

REST API + VUE + Axios

REST API

REST의 정의

“Representational State Transfer” 의 약자 자원을 이름(자원의 표현)으로 구분하여 해당 자원의 상태(정보)를 주고 받는 모든 것을 의미한다. 즉, 자원(resource)의 표현(representation)에 의한 상태 전달

자원(resource)의 표현(representation)

- 자원: 해당 소프트웨어가 관리하는 모든 것
- 자원의 표현: 그 자원을 표현하기 위한 이름

상태(정보) 전달

- 데이터가 요청되어지는 시점에서 자원의 상태(정보)를 전달한다.
- JSON 혹은 XML를 통해 데이터를 주고 받는 것이 일반적이다.

REST의 구체적인 개념

HTTP URI(Uniform Resource Identifier)를 통해 자원(Resource)을 명시하고, HTTP Method(POST, GET, PUT, DELETE)를 통해 해당 자원에 대한 CRUD Operation을 적용하는 것을 의미한다. 즉, REST는 자원 기반의 구조 (ROA, Resource Oriented Architecture) 설계의 중심에 Resource가 있고 HTTP Method를 통해 Resource를 처리 하도록 설계된 아키텍처를 의미한다. 웹 사이트의 이미지, 텍스트, DB 내용 등의 모든 자원에 고유한 ID인 HTTP URI를 부여한다.

CRUD Operation

- Create : 생성(POST)
- Read : 조회(GET)
- Update : 수정(PUT)
- Delete : 삭제(DELETE)
- HEAD: header 정보 조회(HEAD)

HTTP 응답

Status Code	Description
200 (성공)	서버가 요청을 제대로 처리했다는 뜻이다. 이는 주로 서버가 요청한 페이지를 제공했다는 의미로 쓰인다.
201 (콘텐츠 생성)	서버가 요청을 처리 했으며, 자원이 생성되었음을 나타냄
204 (콘텐츠 없음)	서버가 요청을 성공적으로 처리했지만 콘텐츠를 제공하지 않는다.

Status Code	Description
400 (잘못된 요청)	서버가 요청의 구문을 인식하지 못했다.
401 (권한 없음)	이 요청은 인증이 필요하다. 서버는 로그인에 필요한 페이지에 대해 이 요청을 제공할 수 있다. 상태 코드 이름이 권한 없음(Unauthorized)으로 되어 있지만 실제 뜻은 인증 안됨(Unauthenticated)에 더 가깝다.
403 (Forbidden, 금지됨)	서버가 요청을 거부하고 있다. 예를 들자면, 사용자가 리소스에 대한 필요 권한을 갖고 있지 않다. (401은 인증 실패, 403은 인가 실패라고 볼 수 있음)
410 (사라짐)	서버는 요청한 리소스가 영구적으로 삭제되었을 때 이 응답을 표시한다. 404(찾을 수 없음) 코드와 비슷하며 이전에 있었지만 더 이상 존재하지 않는 리소스에 대해 404 대신 사용하기도 한다. 리소스가 영구적으로 이동된 경우 301을 사용하여 리소스의 새 위치를 지정해야 한다.
412 (사전 조건 실패)	서버가 요청자가 요청 시 부과한 사전조건을 만족하지 않는다.

REST API TEST

<https://reqres.in/>

입력

POST /api/users

```
Request
// body
{
  "name": "morpheus",
  "job": "leader"
}

Response
// status 201
{
  "name": "morpheus",
  "job": "leader",
  "id": "827",
  "createdAt": "2021-10-27T02:04:36.632Z"
}
```

수정

PUT /api/users/{id}

```
Request
// body
{
  "name": "morpheus",
  "job": "zion resident"
}

Response
// status 200
{
  "name": "morpheus",
  "job": "zion resident",
  "updatedAt": "2021-10-27T02:04:44.462Z"
}
```

수정(부분)

PATCH /api/users

```
Request
// body
{
  "name": "morpheus"
}

Response
// status 200
{
  "name": "morpheus",
  "job": "zion resident",
  "updatedAt": "2021-10-27T02:05:05.413Z"
}
```

삭제

DELETE /api/users/{id}

```
Response
// status 204
{
  "name": "morpheus",
  "job": "zion resident",
  "updatedAt": "2021-10-27T02:05:05.413Z"
}
```

조회

GET /api/users

```
Request
// query string
page={page}

Response
// status 200
{
  "page": 2,
  "per_page": 6,
  "total": 12,
  "total_pages": 2,
  "data": [
    {
      "id": 7,
      "email": "michael.lawson@reqres.in",
      "first_name": "Michael",
      "last_name": "Lawson",
      "avatar": "https://reqres.in/img/faces/7-image.jpg"
    }
  ]
}
```

Axios

Axios는 브라우저, Node.js를 위한 Promise API를 활용하는 HTTP 비동기 통신 라이브러리이다.

Axios 특징

- 운영 환경에 따라 브라우저의 XMLHttpRequest 객체 또는 Node.js의 HTTP API 사용
- Promise(ES6) API 사용
- 요청과 응답 데이터의 변형
- HTTP 요청 취소 및 요청과 응답을 JSON 형태로 자동 변경

개발

- 'https://reqres.in/' 에서 제공하는 REST API 를 활용하여 CRUD 개발
- vue cli 를 활용하여 vue 프로젝트 생성 vue 2.x
- axios 라이브러리 사용
- 입력, 수정, 등록 각각 화면 개발