**HTTP Fundamentals**

1.What are the basic Features of HTTP?

* HTTP is connectionless. The connection is only established if the client sends the request and after the response is send, the connection is disconnected. So client and server knows about each other during current request and response only. Further requests are made on new connection like client and server are new to each other
* HTTP is media independent. It means, any type of data can be sent by HTTP as long as both the client and the server know how to handle the data content. It is required for the client as well as the server to specify the content type using appropriate MIME-type
* HTTP is stateless. HTTP is connectionless and it is a direct result of HTTP being a stateless protocol. The server and client are aware of each other only during a current request. Afterwards, both of them forget about each other. Due to this nature of the protocol, neither the client nor the browser can retain information between different requests across the web pages.

2. What are request methods in HTTP?

* GET – for fetching some data from server
* POST- for adding some data to the server
* PUT – for replacing a data with another
* PATCH- for making partial modifications in a data
* DELETE – for deleting some data
* HEAD – for getting the header data not response body
* CONNECT – for establishing a tunnel to server identified by the resource
* TRACE - Performs a message loop-back test along the path to the target resource

3.What are the differences between GET and POST methods?

**GET**

* In GET, only **limited amount of data**can be sent because data is sent in header
* GET request is not secured because data is exposed in URL bar.
* GET request can be bookmarked.
* GET request is idempotent

**POST**

* In POST, large amount of data can be sent because data is sent in body.
* POST request is secured because data is not exposed in URL bar.
* POST request cannot be bookmarked
* POST request is not idempotent.

4.What is status code in HTTP?

Status codes are issued by a server in response to a client’s request made to the server. Some common HTTP status codes are;

* 200 - OK – The request succeeded (GET,PUT,DELETE,PATCH)
* 201 - Created – A new resource or object in a collection (POST)
* 304 - Not modified – Nothing was modified by the request (PATCH, PUT)
* 400 - Bad request – The request could not be performed by the server due to bad syntax or other reason in request (All Methods)
* 401 - Unauthorized – Authorization credentials are required or user does not have access to the resource/method they are requesting (All Methods)
* 404 - Resource not found – The URI is not recognized by the server (All Methods)
* 500 - Server error – Generic something went wrong on the server side (All Methods)

5.What are the header fields in HTTP?

HTTP header fields are components of the header section of request and response messages in the Hypertext Transfer Protocol. They define the operating parameters of an HTTP transaction

6.What is URI?

A Uniform Resource Identifier (URI) provides a simple and extensible means for identifying a resource. An URI consist of a HTTP protocol , Location of the resource, Path to the resource, and last the resource.

7.What are Idempotent methods and why do we call them?

We call any methods idempotent if it gave same results when we call it over and over again. GET , HEAD , PUT , DELETE are some idempotent method because they shows the same result if we call it over and over again.

8.Explain HTTP Request & Response Messages.

HTTP messages are how data is exchanged between a server and a client. There are two types of messages:

HTTP requests – These are sent by the client to trigger an action on the server

HTTP responses – These are sent by the server to the client carrying necessary responses which the client requested.

9.What is Session State in HTTP?

Session state is a method to keep track of the user sessions during a series of HTTP requests. Session state allows a developer to store data about a user as he/she navigates through web pages. Session data is stored at the user level

10.What is HTTPS?

Hypertext Transfer Protocol Secure (HTTPS) is an extension of the [Hypertext Transfer Protocol](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol) (HTTP). It is used for [secure communication](https://en.wikipedia.org/wiki/Secure_communications) over a [computer network](https://en.wikipedia.org/wiki/Network_operating_system), and is widely used on the Internet

**Introduction to API**

1.Explain REST and RESTFUL?

Representational State Transfer **(REST)** is an architectural style for developing loosely coupled web applications .It is popular due to its simplicity and it uses the existing features of the HTTP protocols. Here, the client and server exchange the representation of resources using the HTTP protocols.

2.Mention what are the HTTP methods supported by REST?

REST supports all the HTTP methods but CRUD operations(POST,GET,PUT,DELETE,PATCH) are commonly used.

3.Explain the architectural style for creating web API?

There are many architectural styles for creating a web API but REST and SOAP are commonly used. SOAP is a traditional architectural style which consumes more time but more secure. While REST is modern API architectural style which is really fast but less secure while compared to SOAP.

4.Explain the RESTFul Web Service?

It is a lightweight, scalable service that is built on the REST architecture. Using RESTFul web service we can expose API from the application to the client in a secure, uniform and stateless manner. And the client can perform operations using the RESTFul service. HTTP protocols are widely used for this.

5.Explain what is a “Resource” in REST?

In REST, Resource means the data that we need to get from the server. If a client sends a request on a REST web service, he is actually requesting the Resource which is lying anywhere in the server. But the server will not send the Resource to the client. Instead, the server sends the Representation of the Resource which will be a JSON file to the client.

6. Which protocol is used by RESTful web services?

Standard HTTP Protocols are used by the RESTFul web services.

7. What is messaging in RESTful web services?

RESTFul webservices use HTTP protocols as a medium of communication between the client and server. A client sends a message in form of a HTTP Request and the server responds in the form of HTTP Response. This technique is termed as messaging.

8.State the core components of an HTTP Request?

The core components of HTTP Requests are;

* Verb – indicates HTTP methods like GET, POST, PUT, DELETE, etc.
* Uniform Resource Identifier (URI) – to identify the Resource on the server.
* HTTP Version – to indicate the version of HTTP
* Request Header – It contains metadata for the HTTP Request messages as a key-value pair (client type, format of message body, cache stings, format supported by client, etc.).
* Request Body – It is the Representation of the resource or the message content.

9. State the core components of an HTTP Response?

The core components of HTTP Response are;

* Status code – to indicates server status for the requested resource.
* HTTP Version – to indicate the version of HTTP.
* Response Header – It contains metadata for the HTTP Response message as a key-value pair (content length, content type, response date, server type, etc.).
* Response Body – It consist of the response message

10.What do you understand about payload in RESTFul web service?

In RESTFul web service, the payload contains the data that we send to the server can only be send to the server through POST method.

11.Explain the caching mechanism?

Caching refers to storing the server response in the client’s computer. If a client sends request for a resource on the server, that will be stored as cache and If the client request for the same resource again instead of reading the data from it source, the data is read directly from the cache stored on the computer.

12.List the main differences between SOAP and REST?

**SOAP**

* SOAP is a protocol. It includes a WSDL file which has the required information which need to pass along with the request.
* SOAP uses XML for communication
* SOAP is a traditional method
* SOAP is more secured but slow compared to REST
* SOAP cannot make use of REST since SOAP is a protocol and REST is an architectural style

**REST**

* REST is an architecture style
* REST supports different data formats like HTML, XML, JSON etc. but more preferred format is JSON
* REST is a Modern Style
* REST is less secured but fast compared to SOAP
* REST can make use of SOAP because REST is an architectural style and SOAP is an underlying protocol for web services

13.Enlist advantages and disadvantages of ‘Statelessness’.

**Advantages**

* Web services can treat each method request independently
* Any previous communication with the client and server is not maintained and thus the whole process is very much simplified.
* In every HTTP request from the client, the availability of some information regarding the client state is required by the web service

**Disadvantages**

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