**1. HOW HTTPS work behind the scene?**

Ans.HTTPS (Hyper Text Transfer protocol Secure) is basically an HTTP protocol with additional securities.This additional security can be important for the website that takes sensitive data from its users such as passwords, credit card information etc..

It is useful in increasing the security of data transfer.

HTTPS uses encryption protocol named Transport layer Security (TLS) to encrypt communication.

This protocol secures communication by using asymmetric public key encryption.

Behind the scene in HTTPS communication-

(a) A shared symmetric key is generated which can only be known between client and the server , no one else know it.

(b) With that shared symmetric key, client and server able to communicate safely without worrying about anything.

**2. What are different http methods available and what they exactly do.**

Ans. There are various http methods;

1. **Get method**: Get retrieves the representation of the resource at a specified URI. GET should have no side effects on the server.
2. **PUT Method:** put updates a resource at a specified URI . PUT can also be used to create a new resource at a specified URI, if the server allows clients to specify new URIs. for this tutorial ,the API will not support creation through PUT.
3. **POST Method:** POST creates a new resource . The server assigns the URI for the new object and returns this URI as part of the response message.
4. **DELETE Method:** DELETE delete a resource at a specified URI.
5. **CONNECT Method :** It is used to request the server to delete a file at a location specified by the given URL.
6. **OPTION Method:** it is used by the client to find out the HTTP methods and other options supported by a web server
7. **TRACE Method:** IT is used to echo the contents of an HTTP request back to the requester which can be used for debugging purpose at the time of development.

**3.Understand and explain the use of various http response codes.**

**ANS.** There are various response codes:

1. **Information response codes-**

**100 Continue :** This interim response indicates that everything so far is ok and that the client should continue the request ,or ignore the response if the request is already finished.

**101 Switching protocol:** This code is sent in response to an upgrade request header from the client and indicates the protocol the server is switching to.

**102 Processing:** This code indicates that the server has received and is processing the request ,but no response is available yet.

**103 Early Hints:** This status code is primarily intended to be used with the link header, letting the user agent start preloading resource while the server prepares a response.

**Successful**  **responses codes:**

**200 Ok:** The request has succeeded:

**201 Created :** The request has succeeded and a new resource has been created as a result. This is typically the response sent after POST requests, or some PUT requests.

**202 Accepted:** The request has been received but not yet acted upon.It is noncommittal, since there is no way in HTTP to later send an asynchronous response indicating the outcome of the request .It is intended for cases where another process or server handles the request, or for batch processing.

**203 Non -Authoriative information:** This responses code means the returned meta-information is not exactly the same as is available from the origin server, but is collected from a local or a third party copy. This is most used for mirrors or backups of another resource. Except for that specific case, the ‘’200 ok’’ response is preferred to this status.

**204 No content:** There is no content to send for this request, but the headers may be useful.

The user-agent may update its cached headers for this resource with the new ones.

**© Redirection response codes:**

**300 Multiple choice:** The request has more than one possible response. The user-agent or users should choose one of them.

**301 Moved permanently:**  The URL of the requested resource has been changed permanently . The new URL is given inthe response.

**302 Found:** This response code means that the URI of requested resource has been changed temporarily. Further changes in the URI might be made in the future. Therefore,this same URI should be used by the client in future request.

**303 See other:** The server sent this response to direct the client to get the requested resource at another URI with a GET request.

**304 Not modified:** This is used for catching purposes. It tells the client that the response has not been modified,so the client can continue to use the same cached version of the response.

**(D) Client error response codes:**

**400 bad request:** The server could not understand the request due to invalid syntax.

**401 Payment required:** This responses code is reserved for future use. The initial aim for creating this code was using it for digital payment systems, however this status code is used very rarely and no standard convention exists.

**402 Forbidden :** The client does not have access right to the content; that is, it is unauthorized, so the server is refusing to give the requested resource.

Unlike 401, the client’s identify is known to the server.

**(d) Server error responses codes;**

**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 cannot be handled. The only methods that servers are required to suppor must nit return are GET and HEAD .

**502 Bad gateway:** This error response means that the server, while working as a gateway to get a response neded to handle the request, got an invalid response.

**503 Service unavailable:** The server is not readly to handle the request. Common causes are a server that is down foe maintenance or that is overloaded.

**504 Gateway Timedout:** This error response is given when the server is acting as a gateway and cannot get a response in time .

**505 HTTP version not supported:** The HTTP version used in the request is not supported by the server.

**4.What are the different web communication protocols and their use caes.**

**Ans.** There are various web communication protocols used to communicate over internet. Some of them are:

**Hyper Text Transfer Protocol (HTTP) :** It is designed for transferring a hypertext among two or more systems. It is designed on Client-server principles which allow a client system for establishing a connection with the server machine for a making a requests. The server acknowledges the request initiated by the client and responds accordingly.

**Hyper Text Transfer Protocol Secure (HTTPS):** It is a standard protocol to secure the the communication among twocomputers one using the browser and other fetching data from web server. I this proticol transferring of data is done in an encrypted format.

**Telnet :** It is a set of rules designed for connecting one system with another. The system which accepts the connection is the remote computer.

**Transmission Control Protocol (TCP):** It is used for communicating over a network . It divides any messages into series of packets that are sent from source to destination and there it gets reassembled at the destination.

**User Datagram Protocol (UDP): It** is a substitute communication protocol to Transmission Control Protocol implemented primarily for creating loss-tolerating and low- latency linking Between different application. It provides best-effort delivery.

**Post office protocol (POP):** It is designed for receiving incoming E-msils.

**Simple mail transport Protocol (SMTP):** It is designed to send and distribute outgoing E-Mails.

**File Transfer Protocol (FTP):** It allows users to transfer file from one machine to another. Types of files may include program fikes, multimedia files, text files, and documents, etc..

**5. Pros and cons of single page and multipage application:**

**Ans.**

**Pros of single page application :**

1. It is fast, as most resources are only loaded once throughout the lifespan of application.
2. It is easy to debug with chrome.
3. The development is simplified and streamlined.
4. There is no need to write code to render pages on the server.
5. It is easier to make a mobile application because the developer can refuse the same backend code for web application and native mobile application.
6. It provides seamless user experience.
7. It uses low bandwidth because it loads web pages once.

**Cons of single page application:**

1. IT requires a lot of resource from the browser since the browser is doing most of the tasks for single page application.
2. It requires a lot of resources from the browser since the browser is doing most of the tasks for single page application.

**Cons of multi page application:**

1. The development becomes quite complex in it.
2. Frontend and Backend development are tightly coupled.
3. It is slow in performance because server needs to reload most resources.
4. Maintenance and updates are quite difficult in it.

**6.Explain TCP and its working?**

ANS. It stands for transmission control protocol. It is transport layer protocol.

It is responsible for secure transmission between sever and client.

It is also helpful in error detection, reordering of packets.

It introduces latency in a TCP stream.

It is a connection oriented protocol.

It divides file received from the application layer into data packets, creates connection and forward them individually to IP layer.

It is the most common protocol in networks that use IP (Internet protocol).

It uses 3 way handsaking to make connection between server and client by these three steps:

1. Client sends as SYN establish connection.
2. Server responds to the client requests SYN-ACK signal bits set.
3. Now, client acknowledges the response of server.

Now the communication is started.

# 