Introduction to Postman API

Agenda

- Introduction to Postman API
- Importance of Postman API
- Advantages of Postman
- Installing Postman API
- Postman to Execute APIs
- Working with GET and POST Requests
- Parameterize Requests
- Create Postman Tests
- Create Collections
- Run Collections using Collection Runner

Introduction to Postman API

- API:
  - Application Programming Interface is software that acts as an intermediary for two apps to communicate with each other.
  - We use APIs whenever we use an application like Twitter or Facebook, send text messages, or check the weather over the phone.
- HTTP
  - Hypertext Transfer Protocol is the collection of rules for the transmission of data on the WWW like graphic images, text, video, sound, and other multimedia data.
  - The Web users implicitly make use of HTTP as soon as they open their Web browser.

Introduction to Postman API

- Postman is the most popular software testing tool used for API testing. With the help of this tool, developers can easily create, test, share, and document APIs.
- It has a simple GUI for sending and viewing HTTP requests and responses.
- This tool has the ability to make various types of HTTP requests like GET, POST, PUT, and PATCH, and convert the API to code for languages like JavaScript and Python.
- Postman is a software testing tool used for API Testing. API(Application Programming Interface) enables software applications to interact with each other.

Importance Postman API

- **Accessibility –** The Postman tool, can be used by just logging into your own account making it easy to access files anytime, anywhere as long as a Postman application is installed on the computer.
- **Use of Collections –** Postman helps users create collections for their Postman API calls. Each collection can create subfolders and multiple requests helping it in organizing your test suites.
- **Collaboration –** Collections and environments can be imported or exported, making sharing files easy. A direct link can also be used to share collections.
- **Creating Environments –** Having multiple environments helps in less repetition of tests as one can use the same collection but for a different environment.

Importance Postman API

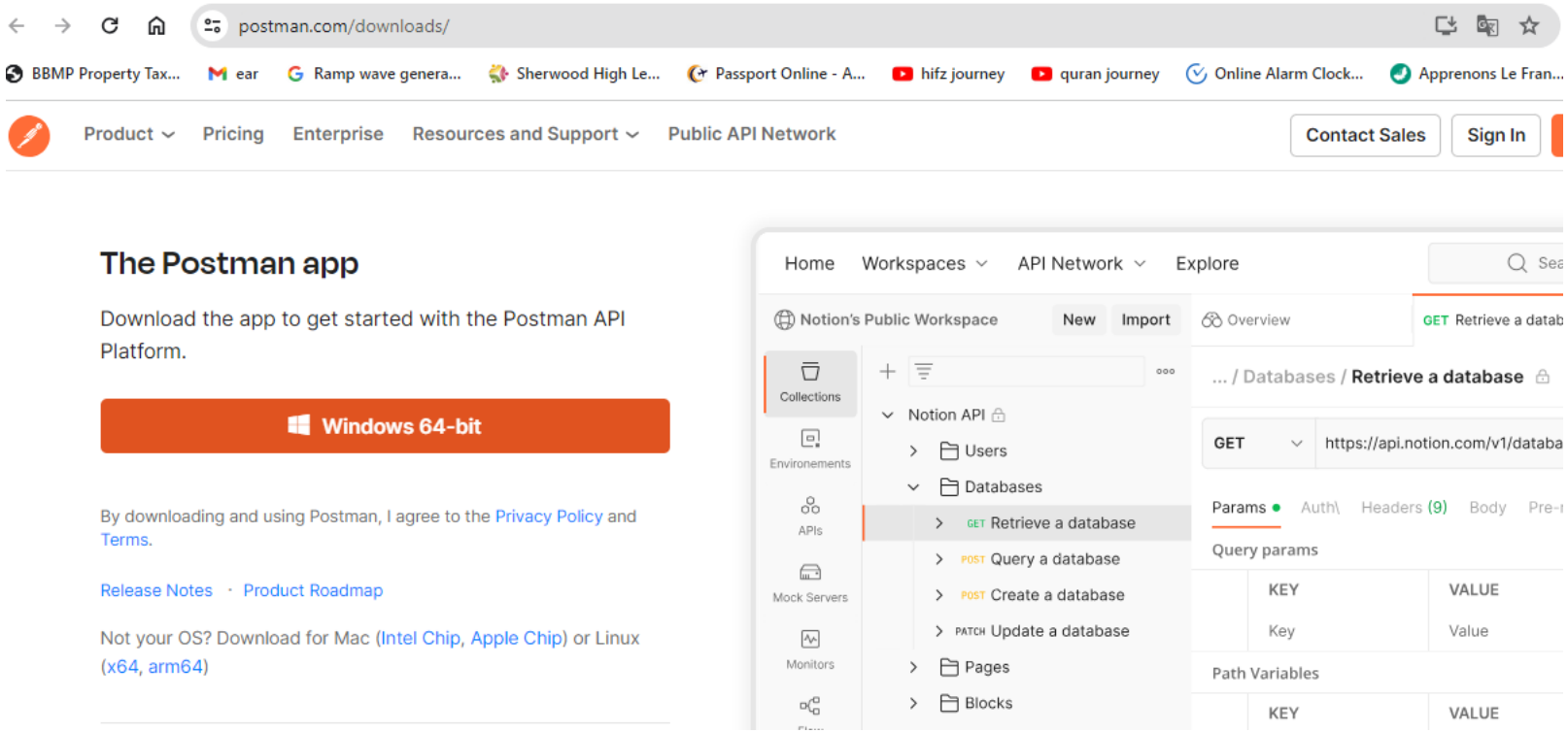
- **Creation of Tests –** Test checkpoints such as verifying for successful HTTP response status can be added to each Postman API call which helps ensure test coverage.
- **Automation Testing –** Using the Collection Runner or Newman, tests can run multiple iterations saving time for repetitive tests.
- **Debugging –** The Postman console helps to check what data has been retrieved making it easy to debug tests.
- **Continuous Integration –** Its ability to support continuous integration maintains development practices.

Advantages of Postman

- Postman enables us to create integration test suites easily.
- Postman permits us to save data for other tests.
- Postman supports integration testing.
- Postman can easily integrate with other build tools like Jenkins, etc.
- Postman eases the movement of environments and tests to code repositories.

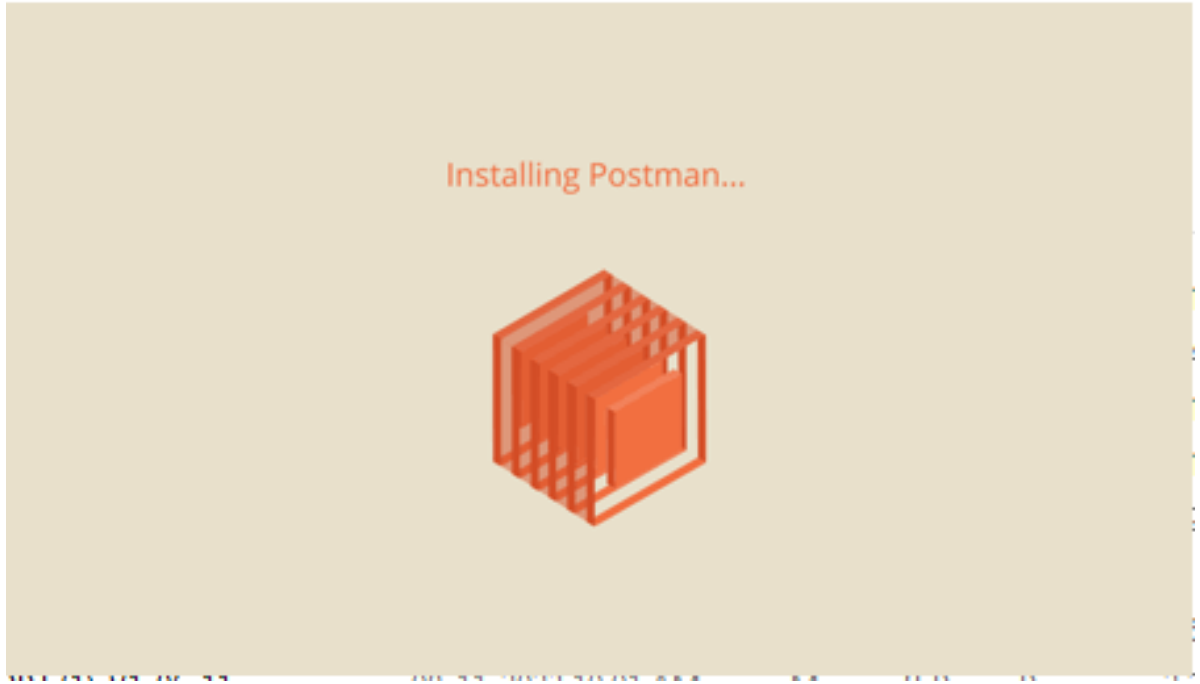
Installing Postman API

- **Step-1:** Go to the link https://www.postman.com/downloads/ and click download for Mac or Windows or Linux based on your operating system.



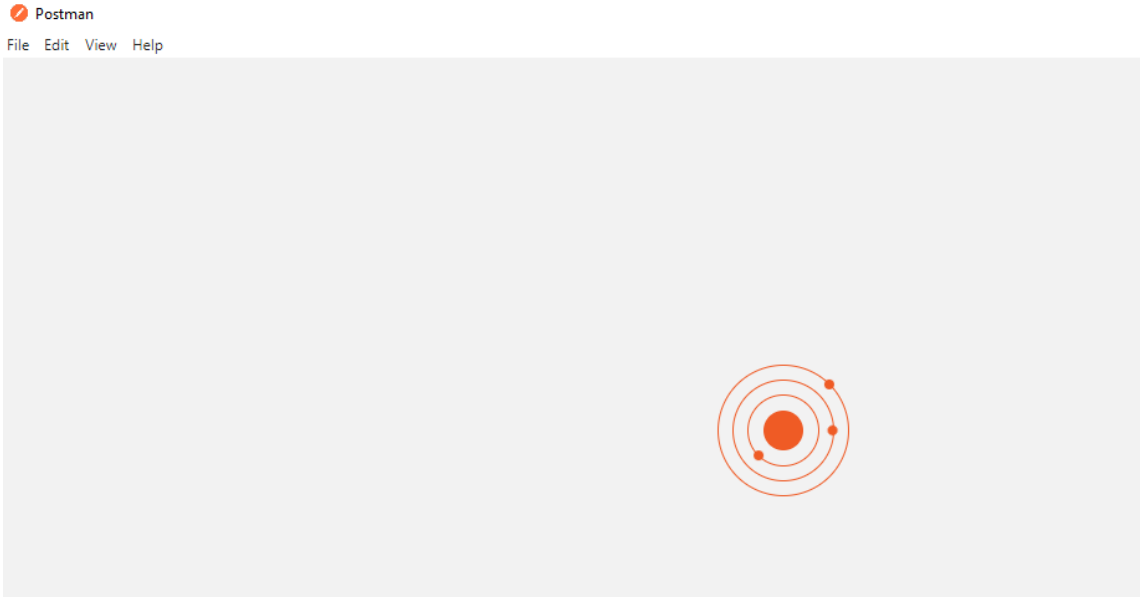
Installing Postman API

- **Step-2:** To download the app for Windows, click on the download button. If you are MAC user download the other options given on the website.
- **Step-3:** Once the .exe file is downloaded, you need to install the application, as shown in the below image.



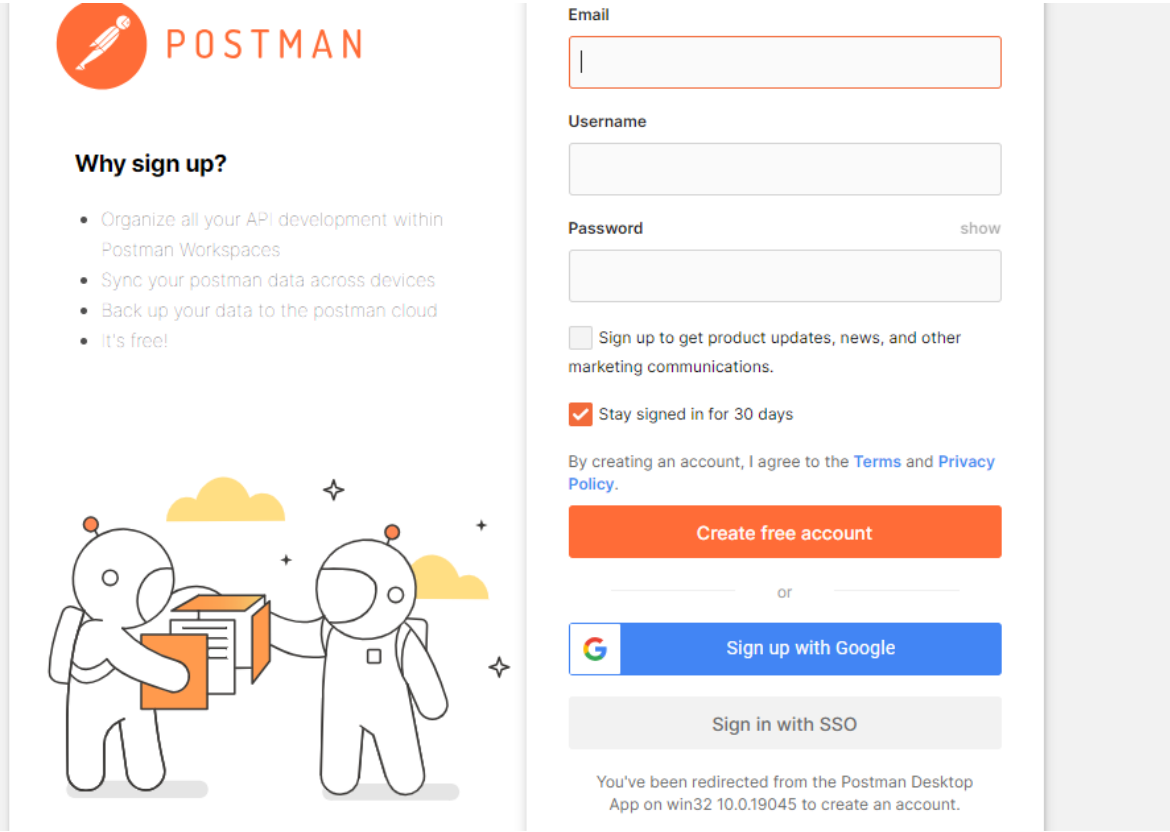
Installing Postman API

- **Step-4:** Once the installation completes, you will be redirected to a window as shown in the image where you can click on **Stop signing in and take me straight to the app** (as this app can also be used without logging in) or otherwise you will get a new window to sign up.



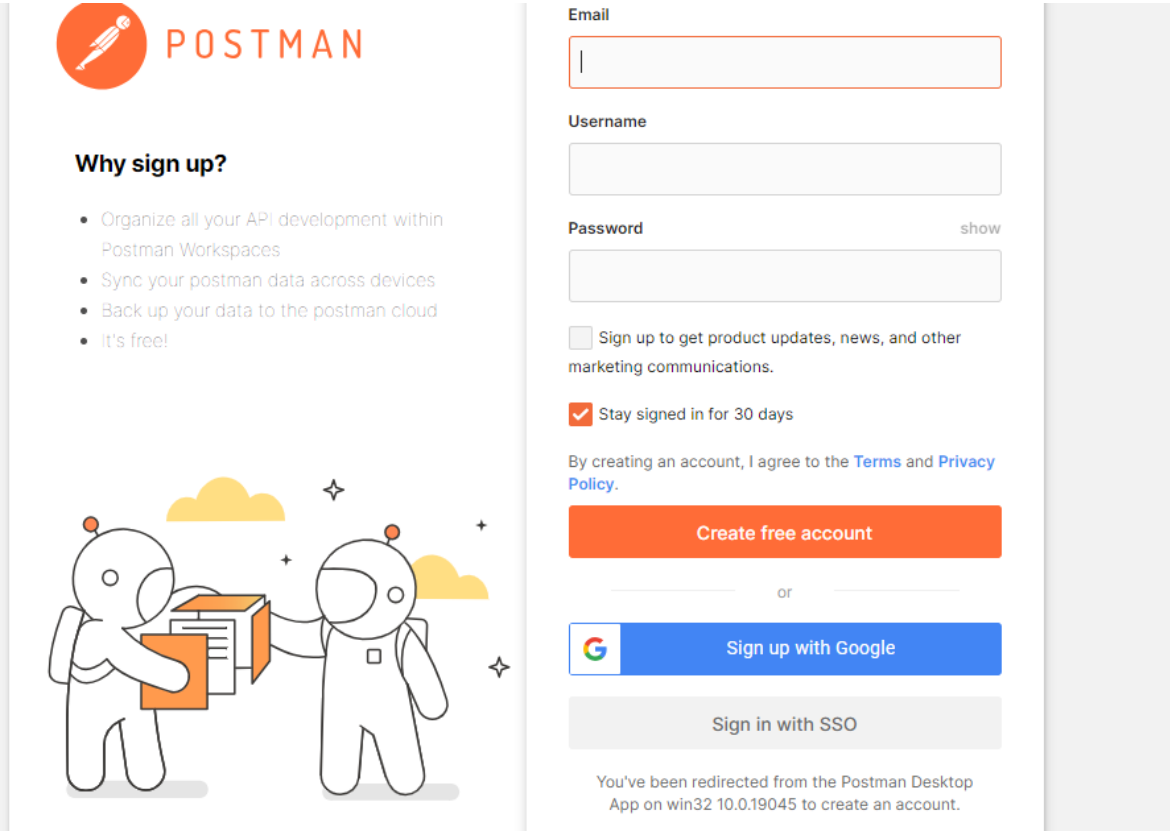
Installing Postman API

- **Step-5:** Create your account with all the required details, or you can also signup with Google, as shown in the image.



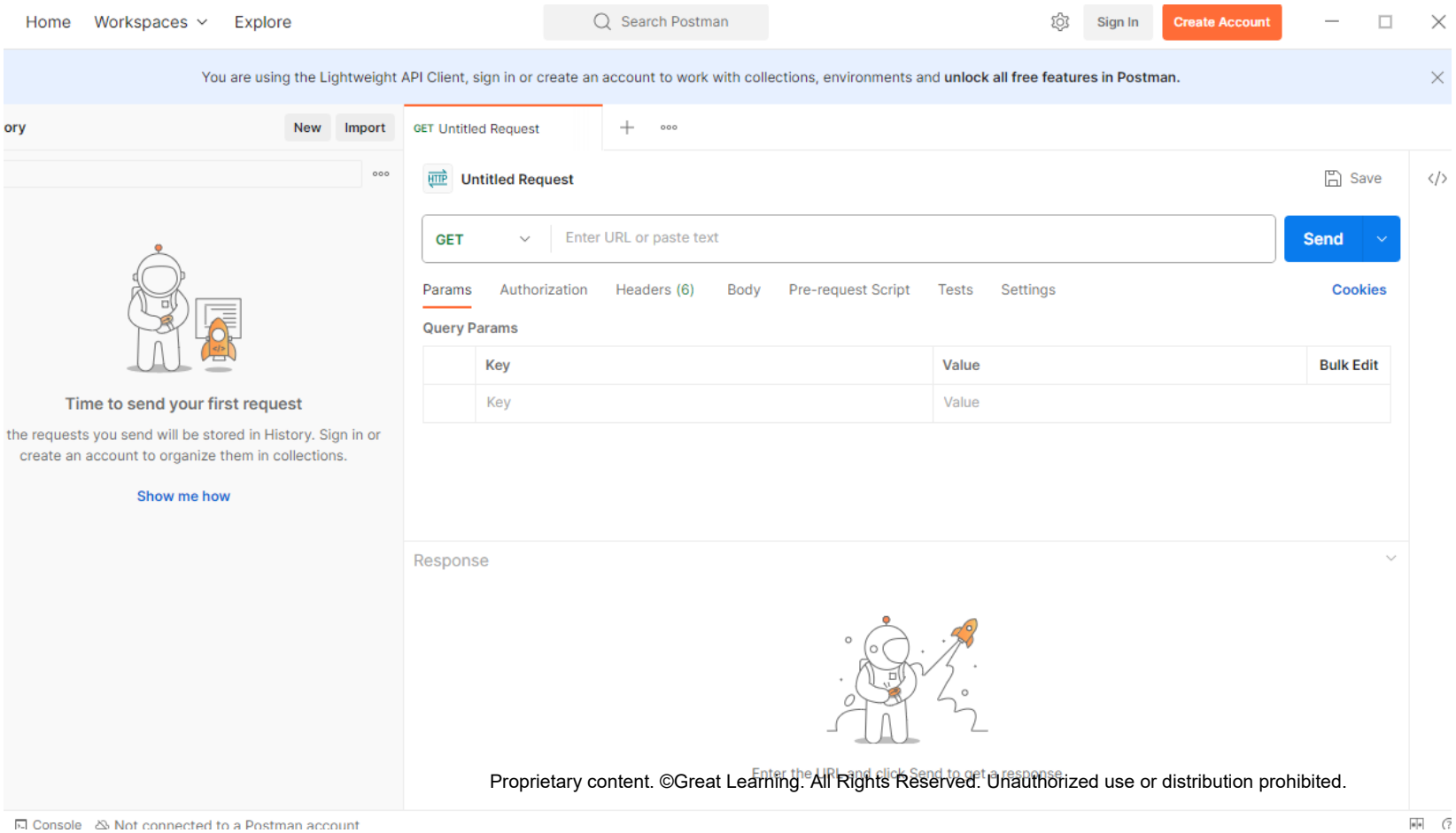
Installing Postman API

- **Step-5:** Create your account with all the required details, or you can also signup with Google, as shown in the image.



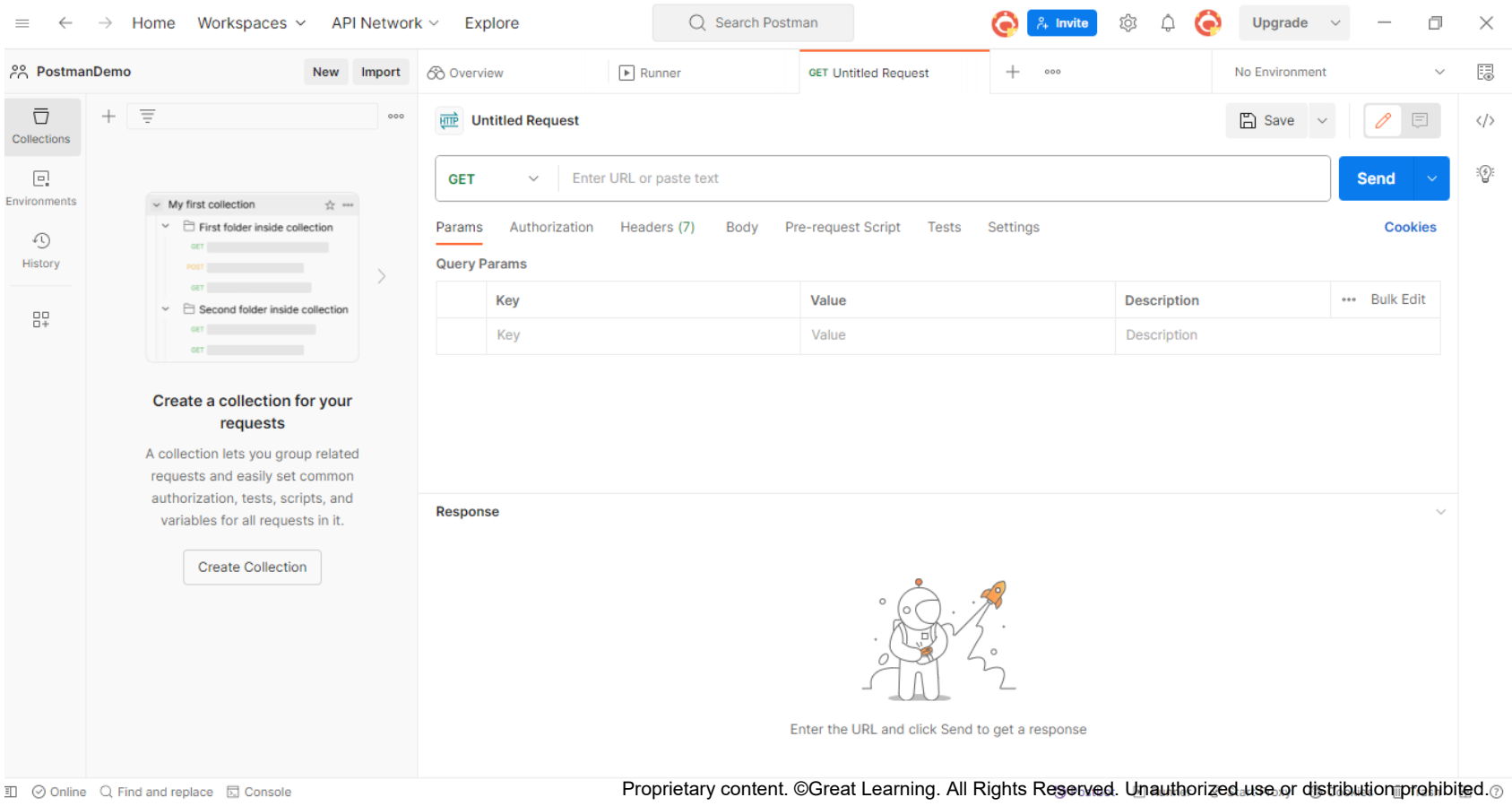
Installing Postman API

- **Step-6:** You will see the following page, and then you are ready to use Postman.



Postman to Execute APIs

- Let’s explore the step-by-step process on How to use Postman and the various features of the Postman tool.



Postman to Execute APIs

- **New** – It is used to create a new request, collection, or environment.
- **Import** – It is used to import a collection or environment. There are options such as import from file, folder, link, or paste raw text.
- **Runner** – Automation tests can be executed through the Collection Runner.
- **Open New** – Open a new tab, Postman Window, or Runner Window by clicking this button.
- **My Workspace** – It is used to create a new workspace individually or as a team.
- **Invite** – It collaborates on a workspace by inviting team members.

Postman to Execute APIs

- **History** – Past requests that you have sent will be displayed in History. This makes it easy to track actions that you have done.
- **Collections** – It organizes your test suite by creating collections. Each collection may have subfolders and multiple requests. A request or folder can also be duplicated as well.
- **Request tab** – It displays the title of the request you are working on. By default, “Untitled Request” is displayed for requests without titles.
- **HTTP Request** – Clicking this would display a dropdown list of different requests such as GET, POST, COPY, DELETE, etc. In Postman API testing, the most commonly used requests are GET and POST.
- **Request URL** – Also known as an endpoint, this is where you will identify the link to where the API will communicate with.
- **Save** – If there are changes to a request, clicking save is a must so that new changes will not be lost or overwritten.

Postman to Execute APIs

- **Params** – This is where you will write parameters needed for a request such as key values.
- **Authorization** – In order to access APIs, proper authorization is needed. It may be in the form of a username and password, etc.
- **Headers** – You can set headers such as content type, JSON depending on the needs of the organization.
- **Body** – This is where one can customize details in a request commonly used in POST requests.
- **Pre-request Script** – These are scripts that will be executed before the request. It is used to ensure that tests will be run in the correct environment.
- **Tests** – These are scripts executed during the request. It is important to have tests as it sets up checkpoints to verify if response status is ok, retrieved data is as expected and other tests.

Working with GET Requests

- GET requests are used to retrieve information from the given URL.
- The below URL will be used as example for this Postman tutorial
  - https://jsonplaceholder.typicode.com/users

Working with GET Requests

- In the workspace
  - Set your HTTP request to GET.
  - In the request URL field, input link
  - Click Send
  - You will see 200 OK Message
  - There should be 10 user results in the body which indicates that your test has run successfully.

Working with POST Requests

- POST requests are different from GET requests as there is data manipulation with the user adding data to the endpoint.
- In the workspace
  - Set your HTTP request to POST.
  - Input the same link in request url: https://jsonplaceholder.typicode.com/users
  - Switch to the Body tab
  - Click raw
  - Select JSON
  - Add a new data
  - Click Send
  - Status: 201 Created should be displayed
  - Posted data are showing up in the body

Parameterize Requests

- Data parameterization is a key feature of Postman.
- It allows for the use of variables instead of hardcoding data into requests.
- Data for these variables can come from external data files or environment variables.
- This technique helps to avoid repetitive tests and is beneficial for automated testing.
- Parameters in Postman are defined using double curly brackets, for example: {{sampledemo}}.

Parameterize Requests

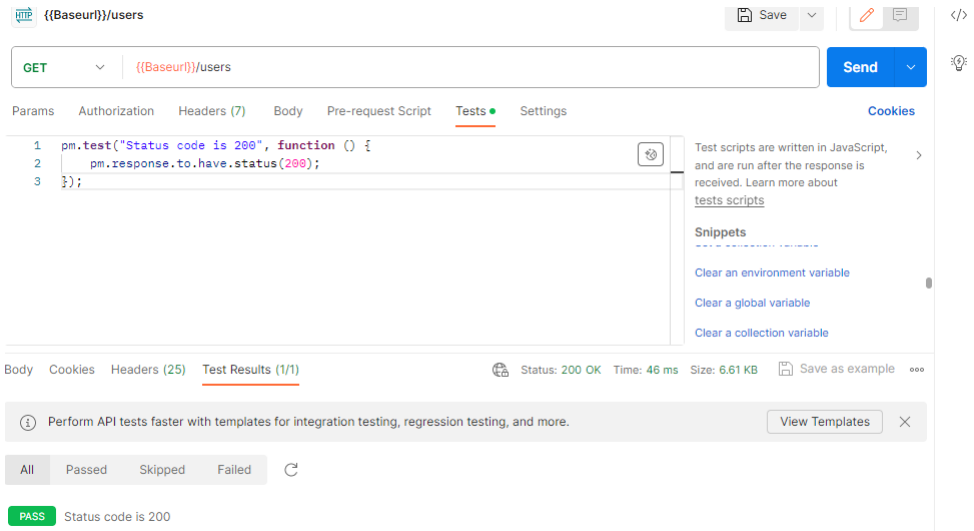
- Now let’s create a parameterized GET request.
  - Set your HTTP request to GET
  - Input this link: https://jsonplaceholder.typicode.com/users. Replace the first part of the link with a parameter such as {{Baseurl}}. Request url should now be {{Baseurl}}/users.
  - Click send.
  - Click the eye icon
  - Click edit to set the variable to a global environment which can be used in all collections.
  - set the name to the url which is https://jsonplaceholder.typicode.com
  - click Save.
  - Go back to your Get request then click send. There should now be results for your request.

Create Postman Tests

- Postman Tests utilize JavaScript for validation of request outcomes.
- They check for conditions like success or failure statuses and expected results comparison.
- The syntax generally begins with pm.test.
- Functionally similar to 'asserts' or 'verify' commands in other testing tools.

Create Postman Tests

- Let’s do some basic API testing using Postman for our parameterized requests:
  - Go to your GET user request.
  - Switch to the tests tab. On the right side are snippet codes.
  - From the snippets section, click on “Status code: Code is 200”.
  - The pane is auto-populated.
  - Now click Send. The test result should now be displayed.



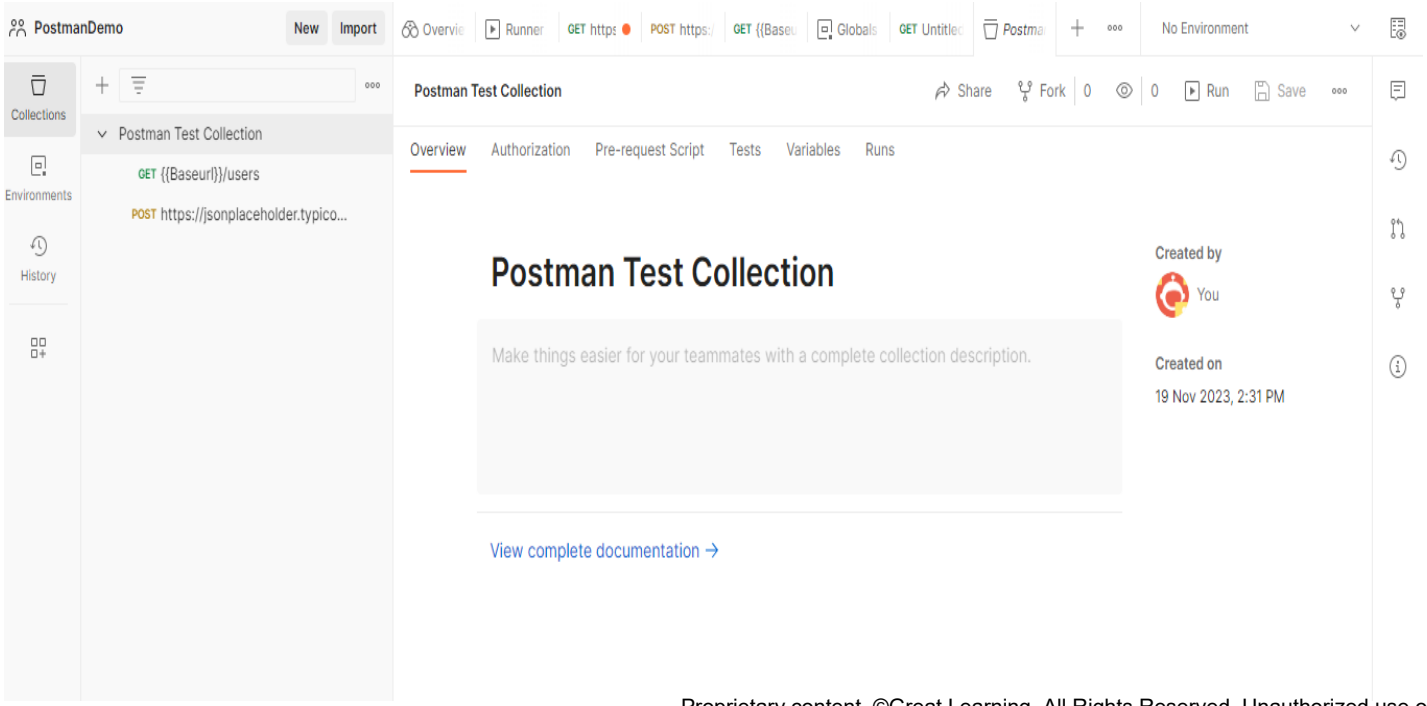
Create Postman Tests

- Let’s add another test. This time we will compare the expected result to the actual result.
- From the snippets section, click on “Response body:JSON value check” and check the test result.

```
pm.test("Check if user with id1 is Leanne Graham", function () {
    var jsonData = pm.response.json();
    pm.expect(jsonData[0].name).to.eql("Leanne Graham");
});
```

Creating Collections

- Collections are central for organizing test suites in Postman.
- They support import and export functions.
- This feature facilitates easy sharing of collections within a team.
- Lets see how collections are saved.

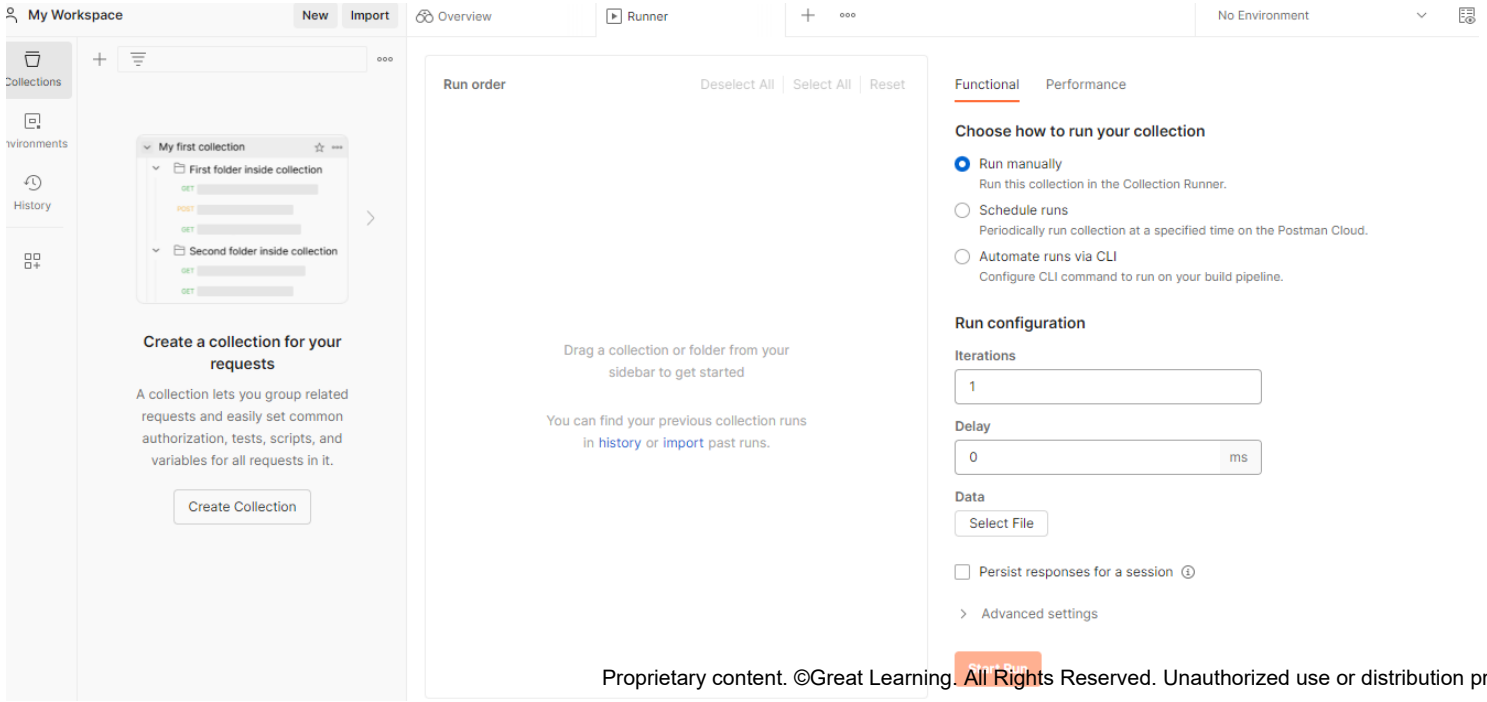


Run Collections using Collection Runner

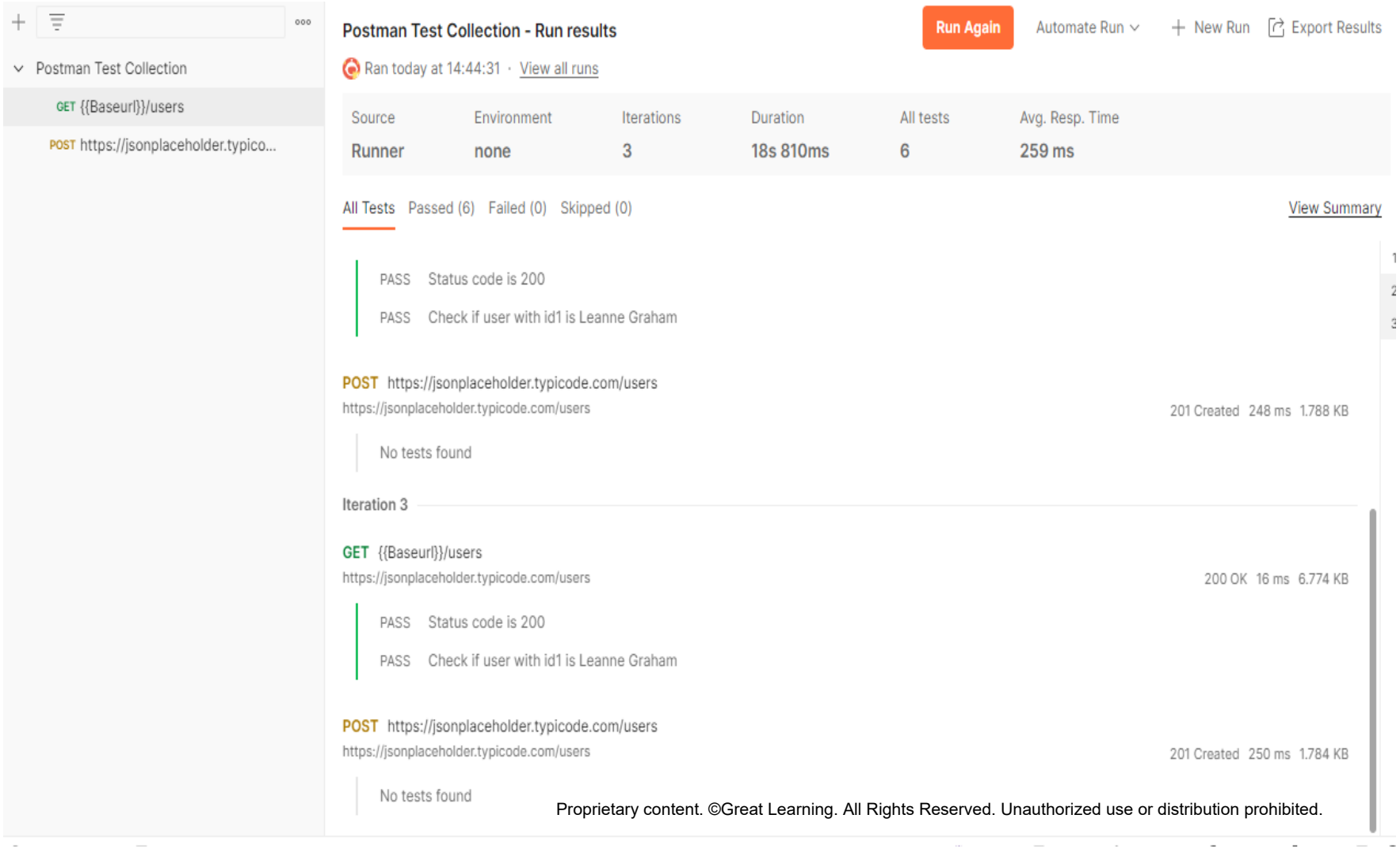
- Collection Runner is a Postman feature for executing multiple requests.
- It allows for running entire collections or selected requests in a sequence.
- Users can set up specific iterations, delays, and log results.
- It's useful for automated testing and debugging workflows.

Run Collections using Collection Runner

- Let’s begin by executing the collection in Collection Runner.
  - Click on the Runner button
  - Choose Postman test collection- Set iterations as 3
  - Set delay as 2500 ms
  - Click on Run button



Run Collections using Collection Runner



Summary

- Application Programming Interface is software that acts as an intermediary for two apps to communicate with each other.
- Postman is a software testing tool used for API Testing. API(Application Programming Interface) enables software applications to interact with each other.
- Postman is used for API Testing due to the following reasons: Availability, Making use of collections, Collaboration, Test Creation, Continuous Integration, Automation Testing and Debugging
- The features of Postman include new, import, runner, collaboration, collections etc.
- Get requests are used to retrieve information from the given URL.
- Post requests are different from Get requests as there is data manipulation with the user adding data to the endpoint.
- Parameterization helps to avoid repetition of the same tests and iterations can be used for automation testing.
- Collection Runner is a Postman feature for executing multiple requests.