API Interview Questions:

**What is API Testing process in your project?**

I am working with API team on one of our projects, the process is we get API documentation through swagger and the dev team provide endpoint URL, parameters, headers, json body for request and expected json body for response. For automation we use Karate framework for current project and also, we have postman to do some manual validation before writing our Karate scripts. We mainly validate Response status code, response headers, and also JSON body. For JSON body we validate each json response node and make sure the data is matching based on expected results. we do end to end testing to validate response against request and request against Database.

**What are the core components of an HTTP request?**

An HTTP request includes five key elements:

* HTTP methods – Set of request methods to perform desired action for a given resource (GET, PUT, POST, DELETE)
* Uniform Resource Identifier (URI) – Describes the resource
* HTTP Version, (example- HTTP v1.1)
* Request Headers, (example- Content-type : application/json, Content-Length : 511)
* Payload – It is basically a Request Body which includes message content.

**What are the Core Components of an HTTP Response?**

Every HTTP response contains four key elements.

* Status/Response Code – These are response codes issued by a server to a client’s request. For example, 404 means Page Not Found, and 200 means Response is OK.
* HTTP Version – describes HTTP version, for example-HTTP v1.1.
* Response Header – Includes information for the HTTP response message. For example, Content-type, Content-length, date, status and server type.
* Response Body – It contains the data that was requested by a client to server.

**In which type of encoding does postman accept authorization credentials?**

Postman accepts Base64 encoding only. This is provided inbuilt in postman

Why does Postman accept Base64 encoding only?

We use base64 particularly because it transmits the data into the textual form and sends it in easier form such as HTML form data. Also, we can rely on the same 64 characters in any encoding language that we use.

**What is meant by the term environment in postman?**

An environment in postman is a set of key value pairs. You can create multiple environments in postman which can be switched quickly with a press of a button. There are 2 types of environments, global and local.

**Can global scope variables have duplicate names in postman?**

Since global variables are global i.e. without any environment, global variables cannot have duplicate names. Local variables can have the same name but in different environments.

**Which one will be preferred in postman- a global variable or a local variable?**

In postman, if 2 variables have the same name( one being local, other global) then the higher priority is of the local variable. it will overwrite the global variable.

**What is a Postman Collection?**

A Postman Collection lets us group individual requests together. Simply it allows us to organize the requests into folders.

**What do you mean by postman monitors?**

The postman monitor is used for running collections. Collections are run till specified time defined by the user. Postman Monitor requires the user to be logged in. Monitor reports are shared by users over email on a daily/monthly basis.

**What do you understand by the term Postman Collection runners?**

A postman collection runner is used to perform Data-driven testing. The group of API requests are run in a collection for the multiple iterations with different sets of data.

**Can local variables be imported in Postman Monitors?**

Yes. Postman monitors allow to import local variables but it does not allow to import global variables.

**What is the purpose of Postman cloud if we are working in a company? Why?**

A Postman cloud is a common repository of companies to access Postman collections. In Postman cloud, work can be saved instantly after logging in. Anyone from the team can access data/collections from anywhere.

**Why is it not preferred to save work in Postman cloud?**

It is not preferred to save your work in Postman cloud as company’s work is not allowed to be leaked and remain confidential. Security breaches can be experienced if Postman cloud is used as Postman cloud requires sign in. Therefore Postman Cloud is discouraged for saving work and team workspace is highly encouraged.

**What is the purpose of status code 304?**

It means NOT MODIFIED. It is used to reduce network bandwidth usage in case of conditional GET requests. Response body should be empty. Headers should have date, location etc.

**Define status code 201?**

It means created, when a resource is successfully created using POST or PUT request. It returns a link to a newly created resource using the location header.

**When do we use global variables, collection variables, and local variables?**

Global variables are general purpose variables, ideal for quick results, and prototyping. They are used while passing data to other requests.

Collection variables can be mostly used for storing some constants that do not change during the execution of the collection. They are used for constants that do not change during the execution and also for URLs / authentication credentials if only one environment exists.

Local variables are only available within the request that has set them or when using Newman/Collection runner during the entire execution. They are used whenever you would like to override all other variable scopes.

**How do you remove local variables?**

Local variables are automatically removed once the tests have been executed.

**How can we stop executing requests or stop the collection run?**

postman.setNextRequest(null);

**Where are query parameters stored in a GET request?**

Query parameters are stored in the URL in a GET request.

**How can we access a Postman variable?**

We can access a Postman variable by entering the variable name as {{var}}

**What is the HTTP response code for a POST request with incorrect parameters?**

400 Bad Request is an ideal response code for request with incorrect parameters.

**How can you iterate a request 100 times in Postman?**

By using Collection Runner

**How can we organize requests in Postman?**

We can organize requests in Postman with the Collections.

**Which programming language is used for Postman tests?**

JavaScript

**What will execute first in a Collection Run?**

Pre-request scripts at the Collection level are executed first in a Collection run.

**What are some of the JS libraries available in Postman?**

Lodash, Moment, GUID

**Which tool can be used to run Postman Collections in Jenkins?**

Newman can be used.

**How can we log requests and responses in Postman?**

We can view requests logs and response logs through the Postman Console window.

**What kinds of bugs does API testing find most commonly?**

Since API testing is testing business logic, we can find duplicated values, missing data types on response, updating wrong data tables.

HTTP Status Codes List: These are important HTPP Status codes. For full series of codes you can check below url:

<https://restfulapi.net/http-status-codes/>

**100 :** An interim response.

Indicates the client that the initial part of the request has been received and has not yet been rejected by the server.

**101 :** Sent in response to an [Upgrade](https://developer.mozilla.org/en-US/docs/Web/HTTP/Protocol_upgrade_mechanism) request header from the client, and indicates the protocol the server is switching to.

**102 :** Indicates that the server has received and is processing the request, but no response is available yet.

**103 :** Primarily intended to be used with the Link header. It suggests the user agent start preloading the resources while the server prepares a final response.

**200**: Indicates that the request has succeeded.

**201**: Created indicates a record was created successfully

**202**: Indicates that the request has been received but not completed yet. It is typically used in log running requests and batch processing.

**203**: Non-Authoritative Information Indicates that the returned metainformation in the entity-header is not the definitive set as available from the origin server, but is gathered from a local or a third-party copy. The set presented MAY be a subset or superset of the original version.

**204**: No Content The server has fulfilled the request but does not need to return a response body. The server may return the updated meta information.

**400 Bad Request**

The request could not be understood by the server due to incorrect syntax. The client SHOULD NOT repeat the request without modifications.

**401 Unauthorized**

Indicates that the request requires user authentication information. The client MAY repeat the request with a suitable Authorization header field

**402 Payment Required (Experimental)**

Reserved for future use. It is aimed for using in the digital payment systems.

**403 Forbidden**

Unauthorized request. The client does not have access rights to the content. Unlike 401, the client’s identity is known to the server.

**404 Not Found**

The server can not find the requested resource.

**405 Method Not Allowed**

The request HTTP method is known by the server but has been disabled and cannot be used for that resource.

**406 Not Acceptable**

The server doesn’t find any content that conforms to the criteria given by the user agent in the Accept header sent in the request.

**407 Proxy Authentication Required**

Indicates that the client must first authenticate itself with the proxy.

**408 Request Timeout**

Indicates that the server did not receive a complete request from the client within the server’s allotted timeout period.

**409 Conflict**

The request could not be completed due to a conflict with the current state of the resource.

**422 un-Processable Entity (WebDAV)**

The server understands the content type and syntax of the request entity, but still server is unable to process the request for some reason.

**500 Internal Server Error**

The server encountered an unexpected condition which prevented it from fulfilling the request.

**501 Not Implemented**

The HTTP method is not supported by the server and cannot be handled.

**502 Bad Gateway**

The server got an invalid response while working as a gateway to get a response needed to handle the request.

**503 Service Unavailable**

The server is not ready to handle the request.

**504 Gateway Timeout**

The server is acting as a gateway and cannot get a response in time for a request.

**505 HTTP Version Not Supported (Experimental)**

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