1. What is Postman?

Answer: Postman is a popular collaboration platform and API development tool that simplifies

the process of designing, testing, and documenting APIs. It provides a user-friendly interface

to make API requests, automate testing, and share API-related information among team members.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

2. Explain the purpose and benefits of using Postman for API testing?

Answer: Postman is a powerful tool for API testing that offers numerous benefits.

Its purpose is to simplify and streamline the process of testing APIs. Some benefits

of using Postman for API testing include:

\* Easy API Exploration:

Postman provides a user-friendly interface for creating and sending requests to APIs,

allowing you to explore and understand their functionalities.

\* Efficient Testing:

With Postman, you can quickly test various scenarios by sending different types of requests

(GET, POST, PUT, DELETE) and examining the responses.

\* Request and Response Validation:

Postman allows you to validate API responses, ensuring they meet the expected criteria

and identifying any errors or inconsistencies.

\* Environment Management:

Postman lets you create and manage environments, making it convenient to switch between

different setups, such as development, staging, and production.

\* Collaboration and Sharing:

Postman facilitates teamwork by enabling the sharing of collections, requests, and test suites,

fostering collaboration among developers, testers, and other stakeholders.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

3. How do you create and send a request in Postman?

Answer: To create and send a request in Postman, follow these steps:

\* Open Postman and create a new request by selecting the appropriate HTTP method

(e.g., GET, POST) from the dropdown menu.

\* Enter the request URL in the address bar.

\* Add any required headers, query parameters, or request body data, depending on

the API endpoint's requirements.

\* Click the "Send" button to send the request to the API server.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

4. What are some key features of Postman that you find most useful in your testing workflow?

Answer: Some key features of Postman that are often found useful in testing workflows include:

Collections:

It allows you to organize and group related requests, making managing and executing test suites easier.

Variables and Environments:

Postman allows you to define variables and environments, which help parameterize requests

and manage different testing environments.

Test Scripts:

Postman's test scripts, written in JavaScript, enable you to automate assertions, extract data

from responses, and perform complex validations.

Pre-request Scripts:

These scripts allow you to perform actions before sending a request, such as setting up

dynamic data or configuring authentication.

Request History:

Postman keeps a history of previously sent requests, allowing you to revisit and retest them

without re-entering the details.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

5. How do you handle authentication and authorization in Postman?

Answer: Postman provides multiple ways to handle authentication and authorization in API requests.

Some commonly used methods include:

\* Basic Authentication: You can include the username and password in the request headers

using the "Authorization" header.

\* Token-based Authentication: Postman allows you to include tokens (such as JWT or OAuth)

in the request headers or as query parameters.

\* API Key: If an API requires an API key, you can pass it as a request header or query parameter.

\* OAuth 2.0: Postman has built-in OAuth 2.0 support, allowing you to configure and authenticate

using various OAuth flows, such as Authorization Code or Client Credentials.

\* Custom Authentication: Postman's scripting capabilities enable you to implement custom

authentication mechanisms by modifying request headers or using specialized libraries.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

6. What can we use to get API information from web developer tools into Postman?

Answer: Copy as cURL allows Postman to receive API data from web developer tools.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

7. Why do we use Postman?

Answer: Following are some of the most important reasons for using Postman:

\* It is software that aids with API testing and is available for free use

\* It aids in managing the complete API lifecycle

\* It provides Runtime Service to assist with managing API collections, workspaces, environments,

and many examples

\* Additionally, Postman may get integrated with CI/CD programs like Circle CI, Jenkins, etc

\* It offers a sizable community forum where you can quickly find solutions to any technical

problems you run across while using the tool

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

8. Where are query parameters stored in a GET request?

Answer: The query parameters for the GET request are saved in Postman's URL.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

9. What is meant by the term environment in Postman?

Answer: An environment in Postman is a group of key-value pairs. Postman allows you to

create various environments that you can switch instantly with a button. There are two

different sorts of environments: global and local. They define the variable's scope so you

can use it in requests.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

10. What are the core components of an HTTP request?

Answer: An HTTP request is made up of five major components:

\* HTTP methods: A collection of request techniques used to carry out particular actions

on resources (GET, PUT, POST, DELETE)

\* URI (Uniform Resource Identifier): locates a resource.

\* Version of HTTP (example- HTTP v1.1)

\* Content-Type: application/JSON, Content-Length: 511) Request Headers

\* Payload: The message content is contained in the request body.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

11. What is GUID?

Answer: Global Unique Identifier is referred to as GUID. It consists of hexadecimal digits

separated by hyphens. This Postman identifier GUID fulfills the goal of uniqueness.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

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

Answer: The correct response code for a request with incorrect parameters is 400 Bad Request.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

13. Can local variables be imported in Postman Monitors?

Answer: Yes, you can import local variables into Postman Monitors. However, global variables

cannot get imported into Postman Monitors.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

14. How can you iterate a request 100 times in Postman?

Answer: With the help of Collection Runner, Postman allows us to iterate a request 100 times.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

21. Which programming language is used for Postman tests?

Answer: Postman tests are run using JavaScript.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

15. What is a Postman Collection?

Answer: We can group several needs using a Postman Collection. It merely enables us to

organize the requests into folders.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

16. What do you understand by the term Postman Collection runners?

Answer: Data-driven testing is carried out by using a Postman collection runner. API

requests are collected for numerous iterations with various data sets.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

17. What is payload in Postman?

Answer: In Postman, a payload refers to the data that is sent with an HTTP request.

It typically includes parameters, headers, and the request body. The payload contains

the information needed by the server to process the request, such as form data, JSON,

or XML data.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

18. How to debug in Postman?

Answer: Debug in Postman using Console logs, Visualizer, Interceptor for breakpoints,

assertions, variable checks, response timings, code snippets, and network logging.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

19. What is HTTP status code?

Answer: HTTP status codes are three-digit numbers returned by a web server to indicate

the success, failure, or other state of a client's request. They convey information about the

outcome of the HTTP request and are categorized into groups, such as 2xx for success,

4xx for client errors, and 5xx for server errors. Examples include 200 for OK, 404 for Not

Found, and 500 for Internal Server Error.