Chapter 5

Communicating with APIS

What is URI in REST?

* Each resource in REST architecture is identified by its URI (Uniform Resource Identifier).
* URI is of the following format − <protocol>://<service-name>/<ResourceType>/<ResourceID> Purpose of an URI is to locate a resource(s) on the server hosting the web service.

What is end point in API?

* API endpoints are the specific digital location where requests for information are sent by one program to retrieve the digital resource that exists there
* Endpoints specify where APIs can access resources and help guarantee the proper functioning of the incorporated software.
* Example: [https://www.testapp.com/customers/{id}](https://www.testapp.com/customers/%7bid%7d)

What is difference between URL, URI, and URN?

URL is used for identifying whereas URI is used for locating

What is Webservice?

* Web services are XML-based information exchange systems that use the Internet for direct application-to-application interaction. These systems can include programs, objects, messages, or documents
* A web service is a collection of open protocols and standards used for exchanging data between applications or systems.
* Software applications written in various programming languages and running on various platforms can use web services to exchange data over computer networks like the Internet in a manner like inter-process communication on a single computer. This interoperability (e.g., between Java and Python, or Windows and Linux applications) is due to the use of open standards.
* Components of Web Services - SOAP, UDDI, WSDL

How does a Web Service Work?

* A web service enables communication among various applications by using open standards such as HTML, XML, WSDL, and SOAP
* XML to tag the data
* SOAP to transfer a message
* WSDL to describe the availability of service.

Explain different http methods?

* GET: It is used to retrieve information from the given server using a given URI. Requesting using GET should only retrieve data and should have no other effect on the data
* POST: It is used to send data to the server example - customer information, file upload etc. using HTML forms
* PUT: It is used to replace all current representations of the target resources with the uploaded content.
* PATCH: It is used to make a partial update on a target resource with the uploaded content.
* DELETE: removes all current representation of the target resources given by uri

What is difference between PUT and POST and PATCH Requests?

Post request is non idempotent whereas put request is idempotent that means when the same request is called multiple times for the put it always produce the same result where as POST request repeated will have side effects of creating the same resource multiple times. Patch request is used when we want to update the record partially whereas put will replace the record completely

What are the different types of API Authentication?

* Password-based authentication
* Multi-factor authentication
* Certificate-based authentication
* Biometric authentication
* Token-based authentication - oAuth1(Transport-independent, Founded in cryptography, especially digital signatures,Messages are each individually cryptographically signed,Basic signature workflow), oAuth2(Transport-dependent,Centered around bearer tokens,Much easier to work with,Much more flexible,Better separation of duties)

What are different types of status codes in API?

**Informational Responses (100-199)**

* 100: **continue** (it states everything is okay so far and the client will continue the request or ignore the response if the request is already finished)
* 101: **Switching Protocol** (this is sent in response to an upgraded request header from the client and indicates the protocol the server is switching to)
* 102: **Processing** (it means server got the request and it is processing the request and no response is available yet)

**Successful Responses (200 - 299)**

* **200: OK** (The request is successful - GET request)
* **201: Created** (the request is successful and new resource is created successfully -POST/PUT request)
* **202: Accepted**(The request is received but not yet acted upon)
* **203: Non Authoritative Information** (This means the returned meta information is not exactly the same as from the available server, but is collected from local or third party copy)
* **204: No Content** (There is no content to send for this request, but the headers may be useful)

**Redirection Messages (300-399)**

* **300: Multiple Choice** (the request has more possible response and the user should choose one of them)
* **301: Moved Permanently** (The url of the request is changed permanently)
* **302: Found** (this means the URI of the requested resource is changed temporarily)
* **304: Not Modified** (This is used for caching purpose and it tells the clients that the response has not been modified so the client can continue to use the same cached version of the response)

**Client errors (400-499)**

* **400: Bad Request** (the server could not understand the request due to invalid syntax)
* **401: Unauthorized** (this means client must authenticate itself to get the requested response- clients identity is unknown)
* **402: Payment Required** (this is reserved for future use with the aim of using it for digital payment systems)
* **403: Forbidden** (The client doesn't have access rights to the content(unauthorized) so the server is refusing to give the requested resources, unlike 401 the clients identity is known to the server)
* **404: Not Found** (The server can't find the requested resource)
* **405: Method not allowed** (the requests method is known by the server but is not supported by the target resource)

**Server errors (500-599)**

* **500: Internal Server Error** (the server has encountered a situation it doesn't know how to handle)
* **501: Not Implemented** (The request method is not supported by the server and can't be handled - the only methods that the server are required to support are GET and HEAD)
* **502: Bad Gateway** (This error response means that the server while working as a gateway to get a response needed to handle the request, got an invalid response)
* **503: Service Unavailable** (The service is not ready to handle the request, the common causes are a server that is down for maintenance or that is overloaded)
* **504: Gateway Timeout** (this means the server is acting as a gateway and cannot get a response in the time)

What are the different types of authorization methods used in REST API?

* Basic Authorization - have username and password which are provided along with the uri
* OAuth(1) - An open data protocol that provides a process for end user to authorize third party access to their server resources without sharing their credentials using user agent redirects
* OAuth(2) - It works by delegating user authentication to the user authentication to the service that hosts a user account and authorizing third-party applications to access that user account.

How to send GET/POST API request using REST Assured?

Text

Description automatically generated

How to validate the status code for the response?

Text

Description automatically generated

How to validate the headers of response?

Text

Description automatically generated

How to validate the response body values?

Text

Description automatically generated

Write code for the sending sample post request?

Endpoint:https://testingsite.com/register

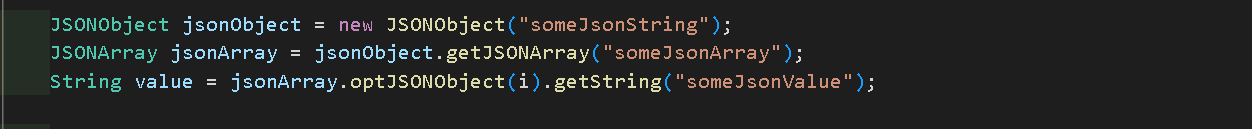
Request Body: { "firstname":"fname","middlename":"mname","lastname":"lname","email":"mail"}

Response: {"desc":"the user details", "region":"temp","Accountdetails":[{"country":"india","data":{"capital":"delhi","currency":"rupee"}}, {"country":"us","data":{"capital":"washington","currency":"rupee"}}]}

Text

Description automatically generated

How to convert the java object to JSON object?

 This is very slow and using jackson Library is much more fastest than this method  
**Using Jackson Library:**

writeValueasString() method in ObjectMapper() class is used to serialize the java object to JSON object.  
ReadValue() method in ObjectMapper class will deserialize a JSON content from given JSON content String.

Text

Description automatically generated

How to pass the json parameters to the request body?

* JSONObject class is available in org.json.simple package.
* add the json parameters as key value pair in the JSONObject
* JSONObject obj = new JSONObject();
* obj.put(key,value)
* To convert this map to string we use toJSONString method.
* obj.toJSONString()