

1) Rest API

Representational State Transfer API의 약어

클라이언트와 서버 간의 통신을 위한 규칙과 제약을 정의하여 자원을 표현하고, 해당 자원에 대한 상태를 주고 받는 방식을 제공

REST API의 주요 특징

1. 자원(Resource) : 모든 것을 자원으로 표현하며, 각 자원은 고유한 식별자(URI)를 가지고 있다.
2. HTTP 메서드 활용 : HTTP 프로토콜의 메서드(GET, POST, PUT, DELETE 등)를 사용하여 자원에 대한 행위를 나타낸다. GET은 자원을 조회할 때, POST는 자원을 생성할 때, PUT은 자원을 수정할 때, DELETE는 자원을 삭제할 때 사용된다.
3. 표현적인 상태 전달 : 클라이언트와 서버 간의 통신에서 상태 정보를 전달하기 위해 표현적인 데이터를 사용합니다. 일반적으로 JSON 또는 XML 형식으로 데이터를 주고 받는다.

2) HTTP 통신

HyperText Transfer Protocol의 약어

클라이언트와 서버 간의 통신을 위해 사용되는 프로토콜

웹 브라우저와 웹 서버 사이의 데이터 전송을 담당하며, 주로 HTML 문서를 주고 받는 데 사용된다.

HTTP는 요청(Request)과 응답(Response)의 형태로 이루어져 있다.

3) URL을 입력하여 서버에서 응답을 받는 과정

URL 입력 -> DNS 조회 -> TCP 연결 -> HTTP 요청 전송 -> 서버 처리 -> HTTP 응답 전송 -> 응답 처리

URL 입력 : 브라우저의 주소창에 요청하고자 하는 웹 페이지의 URL을 입력한다.

DNS 조회 : 브라우저는 입력한 호스트 이름을 IP 주소로 변환하기 위해 DNS 서버에 호스트 이름을 조회한다. DNS 서버는 호스트 이름에 해당하는 IP 주소를 반환한다.

TCP 연결 : 브라우저는 서버의 IP 주소와 포트 번호를 기반으로 TCP/IP 연결을 한다.

HTTP 요청 전송 : TCP 연결이 되면, 브라우저는 HTTP 요청 메시지를 서버로 전송한다.

서버 처리 : 서버는 받은 HTTP 요청을 처리하고, 요청에 따라 필요한 작업을 수행한다.

HTTP 응답 전송 : 서버는 처리된 결과를 HTTP 응답 메시지에 담아 브라우저로 전송한다.

응답 처리 : 브라우저는 받은 HTTP 응답을 처리하여 화면에 표시한다. 응답 본문을 해석하여 웹 페이지를 렌더링 하거나, 다운로드 할 파일을 저장하거나, JavaScript 코드를 실행하는 등의 작업을 수행한다.