Spring Framework 6

Beginner to Guru

Beginners Guide to REST

RESTful Web Services

- Because of their simplicity and versatility, RESTful web services have become the de facto standard for web services.
- REST - Representational State Transfer
  - Representation - Typically JSON or XML
  - State Transfer - Typically via HTTP
  - Established by Roy Fielding from his 2000 doctoral dissertation

What is REST?

- REST APIs use HTTP verbs to create, manage, and delete server resources
- For example:
  - **GET** - is used to get a resource
  - **DELETE** - is used to delete a resource
- Resources are typically data structures represented by JSON or XML
- HTTP Status codes are used to communicate success, failure, or errors
- REST is not a formal standard, more generally agreed upon methods and techniques
  - Unlike SOAP

RESTful Terminology

- Verbs - HTTP Methods: **GET, PUT, POST, DELETE**
- Messages - the payload of the action (JSON/XML)
- URI - Uniform Resource Identifier
  - A unique string identifying a resource
- URL - Uniform Resource Locator
  - A URI with network information - http://www.example.com
  - http://www.example.com/books/1

RESTful Terminology

- Idempotence -
  - Wikipedia “Idempotence is the property of certain operations in mathematics and computer science that they can be applied multiple times without changing the result beyond the initial application.”
  - In other words, you can exercise the operation multiple times, without changing the result.
  - Example: Refreshing a web page (HTTP GET operation)

RESTful Terminology

- Stateless - Service does not maintain any client state
- HATEOAS - Hypermedia As The Engine Of Application State
  - Wikipedia - “a REST client should then be able to use server-provided links dynamically to discover all the available actions and resources it needs. As access proceeds, the server responds with text that includes hyperlinks to other actions that are currently available.”

HTTP - GET

- Use: to read data from resource
- Read only
- Idempotent
- Safe operation - does not change state of resource
- C**R**UD - Read

HTTP - PUT

- Use: to insert (if not found) or update (if found)
- Idempotent - Multiple PUTs will not change result.
  - Like saving a file multiple times
- Not Safe operation - does change state of resource
- **C**R**U**D - Created, Update

HTTP - POST

- Use: to create new object (insert)
- Non-Idempotent - Multiple POSTs is expected to create multiple objects
- Not Safe operation - does change state of resource
- Only Non-Idempotent, Non-Safe HTTP verb
- **C**RUD - Create

HTTP - PATCH

- Use: to partially update an existing resource
- Non-Idempotent - Multiple PATCH requests may change state
  - Typically is Idempotent, but is not required to be
- Not Safe operation - does change state of resource
- CR**U**D - Update

HTTP - DELETE

- Use: to delete an object (resource)
- Idempotent - Multiple DELETEs will have same effect / behaviour.
- Not Safe operation - does change state of resource
- CRU**D** - Delete

SPRING FRAMEWORK

GURU