**How to use Postman efficiently as a Software Developer? -** [Shrishty Chandra](https://sheril.medium.com/?source=post_page-----a4b3ab0ce3f6--------------------------------) Aug 12, 2021

Making API requests is something every developer does frequently in his/her development cycle. And if this can be automated even partially one can save a lot of time. And that is where postman comes to our rescue.

This article is something, I wish I had read before I started working with postman. I hope it will be useful for you.

**Introduction**

Let’s go through a quick introduction of postman terms and terminologies. Please use the image below and image in the subsections as a reference while you read.

Graphical user interface, text, application, email

Description automatically generated

**Collection**

We can group request belonging to a particular domain into a collection. Eg. You can have all API’s related to twitter in a single collection and all apis related to “The COVID Tracking Project” into second collection.

**Requests**

Requests reside inside a collection.

**Folder/Group**

You can create folders or groups based on the type of request. In the image below TweetLookup, UserLookup etc are folders

Graphical user interface, text, letter

Description automatically generated

**HTTP Verb**

You can select the http verb from the following dropdown

Graphical user interface, application

Description automatically generated

**URL**

Just beside the HTTP Verb, goes the complete URL, including the path variables and query params.

Graphical user interface, text, application

Description automatically generated

**Path Variables and Query Params**

Path variable starts with a : and everything after ? in the format key=value separated by & is query param. In the example below id is a path variable and tweet.fields is a query parameter.

Graphical user interface, text, application, email

Description automatically generated

In the image above, the column Key contains the param names and the column Value contains the values corresponding to the keys.

**Send Request**

To send the request you can press the send button or press cmd + enter

**Auth Tab**

Under Auth tab we have multiple options for request authentication. Eg. Bearer Token, Basic Auth, Oauth 2.0 etc.

Graphical user interface, text, application, email

Description automatically generated

The best part about ***Collection*** is that we can define the ***Authorization*** details and select ***inherit from parent*.**This helps in keeping auth related stuff in one place for all the requests. This saves us from repeating the same thing every time we make a request.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

**Header Tab**

All header params can be passed in headers tab. It’s similar to params tab were you can set key value pairs. Most commonly passed header param is Content-Type .

**Body**

In body tab you can enter the request body. It has multiple options like binary, graphql, raw. Most commonly used request body type is raw > json. The raw option also supports multiple other type like json, html, javascript, text.

**Pre-req tab**

In pre-req tab you can have scripts (in javascript) which can run before the request is run. These are really very powerful

**Tests tab**

In tests tab you can have scripts (in javascript) which can run after the request is run.

**Postman Variables**

Variables are my favourite! You can store and reuse the variables across requests which makes making api calls a lot simpler. Variables can be accessed inside headers, body, request\_url and params.

You can access variables using double curly braces with the variable name written inside it. Eg. {{user\_id}}

Variables can be declared at two levels.

1. **Collection Level**
2. **Global Level or Environment Level**

**Collection Level Variables**

This image shows how we can define variables in a collection

Graphical user interface

Description automatically generated with low confidence

Whatever is present in the Current Value column gets picked up while you make the request.

You can access the variables using the double curly brackets (**{{}})**. In the example below BASE\_URL is being accessed like this — {{BASE\_URL}}

Graphical user interface, text, application

Description automatically generated

**Global Level variable**

You can also call this environment variables. You can access global variables by clicking on Manage Environments button on**top right corner**of postman UI.

Graphical user interface, application

Description automatically generated with medium confidence

The eye button is used to access the current environment variables

Dropdown on left of eye lists all the environments that has be setup. Usually software developers has to deal with multiple environments like dev, sandbox, prod etc. In such scenarios we can setup 3 different environments each having its own credentials. **Whenever you have to switch environment you can simply change it from the drop down and you are done. This feature has saved my time a ton of times. It’s awesome.**

Graphical user interface, application

Description automatically generated

If variable with same name is declared both at collection level and global level, global variable is given more preference over collection variable.

**Keyboard Shortcuts**

As a developer I feel its very important to know the keyboard shortcuts of the apps you are using in your day to day development. It **really really** saves your time.

Here are my most used keyboard shortcuts for postman. Go to Postman > Preferences > Shortcut or by using cmd + / to learn about other keyboard shortcuts.

* cmd + \\ To toggle sidebar.
* cmd + n - To create a new collection/request/environment etc.
* cmd + w and cmd + option + w - To close or force close tabs
* shift + cmd + t - Open last closed tab
* cmd + L - Go to request url area
* cmd + enter - Send request
* cmd + e - Rename Item (request/group etc)
* cmd + d - Duplicate Item (request/ group etc)
* option + cmd + e - To open environments list
* cmd + / - Open shortcut help

**Conclusion**

I am listing below few of my favourite features of postman.

* Collection Level variables and global variables — They are life saver, they really help saving lot of time by not repeating ourselves again and again for each and every request
* The way postman handles path variables and query parameters is awesome. You can set a path variable using : and anything after ? is a query parameter and you can define it like key=value in postman.
* You can set the value of query param variable and path variable to a collection level variable or a global variable. And it will save you from so much editing. Try it.
* Postman environment is a life saver. I can not say anything more than that on this one. Just go and check it out.