

. Explain REST?

Ans. REST stands for Representational State Transfer. REST is an architectural style of developing web services which take advantage of the ubiquity of HTTP protocol and leverages HTTP method to define actions. It revolves around resource where every component is a resource which can be accessed by a common interface using HTTP standard methods.

In REST architecture, a REST Server provides access to resources and REST client accesses and presents those resources. Here each resource is identified by URIs or global IDs. REST uses different ways to represent a resource like text, JSON, and XML. XML and JSON are the most popular representations of resources these days.

What Is The Most Popular Way To Represent A Resource In REST?

Ans. REST uses different representations to define a resource like text, JSON, and XML.

JSON is the most popular representations of resources.

Explain What Is A “Resource” In REST?

Ans. REST architecture treats every content as a resource. These resources can be either text files, HTML pages, images, videos or dynamic business data.

REST Server provides access to resources and REST client accesses and modifies these resources. Here each resource is identified by URIs/ global IDs.

Which Protocol Is Used By RESTful Web Services?

Ans. RESTful web services make use of HTTP protocol as a medium of communication between client and server.

What Is Messaging In RESTful Web Services?

Ans. RESTful web services make use of HTTP protocol as a medium of communication between client and server. The client sends a message in the form of an HTTP Request.

In response, the server transmits the HTTP Response. This technique is called Messaging. These messages contain message data and metadata i.e. information about the message itself.

State The Core Components Of An HTTP Request?

Ans. Each HTTP request includes five key elements.

1. The Verb which indicates HTTP methods such as GET, PUT, POST, DELETE.
2. URI stands for Uniform Resource Identifier (URI). It is the identifier for the resource on the server.
3. HTTP Version which indicates HTTP version, for example-HTTP v1.1.
4. Request Header carries metadata (as key-value pairs) for the HTTP Request message. Metadata could be a client (or browser) type, the format that client supports, message body format, and cache settings.
5. Request Body indicates the message content or resource representation.

What is Rest Assured?

In order to test REST APIs, I found **REST Assured library** so useful. It is developed by JayWay Company and it is a really powerful catalyzer for automated testing of **REST-services**. REST-assured provides a lot of nice features, such as **DSL-like syntax**, **XPath-Validation**, **Specification Reuse**, **easy file uploads** and with those features we will handle automated API testing much easier.

How to declare the API details in Rest Assured Test?

Using Given(), When(), Then()

Name The Most Commonly Used HTTP Methods Supported By REST?

Ans. There are a few HTTP methods in REST which are more popular.

1. GET -It requests a resource at the request-URL. It should not contain a request body as it will get discarded. Maybe it can be cached locally or on the server.
2. POST – It submits information to the service for processing; it should typically return the modified or new resource.
3. PUT – At the request URL it updates the resource.
4. DELETE – It removes the resource at the request-URL.
5. OPTIONS -It indicates the supported techniques.
6. HEAD – It returns meta information about the request URL.

2.

Is There Any Difference Between PUT And POST Operations? Explain It.

Ans. PUT and POST operation are almost same. The only difference between the two is in the terms of the result generated by them.

PUT operation is idempotent while POST operation can give a different result.

What Is URI? Explain Its Purpose In REST Based Web Services. What Is Its Format?

Ans. URI stands for Uniform Resource Identifier. URI is the identifier for the resource in REST architecture.

The purpose of a URI is to locate a resource(s) on the server hosting the web service. A URI is of the following format-

<protocol>://<service-name>/<ResourceType>/<ResourceID>

How to compare the response values with Rest Assured Assertion?

Example :

given().

parameters("firstName", "John", "lastName", "Doe").

when().

```
        post("/greetXML").  
then().  
    body("greeting.firstName", equalTo("John")).  
    body("greeting.lastName", equalTo("Doe"));
```

How to Insert cookies in Testing the API using Rest Assured?

```
given().cookie("username",  
"John").when().get("/cookie").then().body(equalTo("username"));
```

How to Insert headers in Testing the API using Rest Assured?

```
given().header("MyHeader", "Something").
```

How to Validate Response Headers with Rest Assured?

```
get("/x").then().assertThat().header("headerName", "headerValue").
```

How to handle Basic Authentication with Rest Assured?

```
given().auth().preemptive().basic("username",  
"password").when().get("/secured/hello").then().statusCode(200);
```

What Do You Understand By Payload In RESTful Web Service?

Ans. Request body of every HTTP message includes request data called as Payload. This part of the message is of interest to the recipient.

We can say that we send the payload in POST method but not in <GET> and <DELTE> methods.