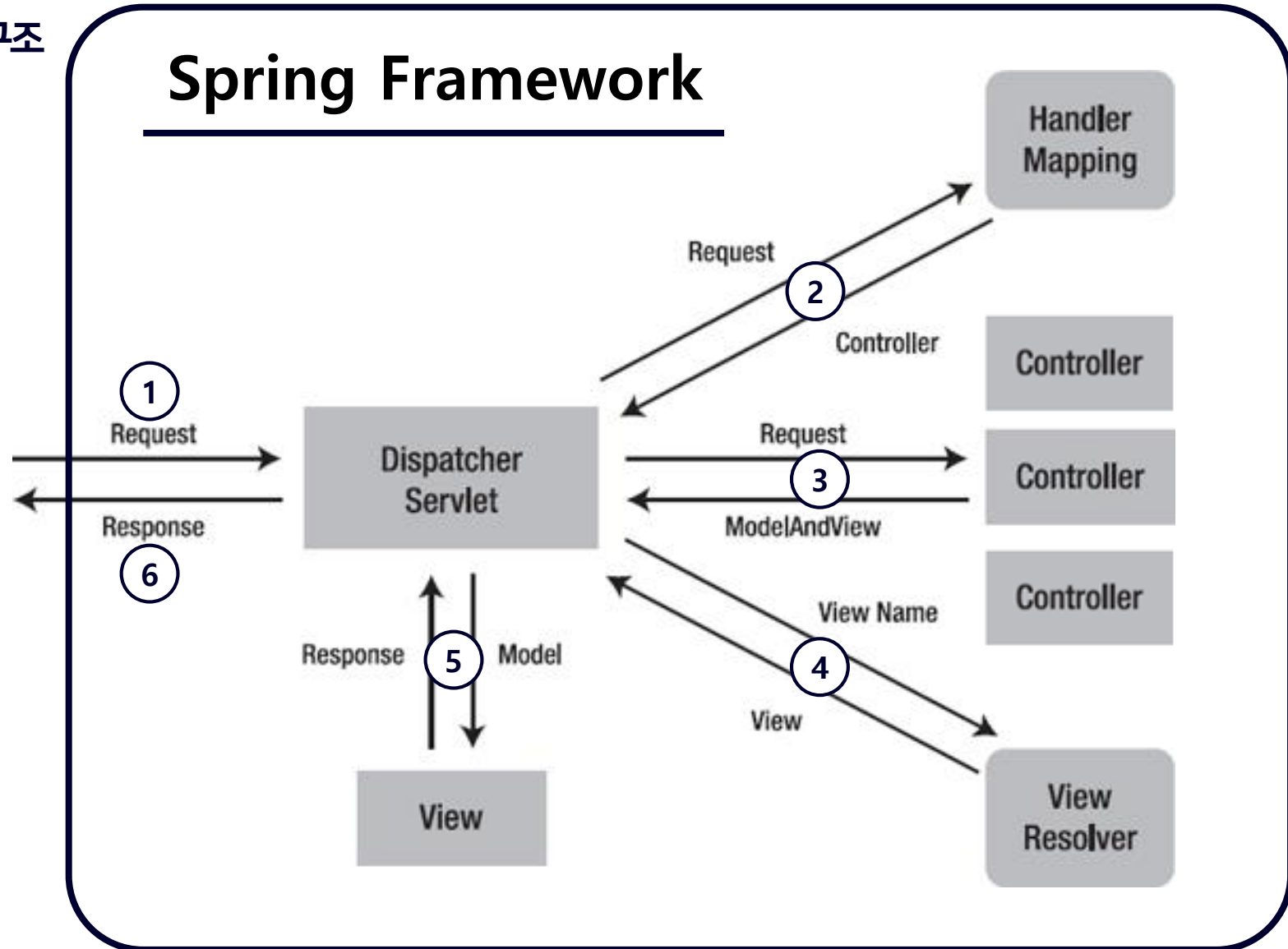


3-Tier Web Application Architecture



01. Server란?

Spring Framework 구조



01. Server란?

Spring Boot



초기 설정의 어려움

- 필요 라이브러리 모두 import
- 라이브러리간 dependency 설정
- xml 형식의 설정(Beans 설정 등)
- Tomcat 따로 설치 및 구동 필요



초기 설정 단순화

- 관련 라이브러리를 버전에 맞추어 import할 수 있음
- Java Annotation으로 간단히 Beans 설정
- 내장 Tomcat 사용

01. Server란?

참고문서

[Front end - Wikipedia](#)

[웹 서버와 웹 애플리케이션 서버 - snippet](#)

[스프링 부트란 \(What is a spring boot?\) :: 엉뚱한 마녀의 상상!?](#)

[5 Benefits of a 3-Tier Architecture - Izenda](#)

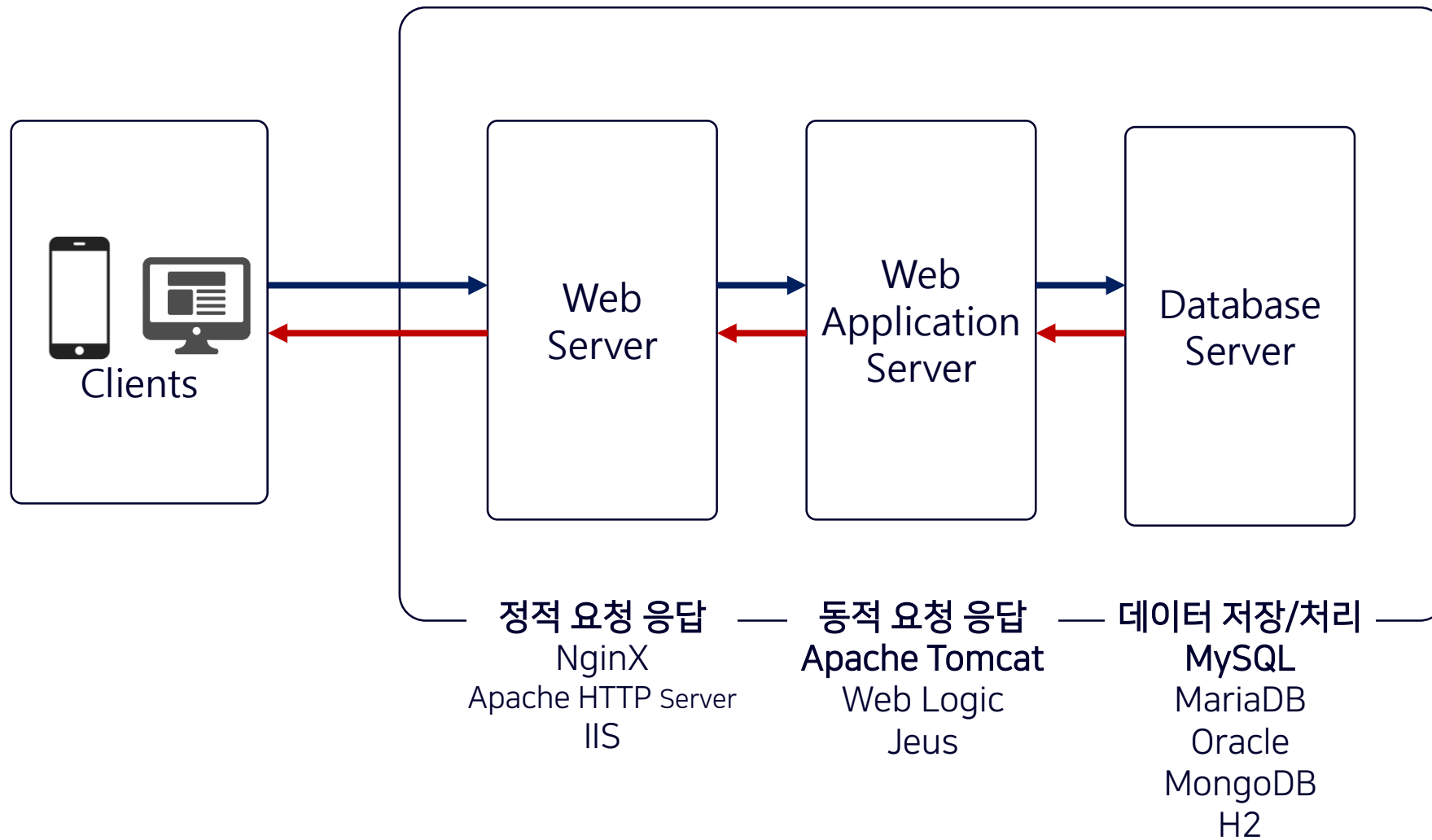
[Front and back ends - Wikipedia](#)

02 실습 (1)

#백견이 불여일타

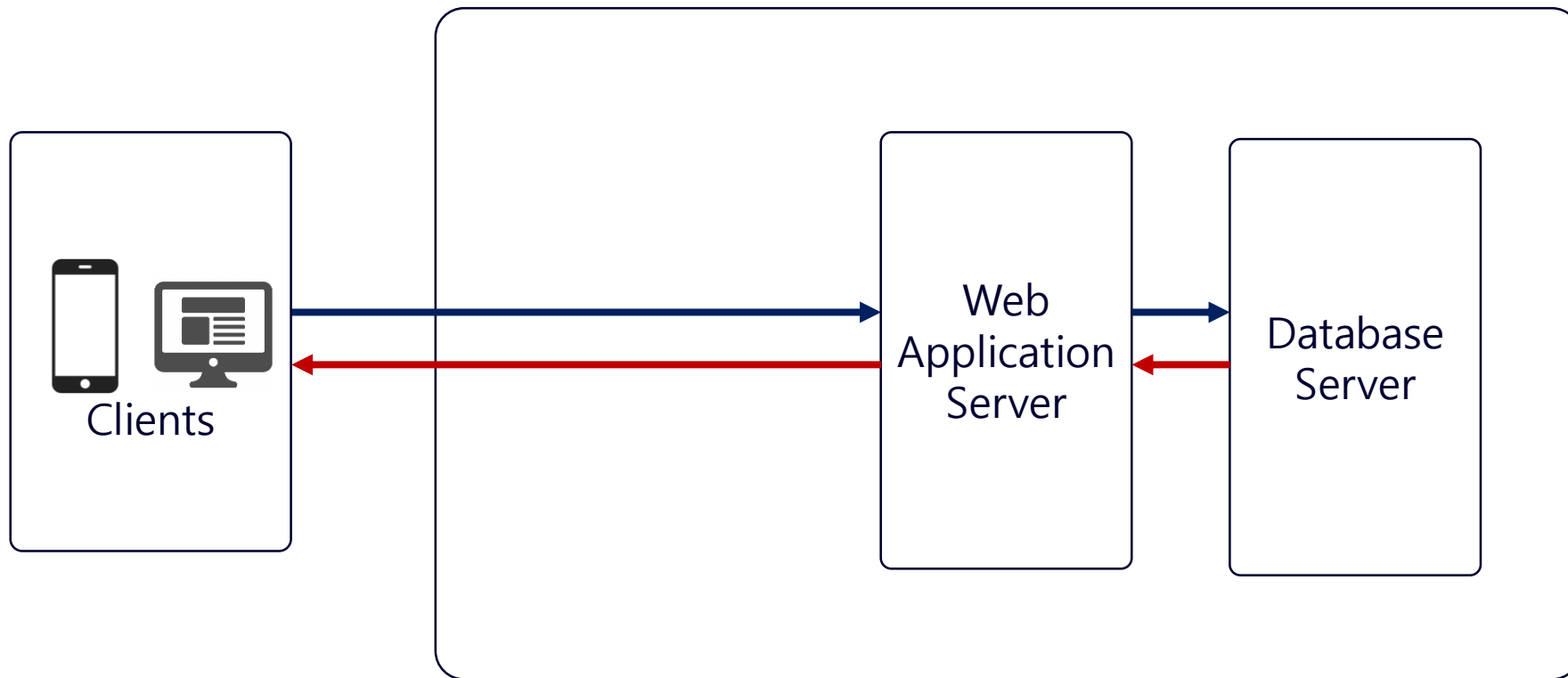
02. 실습 (1)

실습할 아키텍처



02. 실습 (1)

실습할 아키텍처



02. 실습 (1)

개발 환경

백엔드 개발 프레임워크 : Spring Framework, **Spring Boot**

프로그래밍 언어 : **Java** or Kotlin

빌드툴 : Maven or **Gradle**

Persistence Framework : JPA or **MyBatis**

데이터베이스 : **MySQL**

IDE : **IntelliJ** or Eclipse or STS

DB 구축/운영툴 : **MySQL Workbench**

API 테스트툴 : **Postman** or Insomnia

02.

실습 (1)

프로젝트 생성

Spring Initializr Link

<https://start.spring.io>

02.

실습 (1)

프로젝트 생성

Project

Language

Spring Boot

Project Metadata

Dependencies

Maven Project

Gradle Project

Java

Kotlin

Groovy

2.2.0 M5

2.2.0 (SNAPSHOT)

2.1.8 (SNAPSHOT)

2.1.7

Group

com.endlesscreation

Artifact

spring

> Options

Q

☰

4 selected

Search dependencies to add

Web, Security, JPA, Actuator, Devtools...

Selected dependencies

Spring Web Starter

Build web, including RESTful, applications using Spring MVC.
Uses Apache Tomcat as the default embedded container.

✓

Spring Boot DevTools

Provides fast application restarts, LiveReload, and
configurations for enhanced development experience.

✓

MySQL Driver

MySQL JDBC driver.

✓

MyBatis Framework

Persistence framework with support for custom SQL, stored
procedures and advanced mappings. MyBatis couples
objects with stored procedures or SQL statements using a
XML descriptor or annotations.

✓

02. 실습 (1)

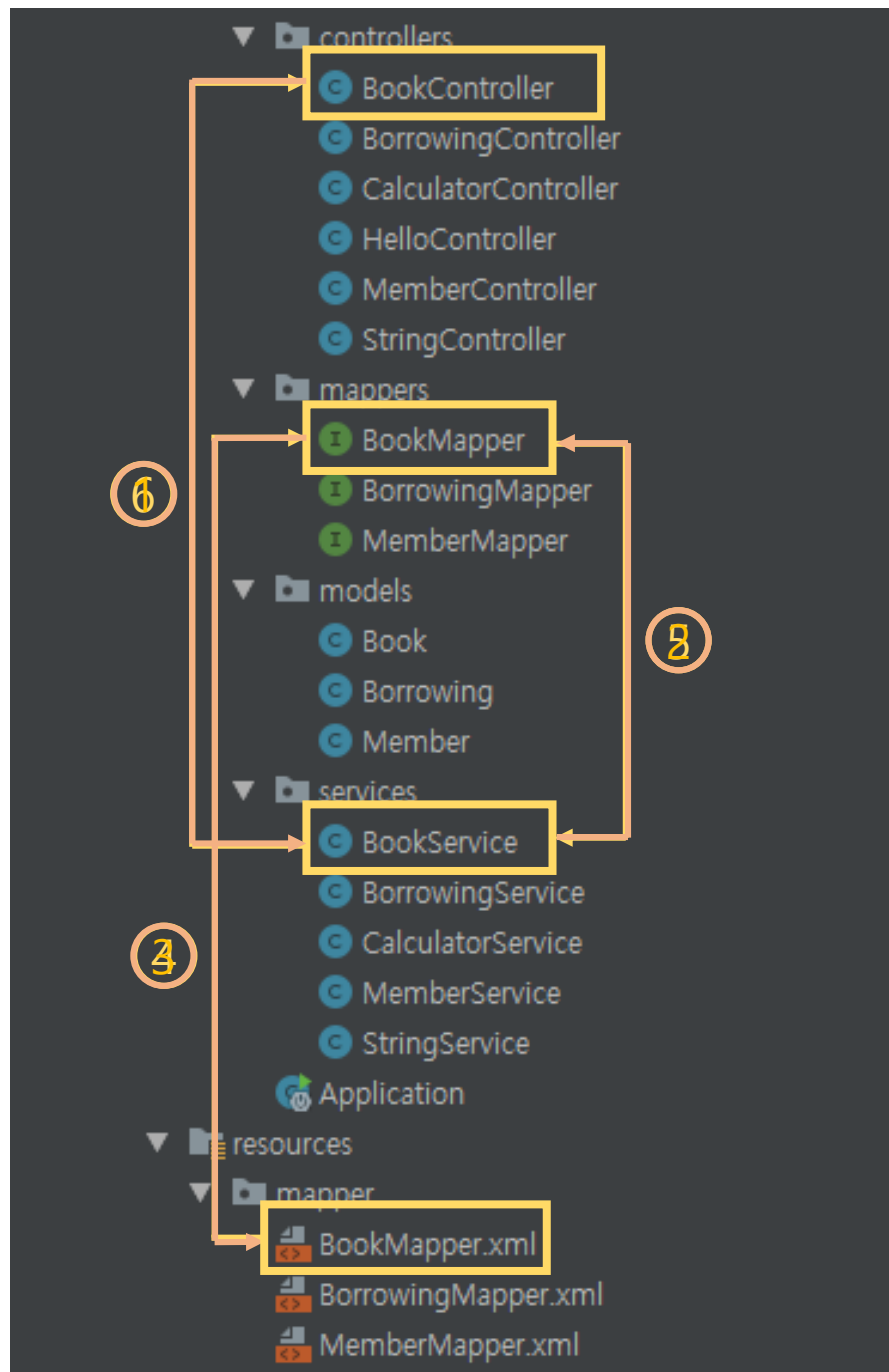
Hello, World!

Hello, World!

02.

실습 (1)

프로젝트 구조 살펴보기



Controller

- Routing 해주는 역할
- URL과 method를 이어주기만 함
- Service를 호출

Service

- 비즈니스 로직 수행
- Mapper를 호출

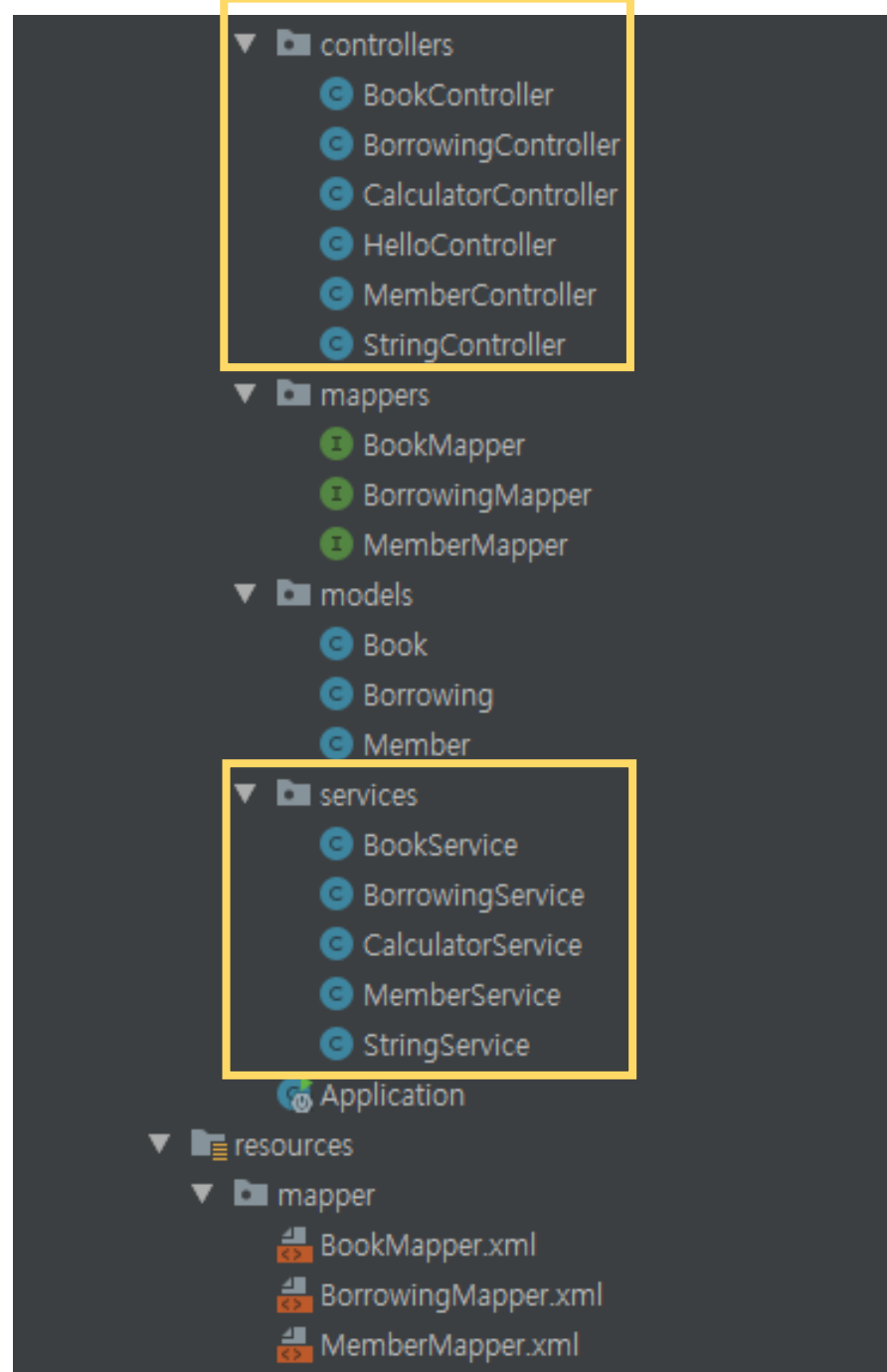
Mapper

- DB에 접근하여 데이터 저장/처리

02.

실습 (1)

프로젝트 구조 살펴보기



Controller

- Routing 해주는 역할
- URL과 method를 이어주기만 함
- Service를 호출

Service

- 비즈니스 로직 수행
- Mapper를 호출

02. 실습 (1)

간단한 API 개발

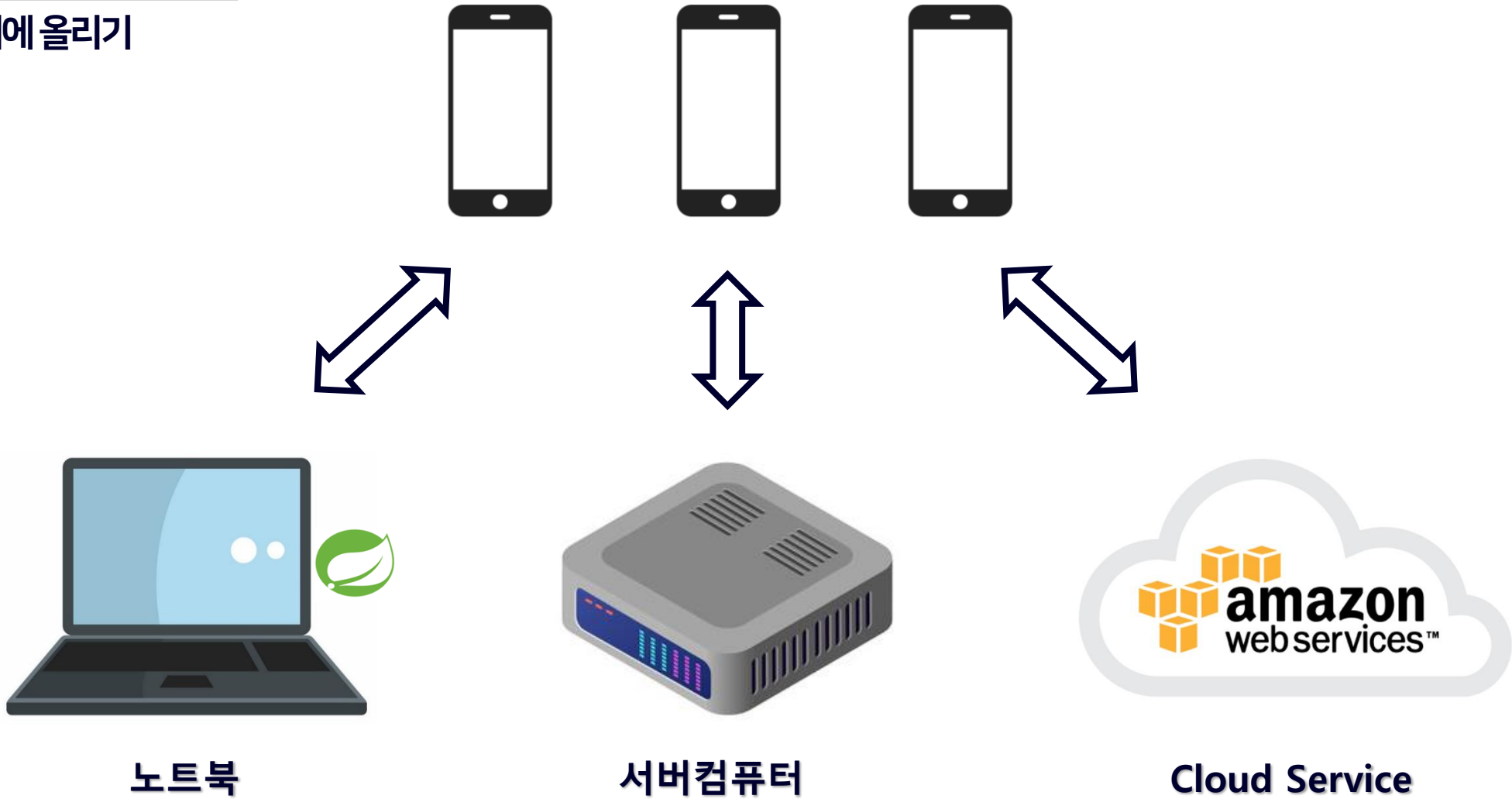
02. 실습 (1)

Controller 나누기

02.

실습 (1)

서버에 올리기



03 배운 내용 정리

#왜 기억을 못하니 $\pi.\pi$

03. 배운 내용 정리

오늘 배운 내용

01 Server란?

Server와 Client
Spring Framework 구조
Spring Boot란?

02 실습 (1)

프로젝트 구조
프로젝트 생성
간단한 API 개발
Controller 나누기

03. 배운 내용 정리

추가로 알아보면 좋을 것들

HTTP (status code, http method, headers, body, etc..)

AOP(Aspect Oriented Programming)

JSON(JavaScript Object Notation) vs XML(eXtensible Markup Language)

WAS(Web Application Server)

IoC(Inversion of Control)

DI(Dependency Injection)

MVC Model(Model-View-Controller)

Bean

기타 디자인 패턴(Singleton, Factory, Proxy, etc..)

RESTful API

Java Annotation

Project Build Tools (Gradle, Maven, Ant, etc...)

AWS (Amazon Web Service)

Persistence Framework (JPA, MyBatis)

03. 배운 내용 정리

어떻게 공부할까?

[백엔드 개발자를 꿈꾸는 학생개발자에게 - NAVER D2 - 네이버](#)

[2018년 웹 개발자가 되기 위한 로드맵 \(번역\) - OKKY](#)

인터넷 강의

- Inflearn / Udemy / Cosera / 생활코딩 / Youtube

Github

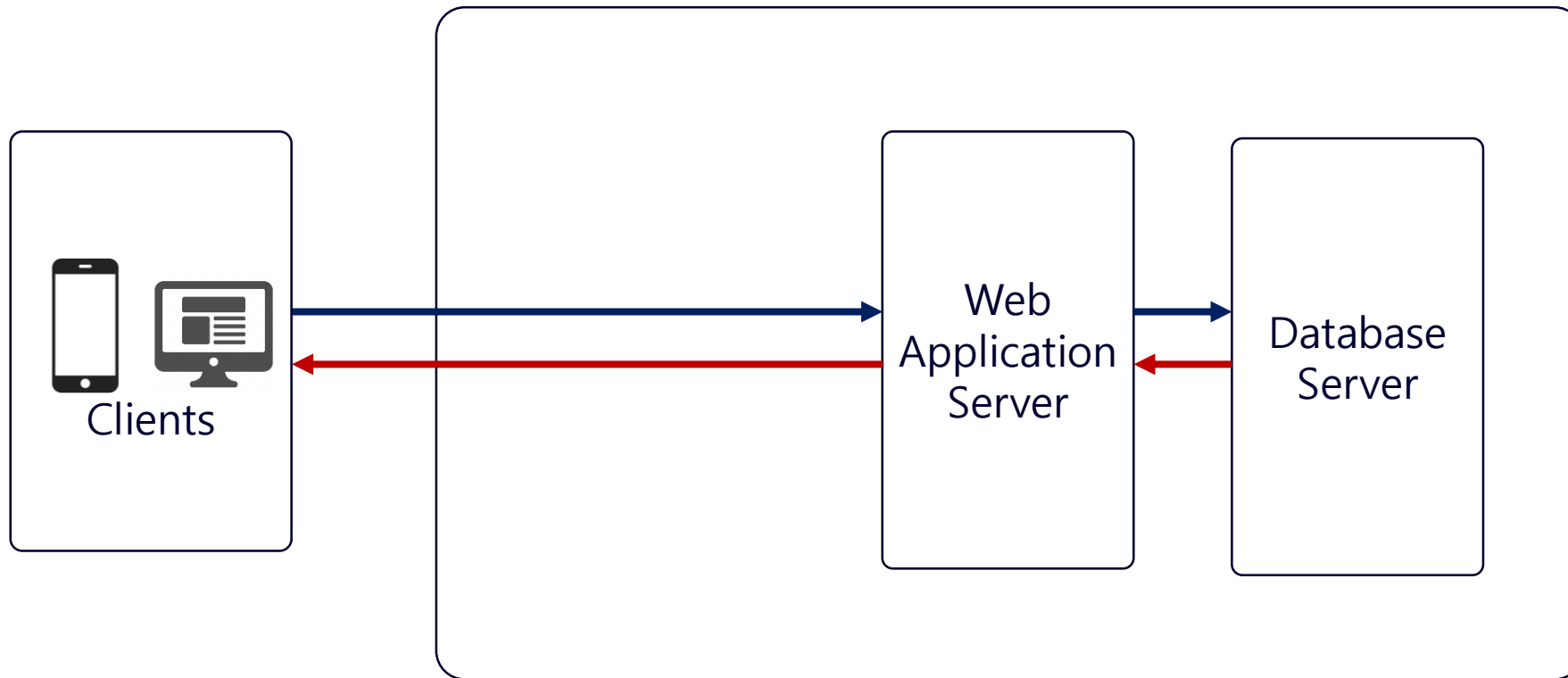
- 다른 사람들의 Github 구경

대외활동

- Naver D2 백엔드 밋업 - 현업 개발자들과 자신과 비슷한 사람들을 만날 수 있음
- Spring Camp - Spring을 사용하는 기업들이 와서 관련 경험과 팁을 발표
- SOPT - 대학생 연합 동아리. 개발, 디자인, 기획 파트로 구성되어 실제 창업 프로젝트 진행
- Nexters - SOPT와 비슷한 연합동아리. 개발자, 디자이너들로 구성
- 인턴 실습
- 공모전
- 등

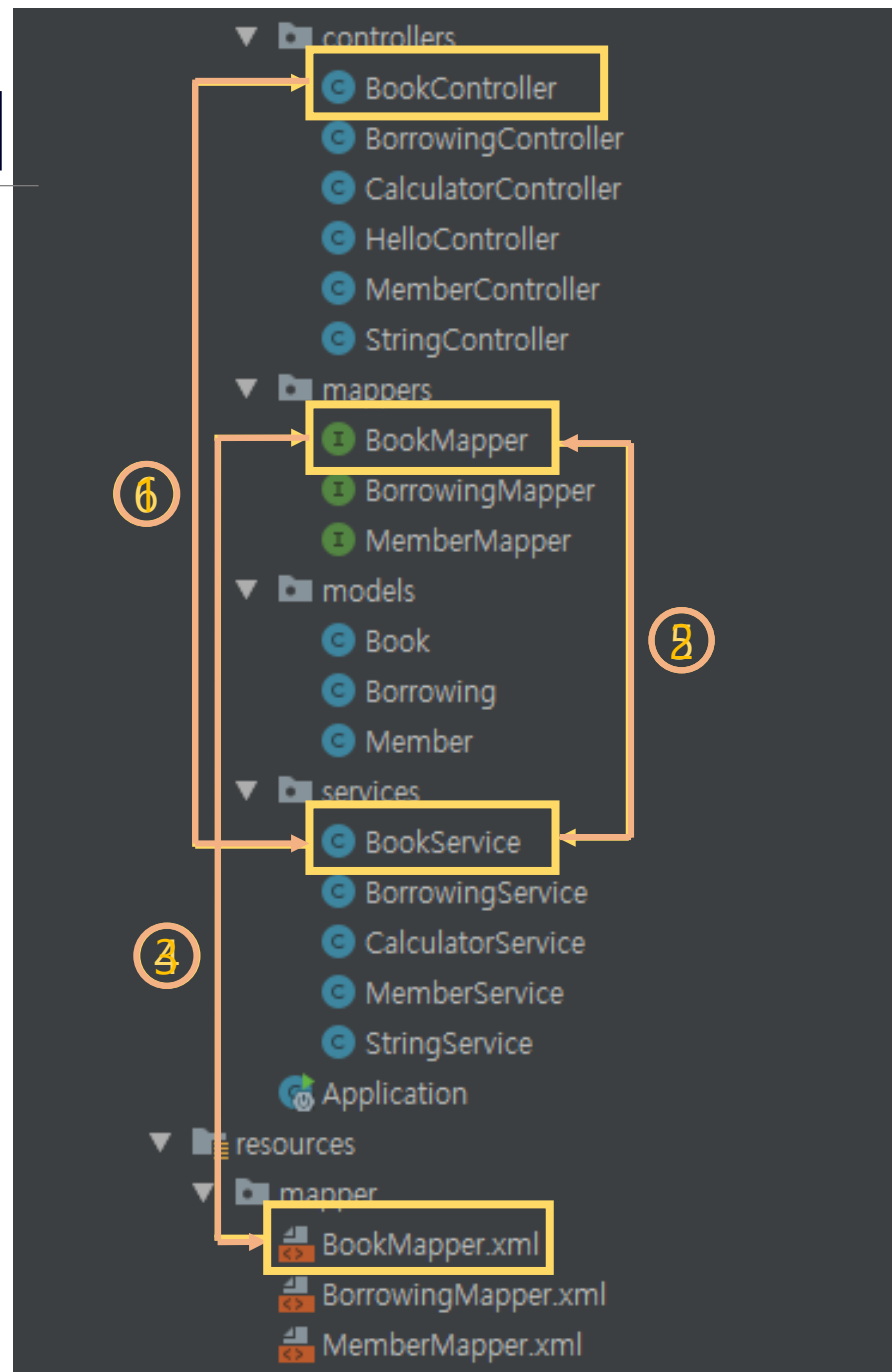
03. 배운 내용 정리

다음 실습내용



03. 배운 내용 정리

다음 실습 내용



Controller

- Routing 해주는 역할
- URL과 method를 이어주기만 함
- Service를 호출

Service

- 비즈니스 로직 수행
- Mapper를 호출

Mapper

- DB에 접근하여 데이터 저장/처리



Thank you

감사합니다

