UNIVERSITY MAKE US CHALLENGE

UMC Server 6주차

REST API & Template

01

과제 점검

02

학습 목표

▪ 학습 목표

- RESTful, REST API 에 대한 이해

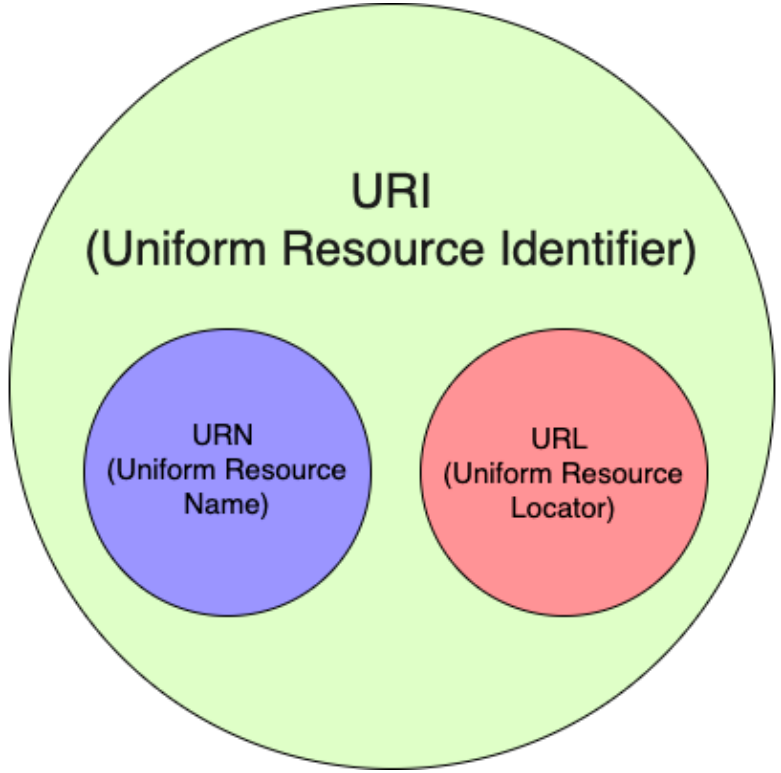
- Server to Server 통신에 대한 이해

- 템플릿 구조 이해와 실제로 적용해보기

03

REST API

- URI? URL? URN?


URI
(Uniform Resource Identifier)
URN
(Uniform Resource Name)
URL
(Uniform Resource Locator)

REST란?

REST란?

HTTP로 정보를 보낼 때,
URI를 어떻게 설계하고 어떤 메서드를 사용할 것인지
표준으로 정해놓은 개발자들 사이의 약속

- **HTTP Method**

GET

POST

PATCH

PUT

DELETE

- **REST API**

URI : 리소스(행위의 목적, 대상) 식별

Method : 해당 리소스에 대한 행위(CRUD)

- **REST API 예시**

  회원목록 조회 API

  회원가입 API

  회원정보 수정 API

  회원탈퇴 API

- **REST API 예시**

회원목록 조회 API		**GET** /members
회원가입 API	→	**POST** /members
회원정보 수정 API		**PATCH** /members
회원탈퇴 API		**DELETE** /members

- REST API

Q.
API를 호출할 때 정보를 담는 방법은?

- **Path Variable / Query String**

GET /students

- **Path Variable / Query String**

GET /students?name=엘리

GET /students?name=엘리&major=컴퓨터공학

GET /students

- **Path Variable / Query String**

GET /students?name=엘리

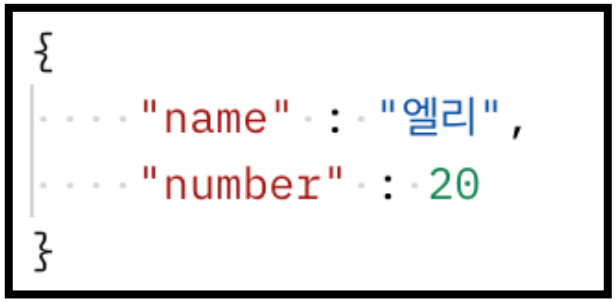
GET /students?name=엘리&major=컴퓨터공학

GET /students

GET /students/20

- **Body**

POST /members



POST /members

```
{
    "name" : "엘리",
    "number" : 20
}
```

POST /posts

```
{
    "title" : "제목",
    "content" : "내용",
    "imgUrl" : ["www.aaa.com", "www.bbb.com"]
}
```

```
{
    "title" : "제목",
    "img" : [
        {
            "url" : "www.aaa.com",
            "content" : "AAA"
        },
        {
            "url" : "www.bbb.com",
            "content" : "BBB"
        }
    ]
}
```

04

RESTful한 설계 방법

- **Restful한 설계 방법**

1. URI에는 되도록 명사를 사용한다.
2. 언더바, 대문자는 사용하지 않는다.
3. 계층관계를 나타낼 때는 '/'를 사용한다.
4. 메소드는 실제 DB에서 동작하는 기준으로 정한다.

05

Validation

06

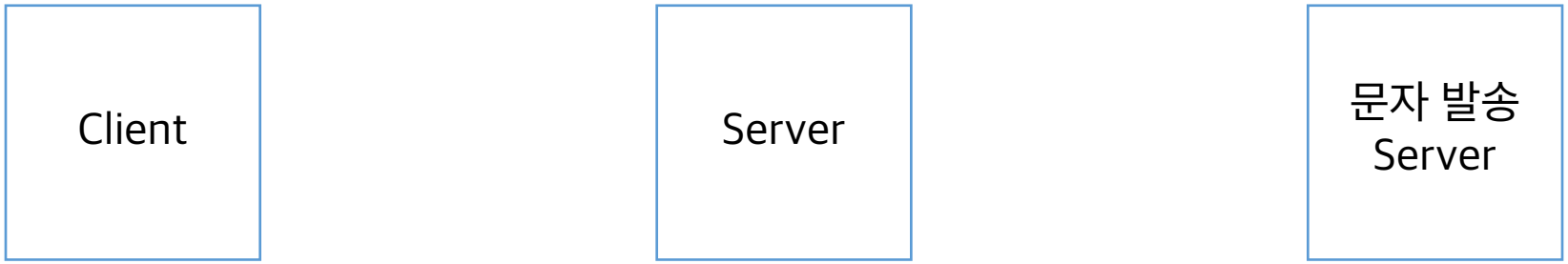
Server to Server

- **Server to server**

Q: 인증번호 보내기, 이메일 보내기

API는 어떻게 만들까?

- **Server to server**



07

Template

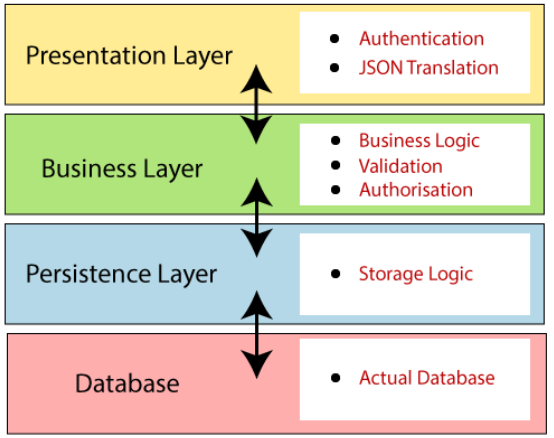
- **프로젝트 구조**

Route

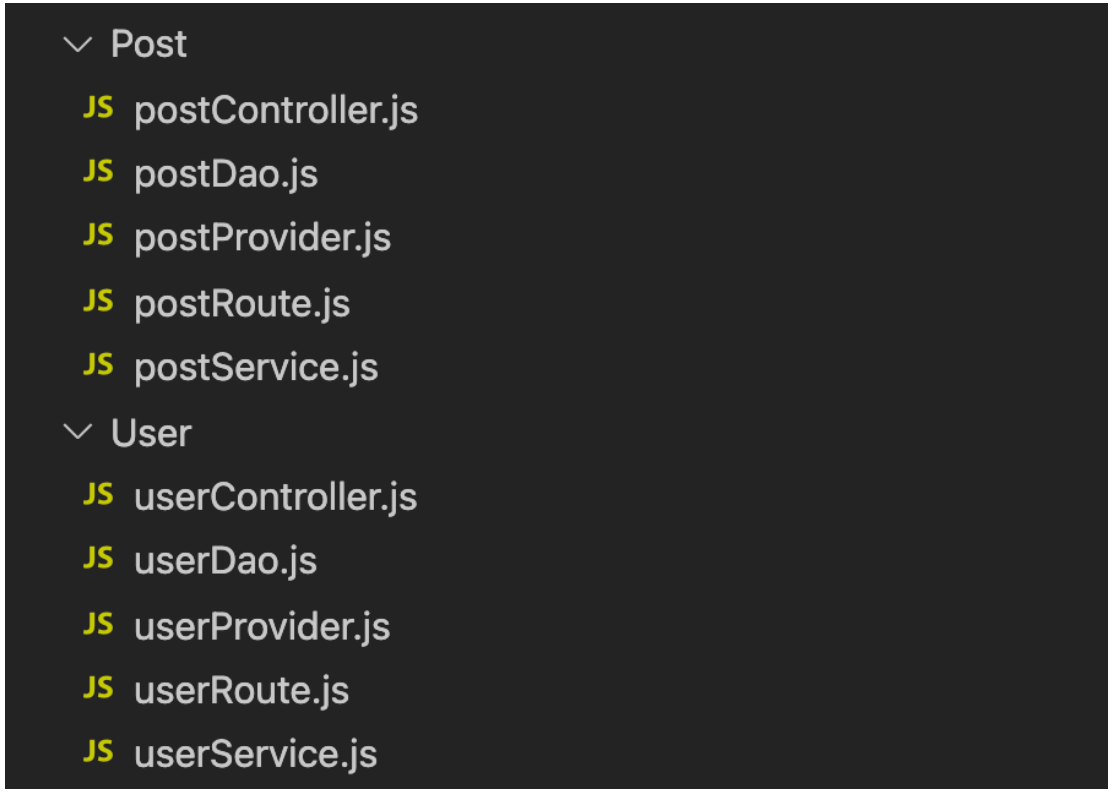
Controller

Service/Provider

DAO



▪ 프로젝트 구조


Post
postController.js
postDao.js
postProvider.js
postRoute.js
postService.js
User
userController.js
userDao.js
userProvider.js
userRoute.js
userService.js

08

실습

09

과제

UNIVERSITY MAKE US CHALLENGE

- **과제**

- 지난번 개발했던 API(+추가 구현) 템플릿에 적용해서 REST API로 발전시키기

- Validation 꼼꼼하게 & 최대한 내부 로직 비슷하게

❗ 최소 10개 ❗

10

NEXT CLASS

Token, Paging, Local Storage