API Design

REST

RE → Representational → basically means that resources on the web are represented in a specific format. The same resource can have different representation based on clients needs.

S → State → basically referes to current condition or attribute of a resource. Each resources has a state that can be transfered between client & server.

T → Transfer → basically indicates the movement of resource between client & server, & they can exchange different representation of the same resource.

$$\underbrace{\text{https}}_{\text{scheme}}\text{://}\underbrace{\text{api.example.com}}_{\text{subdomain}}/\underbrace{\text{v}_1}_{\text{version}}/\underbrace{\text{books}}_{\text{name of resource}}$$

Some industry standard practices are:

☑ resource should always be plural

books ✓ book ✗

☑ don't use space or underscore

☑ use all lowercase & hyphen for space

In simple term we need to use slug

human readable, URL-safe identifier

GET/api/articles/what-is-an-api ✓

GET/api/articles/1234 ✗

Idempotency → is a proparty of an operation where performing it multiple times produces the same result as performing it once.

Idempotent → GET, PUT, DELETE

Non-Idempotent → POST, PATCH (usually)

One more thing about the POST method:

If we have a custom action which don't fall under any of the CRUD operation, we use POST method

Pagination → a technique used to split large set of data into smaller manageable chunks, so you don't overload the server or the client with too much information at once.

Common approaches

☑ Ofset based pagination

```
GET/users?limit=10&offset=20
```

Meaning "Give me 10 users, starting from the 21st user"

☑ Paged-based pagination

```
GET/users?page=3&pageSize=10
```

Meaning "Give me the 3rd page, 10 users per page"

☑ Cursor-based/keyset pagination

```
GET/users?cursor=abcd1234&limit=10
```

Meaning "Give me 10 users after the cursor ab.."