# ✓ RESTful이란?

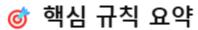
\*\*웹의 자원(Resource)\*\*을 URL로 표현하고,

\*\*행위(Action)\*\*는 HTTP 메서드(GET, POST, PUT, DELETE 등)로 구분하는 방식입니다.

### 👄 햄버거 가게에 비유해서 쉽게!

자원이 "햄버거"라면:

| HTTP 메서드 | URL       | 의미                  |
|----------|-----------|---------------------|
| GET      | /burger   | 햄버거 목록 주세요          |
| GET      | /burger/1 | 1번 햄버거 주세요          |
| POST     | /burger   | 햄버거 하나 만들어 주세요 (주문) |
| PUT      | /burger/1 | 1번 햄버거 내용 바꿔주세요     |
| DELETE   | /burger/1 | 1번 햄버거 없애주세요 (취소)   |



| 개념              | 설명                         |
|-----------------|----------------------------|
| 자원(Resource)    | URL로 표현 (예: /api/persons)  |
| 행위(Method)      | HTTP 메서드로 표현 (GET, POST 등) |
| 무상태성(Stateless) | 서버는 요청자 상태를 기억하지 않음        |
| 일관된 구조          | 모든 자원에 대해 같은 방식 사용         |

## 🧘 RESTful API 예시 (사람 관리)

| 기능       | 메서드    | URL               |
|----------|--------|-------------------|
| 모든 사람 조회 | GET    | /api/persons      |
| 특정 사람 조회 | GET    | /api/persons/{id} |
| 사람 추가    | POST   | /api/persons      |
| 사람 정보 수정 | PUT    | /api/persons/{id} |
| 사람 삭제    | DELETE | /api/persons/{id} |

# 🦞 RESTful을 사용하면?

- URL은 명사만 사용
- 동작은 HTTP 메서드로 구분
- 직관적이고 표준화된 API 설계 가능