Automation Testing

Variables

A Day in the Life of an Automation Testing Engineer

John is assigned the task of making the API testing dynamic.

As part of his assignment, he will create variables and use them to provide different values in the API requests.

After completing this session, John will be able to create, refer, get, and set variables in Postman.

Learning Objectives

By the end of this lesson, you will be able to:

- Identify variables in Postman

- Explain the variable scope in Postman

- List the steps to create and refer a variable

- Explain how to get and set a variable in Postman

Variables In Postman

Variables In Postman

In Postman, variables allow users to save and reuse values. Users can refer to a value stored as a variable throughout their Collections, Environments, requests, and test scripts.
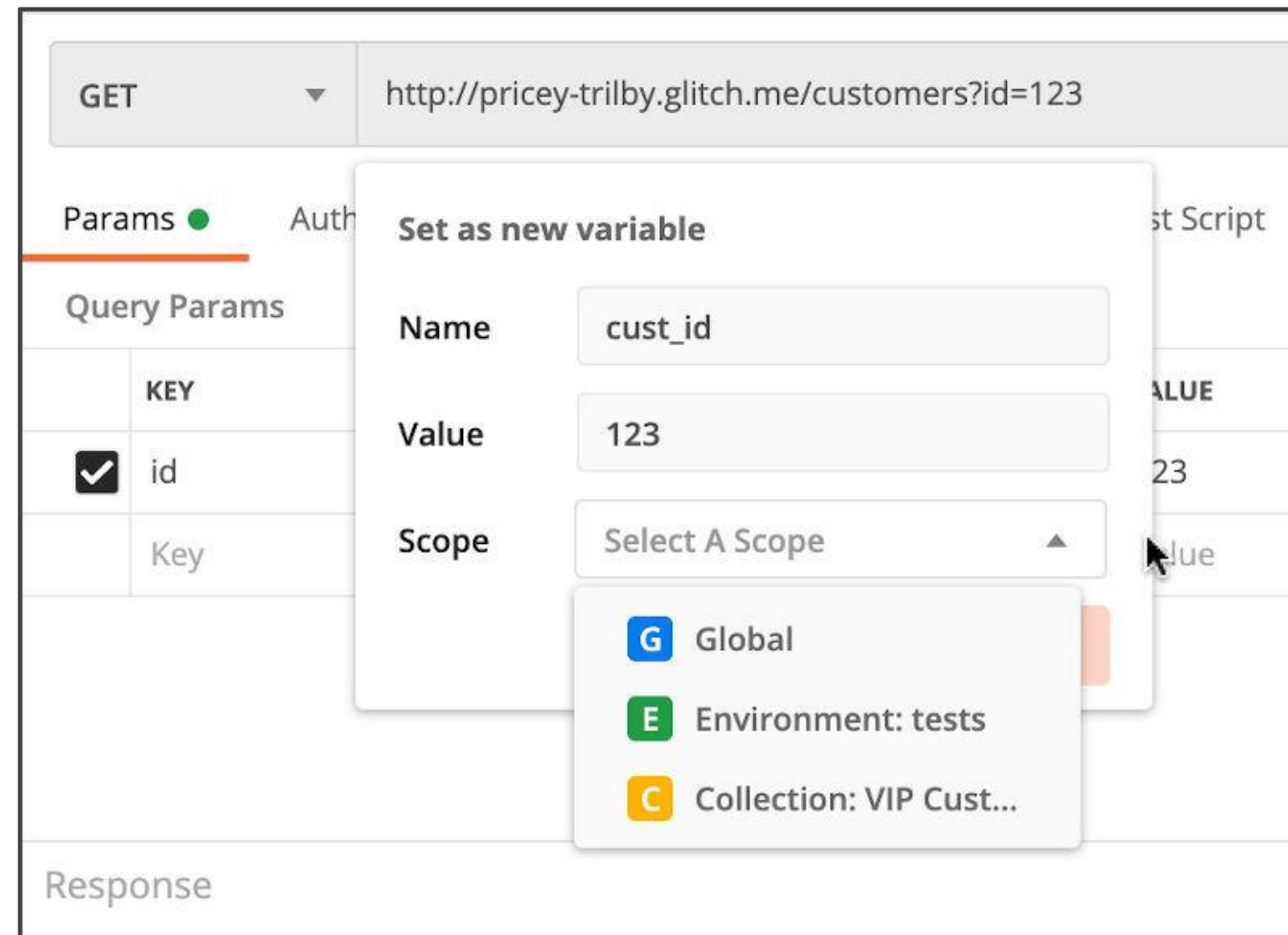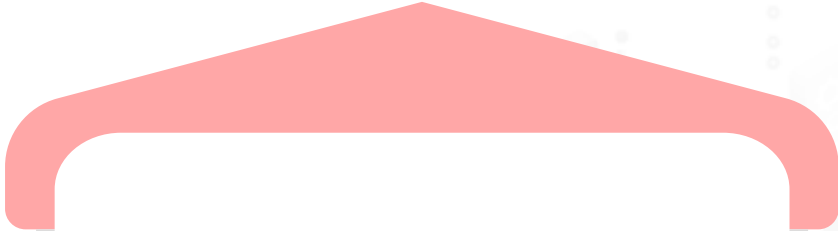
Benefits of Using Variables In Postman

A variable saves time by eliminating the need to duplicate requests.

It stores values used in requests, Environments, Collections, and test scripts.

It reduces errors and boosts the efficiency of Postman.

How to Create a Variable in Postman?

Types of Variables

The two types of variables in the global Environment that users can configure are:

Type	Description
Default	This type of variable is assigned by default and used as plain text with no additional properties.
Secret	This type of variable masks the values and prevents undesirable exposure of sensitive data.

Variable Scope in Postman

Variable Scopes

Variables at various scopes in Postman allow users to customize their processing to a range of tasks related to development, testing, and collaboration. Scopes in Postman relate to the different contexts in which their requests run.
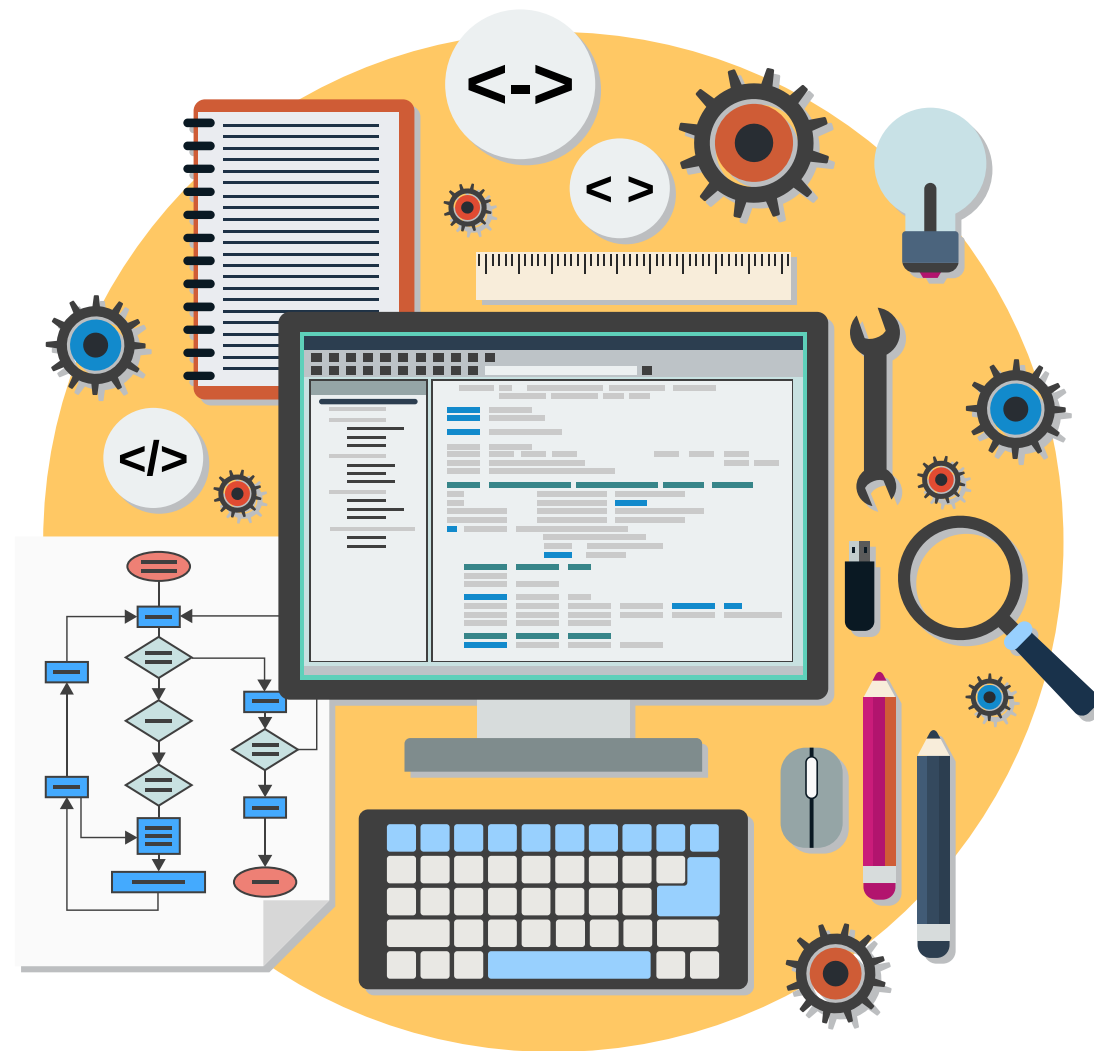
Variable Scopes

There are five types of variable scopes in Postman.

Environment variables

3

Collection variables

2

Data variables

4

Global variables

1

Local variables

5
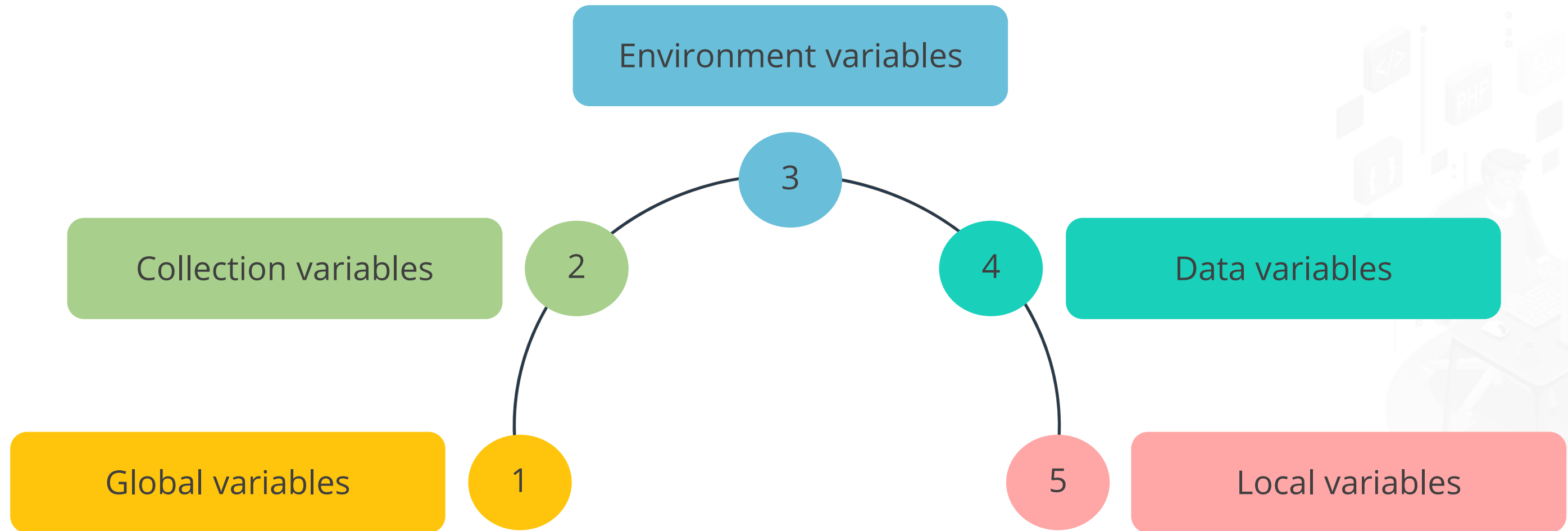
Variable Scopes: Global

An Overview

Users can retrieve data from different Collections, requests, test scripts, and settings using Global variables.

Features

A Global variable of a workplace is accessible from anywhere in the workspace. Global variables are ideal for testing and prototyping because they have the largest reach in Postman.

Variable Scopes: Collection

An Overview

These are available throughout a Collection's request and are not affected by the Environment.

Features

The Collection variables do not change depending on the Environment. These variables are the way to go if users are working in a single Environment.

Variable Scopes: Environment

An Overview

Users can scope their work to different contexts using Environment variables, such as local development versus testing or production. At any given time, only one Environment can be active.

Features

Collection variables in a single Environment can be more economical. However, Environments allow users to establish role-based access permissions.

Variable Scopes: Data

An Overview

These are defined by the external CSV and JSON files and used to define data sets that may be used by Newman or the Collection Runner when running Collections.

Features

The data variables values are temporary and do not remain beyond request or Collection runs.

Variable Scopes: Local

An Overview

These are temporary variables that the request scripts can access. Local variable values are only available for a single request, or Collections run and are gone once the run is over.

Features

Users can use local variables if they require a value to override all other variable scopes but don't want the value to persist after execution.

How to Refer a Variable in Postman?

Referring a Variable in Postman

If users have the same URL in more than one request, but the URL can change later, they can keep the URL in a variable base URL and reference it in their requests using {{base url}}.

If the URL changes, the users can update the variable value, which will be updated wherever the variable name has been used in the Collection.

The same rule applies for any request, provided repetitive data is used. When the requests are executed, whatever value is kept in the variable will be present wherever it has been referred to.

Postman will send the request to the https://postman-echo.com/get link if the base URL value is set to the https://postman-echo.com link and specified in the request URL using the base URL/get.

How to Get and Set a Variable in Postman?
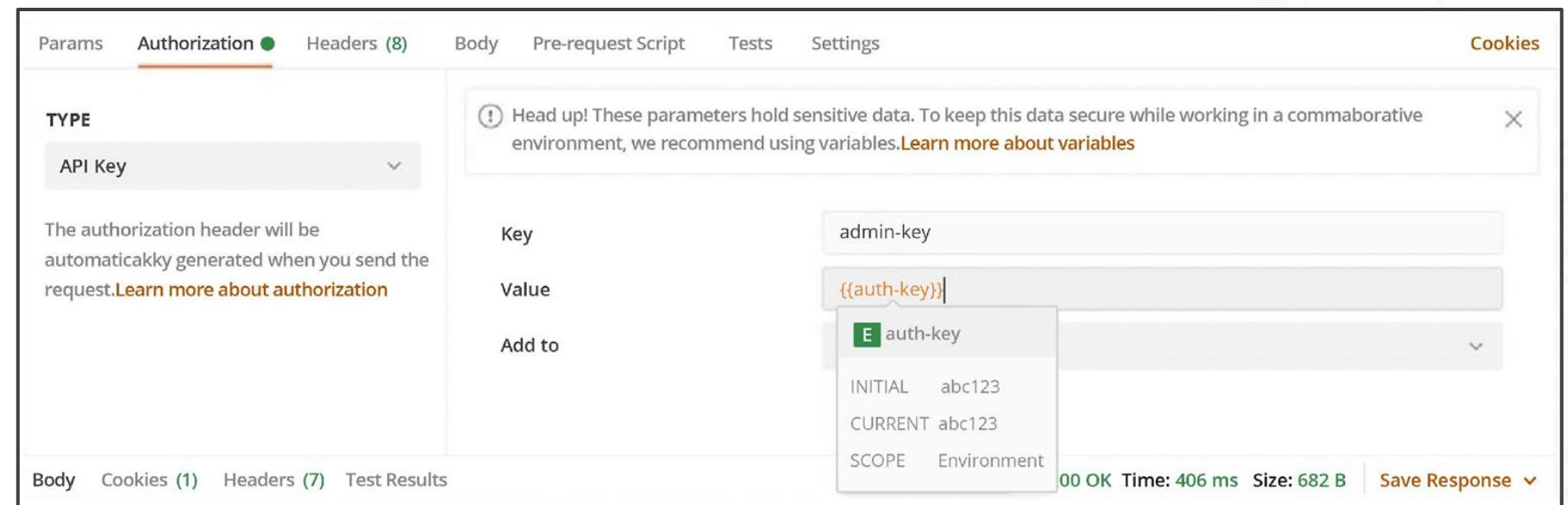
Get Variable in Postman

Get Variable:

Users can get the current value of a variable in their scripts by using the object representing the scope level and the .get method.
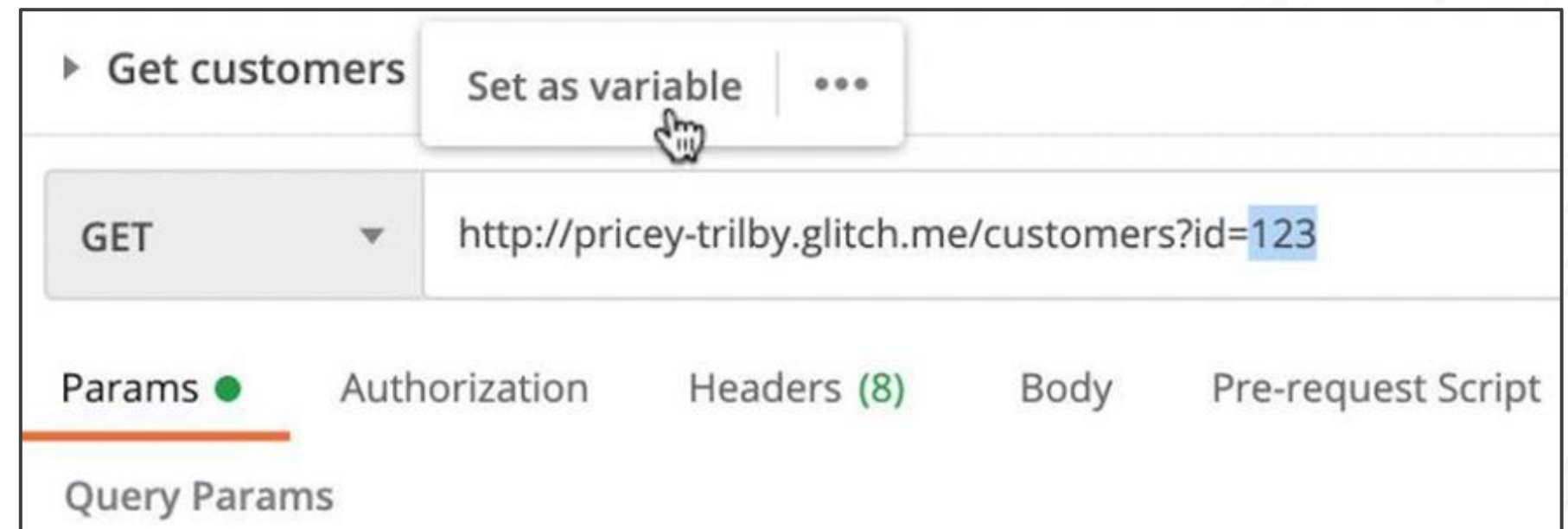
Set Variable in Postman

Set Variable:

Users can set the values for the existing variables to the values from a request's response body.

Params | Authorization ● | Headers (8) | Body | Pre-request Script | Tests | Settings | Cookies

TYPE

API Key ▼

The authorization header will be automatickky generated when you send the request.Learn more about authorization

⚠ Head up! These parameters hold sensitive data. To keep this data secure while working in a commaborative environment, we recommend using variables.Learn more about variables ✕

Key | admin-key

Value | {{auth-key}}|

Add to

E auth-key

INITIAL   abc123
CURRENT abc123
SCOPE   Environment

Body | Cookies (1) | Headers (7) | Test Results | 00 OK Time: 406 ms Size: 682 B | Save Response ⌄

Key Takeaways

- Variables allow users to store and reuse values in Postman.

- Variables support users to work efficiently, connect with their teammates, and organize dynamic workflows.

- Postman allows variables at different scopes, which enables users to streamline their processing to several tasks related to development, testing, and collaboration.

- There are five types of variable scopes: global, Collection, Environment, data, and local.

- Users can get and set Variables at different scopes.