

## Module – 6

### 27. Building Web API

#### 27.1 Understanding HTTP Verbs

- HTTP verbs tell the server what to do with the data identified by the URL.
- The HTTP method is supplied in the request line and specifies the operation that the client has requested.
- If you're going to follow the REST architecture and the HTTP protocol, you must choose from the verbs available in that protocol, the primary or most-commonly-used HTTP verbs are POST, GET, PUT, and DELETE.

#### GET

- The GET method is designed to request a specific resource.
- In essence, it literally “gets” the resource in question, and is pretty limited to just that action.
- GET requests should only retrieve data, leaving other methods to perform the other transformative actions.

#### POST

- POST allows an API to submit an attribute or entity to a given resource.
- this means the targeted resource receives a subordinate resource that is part of a larger collection.

#### PUT

- PUT is somewhat the polar opposite of GET.
- While GET requests a specific resource, PUT places that resource in the remote directory.
- It should be noted that PUT assumes either the resource does not exist or the resource is fine to be overwritten when using PUT, all representations of the target resource will be replaced by the payload.

#### DELETE

- A DELETE request is as simple as its name implies it just deletes, it is used to delete a resource identified by a URI and on successful deletion.
- The important and unique thing about a DELETE request is that the client cannot be guaranteed that the operation has been carried out, even if the status code returned from the origin server indicates that the action has been completed successfully.

## 27.2 Understanding JSON Structure

### Reasons for using JSON:

- XML is expressed in more words than JSON. Hence, JSON is faster to write for programmers.
- JSON is lightweight and easy to write compared to XML.
- We can store values as key/value pair in JSON. Whereas, in XML, we have to store values between opening and closing tags.

### JSON

- JSON stands for JavaScript Object Notation
- JSON is a text format for storing and transporting data.
- e.g. '{" First Name ":"Yash", " Last Name " : "Sanghvi"}'
- You can receive pure text from a server and use it as a JavaScript object or send a JavaScript object to a server in pure text format.
- Simply, it is used for data-interchange. The benefit of JSON is that it has a very compact size as compared to XML documents of the same purpose and data.
- JSON stores the data in the form of key/value pairs.
- The keys are strings and the values are the JSON types. Keys and values are separated by colon. Each entry (key/value pair) is separated by comma.
- Another benefit of JSON is it is language-independent. You can work with JSON data in almost any programming language that can handle string objects.
- The basic structure of JSON document is very much simple. Every document in JSON must have either an object at its root, or an array.

### JSON Strings

- String typed values are the values which contain the character-type values. In many programming environments, strings are called, arrays of characters.
- In JavaScript, you can use either single quotes or double quotes to wrap the string values.
- But, in JSON specification you should always consider using double quotation