**HTML Fundamentals**

1. What are the basic Features of HTTP?

Answer:

* Media independent: Any type of data can be sent by Http and both client and server need to specify the content type using appropriate MIME type.
* Connectionless: Client and server know about each other only during request and response. Client initiate request and wait for response. Server process request and send response back to client, then disconnect the connection
* Stateless: Client and Server will only be in contact during the request. They do not store any information about the request processed. Thus each request will be unique.

1. What are request methods in HTTP?

Answer:

* GET: Get request is used to request data from a specified resource
* POST: Post request is used to send data to server to create a resource
* DELETE: Delete Request is used to delete a specified resource
* PUT: Put request is used to send data to server to replace a resource completely if already available or create new resource
* PATCH: Patch request is used to make partial update on an existing resource
* CONNECT: Used to establish a tunnel to the server identified by the given URI
* HEAD: It is same as Get request but response is without body
* OPTIONS: Method used to describe the communication options for the target resource

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

Answer:

|  |  |
| --- | --- |
| GET | POST |
| Get carry request parameter in the URL | POST carry request parameter in the message body |
| Can be cached | Cannot be cached |
| GET method is less secure compared to POST because data sent is part of URL | POST method is more secure than GET |
| Used for retrieving data | Used for changing data by creating new resource |
| Get is idempotent method | POST can’t be an idempotent method |

1. What is status code in HTTP?

Answer:

Status codes are issued by server in response to the client request. It can be divided into 5 categories.

|  |  |  |
| --- | --- | --- |
| Range | Category | Description |
| 100-199 | Informational | Request received successfully but action pending |
| 200-299 | Success | Action was successful |
| 300-309 | Redirection | Action not success yet, redirect to another page |
| 400-499 | Client error | Client request is invalid which may bad syntax |
| 500-599 | Server error | Request is valid but server failed to fulfil it |

1. What are the header fields in HTTP?

Answer:

Http header fields pass additional information with an Http request or response to client and server. Header field will have a name followed by colon then its value.

Example. Accept: text/plain

Header are of 4 types:

* General header: Apply to both request and response
* Request header: Only applicable to request message, contain more information about resource to be retrieved.
* Response header: Applicable only to response message, have additional information about response like server providing it.
* Entity header: Hold information about the entity body like MIME type.

1. What is URI?

Answer:

Uniform Resource identifier(URI) is a string that uniquely identifies a particular resource.

<http://bank.com> Example of URL which locate all the banks

<http://bank.com/icicibank> Example of URI which identifies the specific bank

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

Answer:

Idempotent methods are operations that will produce the same results even if it is executed multiple times.

Http Methods GET, HEAD, DELETE, PUT are idempotent

1. Explain HTTP Request & Response Messages

Answer:

An Http Client send request to server in the form of Http Request Message. It has the general format of

* Request line: Begins with name of method then Request URI followed by protocol version
* Request Header(Optional): Allowed client to pass additional information about the resource to be fetched.
* Blank line: Separates Request header and Request body
* Request Message Body(Optional):

An Http Server responds with an Http Response Message with the general format

* Status line: Begins with protocol version followed by status code and its text
* Response Header(Optional): Has additional information about the response like server providing it or its location.
* Blank line: Separates Response header and Response body
* Response Message Body(Optional):

1. What is Session State in HTTP?

Answer:

Http protocol is stateless which means it does not store the details of clients while browsing. Session state is a method to save client session during http requests. It allows developers to store user details when they navigate through web pages. Thus session state can make http statefull.

1. What is HTTPS?

Answer:

Hypertext Transfer Protocol Secure is an extension of http used for secure communication in a network. It is more secure version of http because https connections are encrypted.

**Introduction to API**

1. Explain REST and RESTFUL?

Answer:

REST (Representational State Transfer) is an Architectural style that uses Http Protocol. Every component in REST is considered as a Resource. URI’s are used to identify resources. REST defines a set of rules on top of Http to create web services.

RESTful service is an implementation of REST.

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

Answer:

Most commonly used Http methods in REST are:

GET: Retrieve data from a resource

POST: Create new resource

PUT: Update/Replace resource

PATCH: Modify some fields in a resource

DELETE: Delete a resource

1. Explain the architectural style for creating web API?

Answer:

REST is one of the best architectural style for creating web API.

REST stands for Representational State Transfer. It defines a set of rules to be followed for creating web services. Protocol used will be Http. RESTful system will have Client and Server. Client will request for resource and Server has the resource and send response to Client.

Architectural constraints of REST API’s

* Uniform Interface: Server interaction will be uniform irrespective of the type of device
* Caching: Store server responses to cache memory for quick access.
* Stateless: Each request will be considered as an independent one. Session details will not be stored.
* Layered: Architecture is of multiple layers. There will be intermediately servers between client and database server
* Code on Demand: Server can provide code to client upon Request.But it is optional
* Client-Server: REST is based on Client Server Architecture.

1. Explain the RESTFUL Web Service?

Answer:

RESTFUL Web services are web services which are based on REST Architecture. These web services use Http Methods to implement the concept of REST. It allows requesting system to access web resources.

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

Answer:

The main concept of REST is Resource. It can be any information which is referenced between client and server. Text files, html pages, images, videos all are resources. In REST all contents are treated as resource.

1. Which protocol is used by RESTful web services?

Answer:

RESTful web services are REST based. The underlying protocol for REST is HTTP.

1. What is messaging in RESTful web services?

Answer:

RESTful web services use HTTP protocol for communication between client and server. Client send an HTTP Request and Server send back response through HTTP Response. This is messaging in Web services.

1. State the core components of an HTTP Request?

Answer:

HTTP Request

Http Version

URI

Http Method

Request Header

Request Body

* Http Method: Indicate the Http Methods such as GET, PUT, POST, DELETE
* URI: Uniform Resource Identifier to uniquely identify resource
* Http Version: Indicate the current Http version
* Request Header: Allows client to pass additional information about the resource to be fetched.
* Request Body: Message content

1. State the core components of an HTTP response?

Answer:

* Status Code: Indicate the status of the requested resource (eg.200 for OK)
* Http Version: Indicate Http version
* Response Header: Has additional information about the response like server providing it or its location
* Response Body: Response Message content

Http Response

Http Version

Status/Response Code

Response Header

Response Body

1. What do you understand about payload in Restful web service?

Answer:

Payload is data that is transmitted across Client and Server. It can be part of either Client Request or Response from the Server.

1. Explain the caching mechanism?

Answer:

Caching means storing the result from the server to a cache memory. When the same request comes from client instead of fetching the result from database, data will be fetched from cache. Caching mechanism allows to effectivity reuse previously retrieved data

1. List the main differences between SOAP and REST?

Answer:

|  |  |
| --- | --- |
| SOAP | REST |
| SOAP-Simple Object Access Protocol | REST-Representational State Transfer |
| SOAP is a Protocol | REST is an Architectural style |
| Data is available as services-Function Driven | Data is available as resources- Data Driven |
| Data cannot be cached | Data can be cached |
| Uses only XML Data format | Allows plain text, Html, XML ,JSON |
| Require more resources and bandwidth | Require fewer resources |
| More secure. WS security with SSL support | Less secure  Support HTTPS and SSL |
| Mainly used in financial services, payment gateways, high security apps | Recommended for social network, mobile services |

1. Enlist advantages and disadvantages of ‘Statelessness’.

Answer:

Advantages:

* + As server do not need to manage any sessions, scalability can be achieved by increasing number of servers
  + Improves visibility as each request is considered as unique
  + Less complexity

Disadvantage:

* As each request is considered as new ,it is required to pass additional information and the server has to process it again