What is REST

REST stands for **Representational State Transfer**. It's an architectural style for designing networked applications.

👉 The Core Concepts of REST

1. Resources: Data as Nouns, Not Verbs

When we build web applications, we primarily work with data - creating it, reading it, updating it, and deleting it (CRUD operations). In REST:

- Each type of data on the server is considered a **resource**
- Resources are identified using **nouns** in URLs, not verbs or actions
- Example resources in a Job Application: Jobs, Employees, Admin profiles

- **Traditional URL (verb-based):**

  /viewAllJobs

  /addJob

  /deleteJob

- **REST URL (noun-based):**

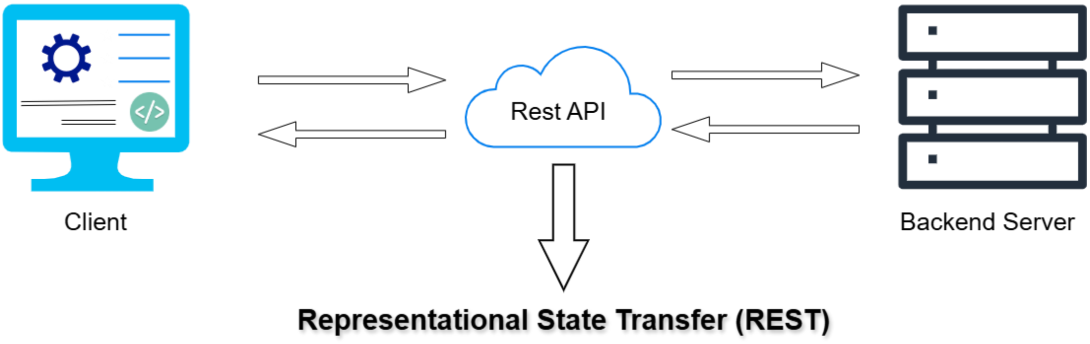
  /jobs (to access all jobs)

  /jobs/42 (to access a specific job with ID 42)

2. State and Representations

- **State:** The current values of a resource at a specific moment
- Example: An employee's "CurrentEmployer" field might be "Microsoft" now, but "Telusko" a few months later
- Each time you access the resource, you get its current state

- **Representation**: How this state is formatted and presented
- The same resource can be represented in different formats (JSON, XML)
- The server transfers these representations to the client



3. Statelessness

REST is stateless, which means:

- Each request from client to server must contain all information needed to understand and process the request
- The server doesn't store any client context between requests
- Each request is treated as a standalone interaction

Think of it like this: If a client makes a request today and the same request tomorrow, the server treats them as completely separate events, with no memory of the previous interaction.

4. HTTP Methods Instead of Different URLs

Instead of using different URLs for different actions, REST uses the same URL with different HTTP methods:

HTTP Method	Purpose	Example
GET	Retrieve data	GET /jobs (list all jobs) or GET /jobs/42 (get specific job)
POST	Create new data	POST /jobs (create a new job)
PUT	Update existing data	PUT /jobs/42 (update job with ID 42)
DELETE	Remove data	DELETE /jobs/42 (delete job with ID 42)

👉 Data Formats in REST

REST APIs typically return data in standardized formats:

1. JSON (JavaScript Object Notation)

- Modern, lightweight format
- Easier to read and write
- Better performance
- Native support in JavaScript

Example:

```
{
 "id": 42,
 "title": "Software Developer",
 "company": "Telusko",
 "location": "Bangalore"
}
```

2. XML (eXtensible Markup Language)

- Older format, still used in some enterprise systems
- More verbose than JSON

Example:

```
<job>
 <id>42</id>
 <title>Software Developer</title>
 <company>Telusko</company>
 <location>Bangalore</location>
</job>
```

👉 Summary

REST is an approach to building web services that:

- Treats data as resources with unique URLs
- Uses standard HTTP methods for different operations
- Returns the current state of resources in a standardized format
- Maintains no session state between requests