**코멘토 직무부트캠프 2주차 과제 (2)**

김동오

**REST API란?**

* RESTful한 API를 의미한다. REST원칙을 준수하는 API라고도 볼 수 있다.
* REpresentational State Transfer의 약자이다.
* HTTP 통신을 활용하기 위해 고안된 아키텍쳐이다.
* 클라이언트는 URI로 표현된 자원을 HTTP 메서드를 사용해 CRUD 연산을 할 수 있다.
* State Transfer는 자원의 상태를 주고받는 것, 즉 요청받은 자원의 상태를 전달하는 것을 의미한다.
* 결국 REST는 자원을 명시해 연산을 수행하고 상태를 주고받는 것을 의미한다.

Cf) REST는 다음과 같은 특징이 있다.

* **일관된 인터페이스** : 자원을 나타내는 URI를 HTTP 메서드로 조작하는 일관된 인터페이스를 사용한다. 따라서 HTTP를 따르는 모든 플랫폼에서 REST를 사용할 수 있다.
* **클라이언트/서버 구조** : 클라이언트와 서버 간에 요청-응답의 독립적인 구조를 갖는다. 클라이언트는 서버에 요청을 보내고 응답을 대기한다. 서버는 자원을 가지고 있으며 클라이언트의 요청에 응답한다.
* **무상태성** : 서버에서는 클라이언트의 요청을 저장하거나 관리하지 않는다. 서버는 클라이언트의 요청에 대한 처리와 응답만 한다. 사용자 인증, 로그인 정보 등은 클라이언트에서 직접 관리한다.
* **캐싱 가능** : HTTP 표준을 사용하므로 클라이언트는 이전에 서버로부터 받은 응답을 저장 및 재사용하는 캐싱을 할 수 있다. 캐싱은 클라이언트의 많은 요청으로부터 서버 부하를 줄여 주고, 클라이언트는 비교적 빨리 응답을 받을 수 있게 한다.
* **자체 표현 구조** : REST API는 자원, 행위, 표현으로 구성되어 REST API를 보고 어떤 요청을 하는지 알 수 있다.

**1) HTTP 통신이란?**

* 어플리케이션 계층(OSI 7계층), 즉 유저 어플리케이션 단에서 통신하는 방법 중 하나
* 그 중 HTTP protocol을 따르는 통신 방법
* CRUD기능(CREATE, READ, UPDATE, DELETE)을 HTTP method로 요청할 수 있음
* 각각 POST, GET, PUT/PATCH, DELETE로 대응됨

// PUT : 전부 수정 vs PATCH : 일부 수정

-

**2) 브라우저에 URL을 입력 후 요청하여 서버에서 응답하는 과정**

2-1) 사용자가 브라우저에 URL로 자원 요청

2-2) 브라우저가 URL을 DNS 서버에 전달

2-3) DNS서버에서 URL에 맞는 IP 주소를 반환

2-4) 브라우저는 반환받은 IP 주소로 웹 서버와 TCP/IP 연결을 하고 HTTP요청을 보냄

2-5) 웹 서버는 받은 HTTP요청에 응답(웹 페이지와 필요한 리소스를 반환)

2-6) 웹 브라우저는 받은 응답을 바탕으로 사용자에게 웹 페이지를 보여줌