



# Chap44 REST API

## ▼ REST

: HTTP를 기반으로 클라이언트가 서버의 리소스에 접근하는 방식을 규정한 아키텍처

## ▼ REST API

: REST를 기반으로 서비스 API를 구현한 것

## ▼ RESTful

: REST의 기본 원칙을 성실히 지킨 서비스 디자인

## ▼ REST API의 구성

REST API는 자원, 행위, 표현의 3가지로 구성된다. REST는 자체 표현 구조로 구성되어 REST API 만으로 HTTP요청의 내용을 이해할 수 있다.

구성요소	내용	표현방법
자원	자원	URI(엔드포인트)
행위	자원에 대한 행위	HTTP 요청 메서드
표현	자원에 대한 행위의 구체적 내용	페이로드

## ▼ REST API 설계원칙

1. URI는 리소스를 표현해야 한다.

### ▼ URL과 URI의 차이

URI= 식별자, URL=식별자+위치

elancer.co.kr은 URI다. 왜냐하면 리소스의 이름만 나타내기 때문입니다.

반면, https://elancer.co.kr은 URL이다. 이름과 더불어, 어떻게 도달할 수 있는지 위치까지 함께 나타내기 때문이. (프로토콜 'https' 포함)

URI는 리소스를 표현하는 데 중점을 두어야한다. 리소스를 식별할 수 있는 이름은 동사보다는 명사를 사용한다.

```
#bad
GET /getTods/1
GET /todos/show/1

#good
GET /todos/1
```

## 2. 리소스에 대한 행위는 HTTP요청 메서드로 표현한다.

HTTP 요청 메서드는 클라이언트가 서버에게 요청의 종류와 목적을 알리는 방법이다. 주로 5가지 요청 메서드(GET, POST, PUT, PATCH, DELETE 등)를 사용하여 CRUD를 구현한다.

HTTP요청 메서드	종류	목적	페이로드
Get	index/retrieve	모든/특정 리소스취득	x
POST	create	리소스 생성	o
PUT	replace	리소스 전체 교체	o
PATCH	modify	리소스의 일부 수정	o
DELETE	delete	모든/특정 리소스 삭제	x