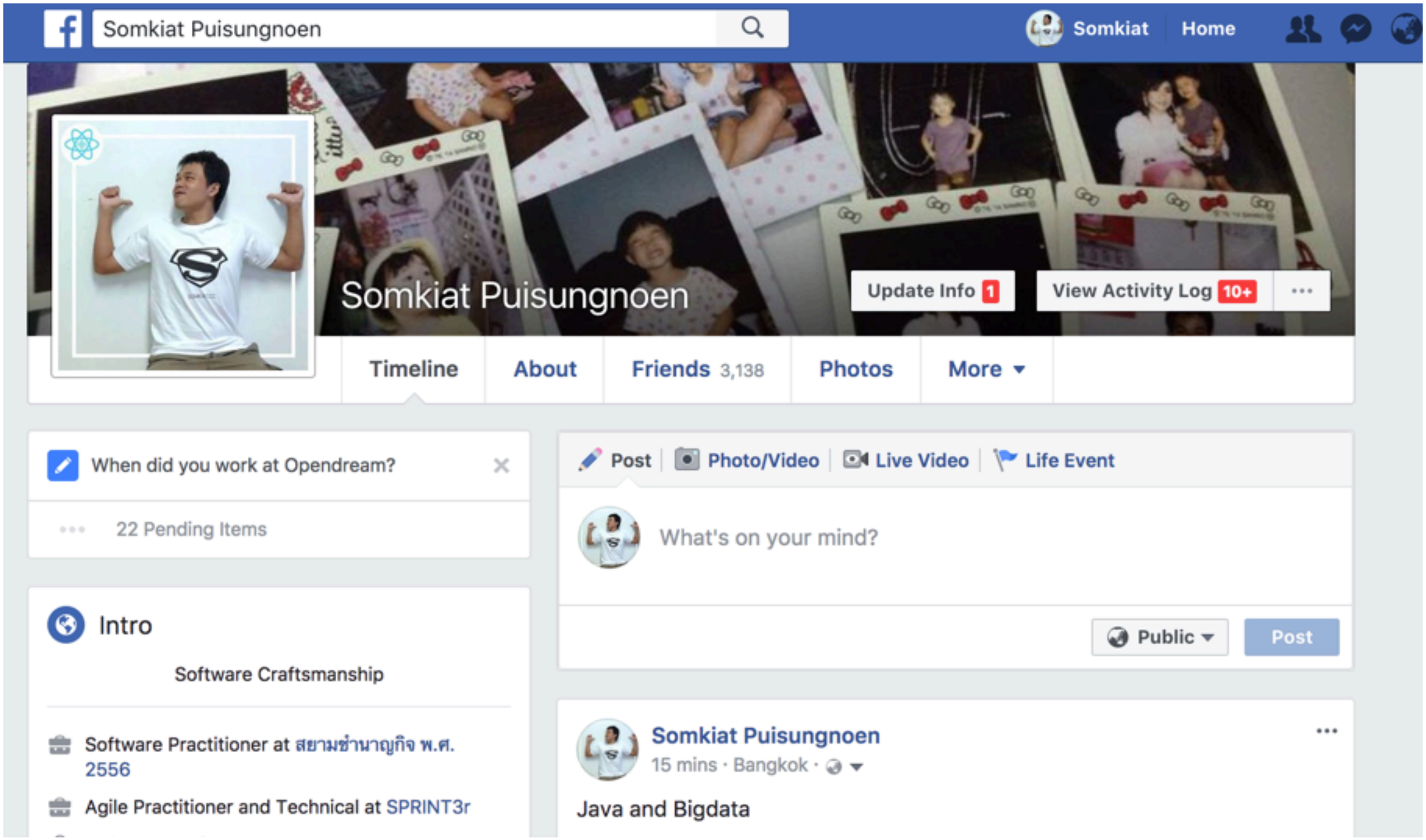
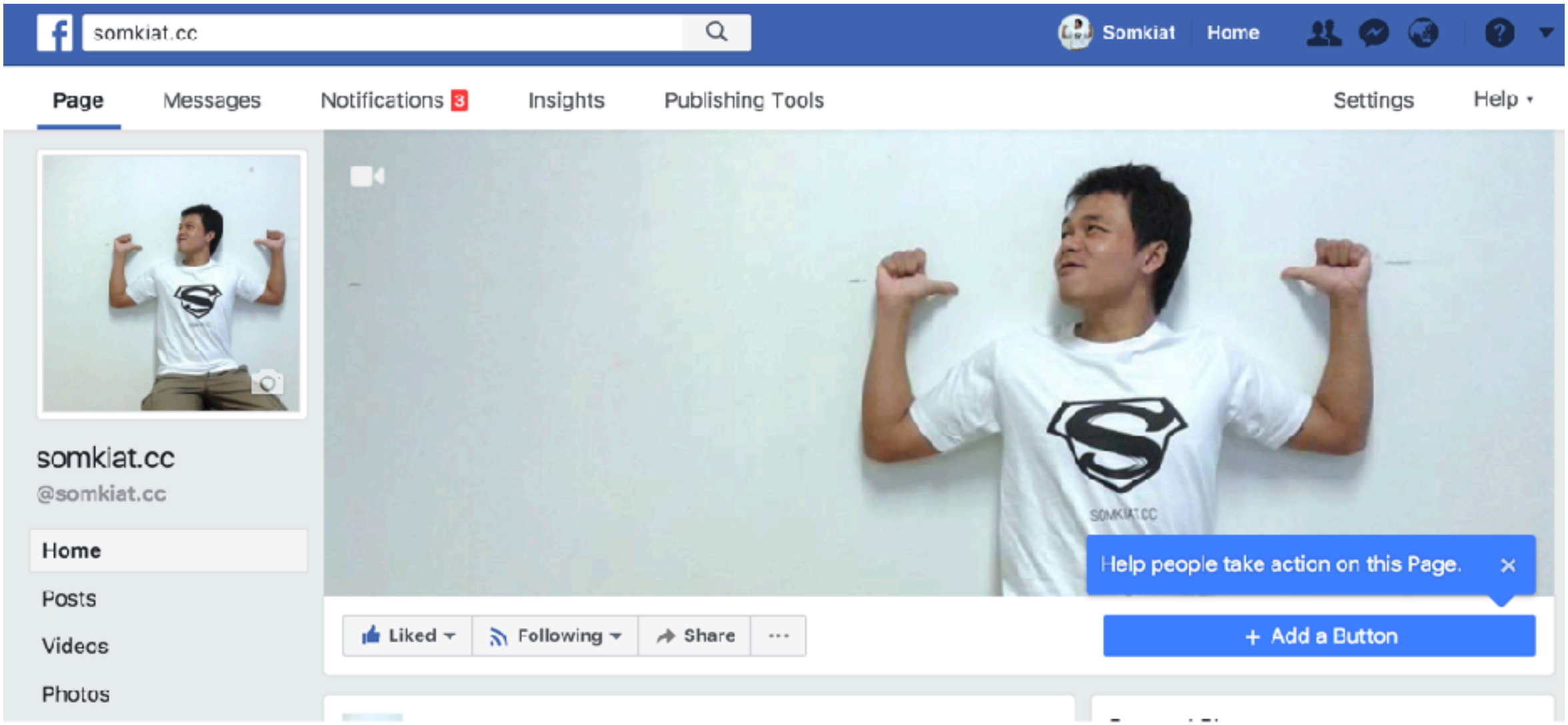
สยามชำนาญกิจ
SIAM CHAMNAN KIT

Postman in the Right Way





Postman in the right way

https://github.com/up1/workshop-postman

Testing

Fast feedback

High quality

Continuous Testing

API Design

API Testing

Postman

Basic of Postman

Postman

PUBLISH

Onboard developers to your API faster with Postman collections and documentation

DESIGN & MOCK

Design in Postman & use Postman's mock service

MONITOR

Create automated tests to monitor APIs for uptime, responsiveness, and correctness

DEBUG

Test APIs, examine responses, add tests and scripts

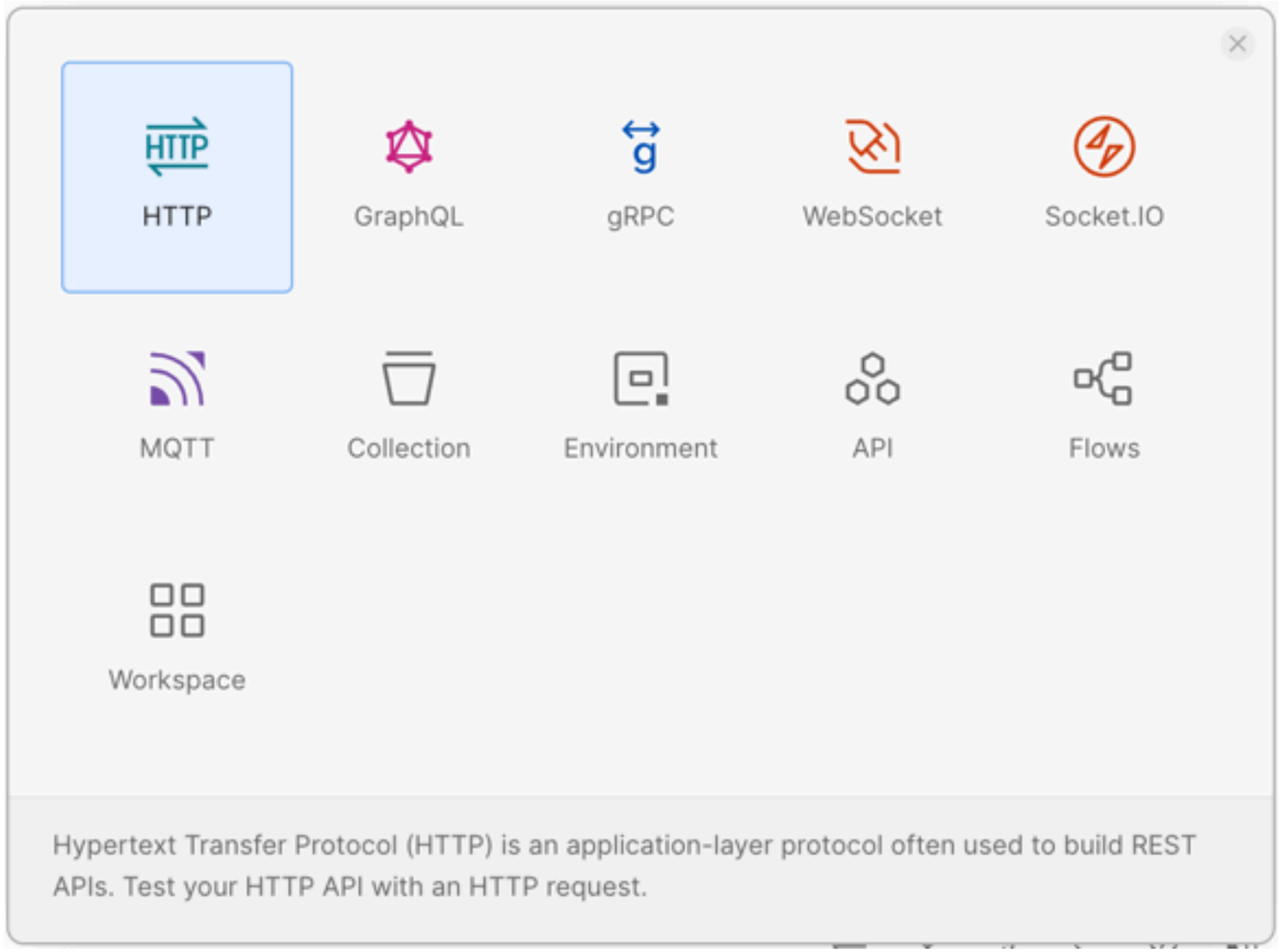
DOCUMENT

Create beautiful web-viewable documentation

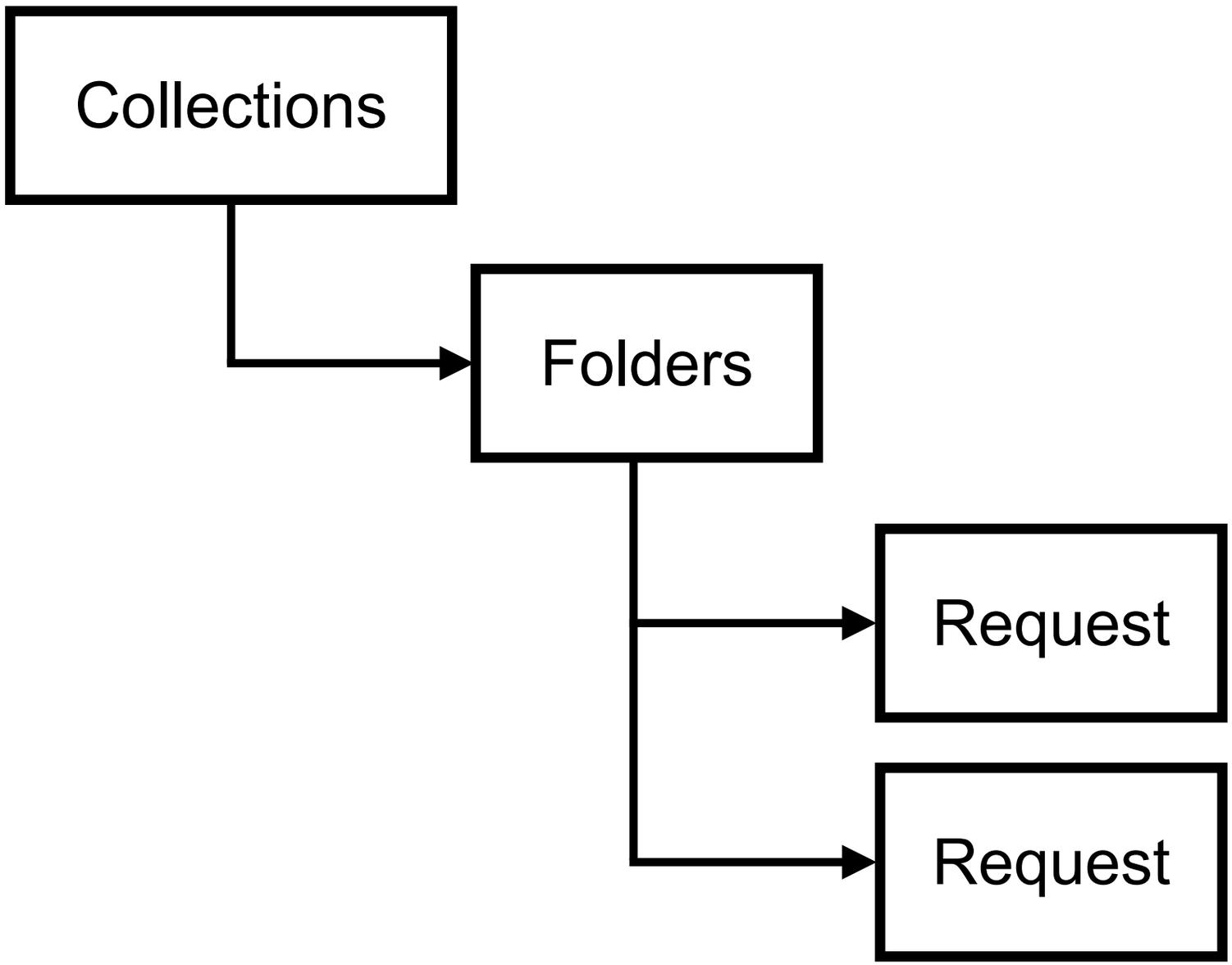
AUTOMATED TESTING

Run automated tests using the Postman collection runner

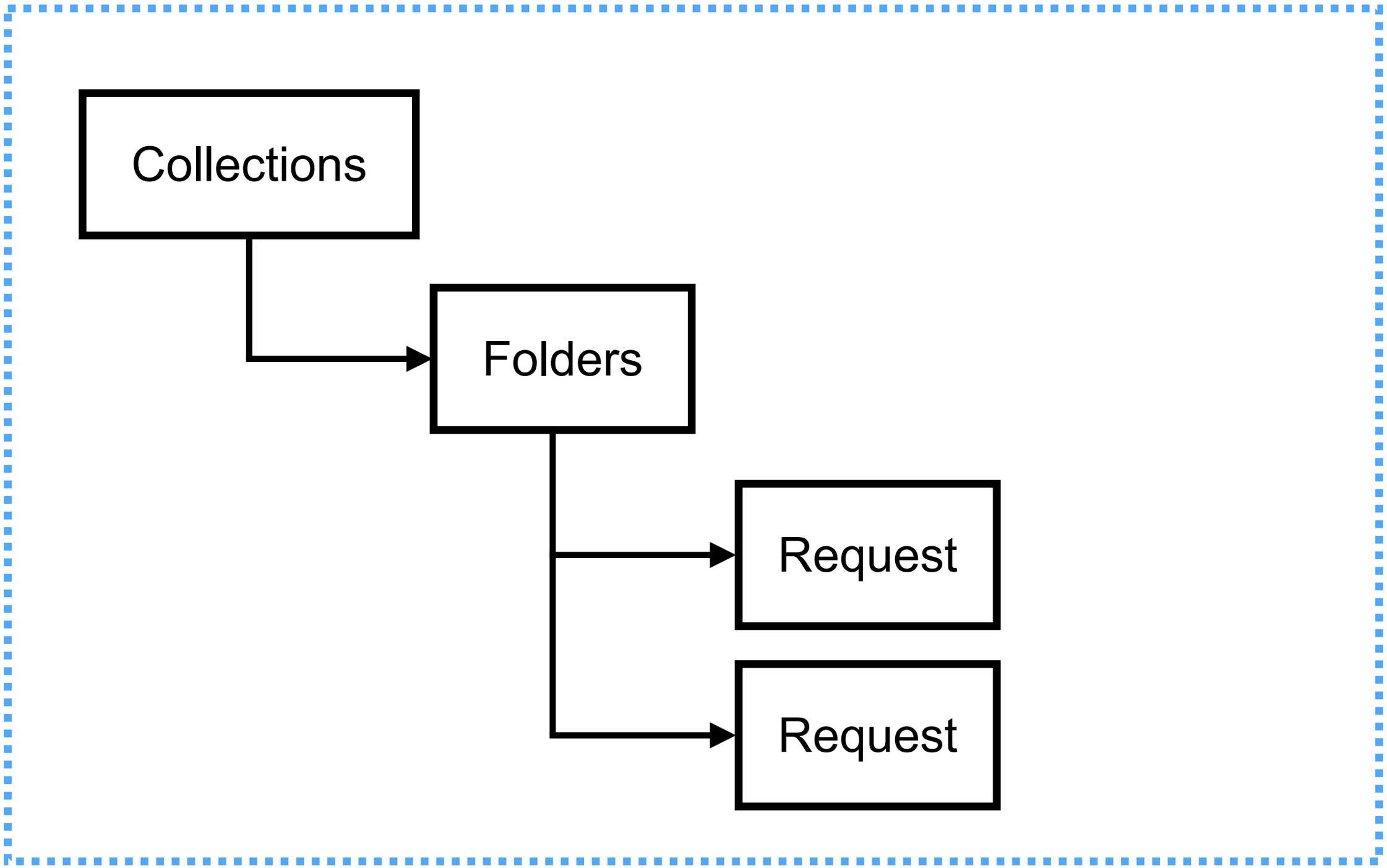
Postman



Postman structure



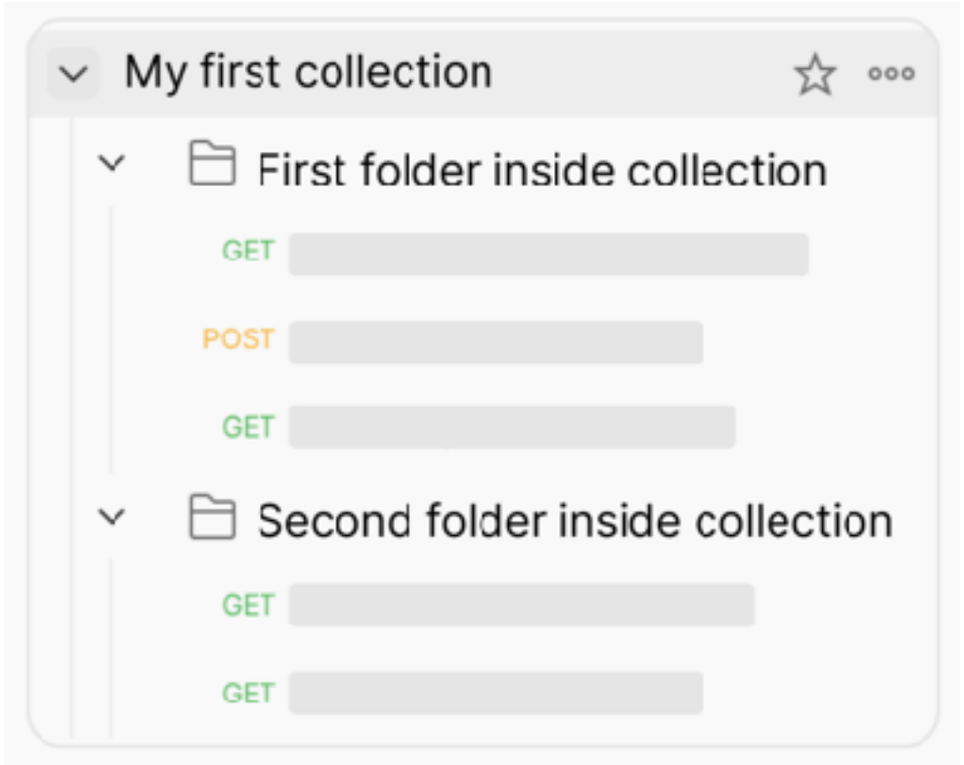
Workspace



Collections

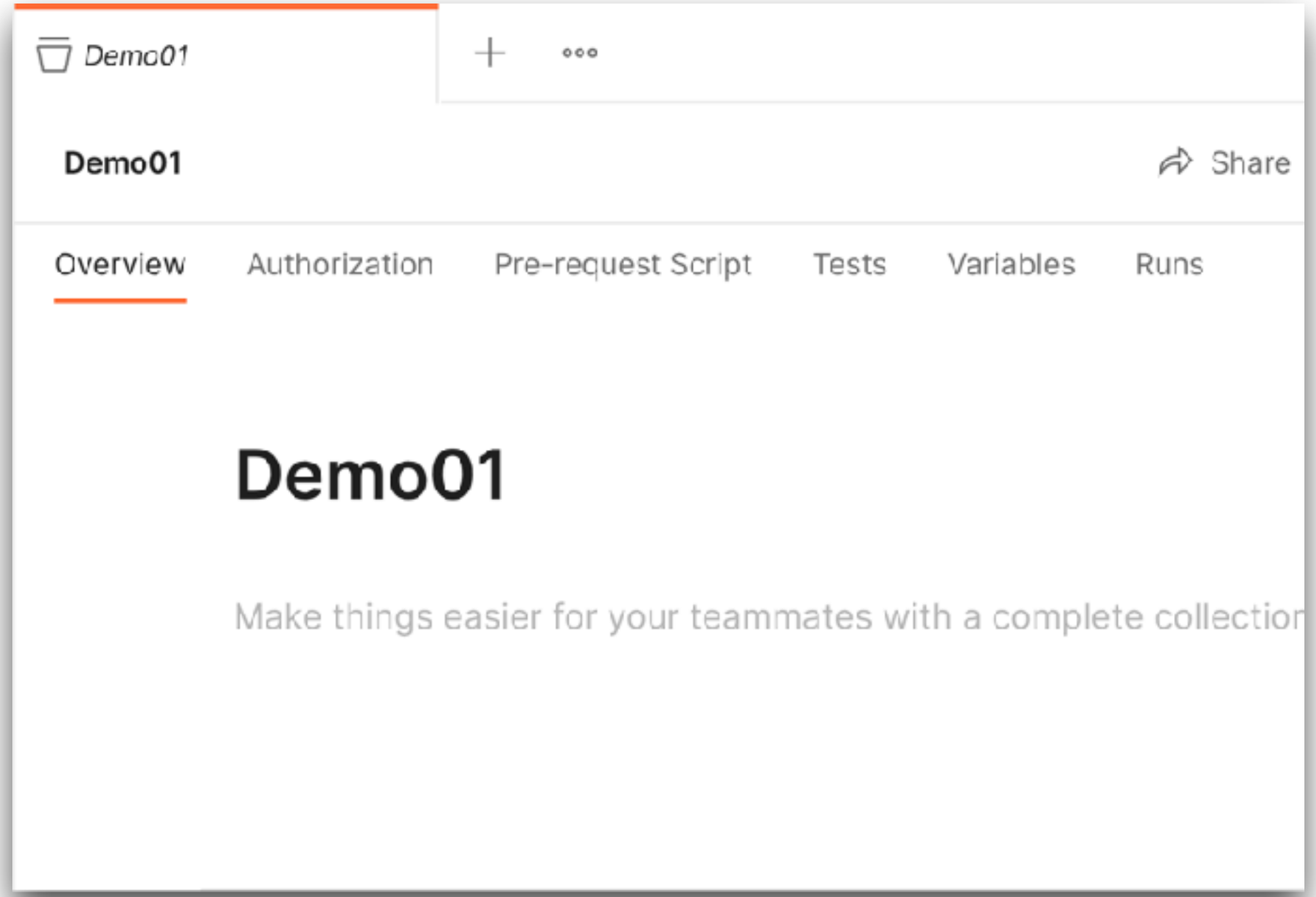
Postman collections

Grouping requests

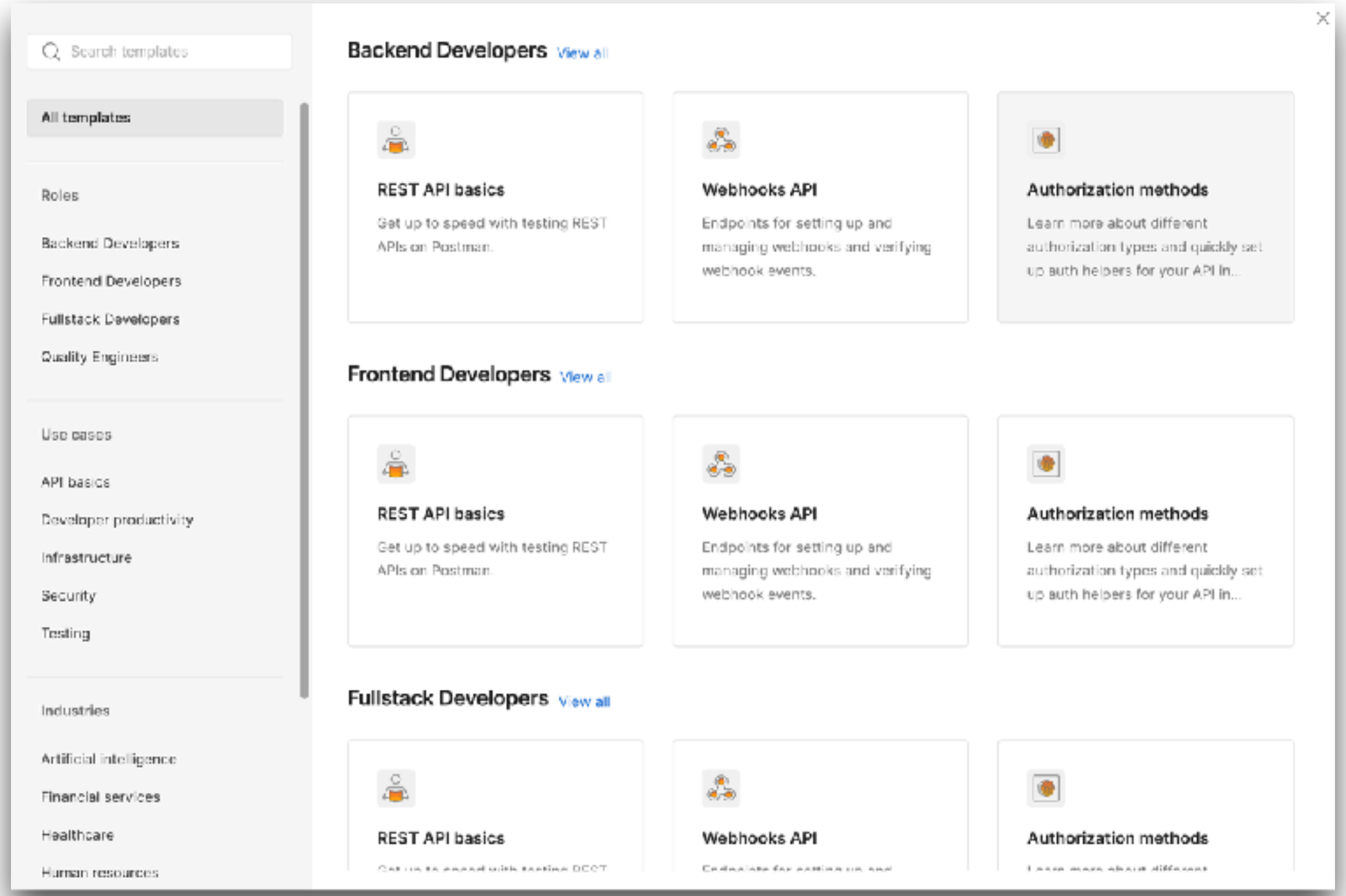


https://www.postman.com/collection/

Collections

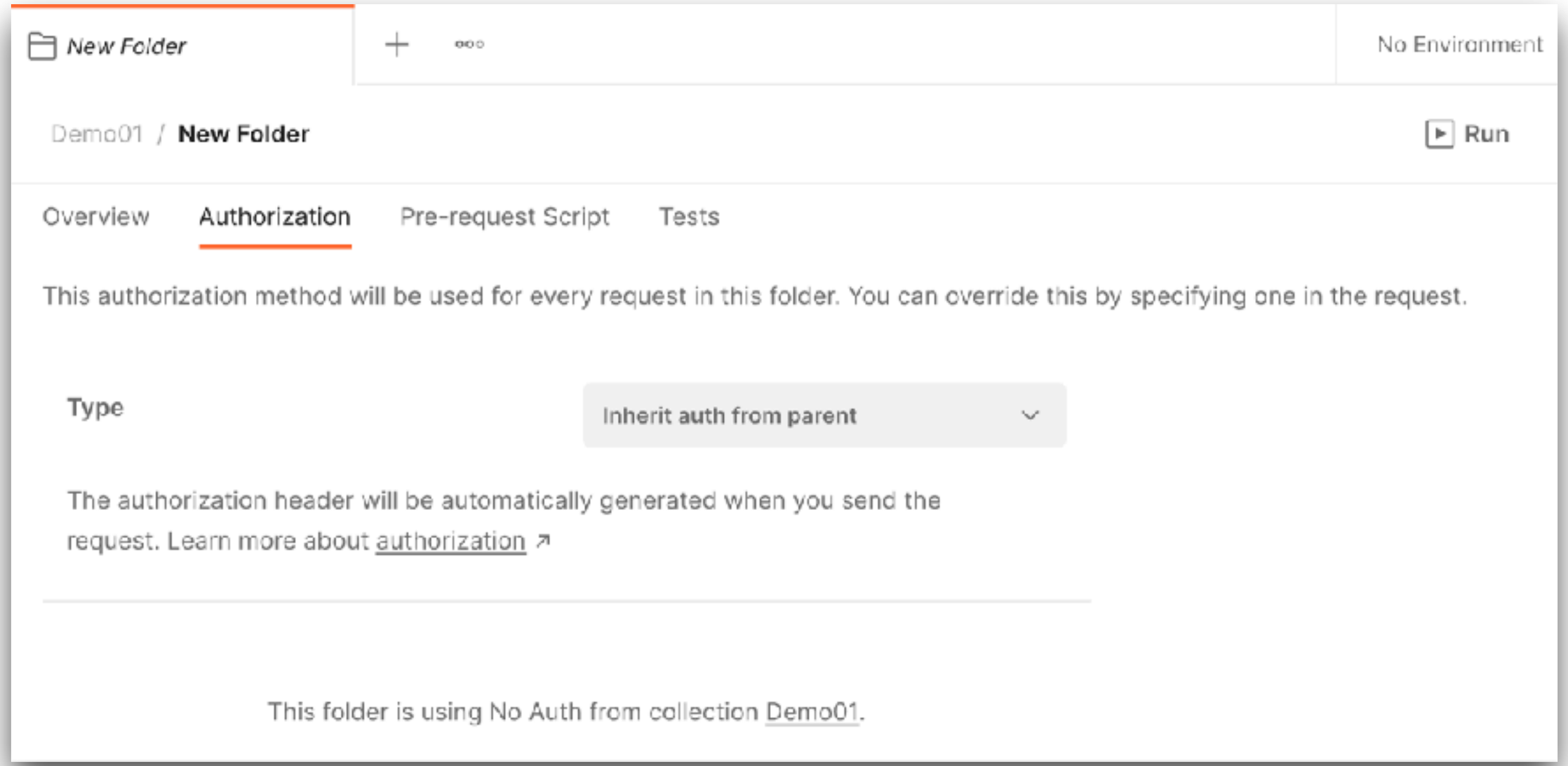


Collection Template !!



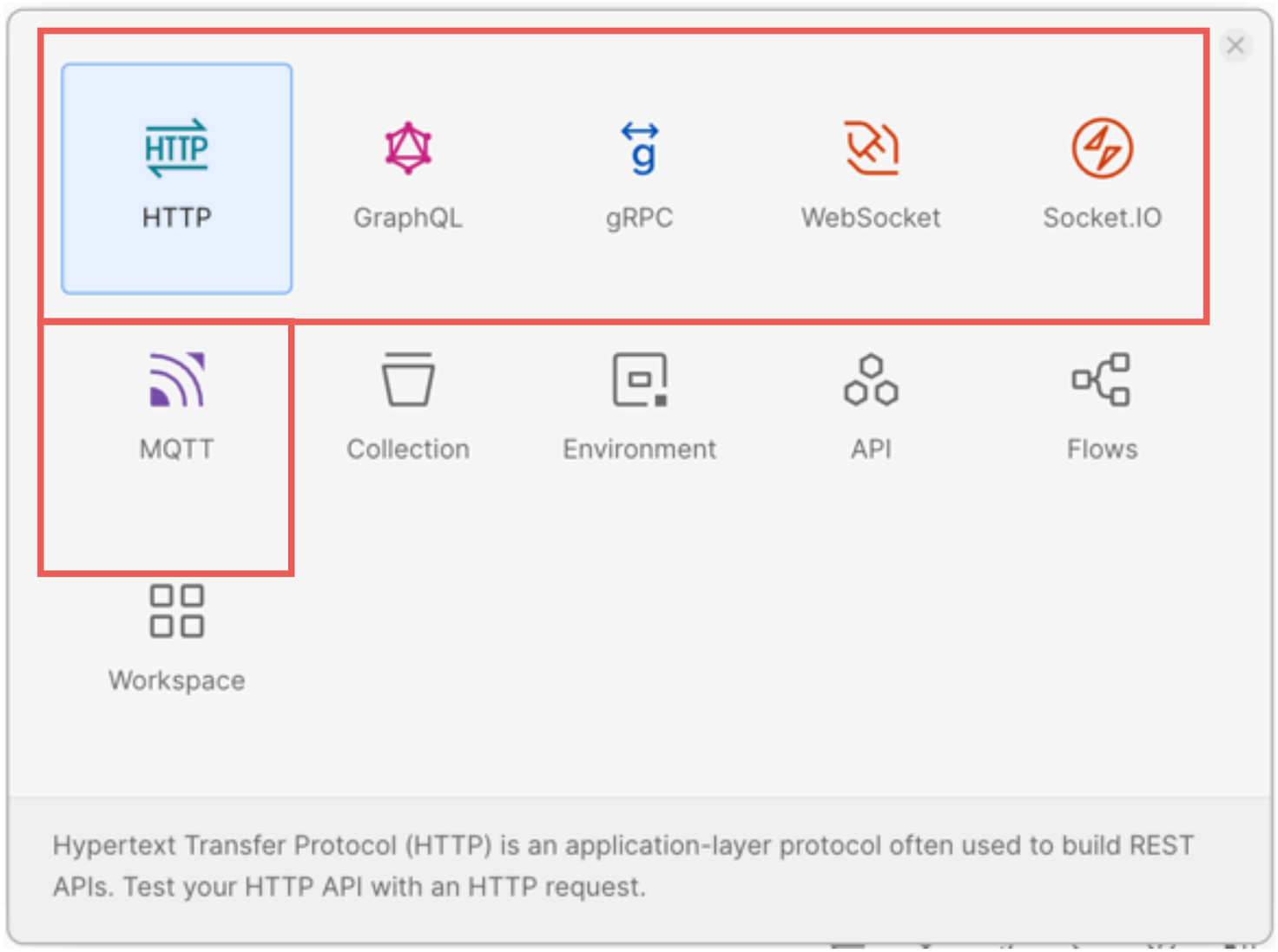
Folders

Folders

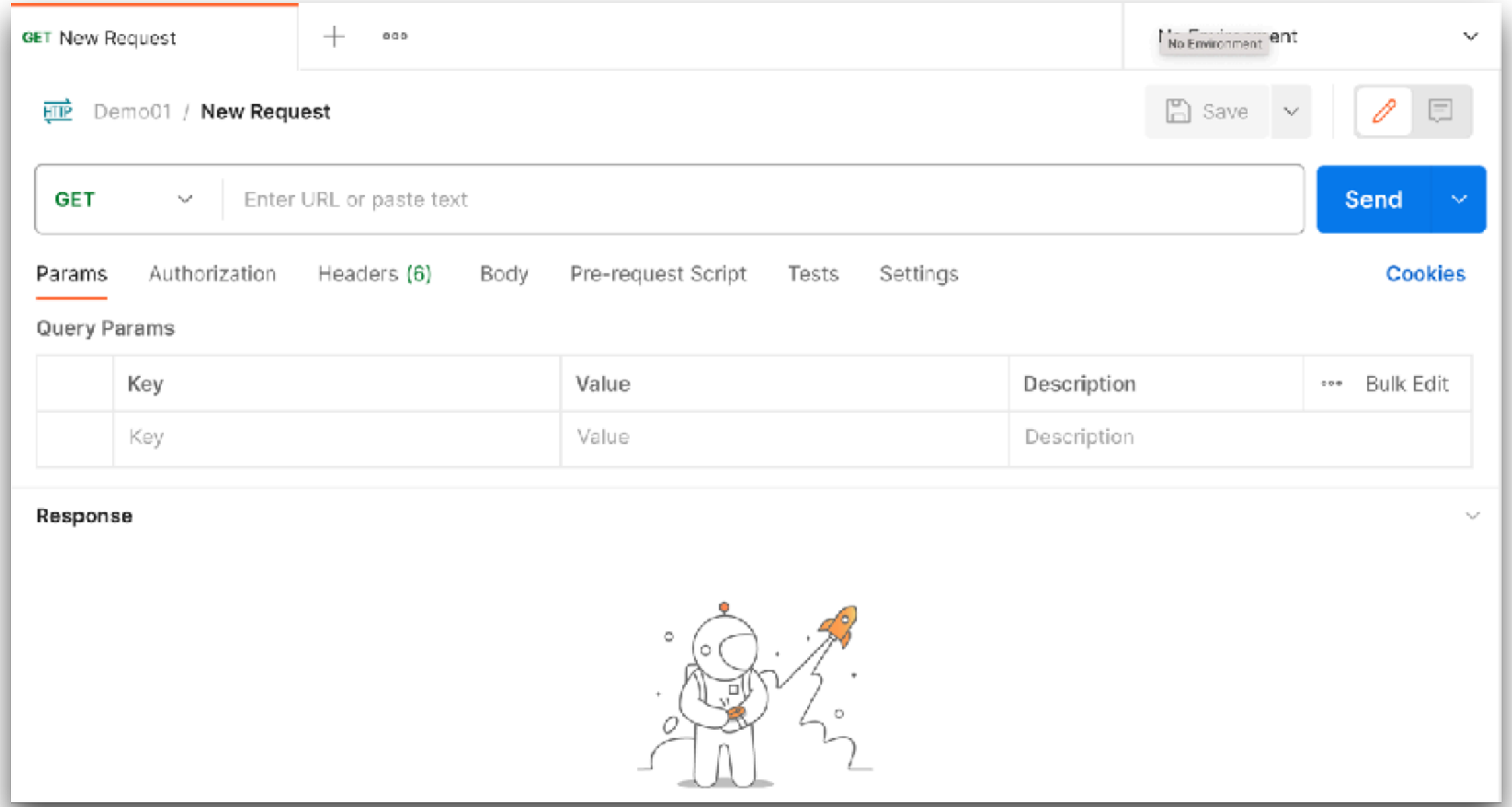


Requests

Request

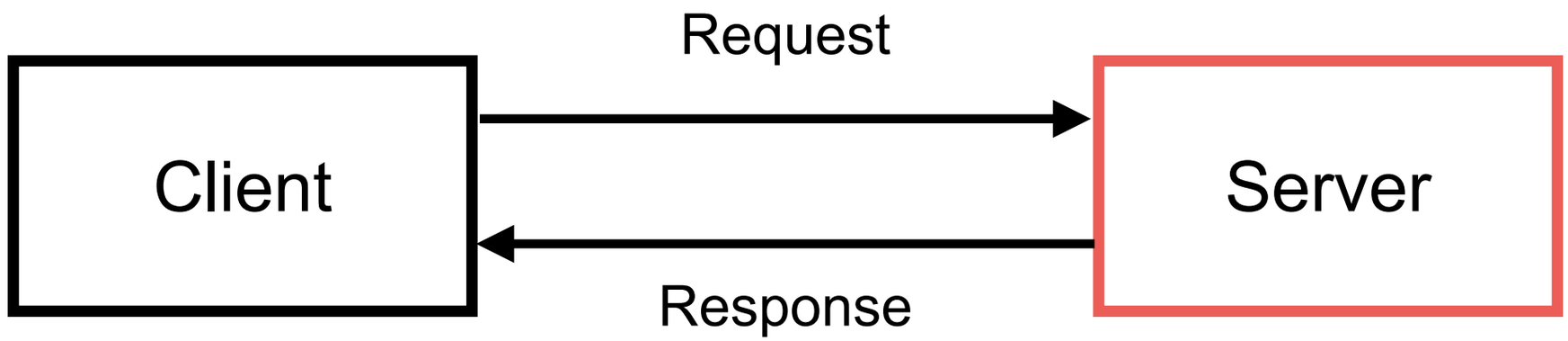


HTTP Request



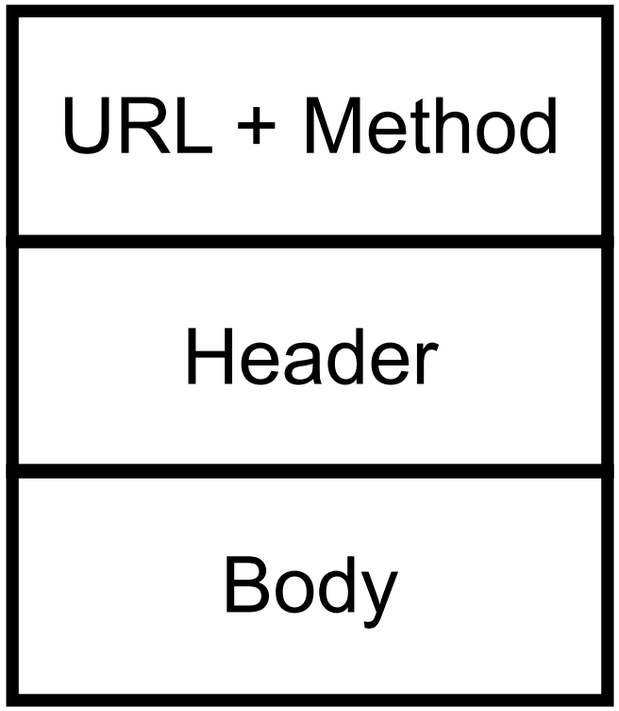
HTTP (Hyper Text Transfer Protocol)

Protocol to exchange information over network



HTTP Request

URL
HTTP methods
Request headers
Request body



HTTP Request

```
GET /api/data HTTP/1.1
Host: example.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
Accept: application/json
Accept-Language: en-US,en;q=0.5
Authorization: Token abc123
Cache-Control: no-cache
Connection: keep-alive
Referer: https://www.google.com/
Pragma: no-cache
```

URL in HTTP Request

URL (Uniform Resource Locator)

URL

https://www.youtube.com/watch?v=9JunaO2hGK0

URL in HTTP Request

URL (Uniform Resource Locator)

URL

https://www.youtube.com/watch?v=9JunaO2hGK0

URI

https://www.youtube.com/watch?v=9JunaO2hGK0

URL in HTTP Request

URL (Uniform Resource Locator)

URL

https://www.youtube.com/watch?v=9JunaO2hGK0

URI

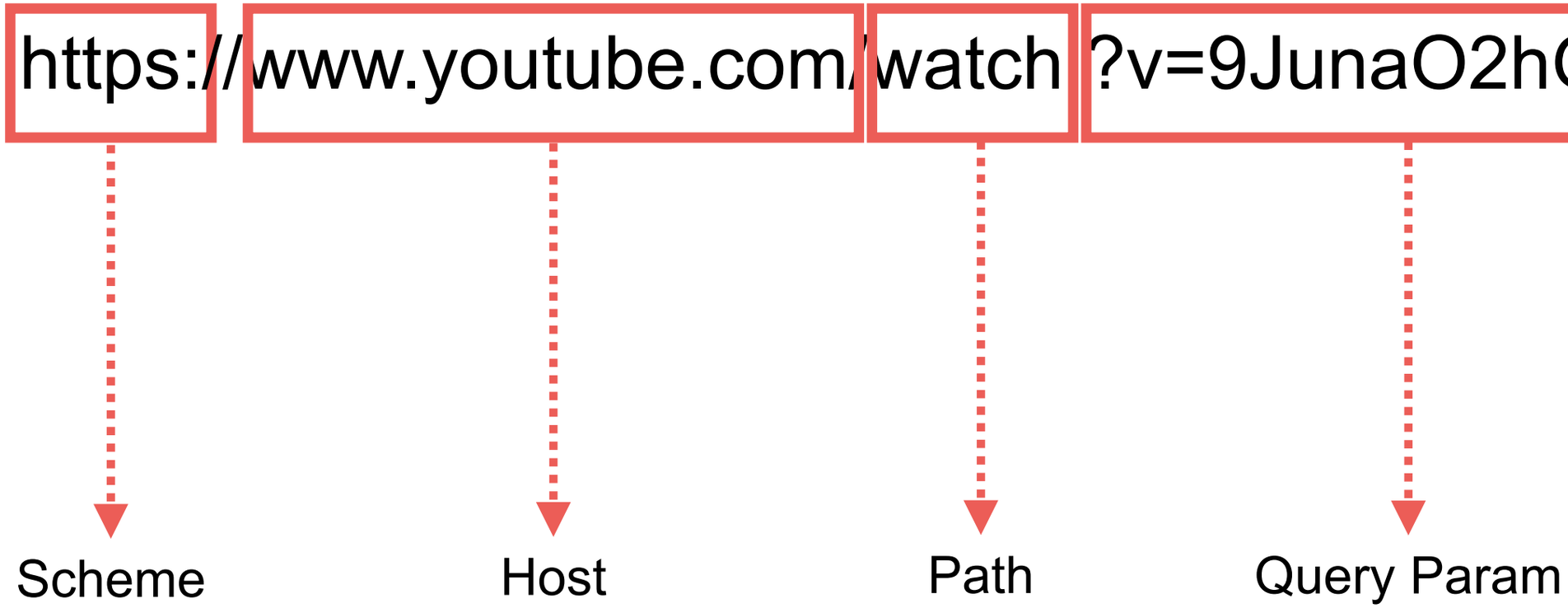
https://www.youtube.com/watch?v=9JunaO2hGK0

URN

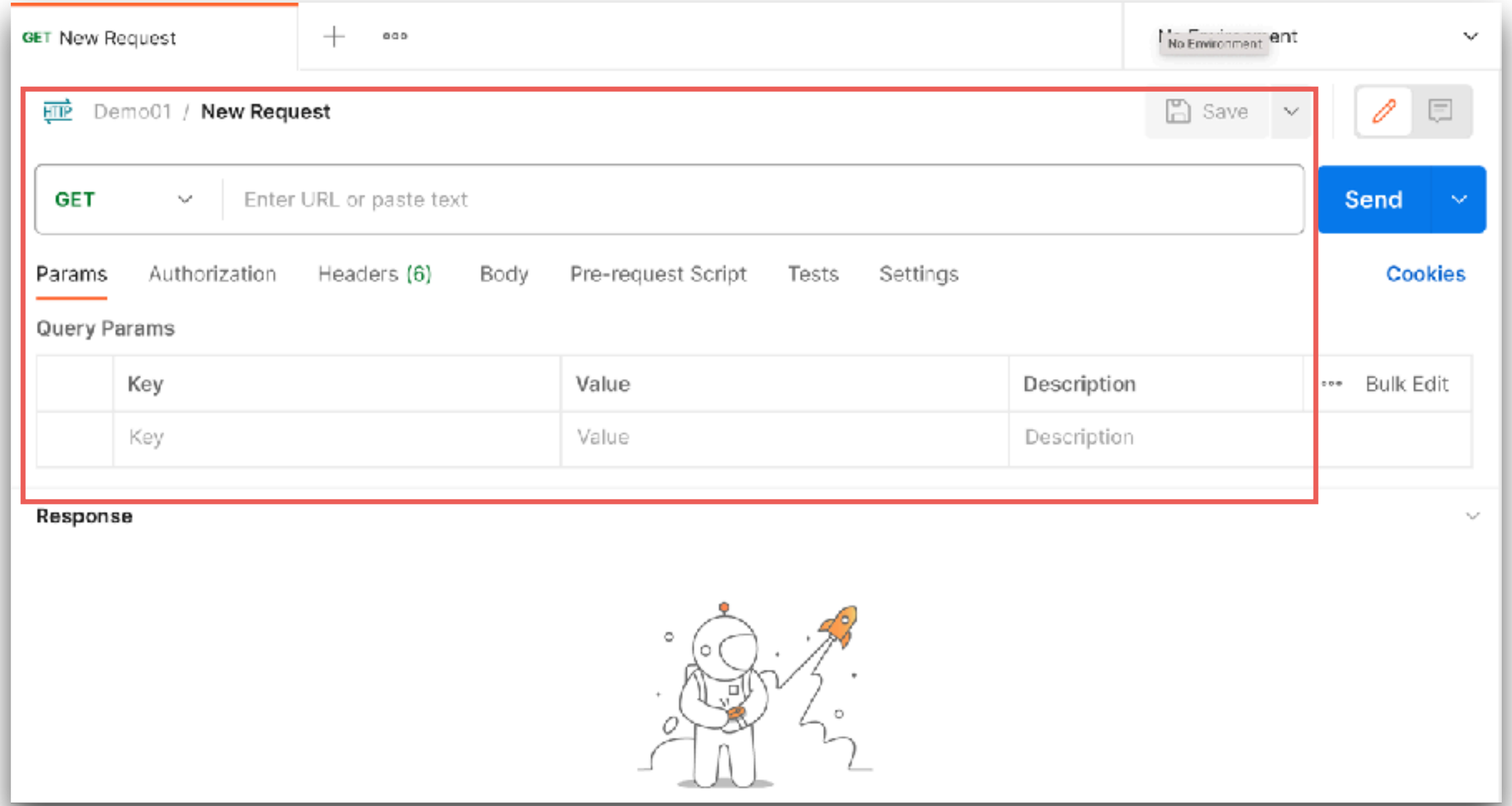
https://www.youtube.com/watch?v=9JunaO2hGK0

URL in HTTP Request

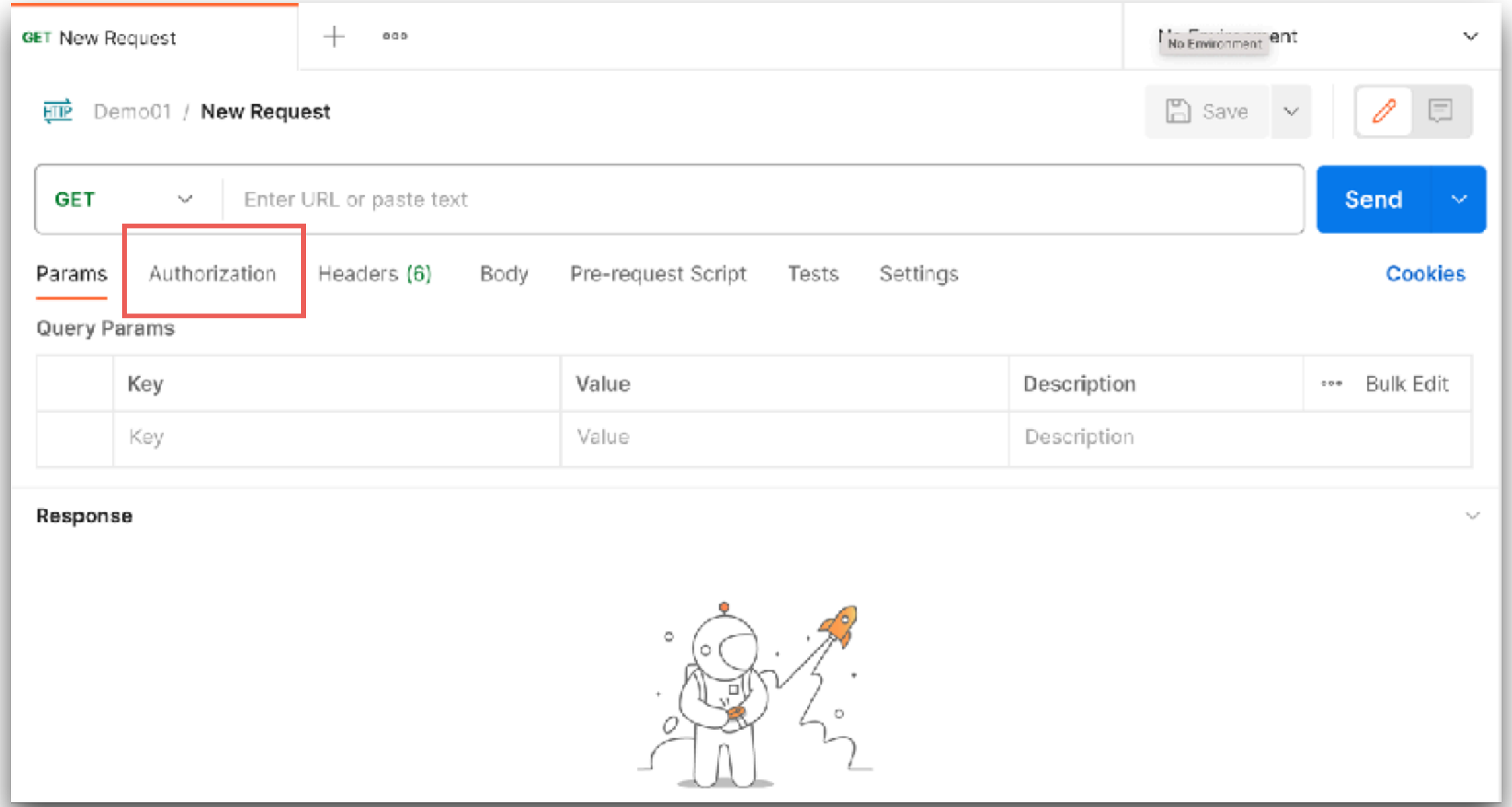
URL (Uniform Resource Locator)



HTTP Request in Postman



Authorization in request ?

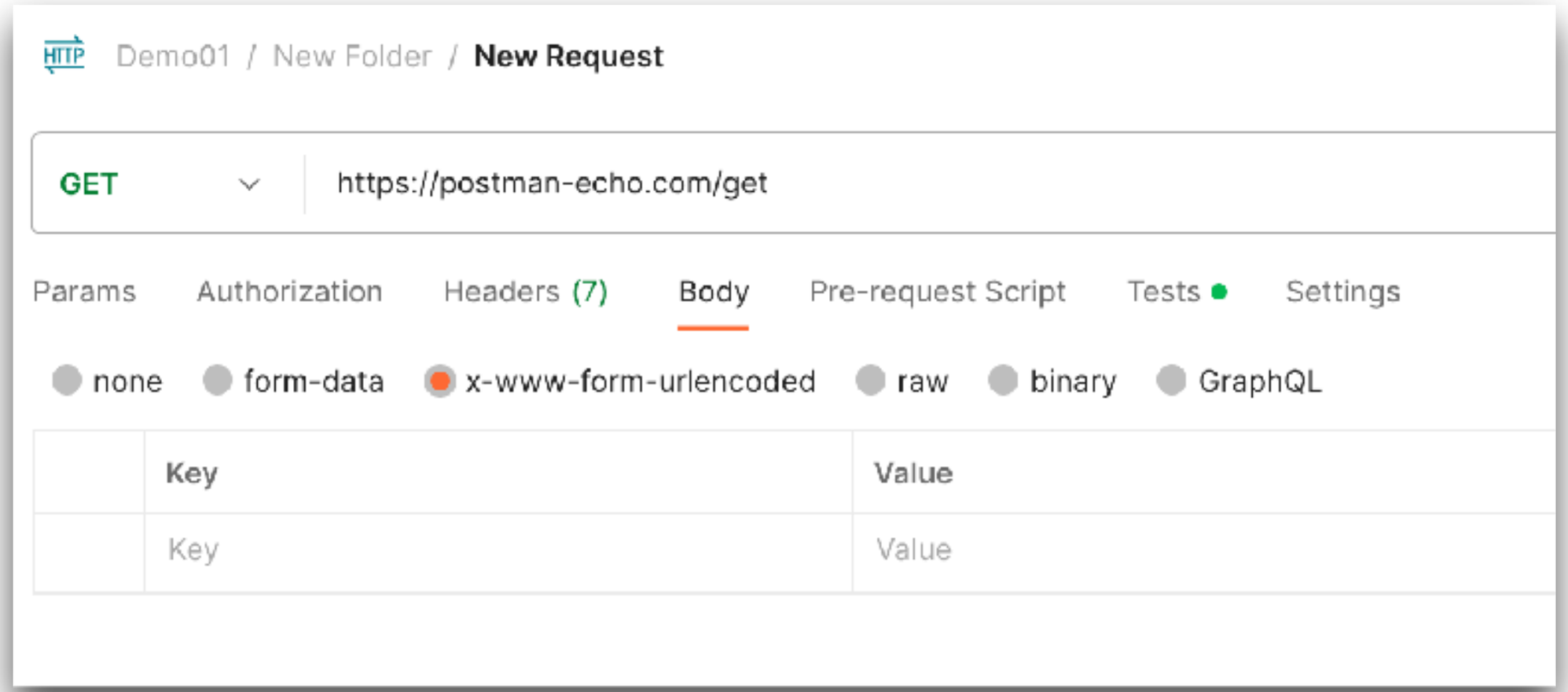


HTTP Methods

Method	Description
GET	Ask the server to retrieve a resource
POST	Ask the server to create a new resource
PUT	Ask the server to update an existing resource
DELETE	Ask the server to delete a resource
HEAD	Ask the server for status of a resource

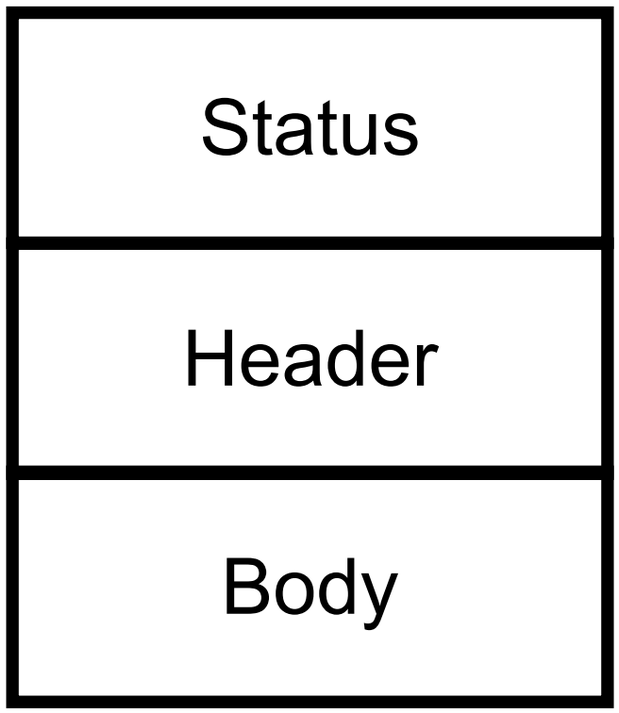
Request Body

Data sent from client to server
Text/HTML/JSON/XML/Multipart



HTTP Response

Message that server sent back to a client



HTTP Response

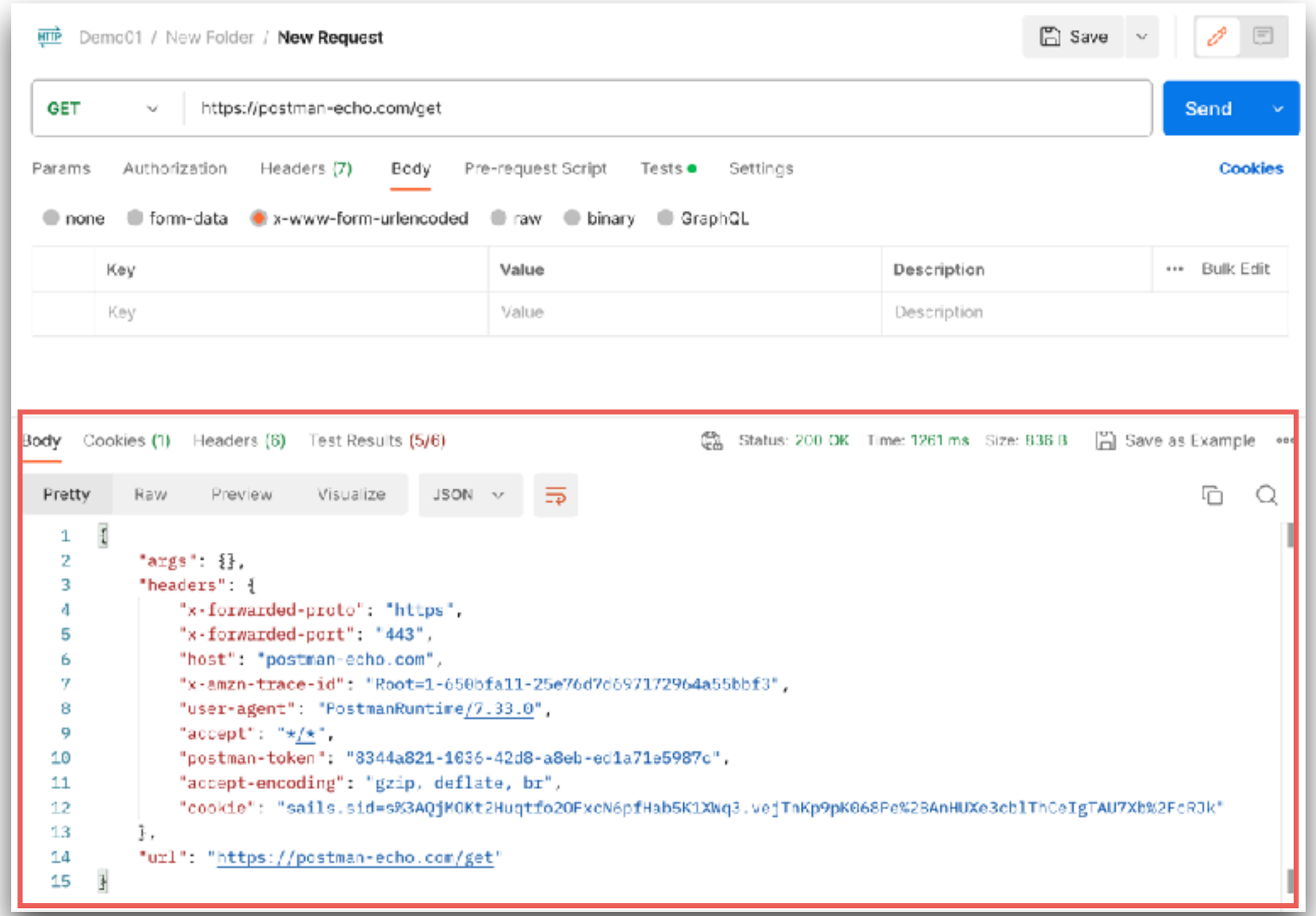
```
HTTP/1.1 200 OK
Date: Sun, 28 Mar 2023 10:15:00 GMT
Content-Type: application/json
Server: Apache/2.4.39 (Unix) OpenSSL/1.1.1c PHP/7.3.6
Content-Length: 1024

{
    "name": "John Doe",
    "email": "johndoe@example.com",
    "age": 30,
    "address": {
        "street": "123 Main St",
        "city": "Anytown",
        "state": "CA",
        "zip": "12345"
    }
}
```

Status Code

Code	Description
1xx	Information
2xx	Successfu
3xx	Redirection
4xx	Client error
5xx	Server error

https://developer.mozilla.org/en-US/docs/Web/HTTP/Status

HTTP Response in Postman



Test in Postman

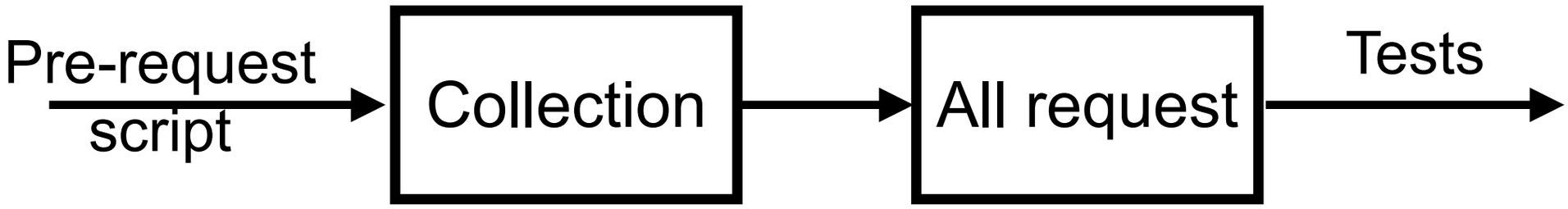
Pre-request script

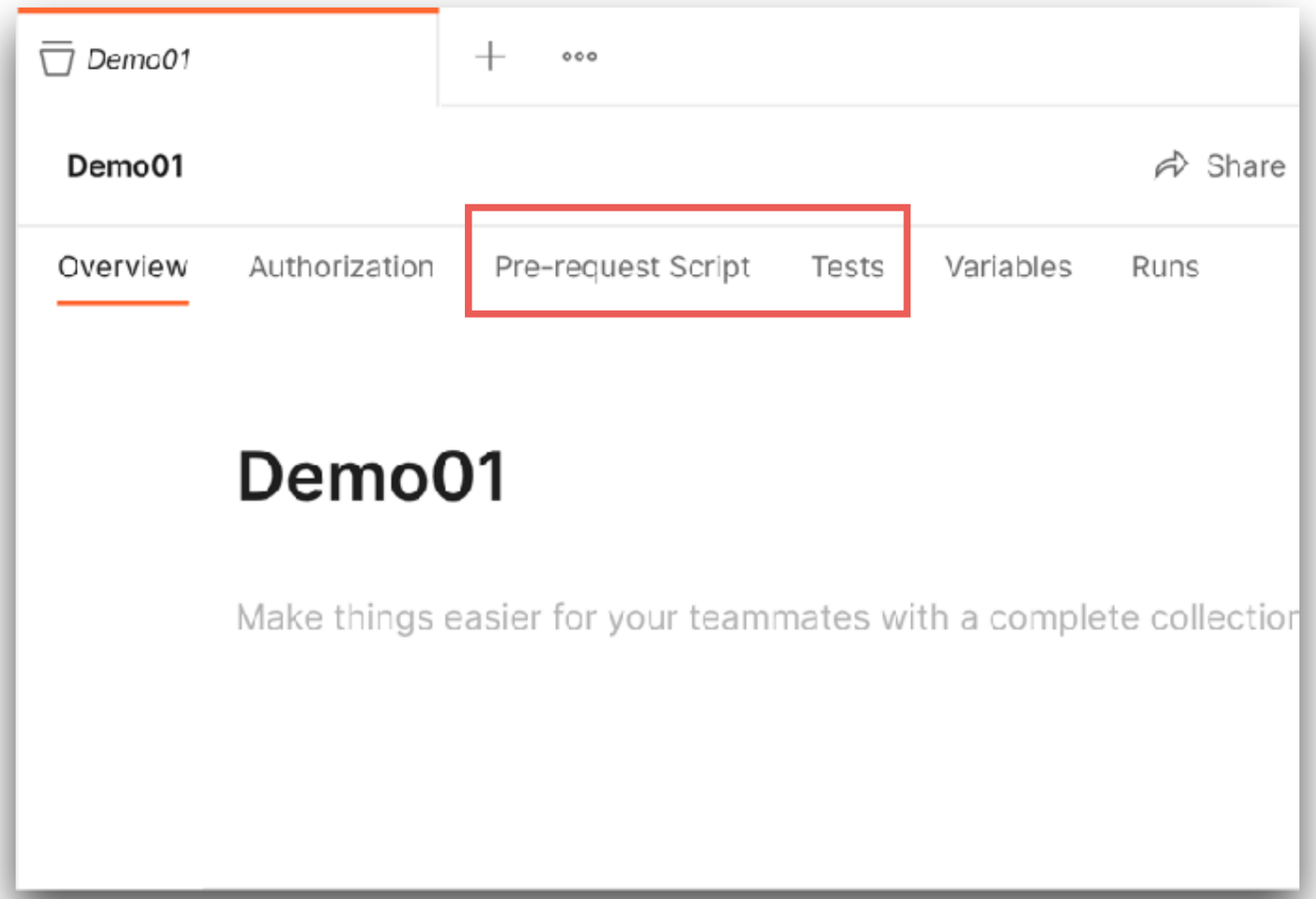
Test response

https://learning.postman.com/docs/writing-scripts/pre-request-scripts/

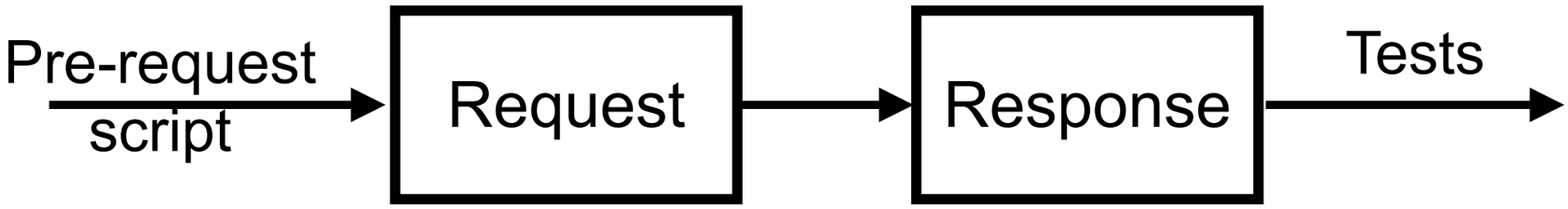
https://learning.postman.com/docs/writing-scripts/test-scripts/

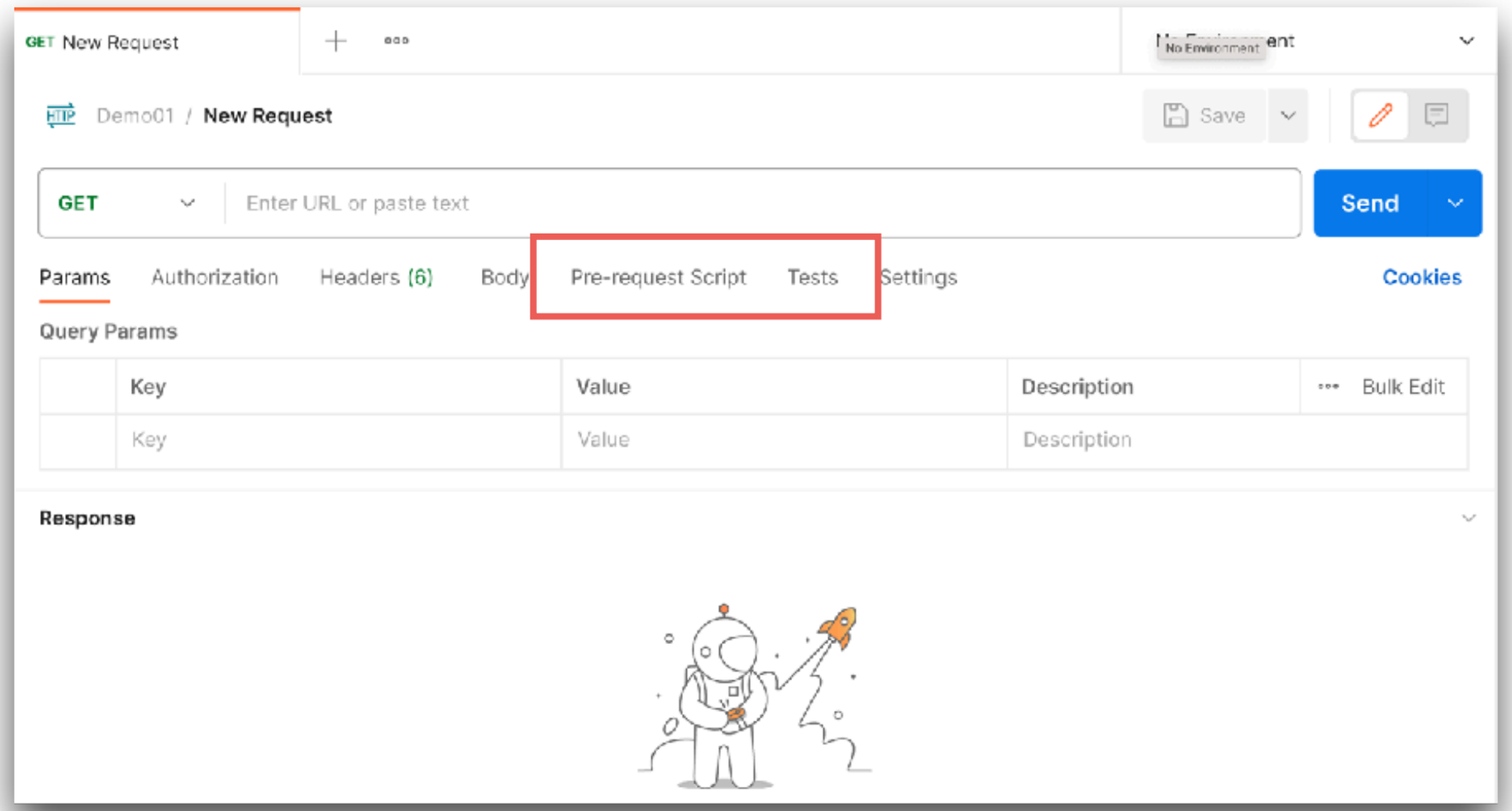
Test in collection





Test in request





What to test ?

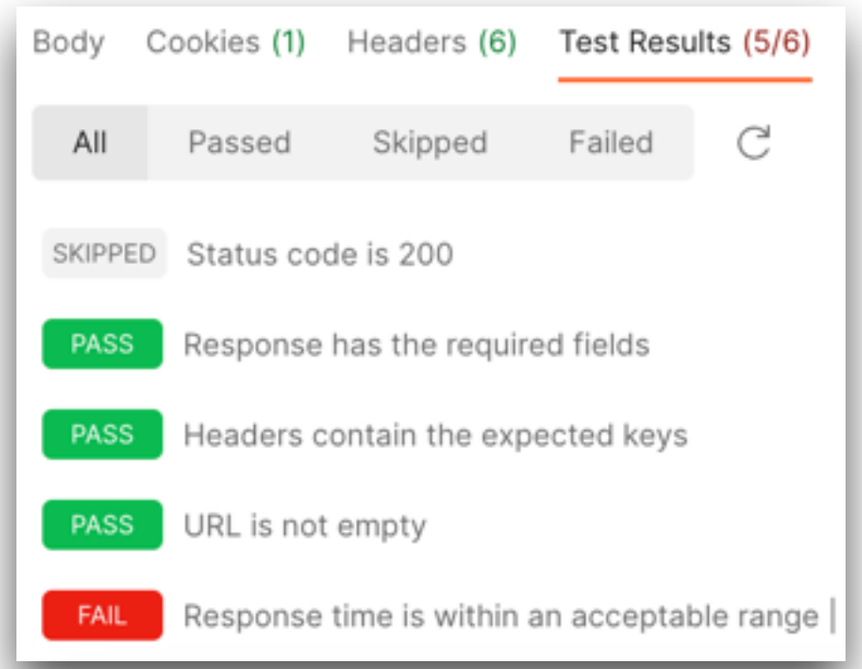
```
HTTP/1.1 200 OK
Date: Sun, 28 Mar 2023 10:15:00 GMT
Content-Type: application/json
Server: Apache/2.4.39 (Unix) OpenSSL/1.1.1c PHP/7.3.6
Content-Length: 1024

{
    "name": "John Doe",
    "email": "johndoe@example.com",
    "age": 30,
    "address": {
        "street": "123 Main St",
        "city": "Anytown",
        "state": "CA",
        "zip": "12345"
    }
}
```

Write test with Postbot

https://learning.postman.com/docs/getting-started/basics/about-postbot/

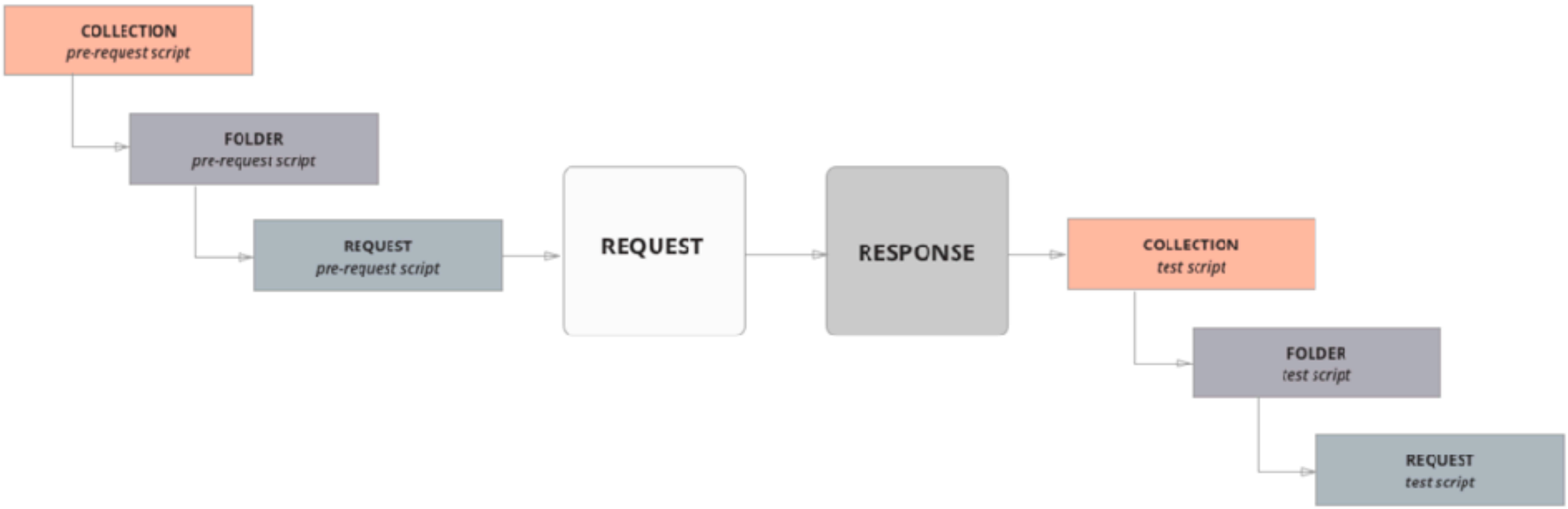
Skip test !!

```
pm.test.skip("Status code is 200", function () {

    pm.response.to.have.status(200);

});
```



Test Execution

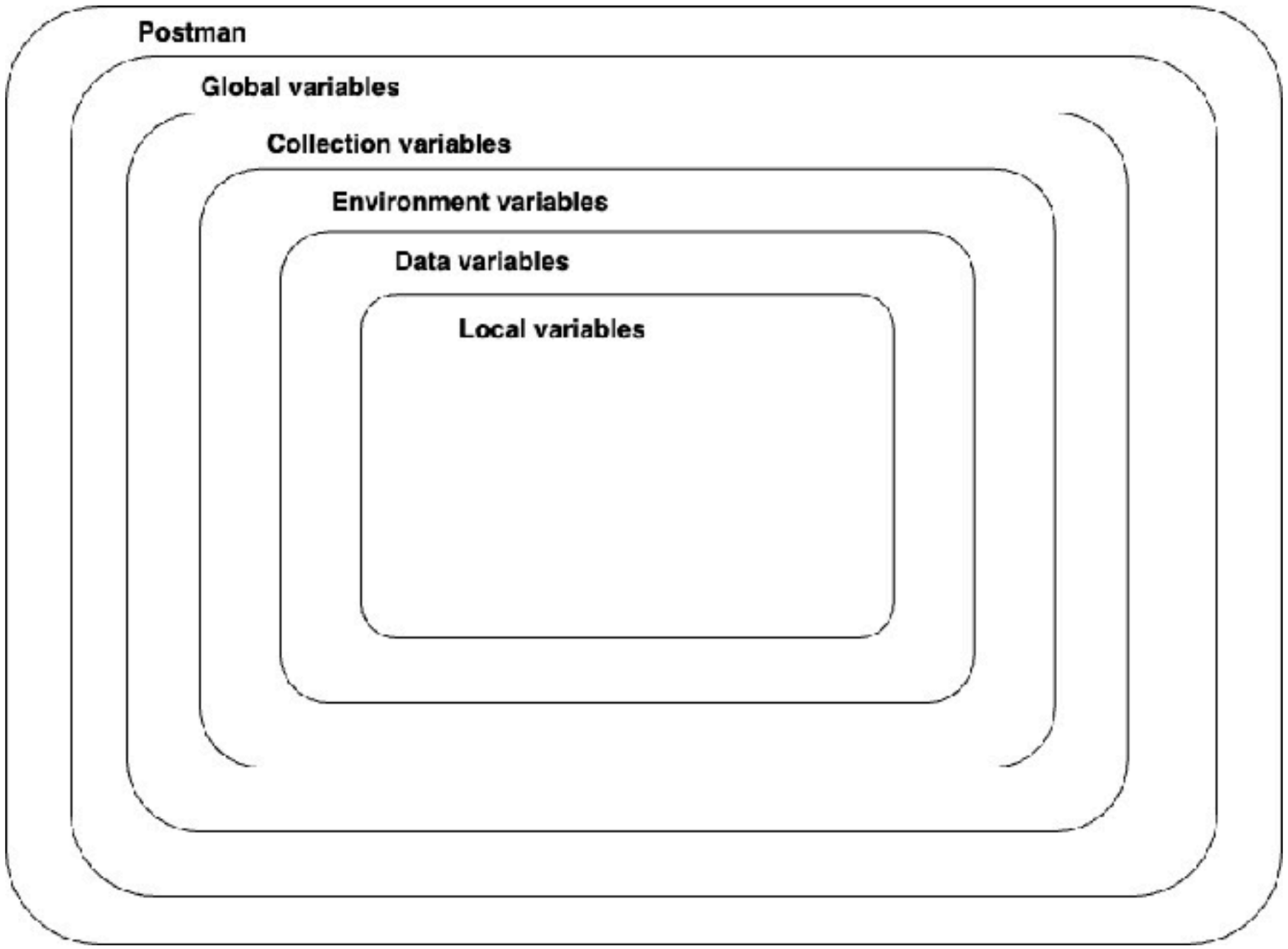
Test Execution



Variables

https://learning.postman.com/docs/sending-requests/variables/

Variables Scopes



https://learning.postman.com/docs/sending-requests/variables/#variable-scopes

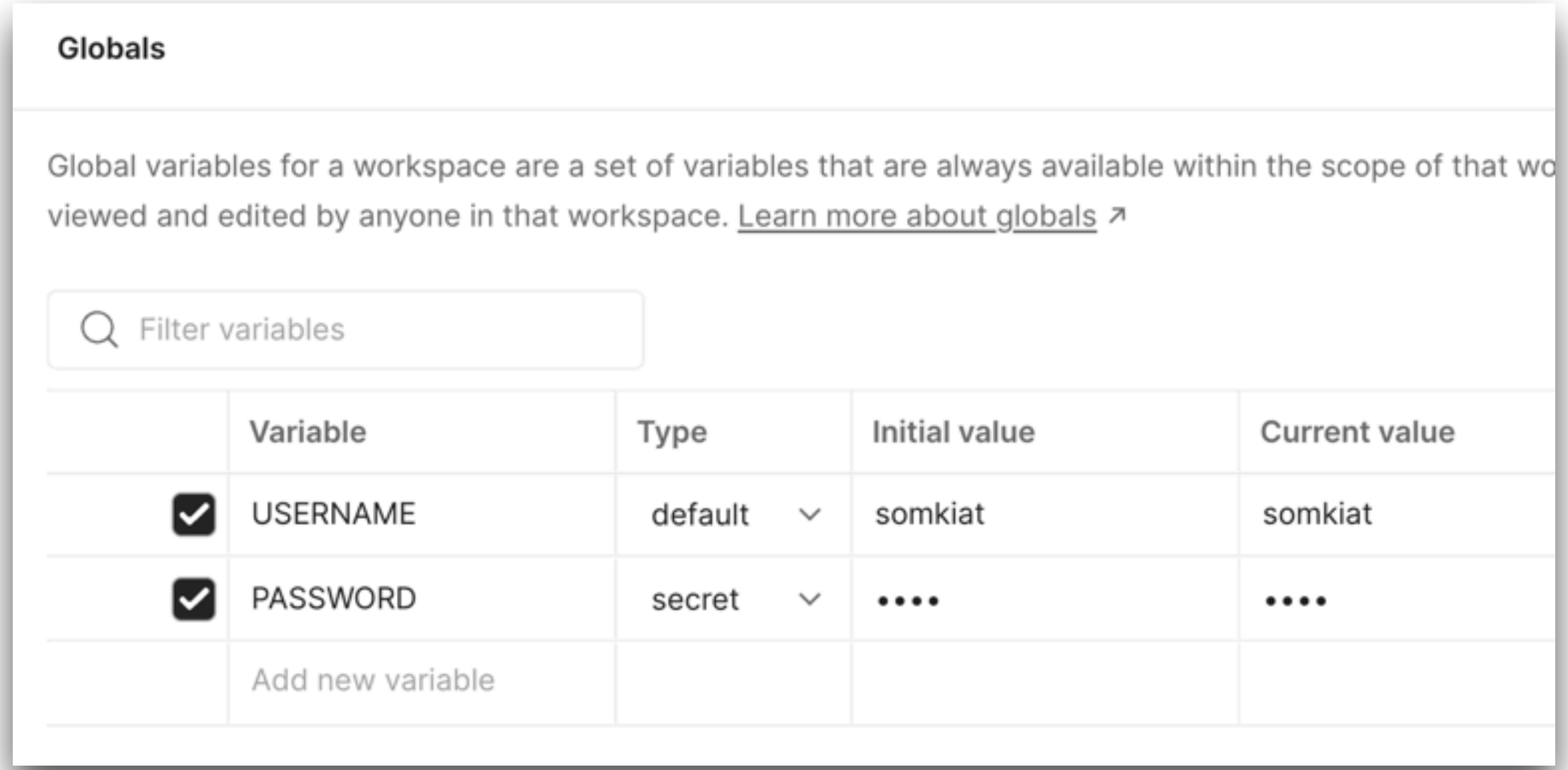
Variables

Global variables
Collection variables
Local variables

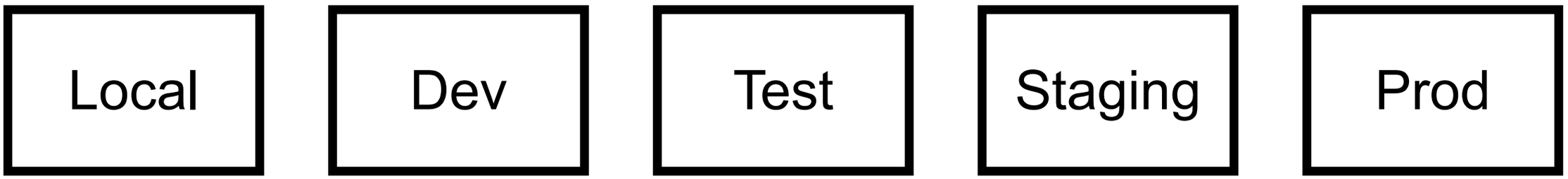
{{VARIABLE_NAME}}

Variable Types

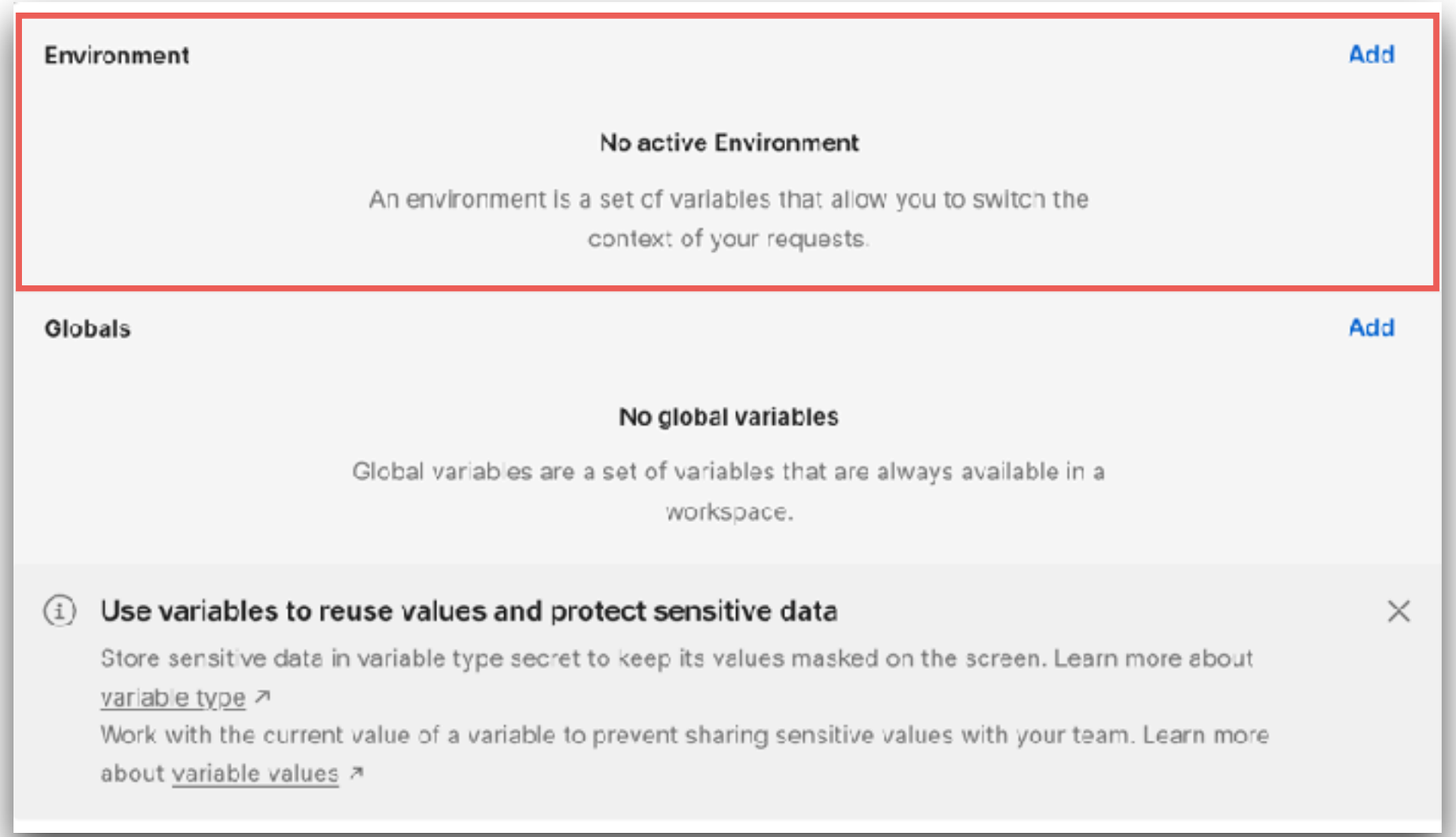
Plain text
Secret for sensitive data



Environment to test



Environment in Postman



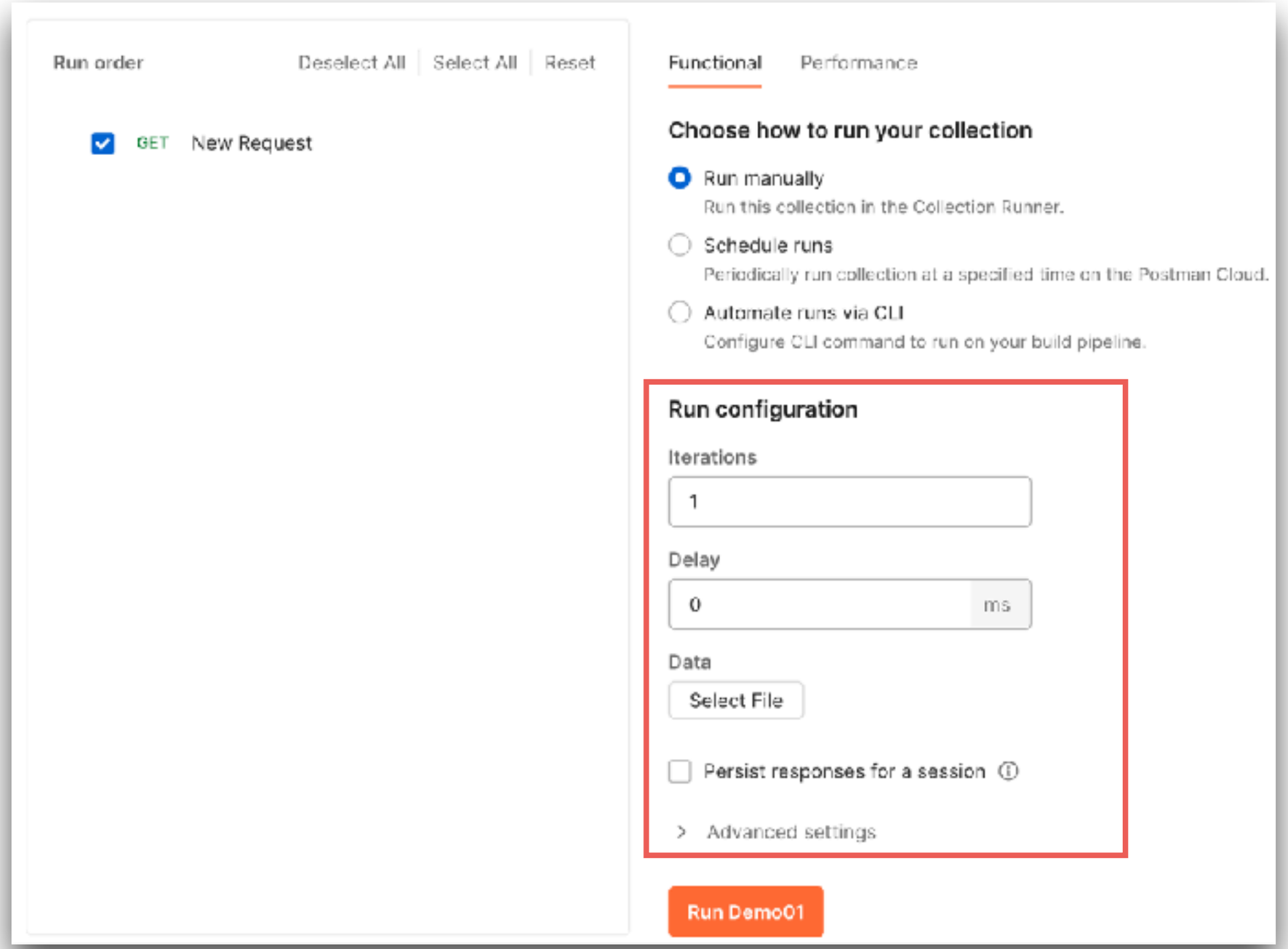
Working with more data

Data-Driven Testing

Duplicate requests	Use external data files	Use pre-request script

CSV, JSON

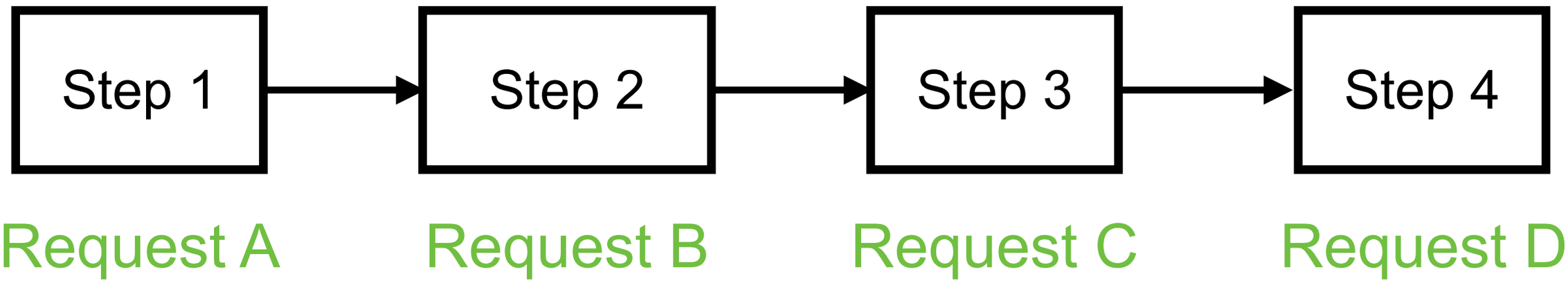
Working with CSV/JSON



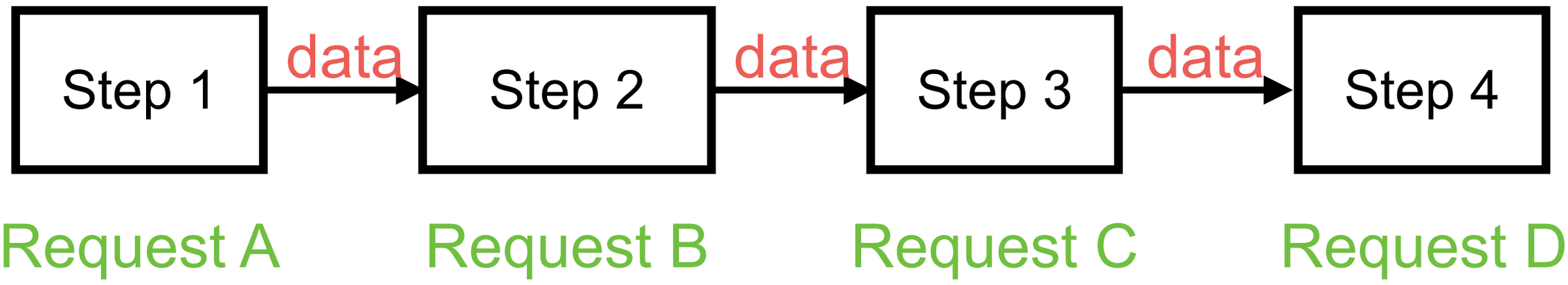
https://learning.postman.com/docs/collections/running-collections/working-with-data-files/

Workflow in postman

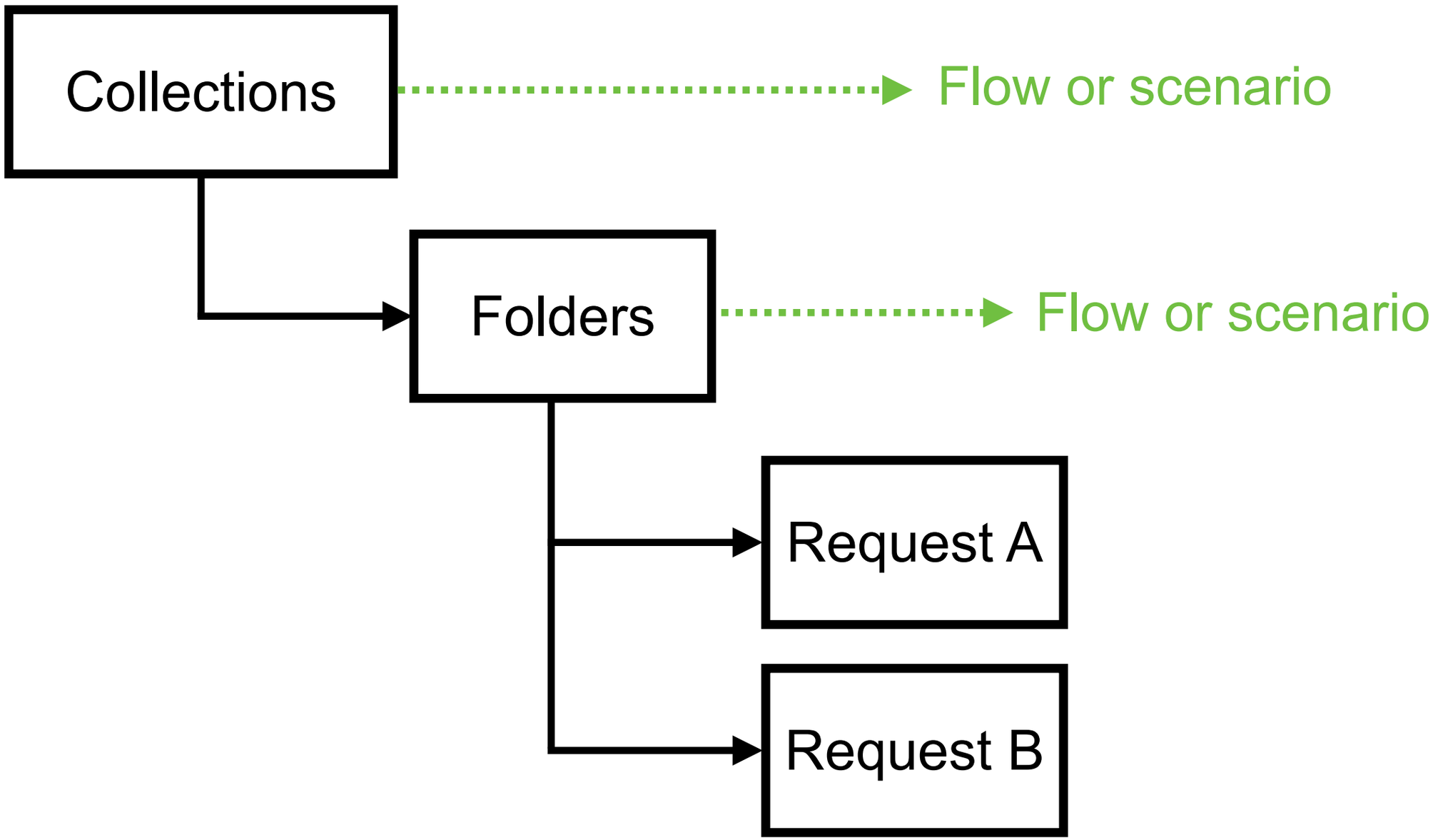
Workflow



Workflow in postman ?



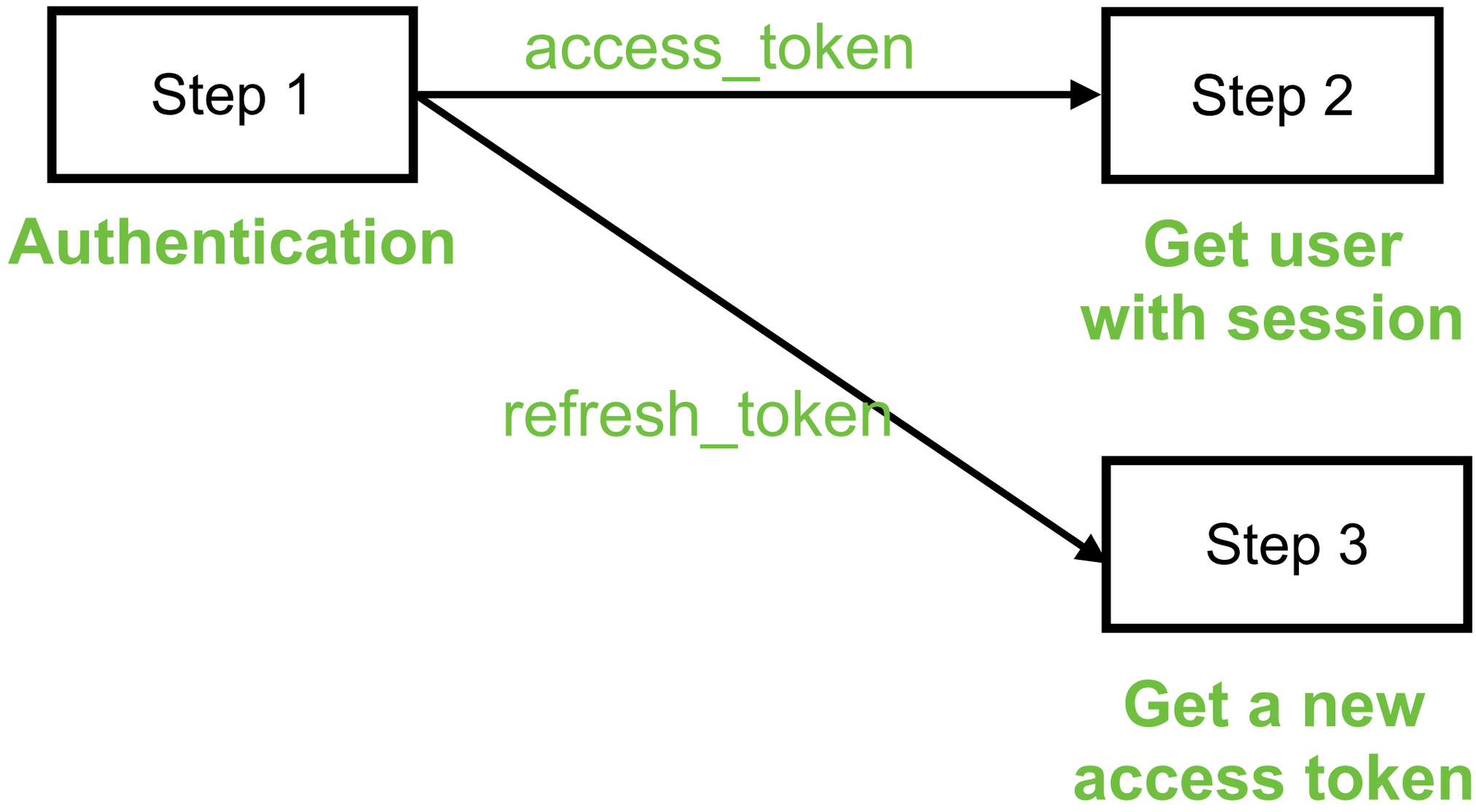
Postman structure



Workshop

https://fakeapi.platzi.com/

Workflow



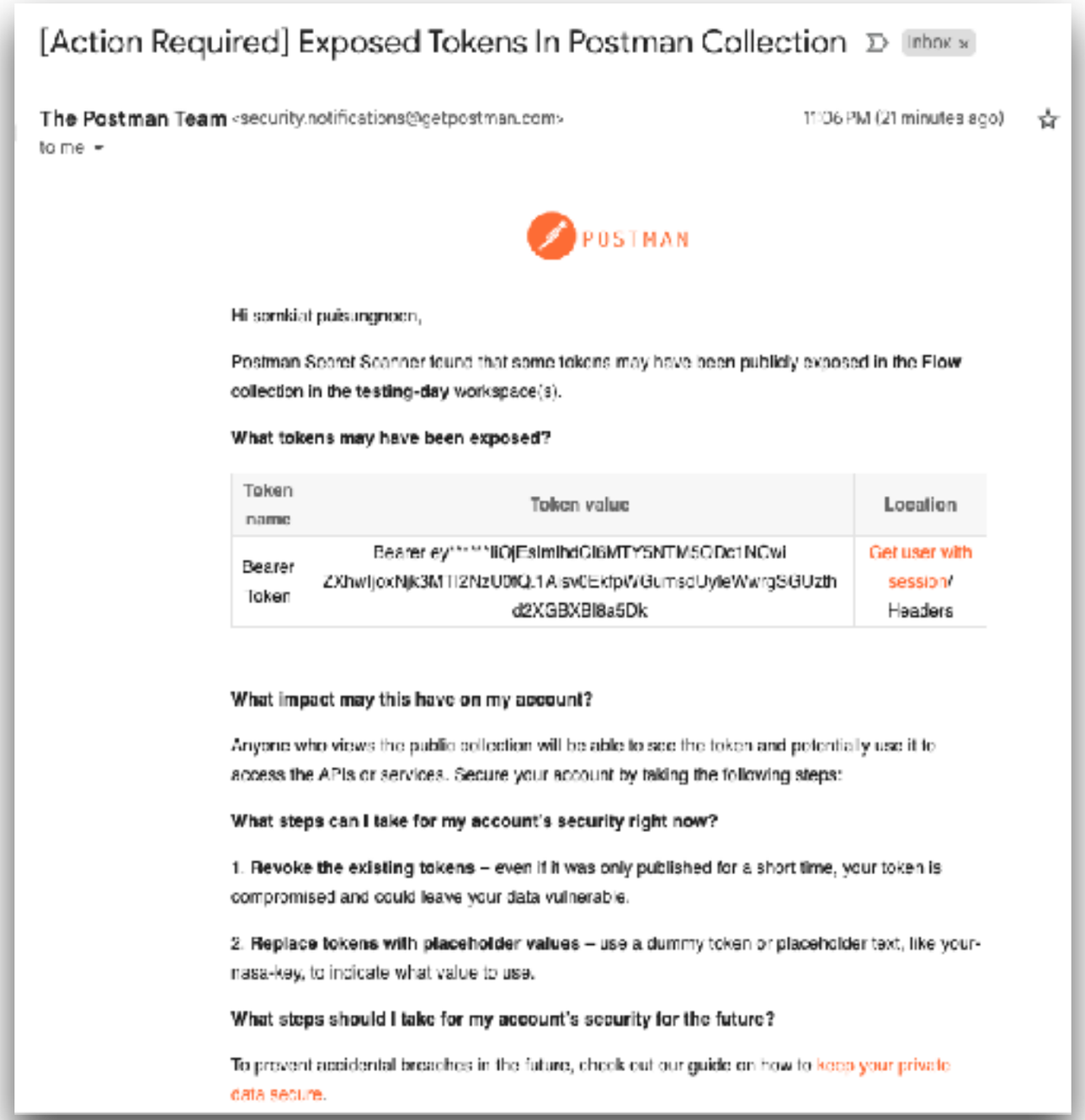
Solutions

Hard code !!

Variables and Test scripts

Call javascript function

Exposed Tokens in Postman



[Action Required] Exposed Tokens In Postman Collection Inbox x

The Postman Team <security.notifications@getpostman.com> 11:06 PM (21 minutes ago)

to me

POSTMAN

Hi sarnkiat puisungnoen,

Postman Secret Scanner found that some tokens may have been publicly exposed in the Flow collection in the testing-day workspace(s).

What tokens may have been exposed?

Token name	Token value	Location
Bearer token	Bearer ey[redacted]ZXhwIjoxNjk3M1I2NzU0fQ.1A svCEkfpWGumsoUyfeWwrgSGUzthd2XGBXBl8a5Dk	Get user with session Headers

What impact may this have on my account?

Anyone who views the public collection will be able to see the token and potentially use it to access the APIs or services. Secure your account by taking the following steps:

What steps can I take for my account's security right now?

1. **Revoke the existing tokens** – even if it was only published for a short time, your token is compromised and could leave your data vulnerable.

2. **Replace tokens with placeholder values** – use a dummy token or placeholder text, like your-nasa-key, to indicate what value to use.

What steps should I take for my account's security for the future?

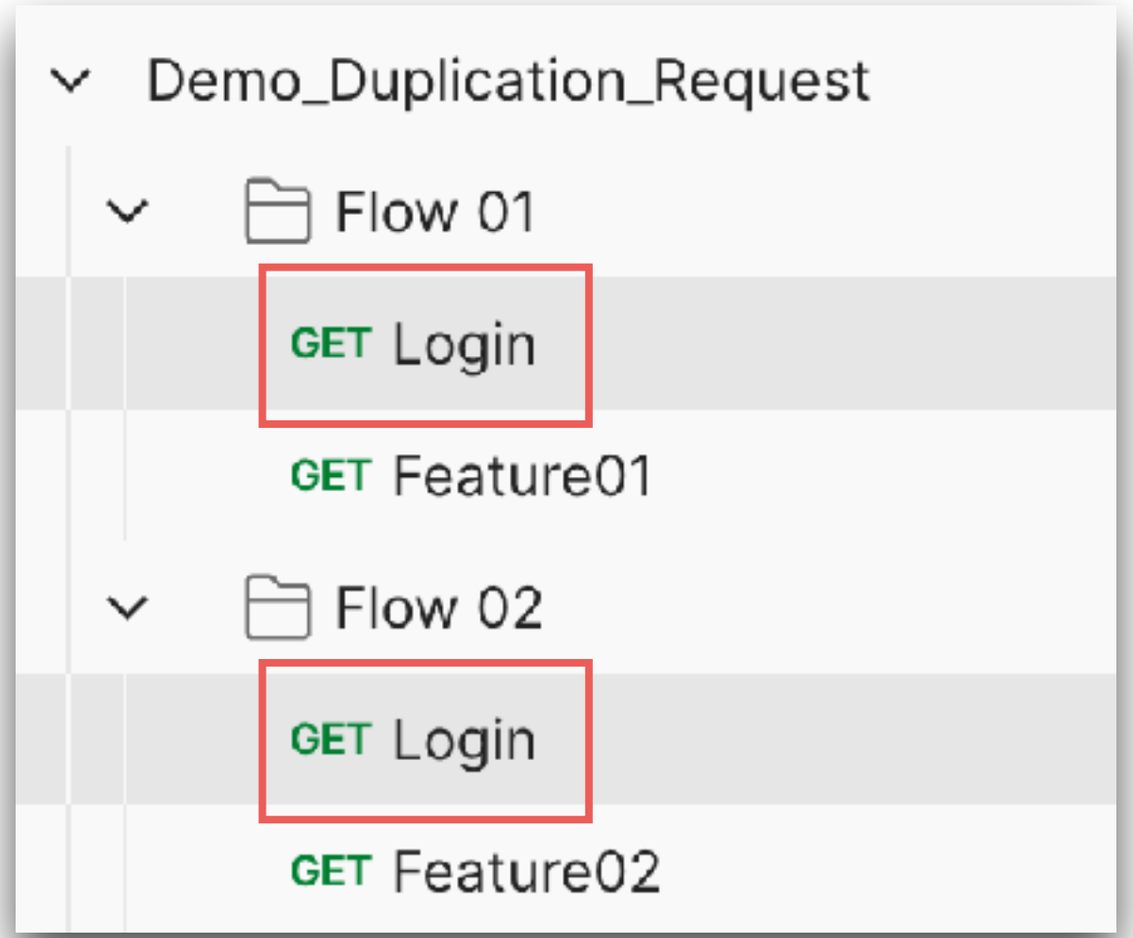
To prevent accidental breaches in the future, check out our guide on how to keep your private data secure.

Use Variables in test script

Collection variables

```
var jsonData = pm.response.json();
var accessToken = jsonData.access_token;
var refreshToken = jsonData.refresh_token;

pm.collectionVariables.set("accessToken", accessToken);
pm.collectionVariables.set("refreshToken", refreshToken);
```

Duplication Requests !!



https://learning.postman.com/docs/writing-scripts/script-references/postman-sandbox-api-reference/#scripting-with-request-data

Write code

```
const postRequest = {
  url: 'https://postman-echo.com/post',
  method: 'POST',
  header: {
    'Content-Type': 'application/json',
    'X-Foo': 'bar'
  },
  body: {
    mode: 'raw',
    raw: JSON.stringify({ key: 'this is json' })
  }
};
pm.sendRequest(postRequest, (error, response) => {
  console.log(error ? error : response.json());
});
```

https://github.com/up1/workshop-postman/wiki/Workshop-Code-Snippets

Create JS Function

```
var getToken = () => {
    const postRequest = {
        url: 'https://api.escuelajs.co/api/v1/auth/login',
        method: 'POST',
        header: {
            'Content-Type': 'application/json'
        },
        body: {
            mode: 'raw',
            raw: JSON.stringify({
                "email": "john@mail.com",
                "password": "changeme"
            })
        }
    }
    pm.sendRequest(postRequest, (error, response) => {
        var response = response.json();
        pm.collectionVariables.set("accessTokenV3", response.access_token);
        pm.collectionVariables.set("refreshTokenV3", response.refresh_token);
    });
}

pm.globals.set("getToken", getToken.toString());
```

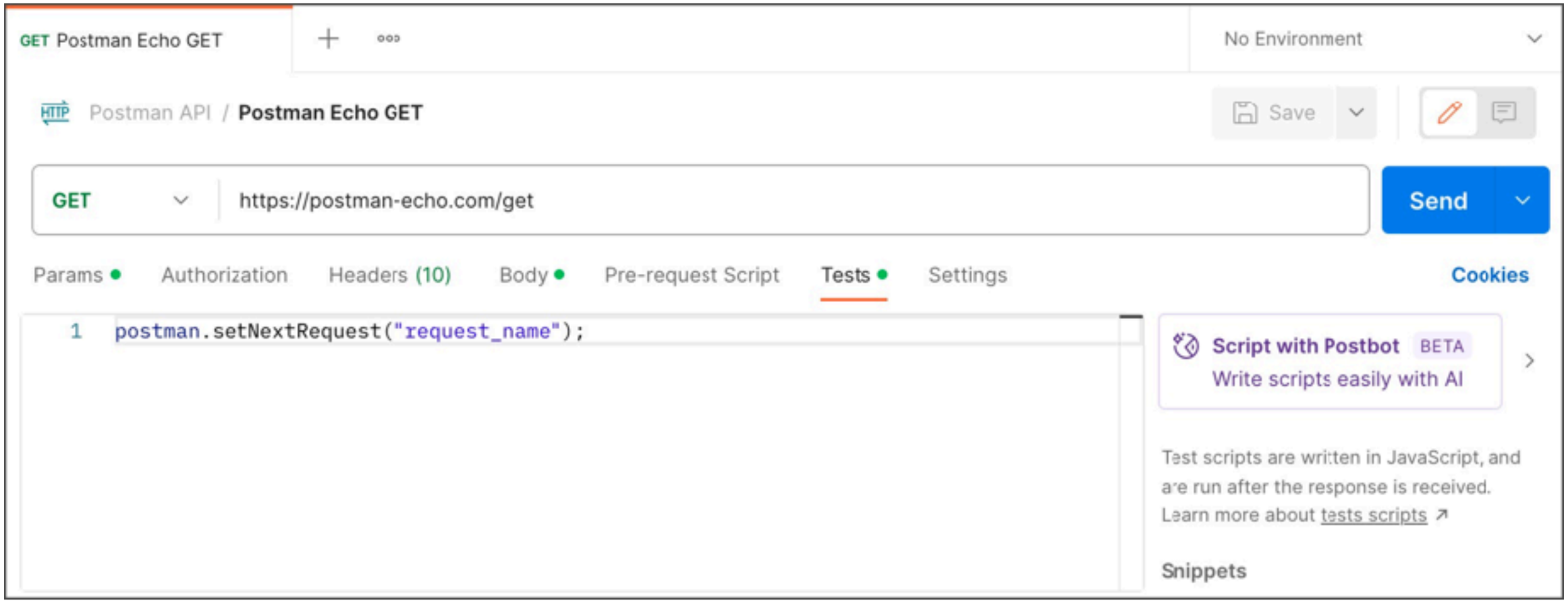
https://github.com/up1/workshop-postman/wiki/Workshop-Code-Snippets

Eval function

```
eval(pm.globals.get("getToken"))();
```

https://github.com/up1/workshop-postman/wiki/Workshop-Code-Snippets

Call request in collection



https://learning.postman.com/docs/collections/running-collections/building-workflows/

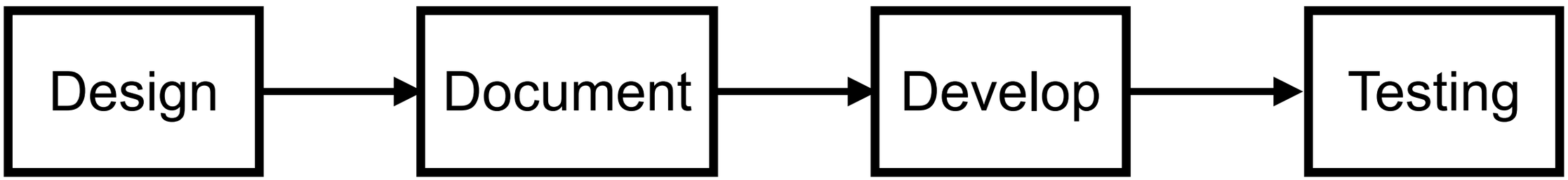
Terminate execution !!

postman.setNextRequest(null);

https://learning.postman.com/docs/collections/running-collections/building-workflows/

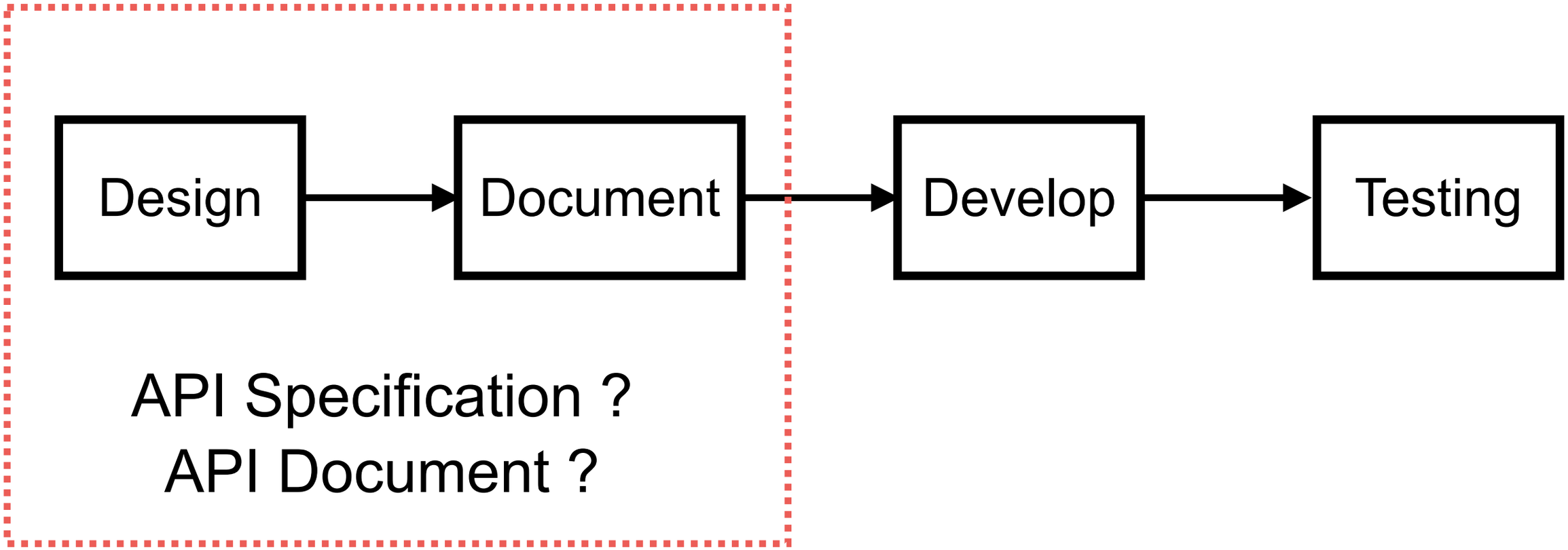
Software Delivery Process

Software Delivery Process

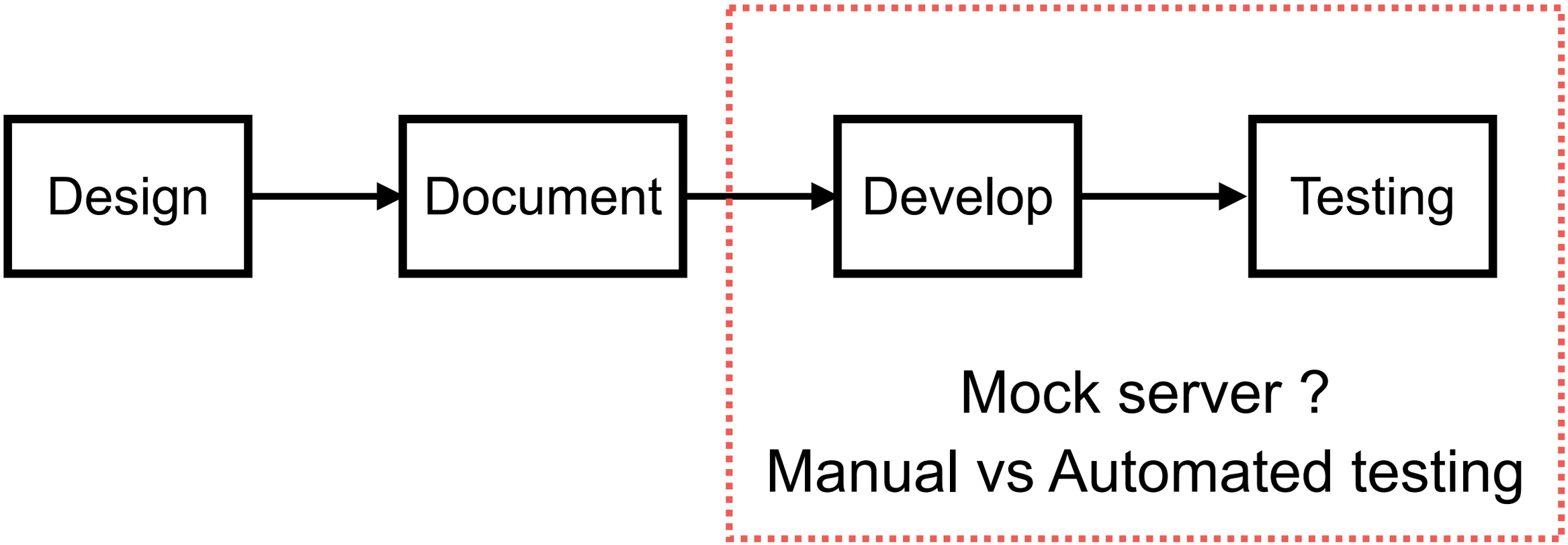


API Development

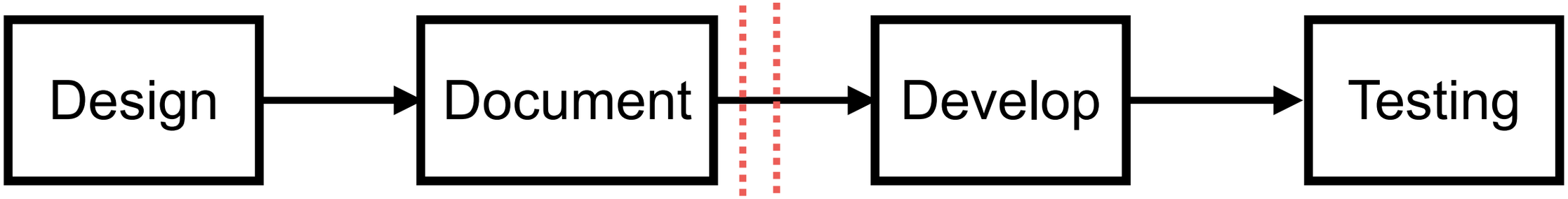
APIs



APIs



Working with Postman

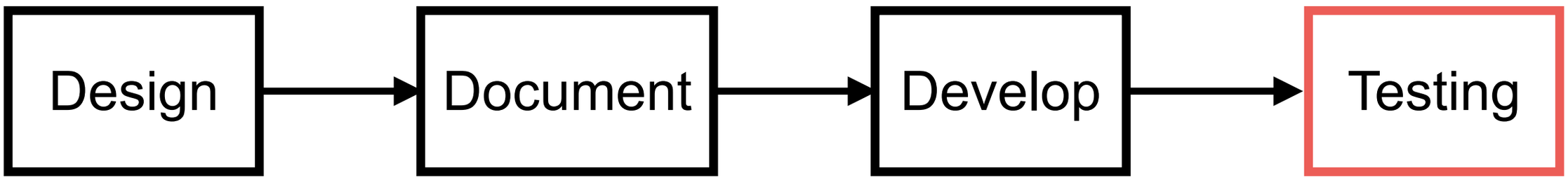


API Specification ?
API Document ?

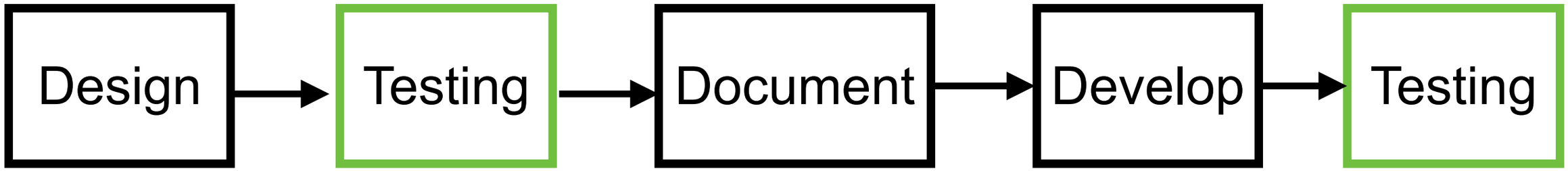
Mock server ?
Manual vs Automated testing

POSTMAN

ทำก่อนคิด (ทดสอบอย่างไร)



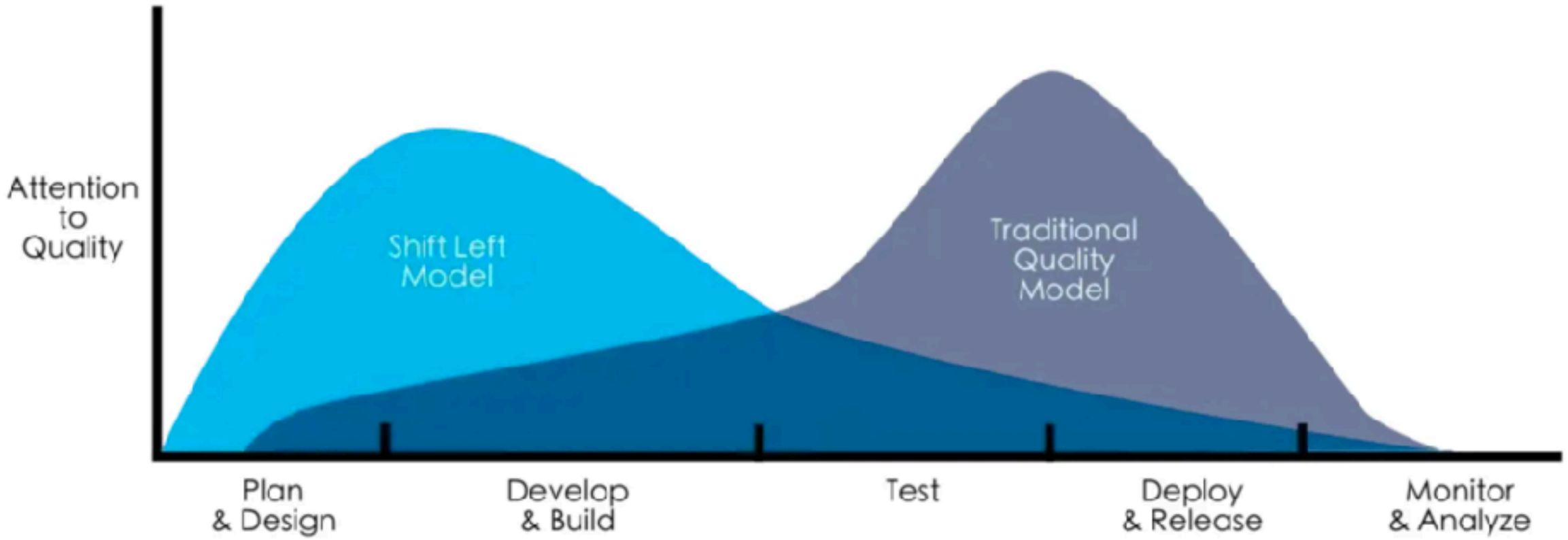
คิดก่อนทำ



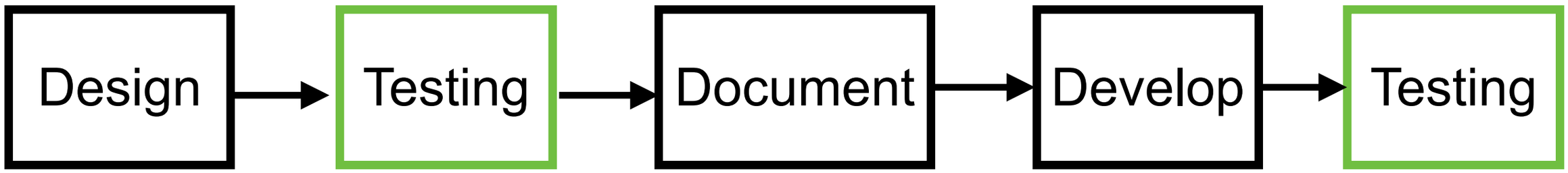
Why and What to test ?

Validation
Verification

คิดก่อนทำ



High Quality



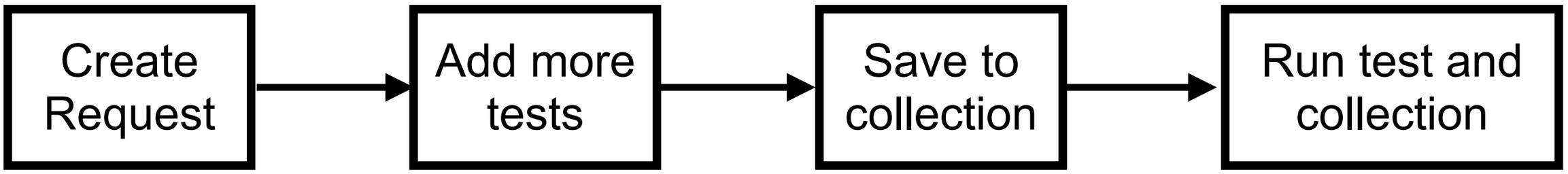
Validation
Verification

Why and What to test ?

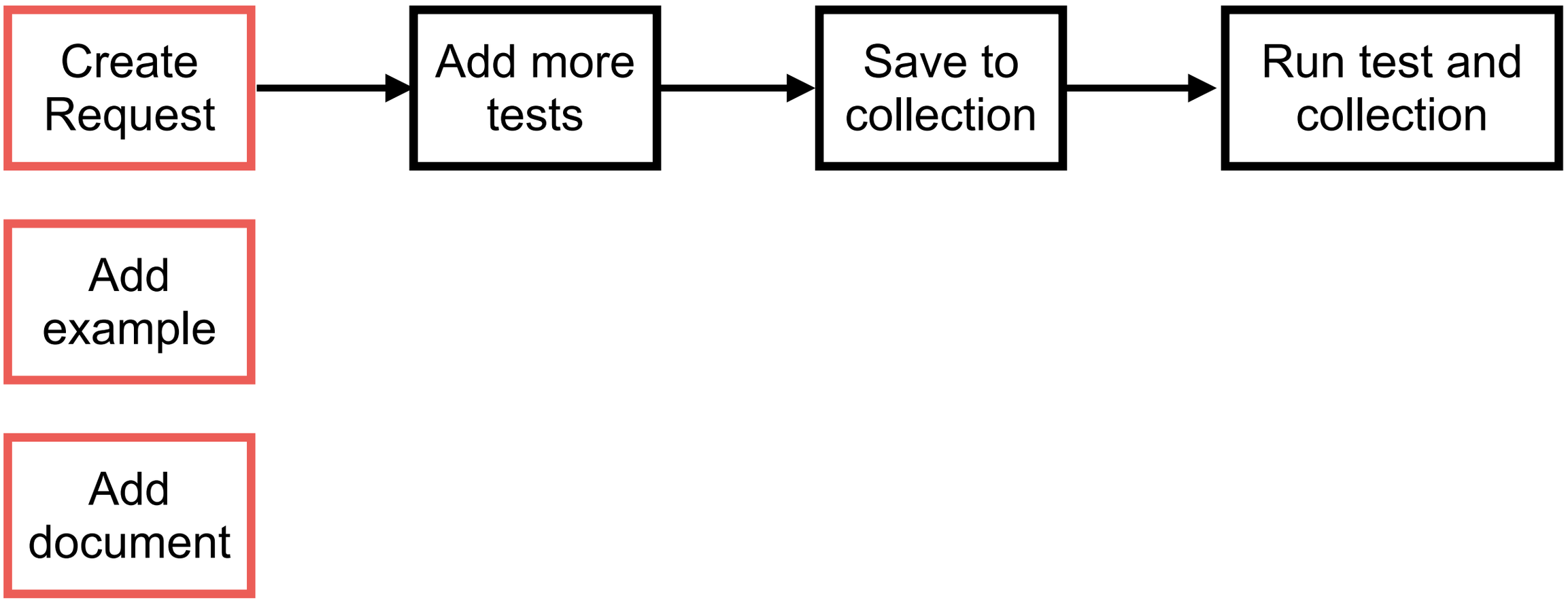
POSTMAN

Workshop

Simple Process



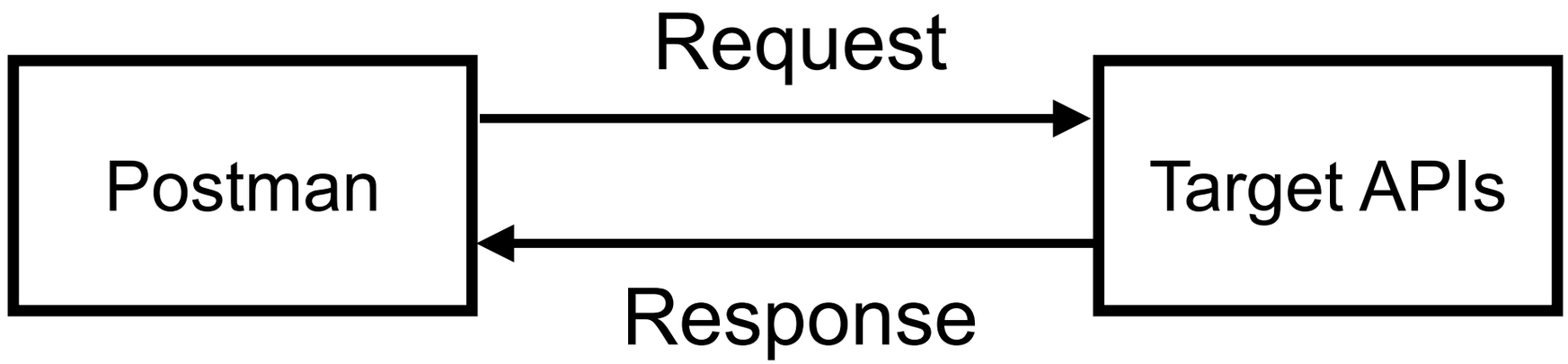
Design first with Postman



Add more tests !!

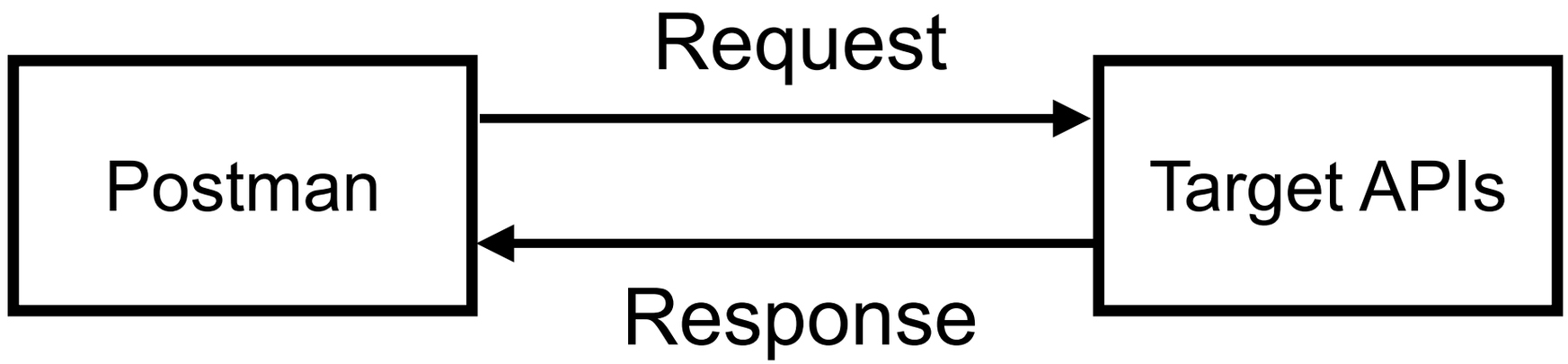
Why and What to test ?

Thinking about testing (API)



Why and What to test ?

Thinking about testing (API)



Status code	Response time	Response body	Response schema

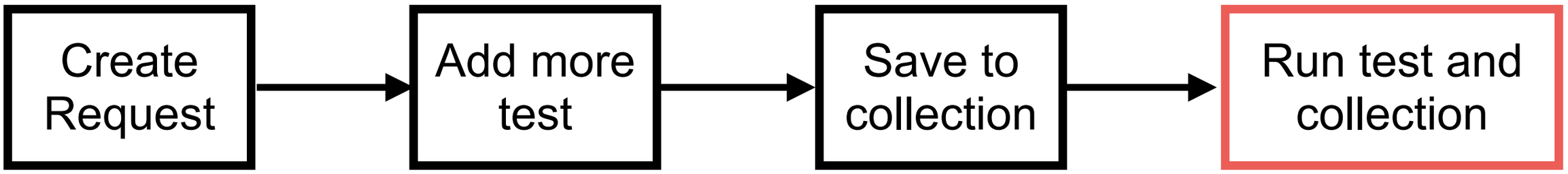
Add test cases in request and collections

JavaScript

https://learning.postman.com/docs/writing-scripts/test-scripts/

Automation process with Postman

Testing process ?



Testing with Postman

Collection runner

Postman CLI

Newman CLI

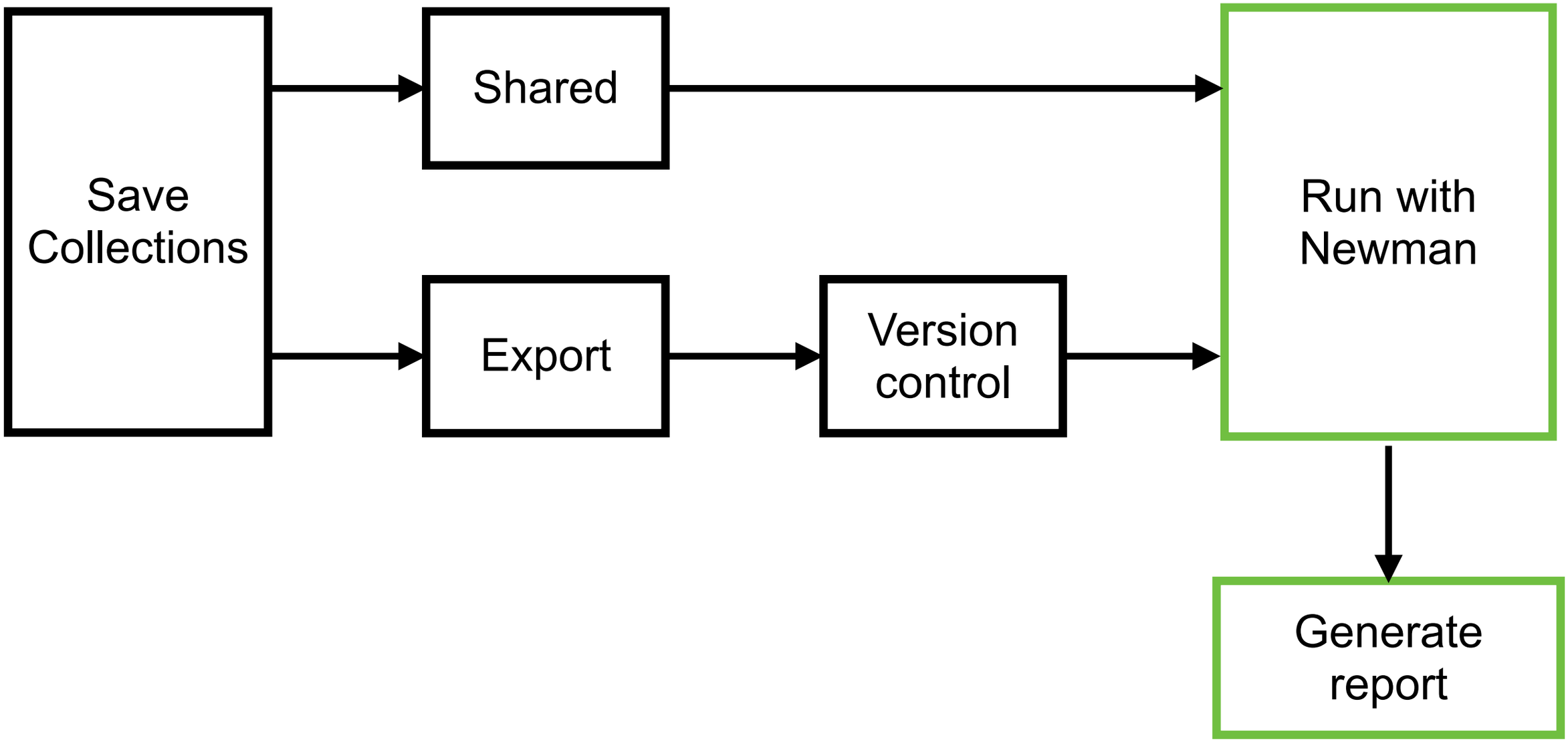
Newman CLI

Send request and receive response
Run collections in command line
Integrate with CI/CD tools

$npm install -g newman

https://github.com/postmanlabs/newman

Working with Newman



Newman CLI

Run collection
Run requests in folder
Working with data
Working with environments
Working with OS variables

Working with OS variables

Test script

```
var user = pm.globals.get(“demo_username”)
var pass = pm.globals.get(“demo_password”)
```

Run in command line

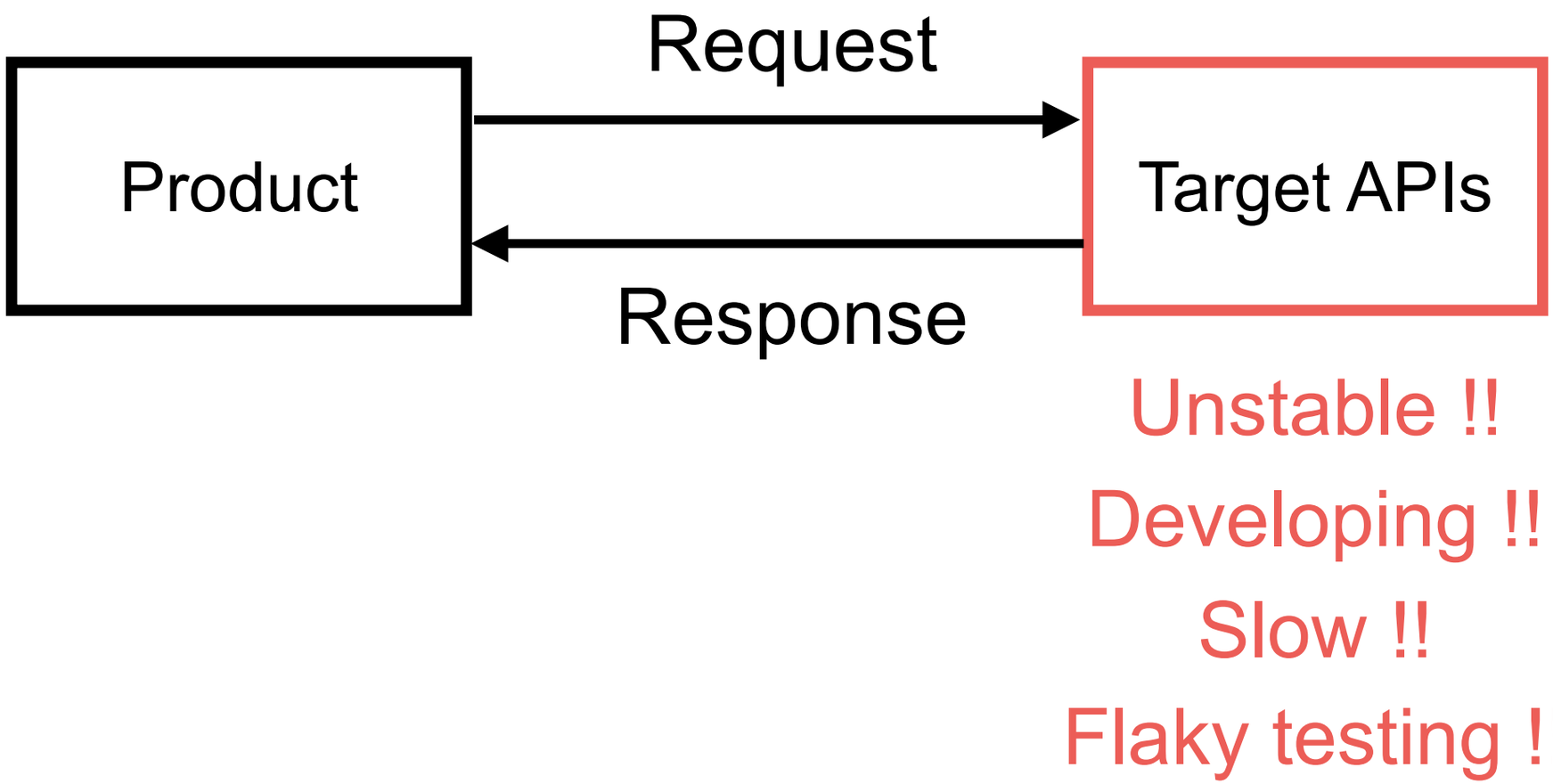
```
$newman run your-collection.json
  --global-var demo_username=demo user
  --global-var demo_password=demo password
```

Test Report

CLI
JUNIT
JSON
HTML

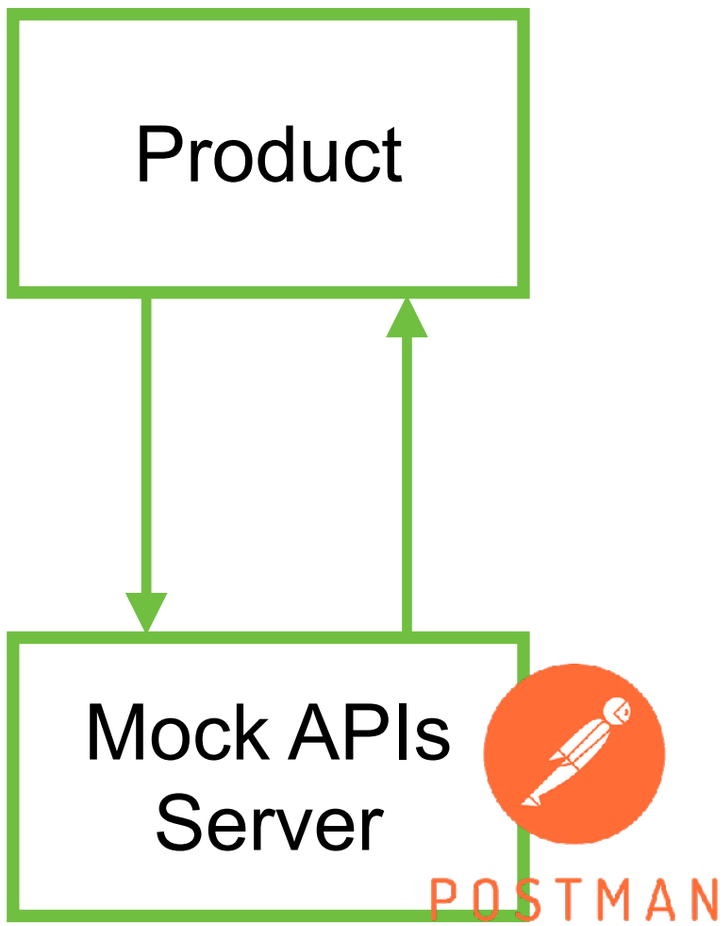
Mock Server

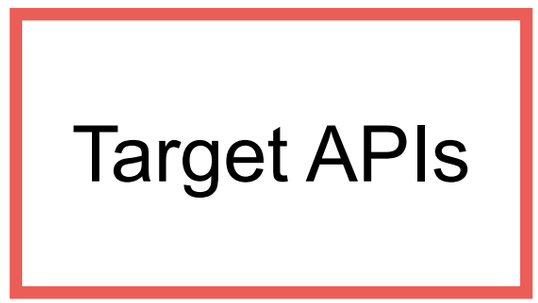
Problems ?



https://learning.postman.com/docs/designing-and-developing-your-api/mocking-data/setting-up-mock/

Mock Server with Postman





Unstable !!

Developing !!

Slow !!

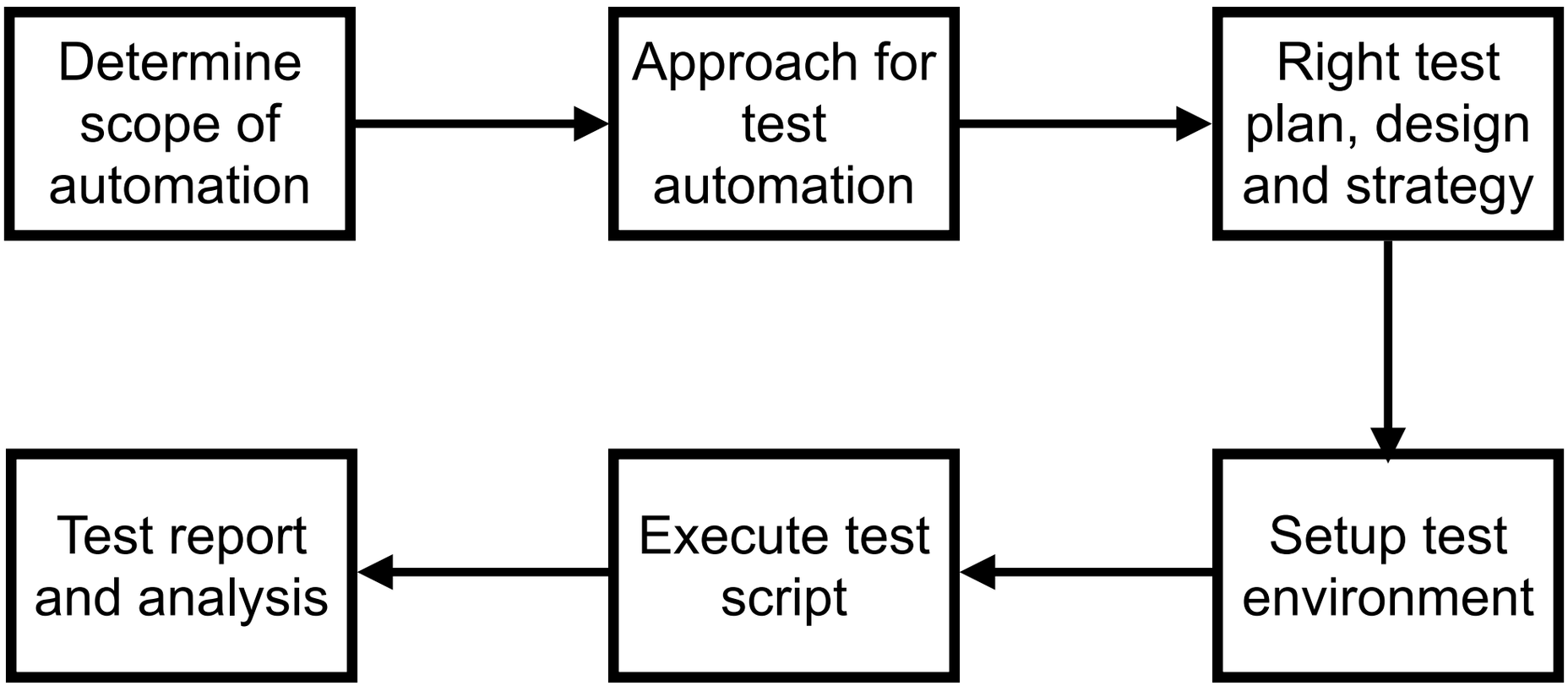
Flaky testing !!

https://learning.postman.com/docs/designing-and-developing-your-api/mocking-data/setting-up-mock/

Workshop MockServer

Automated Testing Lifecycle

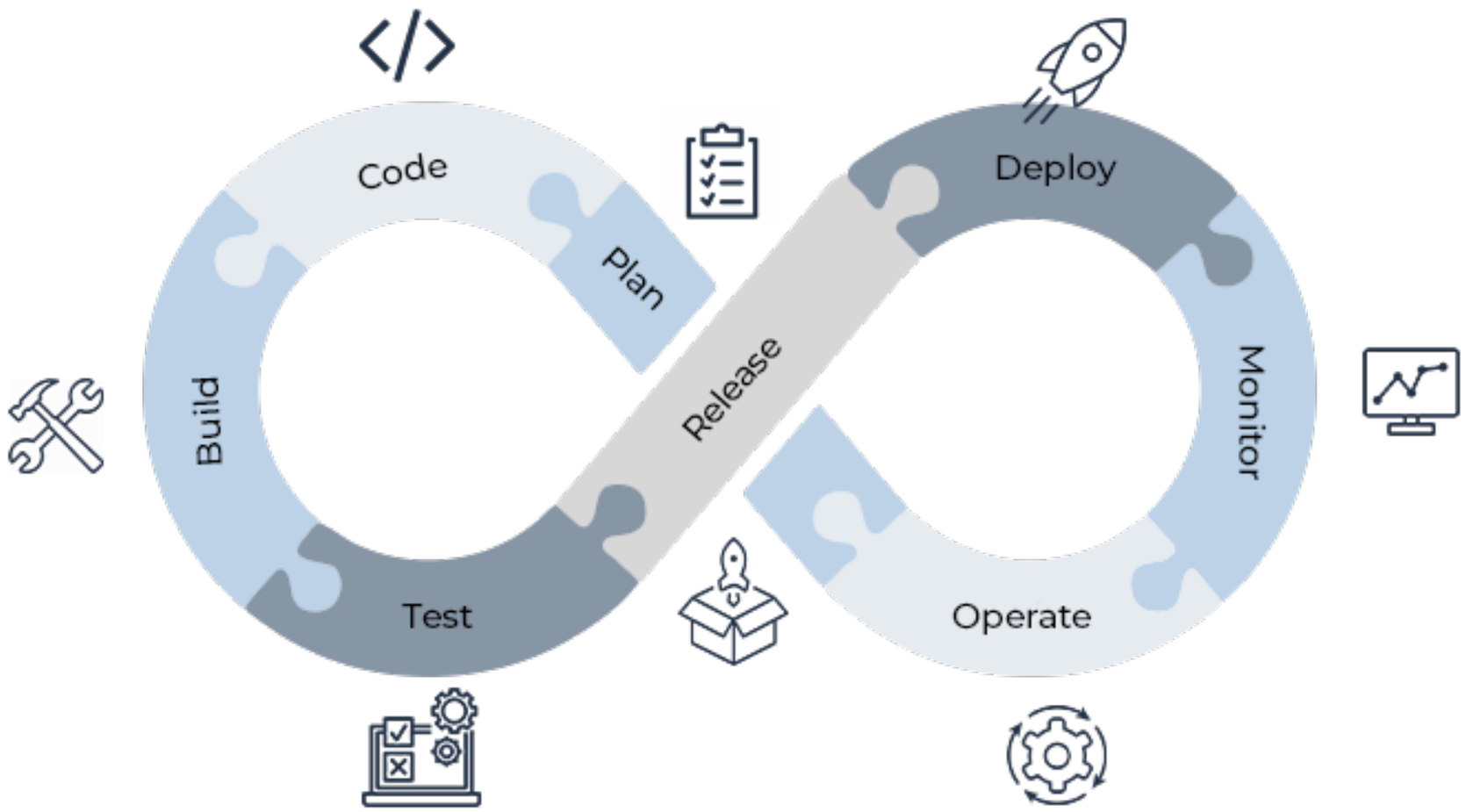
Automated Testing Lifecycle



Continuous Integration

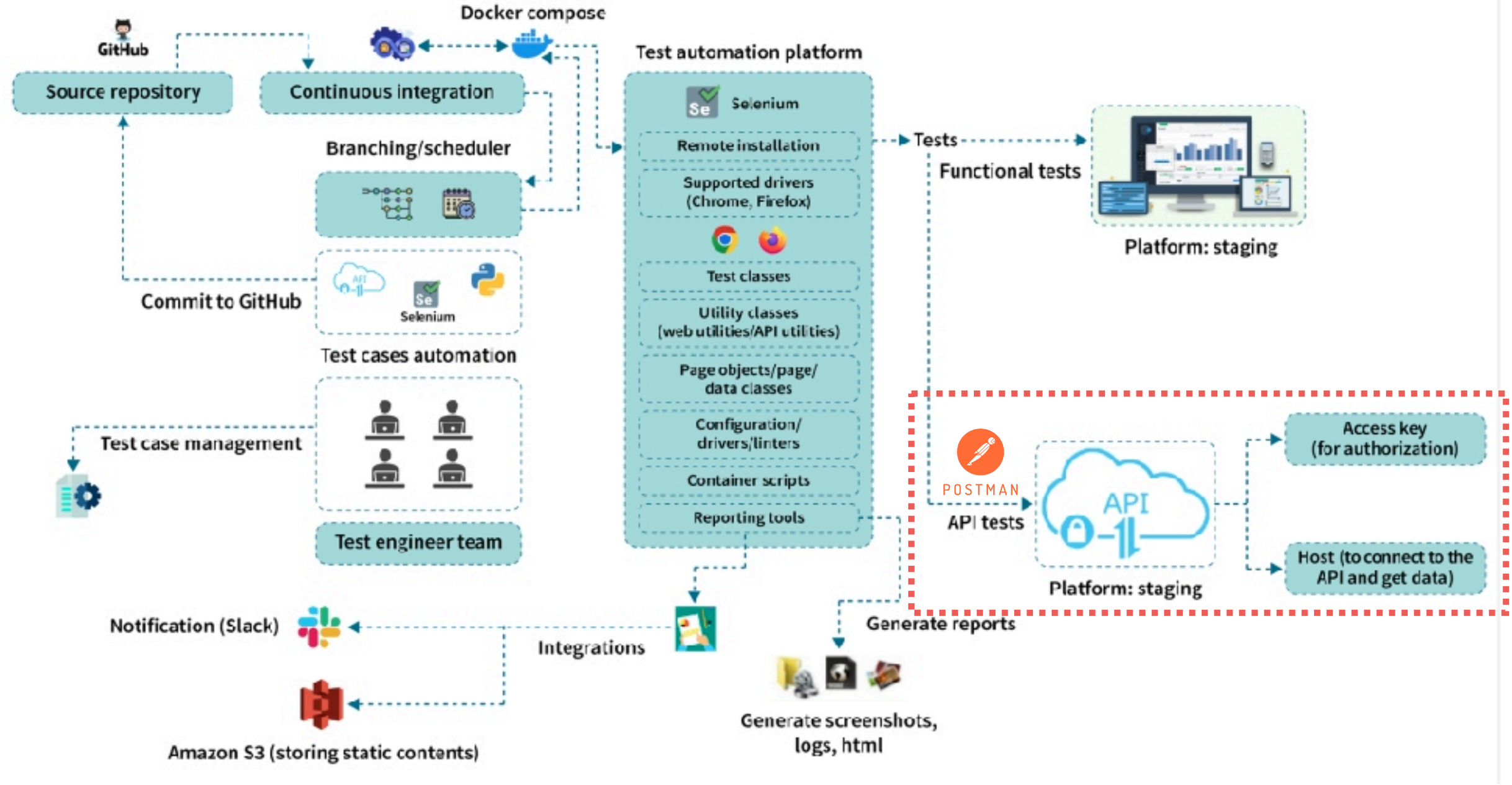
Continuous Testing

DevTestOps



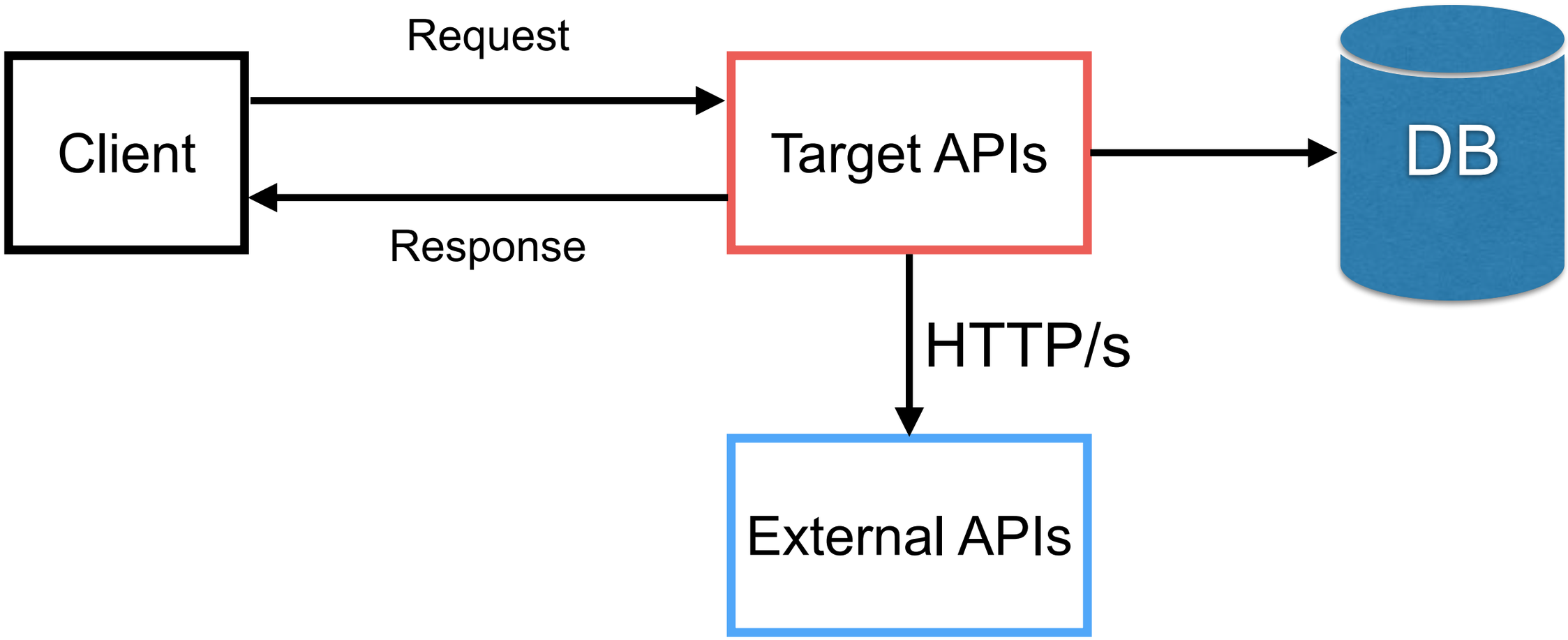
https://www.xenonstack.com/insights/devtestops-advantages

Example

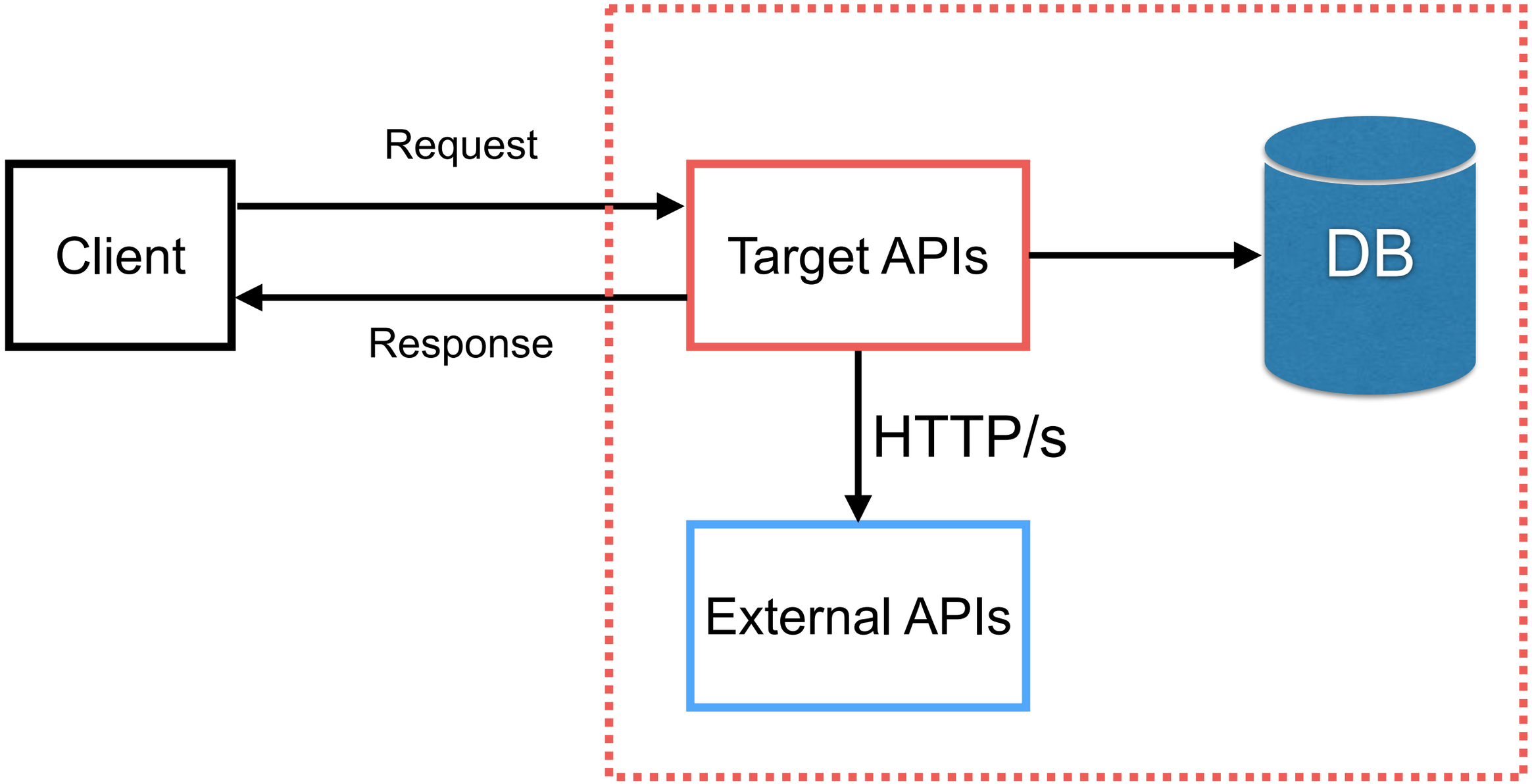


Test Strategies ?

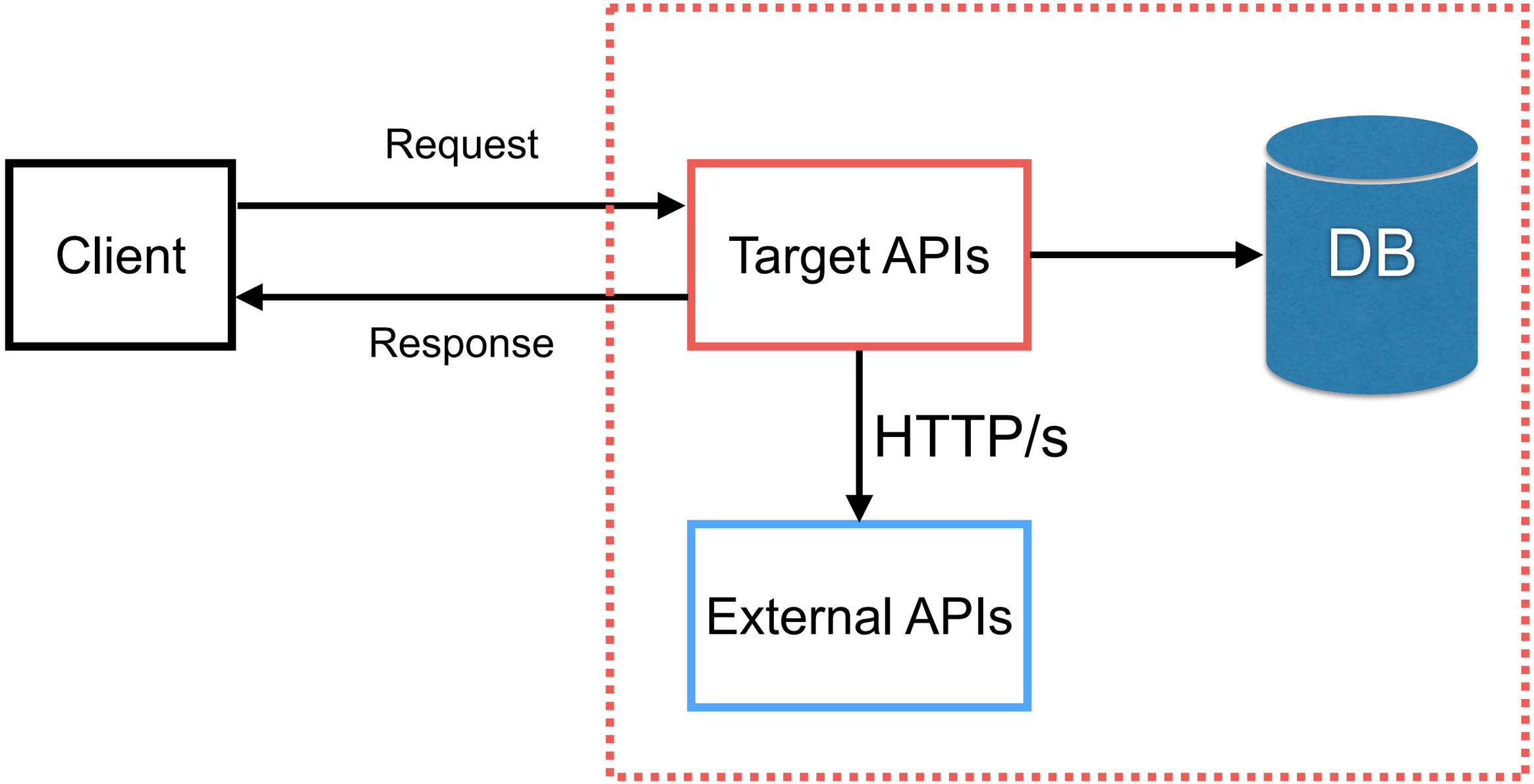
Architecture



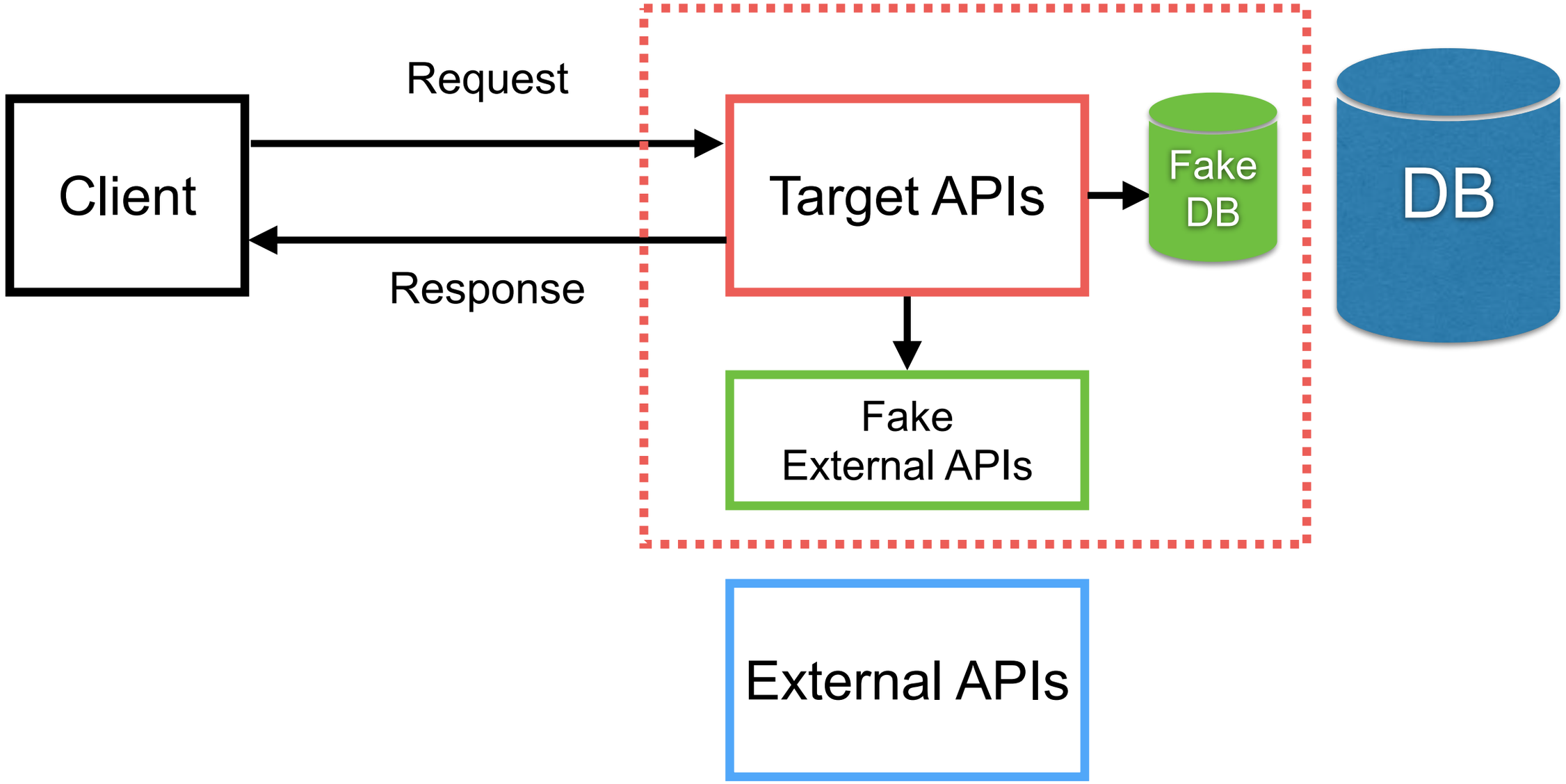
What and Why to test ?



Testing with real dependencies

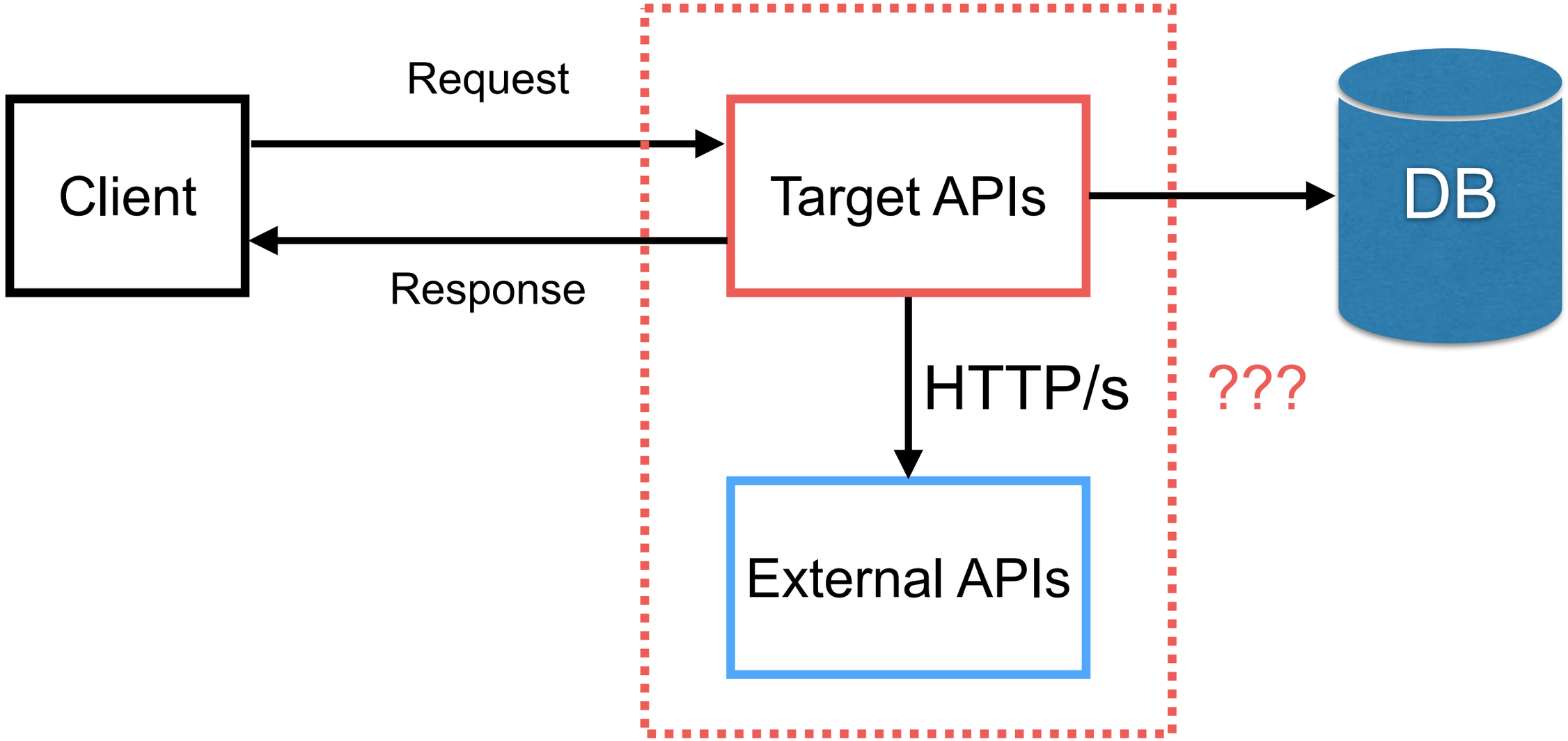


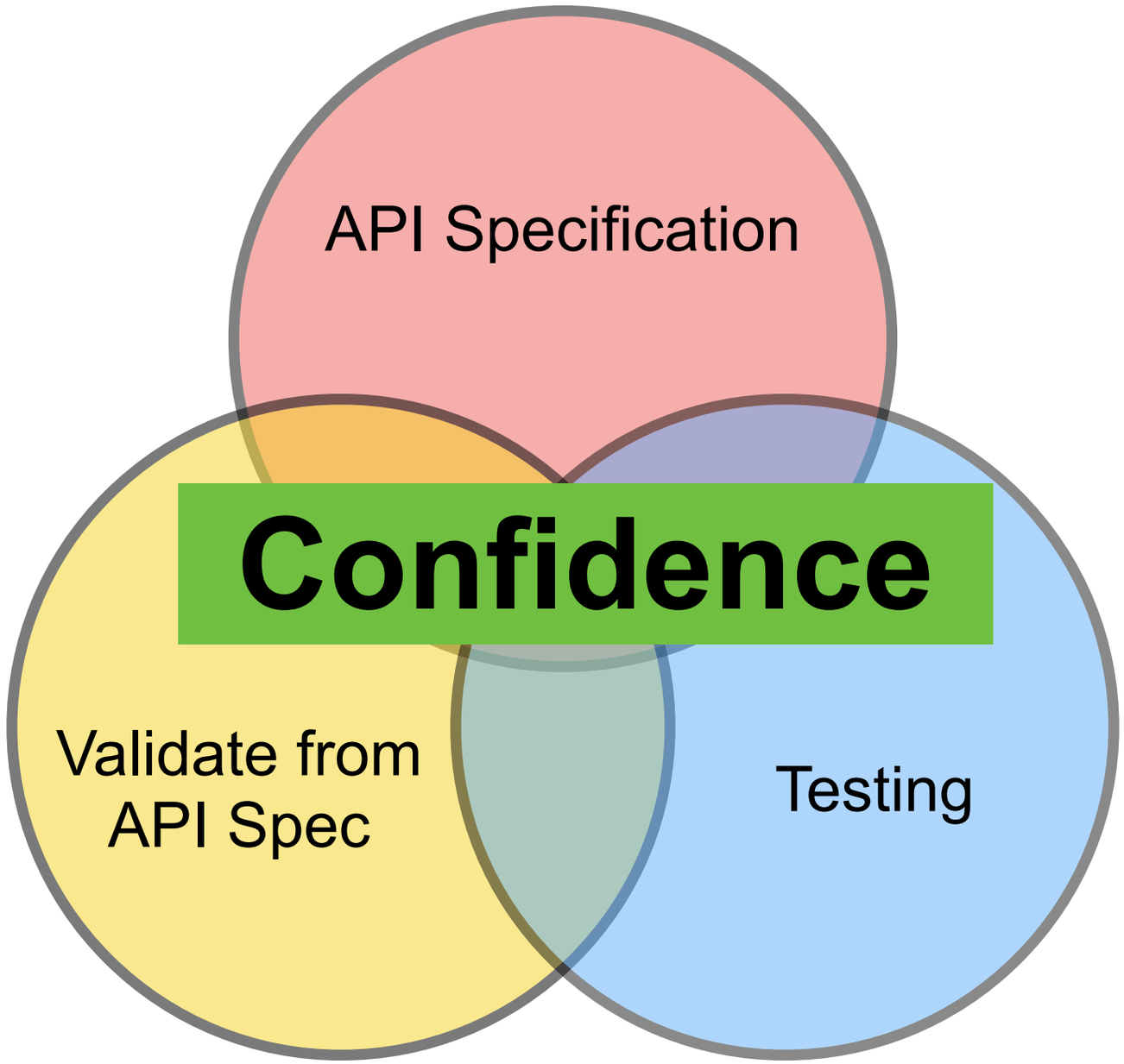
Testing with fake dependencies



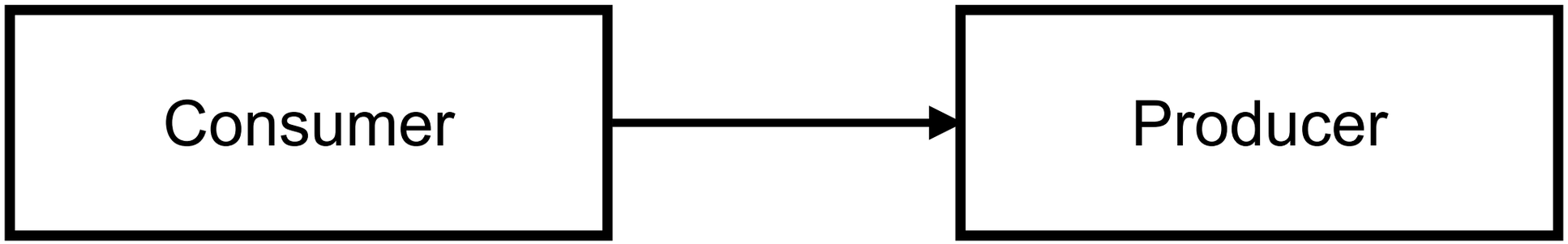
Contract Testing

Contract Testing ?





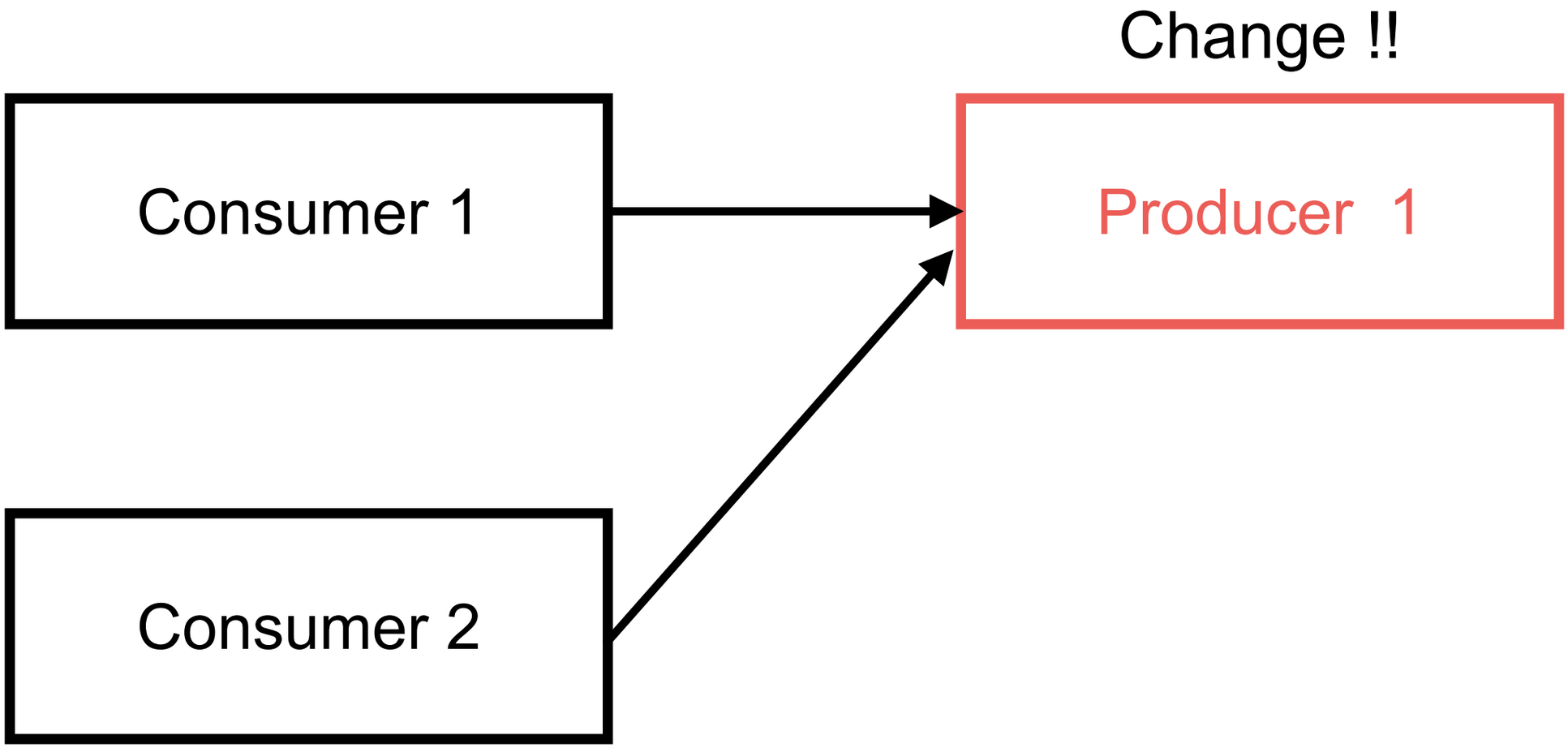
Testing ?



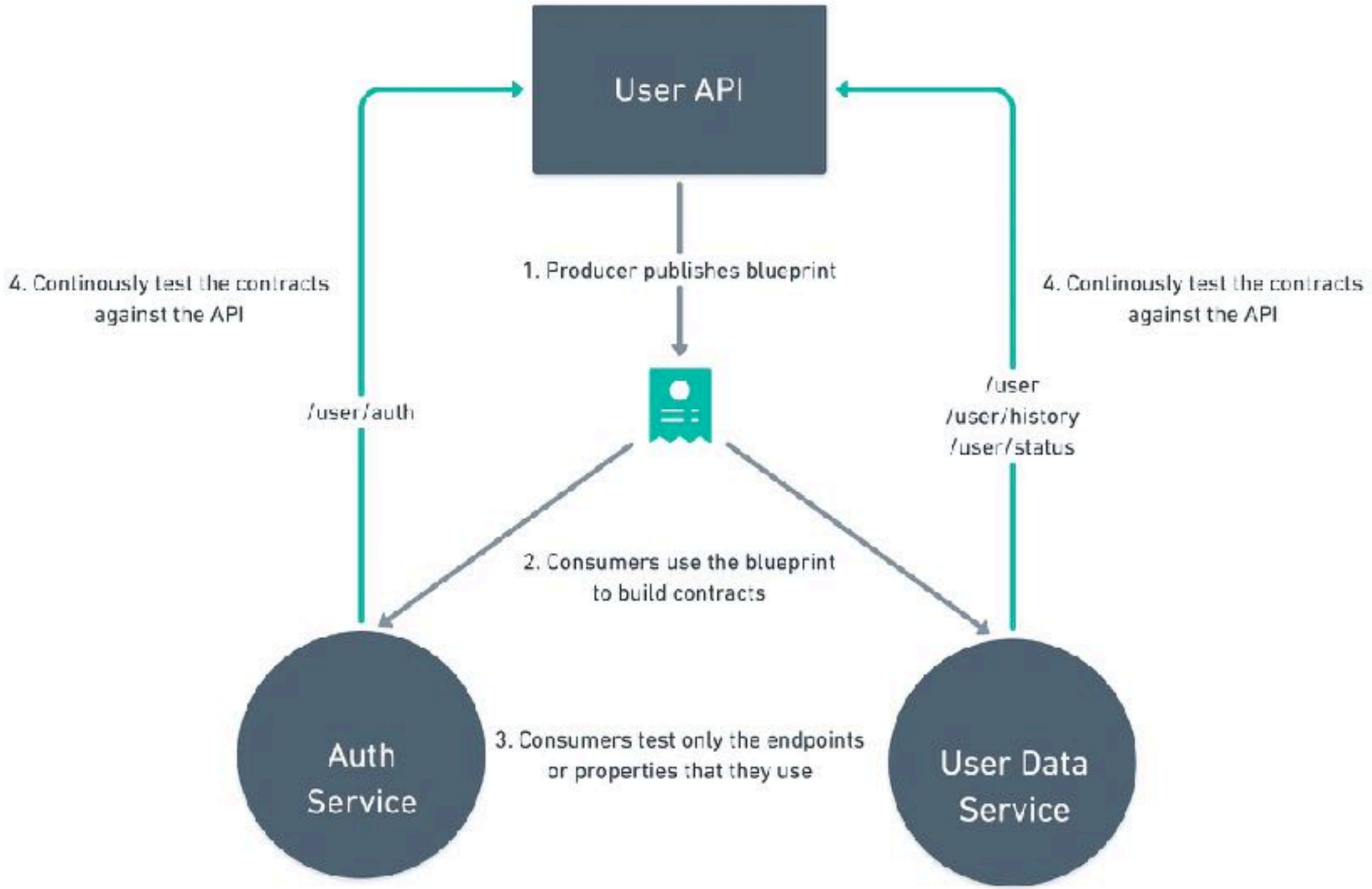
How to call/use API ?

How consumer use my API ?

Testing ?



Example



https://medium.com/better-practices/consumer-driven-contract-testing-using-postman-f3580dba5370

Workshop

https://postman-echo.com

1. Producer publish blueprint

Design API Specification
Validate schema

API
GET /get?key1=value1&key2=value2
POST /post key1=value1 key2=value2

2. Consumer create contract

Call APIs from provider
Define inputs and expected results
Create requests
Write test scripts to verify

3. Mock provider in consumer-side

Easy to test and develop in consumer-side
Define inputs and expected results

More testing !!

Snapshot testing

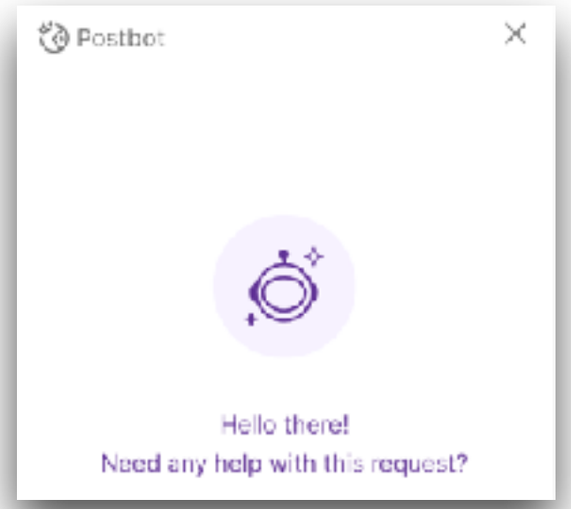
Performance testing

Security testing

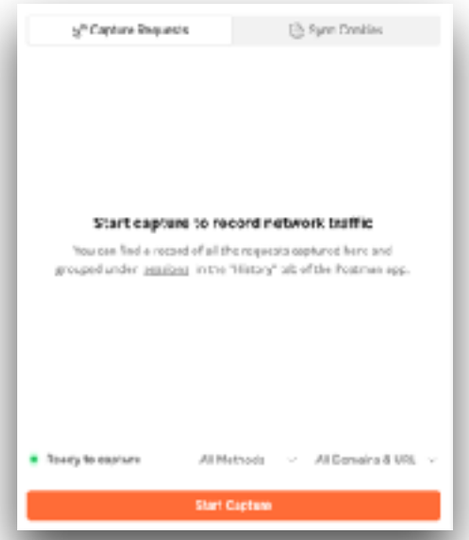
More features

POSTMAN

More features

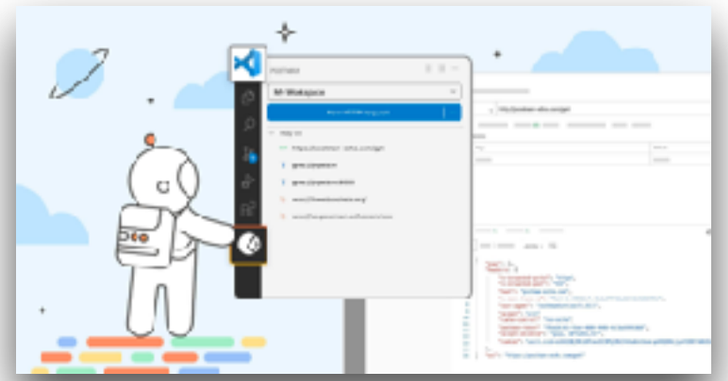


AI assistant for API testing



Postman interceptor

Working with OpenAPI/Swagger



Postman extension in VSCode

Summary

Better delivery process

Think about testing
Fast feedback
High quality
Working as a team

Q/A

https://www.facebook.com/beSkilledSoftwareTester