Why Postman?

During the API creating work, it is necessary to test API calls. We can do this in many ways. One of the good methods could be just using browser to enter URL in address bar. However, using this method, we can test only GET requests. Also, there is problem with setting up headers.

Fortunately, it exists a program Postman. It is a software for all kind of HTTP request work. It can be installed as a standalone application and as a Chrome plug in.

"*Postman is a tool for testing APIs, by sending request to the web server and getting the response back.*"

What are the reasons to use Postman as a testing tool for APIs?

1. Easily create test suites.

Postman allows you create collections of integration tests to ensure your API is working as expected. Tests are run in a specific order with each test being executed after the last is finished. For each test, an HTTP request is made and assertions written in JavaScript are then used to verify the integrity of your code. Since the tests and test assertions are written in JavaScript, we have freedom to manipulate the received data in different ways, such as creating local variables or even creating loops to repeatably run a test.

2. Store information for running tests in different environments.

You wrote your test collection and it all works perfectly. You can run it again and again against your local environment and every test passes every time. But your local environment is usually configured a little differently than a test server. Luckily, Postman allows you to store specific information about different environments and automatically insert the correct environment configuration into your test. This could be a base URL, query parameters, request headers, and even body data for an HTTP post.

3. Store data for use in other tests.

Postman also allows you to store data from previous tests into global variables. These variables can be used exactly like environment variables. For example, you may have an API that requires data received from another API. You can store the response (or part of the response, since it is JavaScript) and use that as part of a request header, post body, or URL for the subsequent API calls.

4. Integrates with build systems, such as Jenkins using the Newman command line tool.

Postman has a command line interface called Newman. Newman makes it easy to run a collection of tests right from the command line. This easily enables running Postman tests on systems that don't have a GUI, but it also gives us the ability to run a collection of tests written in Postman right from within most build tools. Jenkins, for example, allows you to execute commands within the build job itself, with the job either passing or failing depending on the test results.

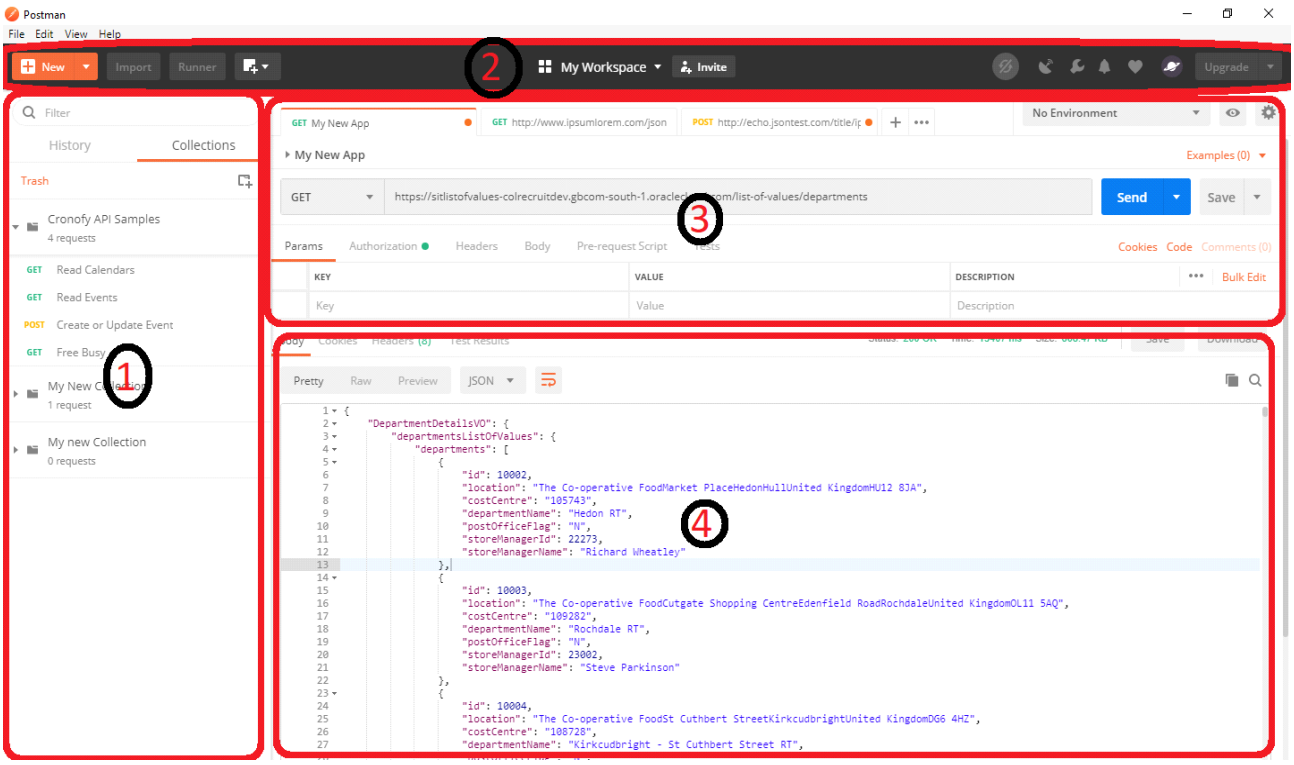
5. Easily move tests and environments to code repositories.

Postman makes it easy to share and move tests into different systems and environments by encapsulating collections in a "test" file and an "environment configuration" file. You can also export a test collection and use traditional filesharing to move it from environment to environment. This also makes it extremely simple to use code repository tools such as Git or SVN to manage the Postman tests and environment files, as well as maintaining version history in case you need to roll back changes.

6. On-the-fly testing

Any time you write some code, you'll want to incrementally run it to make sure there aren't any bugs or typos. Postman makes it easy to do just that. Postman lets you write a test without needing to put it in a collection, save it, or back it up. You just make a request to a URL, then see the result. Postman even parses JSON data to make it a little easier to read.

Postman UI



3.1. Side Bar Section

- History
- Collections

History Tab

Postman records history of your API request just like any other web browser automatically. As soon as you invoke a REST request, it is saved in the history and can be seen below the History Tab.

Collections

The concept of grouping requests is called Collections and each Collection is displayed under the Collection Tab. A collection in Postman can be imagined similar to a folder in your system. You create a folder, for example movies, and keep movies in it so that you know where all your movies are. Similarly, in Postman we save the similar kind of requests under some collection name.

3.2. Header section

- New
- Import
- Interceptor
- Sync

New

Choosing this option will let you choose what new you want to start. New option lets you create the following:

- Request
- Collection
- Environment
- Documentation
- Mock Server
- Monitor

Import

Import option lets you import files of different format. Importing means choosing the files located in your system or through a link and running it through Postman.

Interceptor

You can set proxy server here to capture all the API request that you send through your browser. A proxy server can be used to capture all the requests that you send through your browser or from your phone or any other system.

Sync

Sync option is for synchronizing the API requests that you have sent on any machine to the Postman cloud. When you are working in Postman and making changes or sending requests, if you Sync is on, it will automatically be saved in your Postman's cloud storage.

3.3. Builder Section

A builder part of the Postman is basically what a cpu is to a computer. It is the main part that controls all the functionalities and methods to be incorporated inside the API. These items will help use create a new Request.

- Request Type
- Endpoint Address Bar
- Params

Request Type

This is the request type method for the API. It indicates the type of HTTP Request that has been sent. There are different kinds of requests

GET: Read a specific resource (by an identifier) or a collection of resources.

PUT: Update a specific resource (by an identifier) or a collection of resources. Can also be used to create a specific resource if the resource identifier is known before-hand.

DELETE: Remove/delete a specific resource by an identifier.

POST: Create a new resource. Also, a catch-all verb for operations that don't fit into the other categories.

Endpoint Address Bar

This is the box, beside request type option, to enter the EndPoint (API). It acts just like a browser with the similar interface for New tab. We enter our required endpoint into the bar which is our main URL.

Params

Params is the parameter option which allows us to write the parameters of the URL. The parameters are embedded into a URL and are very important to get the desired result. They also help us in getting efficient usage of the memory and bandwidth.

Authorization

The authorization process verifies whether you have permission to access the data you want from the server. Not all data is available for everyone inside a company, so there lies the solution as Authorization. With the authorization, server first checks whether the data you are asking can be shown to you. If it can be, you get the desired response.

Header

A header in the HTTP request or response is the additional information that is needed to be conveyed between the client-server. HTTP headers are mainly intended for the communication between the server and client in both directions.

3.4. Response Section

A response box is the box which shows the response from the server that we receive after requesting through API.

How to use Postman to execute APIs?

1. Enter the API endpoint where it says 'Enter request URL' and select the method (the action type) on the left of that field.

2. Add authorization tokens/credentials according to the server-side requirements. The different methods/protocols Postman supports are No Authentication, Basic Authentication (provide username and password only), Digest Authentication, OAuth 1.0, OAuth 2.0, Hawk Authentication and AWS Signature.

3. Enter headers in case they are required.

4. Enter a post body in case it is required.

5. If you wish to execute this API, hit the ‘Send’ button, which is located to the right of the API request field. You can also click on the ‘Save’ button beside it to save that API request to your library.

Load Testing with Postman

Postman is one of the best-in-market tools for functional testing of APIs. Postman makes it easy for you to develop, document, and test your APIs. But what about when you want to test those APIs beyond single requests. That is where the Load Impact Postman converter can help.