**Q #1) What is your understanding of what are RESTful web services?**

Just like SOAP (Simple Object Access Protocol), which is used to develop web services by XML method, RESTful web services use web protocol i.e. HTTP protocol method. They have the feature like scalability, maintainability, help multiple application communication built on various programming languages etc.

RESTful web service implementation defines the method of accessing various resources which are required by the client and he has sent the request to the server through the web browser. The important aspects of this implementation include:

* Resources
* Request Headers
* Request Body
* Response Body
* Status codes

**Q #2) Name the protocol which is used by RESTful web services.**

RESTful web services use a famous web protocol i.e. HTTP protocol. This serves as a medium of data communication between client and server. HTTP standard methods are used to access resources in RESTful web service architecture.

**Q #3) Explain the term ‘Addressing’ with respect to RESTful WEB service.**

Just like we require address with postal code to reach any person, in the same way, ‘Addressing’ locates resources that are present on the server for the purpose of hosting web services. This is usually done with URI i.e. Unified Resource Identifier.

**Q #4) Enlist features of RESTful web services.**

Every RESTful web services should have following features and characteristics that are enlisted below:

* Based on the Client Server representation.
* Use of HTTP protocol for performing functions like fetching data from the web service, retrieving resources, execution of any query, etc.
* The communication between the server and client is performed through the medium known as ‘messaging’.
* Addressing of resources available on the server through URIs.
* Based on the concept of statelessness where every client request and the response is independent of the other with complete assurance of providing required information.
* Uses the concept of caching.
* Works on Uniform interface.

**Q #5) Explain messaging technique.**

Messages are the mode of exchanging data for any type of communication to take place. In the same way, HTTP protocol plays the role of message communication between the client and server through HTTP Request and Response methods. HTTP request is sent by the client who contains information about the data and in turn, receives HTTP Response from the server.

Messages are the collection of information about the data i.e. Metadata.

**Q #6) What are the core components of HTTP request and HTTP response?**

The core components that come under HTTP Request are:

* **Verb:** Includes methods like GET, PUT, POST, etc.
* Uniform Resource Identifier for identifying the resources available on the server.
* HTTP Version for specifying the HTTP version.
* HTTP Request header for containing the information about the data.
* HTTP Request body that contains the representation of the resources in use.

**The core components that come under HTTP Response are:**

* **Request Code:** This contains various codes which determine the status of the server response.
* HTTP Version for specifying the HTTP version.
* HTTP Response header for containing the information about the data.
* HTTP Response body that contains the representation of the resources in use.

**Q #7) Explain the term ‘Statelessness’ with respect to RESTful WEB service.**

In REST, ST itself defines State Transfer and Statelessness means complete isolation. This means, the state of the client’s application is never stored on the server and is passed on. In this process, the clients send all the information that is required for the server to fulfill the HTTP request that has been sent. Thus every client request and the response is independent of the other with complete assurance of providing required information.

Every client passes a ‘session identifier’ which also acts as an identifier for each session.

**Q #8) Enlist advantages and disadvantages of ‘Statelessness’.**

In the above question, we have understood the meaning of statelessness with respect to the client-server communication. Now, let us see some of its advantages and disadvantages.

**Advantages:**

* Every method required for communication is identified as an independent method i.e. there are no dependencies to other methods.
* Any previous communication with the client and server is not maintained and thus the whole process is very much simplified.
* If any information or metadata used earlier in required in another method, then the client sends again that information with HTTP request.
* HTTP protocol and REST web service, both shares the feature of statelessness.

**Disadvantages:**

* In every HTTP request from the client, the availability of some information regarding the client state is required by the web service.

**Q #9) Enlist some important constraints for RESTful web services.**

Every constraint has positive as well as negative impacts and to produce an overall architecture, there should be the balance between both of them. Below mentioned are some important constraints for RESTful web service:

* There should be separate concerns for each server and client which will help to maintain the modularity within the application. This will also reduce the complexity and increase the scalability.
* The client-server communication should be stateless, which means no previous information is used and the complete execution is done in isolation. In cases of failure, it also helps the client to recover.
* In client-server communication, the HTTP response should be cacheable so that when required cached copy can be used which in turn enhances the scalability and performance of the server.
* The fourth constraint is the uniform interface which allows client-server interaction to be easily understood. This constraint is further divided into four sub-constraints as:
  + Resource Identification
  + Resource manipulation
  + Each message is easily understood and is self-descriptive.
  + Hypermedia, which is defined as the text with hyperlinks and when clicked it moves to another application state.
* Client-server communication should be done on a layered system and thus the client should only have knowledge about the intermediate level with which communication is being done,

**Q #10) What is a ‘Resource’?**

Just like the ‘Object’ instance, we have learned in object orient programming Language, in the same way, ‘Resource’ is defined as an object of a type which can be an image, HTML file, text data, and any type of dynamic data. There are varieties of representation formats available in order to represent a resource.

**Some most common are enlisted below:**

* JSON
* YAML
* XML
* HTML

**Q #11) Why proper representation of Resource is required?**

Representation is very important because it determines the easy identification of resources. With proper representations of resource in the proper format, allows the client to easily understand the format.

**Q #12) Enlist some important points that should be kept in mind while designing Resources representation for RESTful web services.**

As there are no restrictions on the format in which the resource representation is done but just that the main requirement is the format of the representation should be as per the client requirement. A good resource representation is designed by considering the following main points:

* The resource representation format should be easily understood by the client and server.
* The representation should be complete regardless of its format structure, which may be complex or simple.
* In the case of the link of the resources to other resources, such cases should also be considered and handled.

**Q #13) What is Caching?**

Caching is the process in which server response is stored so that a cached copy can be used when required and there is no need of generating the same response again. This process not only reduces the server load but in turn increase the scalability and performance of the server. Only the client is able to cache the response and that too for a limited period of time.

Mentioned below are the header of the resources and their brief description so that they can be identified for the caching process:

* Time and Date of resource creation
* Time and date of resource modification that usually stores the last detail.
* Cache control header
* Time and date at which the cached resource will expire.
* The age which determines the time from when the resource has been fetched.

**Q #14) Explain Cache-control header.**

A standard Cache control header can help in attaining cache ability. Enlisted below is the brief description of various cache control header:

* **Public:** Resources that are marked as the public can be cached by any intermediate components between the client and server.
* **Private:** Resources that are marked as private can only be cached by the client.
* No cache means that particular resource cannot be cached and thus the whole process is stopped.

**Q #15) What are the best practices that are to be followed while designing RESTful web services?**

To design a secure RESTful web service, there are some best practices or say points that should be considered. These are explained as follows:

* Every input on the server should be validated.
* Input should be well formed.
* Never pass any sensitive data through URL.
* For any session, the user should be authenticated.
* Only HTTP error messages should be used for indicating any fault.
* Use message format that is easily understood and is required by the client.
* Unified Resource Identifier should be descriptive and easily understood.

**Q #16) What is Payload?**

The request data which is present in the body part of every HTTP message is referred as ‘Payload’.  In Restful web service, the payload can only be passed to the recipient through POST method.

There is no limit of sending data as payload through POST method but the only concern is that more data with consuming more time and bandwidth. This may consume much of user’s time also.

**Q #17) Enlist some of the HTTP methods with description.**

Mentioned below is the list of HTTP methods with their descriptions:

* **GET:** This is a read only operation which fetches the list of users on the server.
* **PUT:** This operation is used for the creation of any new resource on the server.
* **POST:** This operation is used for updating an old resource or for creating a new resource.
* **DELETE:** As the name suggests, this operation is used for deleting any resource on the server.
* **OPTIONS:** This operation fetches the list of any supported options of resources that are available on the server.

**Q #18) What is the difference between PUT method and POST method?**

The major difference between the PUT and POST method is that the result generated with PUT method is always same no matter how many times the operation is performed. On the other hand, the result generated by POST operation is always different every time.

**Q #19) What is your understanding about JAX-RS?**

JAX-RS is defined as the Java API for RESTful web service. Among multiple libraries and framework, this is considered as the most suitable Java programming language based API which supports RESTful web service.

**Some of the implementations of JAX-RS are:**

* Jersey
* RESTEasy
* Apache CFX
* Play

Among these, Jersey is the most popular framework.

**Q #20) What are HTTP status codes? Enlist few with meaning.**

HTTP status codes basically are the representation of the status of the task that has been performed on the server, with the mode of some codes. Every code has their own meaning.

**Some of the HTTP status codes with their meaning are as follows:**

* **Code 200:** This indicates success.
* **Code 201:** This indicates resource has been successfully created.
* **Code 204:** This indicates that there is no content in the response body.
* **Code 404:** This indicates that there is no method available.

**1) Explain what is REST and RESTFUL?**

REST represents REpresentational  State Transfer; it is a relatively new aspect of writing web API.

RESTFUL is referred for web services written by applying REST architectural concept are called RESTful services, it focuses on system resources and how state of resource should be transported over HTTP protocol to different clients written in different language.  In RESTFUL web service HTTP methods like GET, POST, PUT and DELETE can be used to perform CRUD operations.

**2) Explain the architectural style for creating web API?**

The architectural style for creating web api are

* HTTP for client server communication
* XML/JSON as formatting language
* Simple URI as the address for the services
* Stateless communication

**3) Mention what tools are required to test your web API?**

SOAPUI tool for SOAP WS and Firefox “poster” plugin for RESTFUL services.

**4) Mention what are the HTTP methods supported by REST?**

HTTP methods supported by REST are:

* **GET:**It requests a resource at the request URL. It should not contain a request body as it will be discarded. Maybe it can be cached locally or on the server.
* **POST:**It submits information to the service for processing; it should typically return the modified or new resource
* **PUT:**At the request URL it update the resource
* **DELETE:**At the request URL it removes the resource
* **OPTIONS:**It indicates which techniques are supported
* **HEAD:**About the request URL it returns meta information

**5) Mention whether you can use GET request instead of PUT to create a resource?**

No, you are not supposed to use POST or GET.  GET operations should only have view rights

**6) Mention what are resources in a REST architecture?**

Resources are identified by logical URLs; it is the key element of a RESTful design.  Unlike, SOAP web services in REST, you view the product data as a resource and this resource should contain all the required information.

**7) Mention what is the difference between AJAX and REST?**

|  |  |
| --- | --- |
| **AJAX** | **REST** |
| * In Ajax, the request are sent to the server by using XMLHttpRequest objects. The response is used by the JavaScript code to dynamically alter the current page * Ajax is a set of technology; it is a technique of dynamically updating parts of UI without having to reload the page * Ajax eliminates the interaction between the customer and server asynchronously * REST requires the interaction between the customer and server | * REST have a URL structure and a request/response pattern the revolve around the use of resources * REST is a type of software architecture and a method for users to request data or information from servers * REST requires the interaction between the customer and server |

**7) Mention some key characteristics of REST?**

Some key characteristics of REST includes

* REST is stateless, therefore the SERVER has no state (or session data)
* With a well-applied REST API, the server could be restarted between two calls as every data is passed to the server
* Web service mostly uses POST method to make operations, whereas REST uses GET to access resources

**8) Mention what are the different application integration styles?**

The different integration styles include

* Shared database
* Batch file transfer
* Invoking remote procedure (RPC)
* Swapping asynchronous messages over a message oriented middle-ware (MOM)

**9) Explain how JAXB related to RESTful web API?**

JAXB stands for java arch for XML binding.

**10) Mention what is the difference between PUT and POST?**

“PUT” puts a file or resource at a particular URI and exactly at that URI.  If there is already a file or resource at that URI, PUT changes that file or resource.  If there is no resource or file there, PUT makes one

POST sends data to a particular URI and expects the resource at that URI to deal with the request.  The web server at this point can decide what to do with the data in the context of specified resource

PUT is idempotent meaning, invoking it any number of times will not have an impact on resources.

However, POST is not idempotent, meaning if you invoke POST multiple times it keeps creating more resources

**11) Mention which markup language can be used in restful web api?**

JSON and XML are the two markup language that can be used in restful web api

**12) Mention what is the difference between RPC or document style web services? How you determine to which one to choose?**

In document style web services, we can transport an XML message as part of SOAP request which is not possible in RPC style web service.  Document style web service is most appropriate in some application where XML message behaves as document and content of that document can alter and intention of web service does not rely on the content of XML message.

**13) Mention what is JAX-WS and JAX-RS?**

Both JAX-WS and JAX-RS are libraries (APIs) for doing communication in various ways in Java.  JAX-WS is a library that can be used to do SOAP communication in JAVA, and JAX-RS lets you do the REST communication in JAVA.

**14) List out the tools or API for developing or testing web api?**

Testing tools for web services for REST APIs includes

* Spring REST web service using MVC
* Jersey API
* CFX
* Axis
* Restlet,

**15) Mention what is the difference between SOAP and REST?**

|  |  |
| --- | --- |
| **SOAP** | **REST** |
| * SOAP is a protocol through which two computer communicates by sharing XML document * SOAP permits only XML * SOAP based reads cannot be cached * SOAP is like custom desktop application, closely connected to the server * SOAP is slower than REST * It runs on HTTP but envelopes the message | * Rest is a service architecture and design for network-based software architectures * REST supports many different data formats * REST reads can be cached * A REST client is more like a browser; it knows how to standardized methods and an application has to fit inside it * REST is faster than SOAP * It uses the HTTP headers to hold meta information |