By OnlineInterviewQuestions.com

Rest API Interview Questions

Q1. Explain What is REST?

REST, expanded as Representational State Transfer, is an architectural style developed by **Roy Fielding** in **2000** . REST presents a set of constraints to be used in the creation of web services. The services that use REST constraints are called as RESTful Web Services. For an interface to be referred to as RESTful, it should satisfy the six-guiding constraints. These constraints are names as the client-server, **stateless**, **cacheable**, **uniform interface**, **layered system**, and **code** on demand.

After satisfying the constraints, the RESTful web services can be used to provide an interface between the computer system and the internet. Using this interface, a system can access and manipulate the resources present on the web by using a predefined set of operations.

Q2. Enlist different types of HTTP requests supported by REST API?

REST supports a group of **HTTP** methods to manipulate the request and response. They are,

GET – It is used to read the representation of a resource. It usually returns the resource in XML or JSON type with the HTTP response code 200 on successful creation.

POST – It is used to create new resources. It is usually used to create a subordinate resource and returns HTTP code 201 on successful creation.

PUT – This method mainly used to update the resource. It can also be used to create a resource when the resource ID is selected by the client. It returns response code 200 on a successful update.

PATCH – It is used to modify the resource. It only contains the change needed to be modified in the resource.

DELETE – It deletes a resource. The resource needed to be deleted is identified by a URI. It returns an HTTP status 200 on successful completion.

Q3. What is resource in REST API?

In the REST architecture, every content is a resource. It can be a text file, **HTML pages**, **images**, **videos**, or **business data**. These resources are identified by the **URI** or global **ID's** which can be used by the REST client to access or modify.

The most used representation of the resources is **XML** and **JSON**. Resources can also be grouped into a

collection of the same type.

Q4. What is options in REST API?

The options the REST are annotations that are used to indicate whether a method responds to the **HTTP OPTIONS** request only. It allows the client of the **REST API** to determine what HTTP methods (GET, HEAD, POST, PUT, DELETE) can be used for the resource identified by the requested URI. The client determines without initiating a resource request.

The REST OPTIONS method is also used for the **CORS** (**Cross-Origin Resource Sharing**) request.

Q5. What is difference between Ajax and REST API?

AJAX (Asynchronous JavaScript and XML) is a set of technologies that are used for **updating** the data and **web page** without reloading the page. Using this, you can send data in the background, request/receive data from a server after the page has loaded.

REST (REpresentational State Transfer) uses **HTTP** requests to transfer data between the client and the server. It defines a set of constraints for the successful creation of the RESTful web services.

AJAX and REST are completely orthogonal. You can use REST for creating an AJAX call. REST is one way of implementing AJAX.

Q6. What is an URI in REST?

URI (**Uniform Resource Identifiers**) is used to identify each resource in the REST. An HTTP operation is called by the client application to access the resource. In the construction of a URI to identify a resource, some rules are to be followed. The resource should be defined in the plural noun. The URI is case0sensitive, so use a lowercase letter when defining a URI. There should be no spaces included in the URI. The created URI should be backward compatible.

//FORMAT for creating a URI

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

Q7. What is Postman?

Postman is a popular test and development tool to simplify the API workflow. It provides the tool to manage every stage of the API lifecycle and makes the development of the API simple.

With postman, you can **design**, **debug**, **test**, **document**, **monitor**, and publish the **API** from one place. It also

provides version control and tagging to maintain multiple versions of the API. It also provides a testing tool to automate the testing process.

Postman can also be used to create a **mock server** to simulate the endpoints and the response.

Q8. What is a Payload in Rest?

The payload is used by the REST API to pass and return data structures. It is used when the data is too large to be passed as a parameter. The input payload is a filter definition passed in the request to test result resources. The output payload is a set of test results. Payloads have a predefined structure to easily create, consume, manipulate and present by the client tools.

Q9. List major difference between SOAP and REST?

SOAP (**Simple Object Access Protocol**) is used to exchange data between different platforms easily. It has a **specification** and **WSDL file** that has information about the location and function of the web service. It uses a service interface to expose its functionality to the clients.

REST is an architectural style pattern to create a RESTful web service. It uses normal HTTP requests to **receive/request** the resource on the web. **REST** uses **SOAP** as an underlying protocol for the creation of **web services** as it is just an **architectural design**.

Q10. Which markup language is used in REST API?

The data in the RESTful web service is mainly represented in **XML** or **JSON** form.

XML (Extensible Markup Language) is used to **store and transfer** data.

JSON (JavaScript Object Notation) is used to serialize and transfer data. It is lighter than XML and can transfer the same amount of data as the XML with less bandwidth.

Q11. List major HTTP response codes returned by REST API?

The status code in the REST API is divided into five categories. **They are**,

1xx – It is used to communicate the transfer protocol-level information.

2xx – It is used to indicate the request was accepted successfully. Some codes are,

- 200 (**OK**) - It indicates the request is successfully carried out.
- 201 (**Created**) - It is returned when a resource is created inside the collection.

- 202 (**Accepted**) - It indicates the request has been accepted for processing.
- 204 (**No Content**) - It indicates when a request is declined.

3xx – It indicates the client must take additional action to complete the request.

4xx – It is the client error status code.

5xx – It is the server error status code.

Q12. What is app engine in gcp?

Google App Engine (GAE / App Engine) in **Google Cloud Platform** is a Platform as a Service (PaaS) for developing and hosting web applications in Google-managed data centers. Applications are sandboxed and operated via multiple servers and an app engine allows automatic scaling for web applications as the number of requests progress for an application, it automatically allocates more resources for the web application to manage the additional demand.

Q13. What is a stateless server?

A stateless server is a server that keeps no state information and stateless file servers do not store any session state. Therefore, every client request is treated independently, and not as a part of a new or existing session. A stateless server does not need a client to first establish a connection to the server. So, it views a client request as an independent transaction and responds to it.