# POSTMAN COLLECTIONS

**WHAT TO EXPECT**

PROGRESS
- NOT -
Perfection

# What Are Postman Collections?

A Postman Collection is a structured group of API requests that can be executed sequentially or independently.

- Organize related requests into folders (test cases).

- Build comprehensive test suites (e.g., Smoke, Regression, Functional).

- Enable collaboration with team members.

# Structuring Test Cases and Test Suites

### 1. Create Test Cases

- Group related API requests into folders (e.g., Login APIs, User Management).
- Add descriptive names for clarity.

### 2. Build Test Suites

- Create folders as "Suites" (e.g., Smoke Tests, Regression Tests).
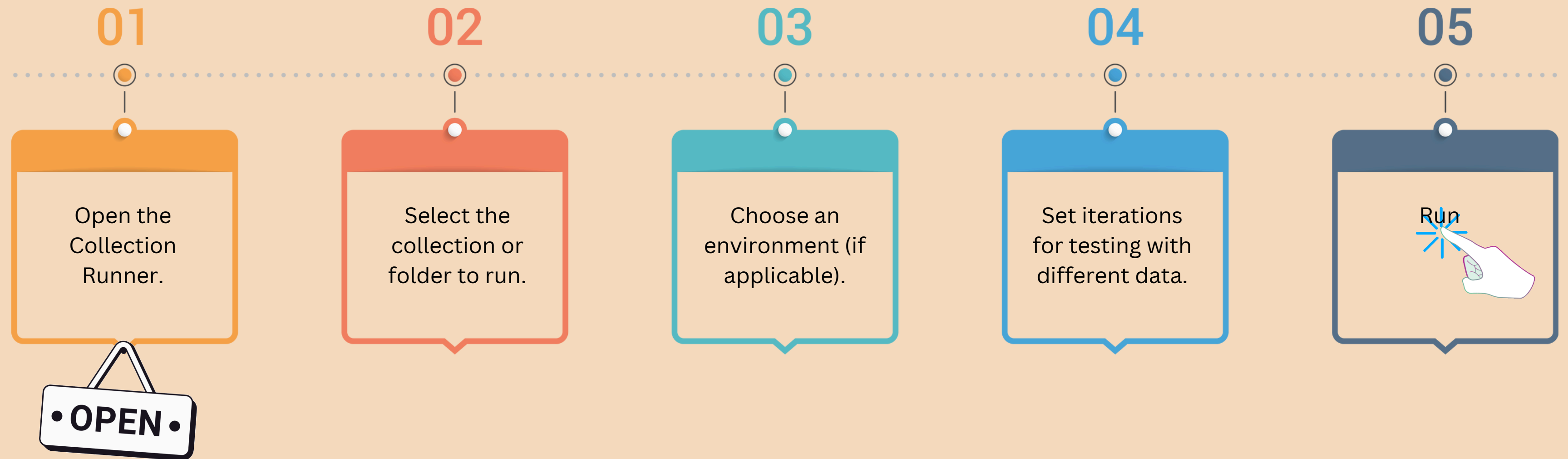- Reuse requests across multiple suites using duplication

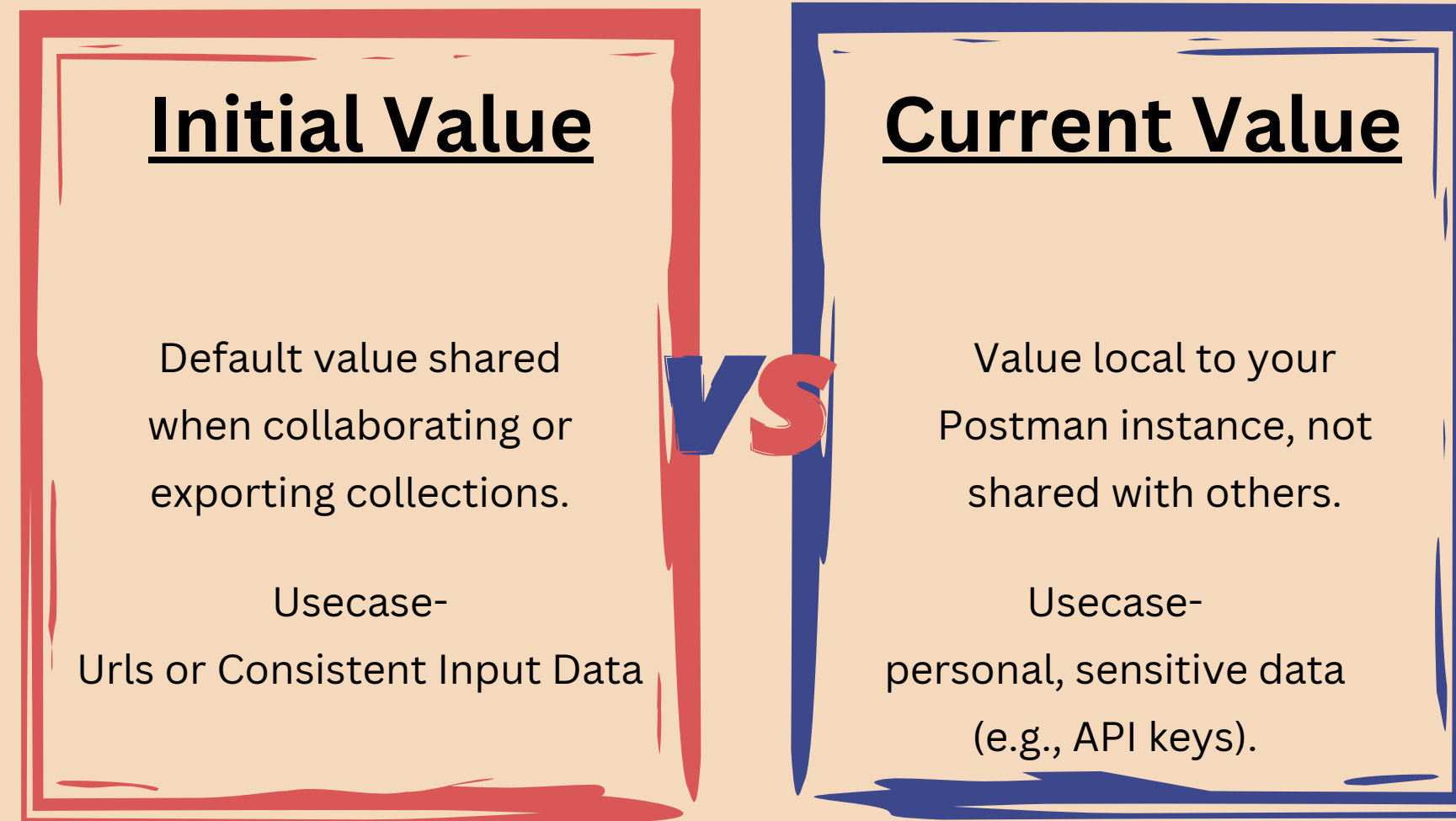### 3. Add Tests to Requests

- Use the Tests tab for assertions.

**Why?**

Organizing test cases and suites in collections ensures scalability. You can group similar requests, reuse them, and add tests directly to each request to validate the API's behavior.

# Automating Test Suites with the Collection Runner

**01**

Open the Collection Runner.

OPEN

**02**

Select the collection or folder to run.

**03**

Choose an environment (if applicable).

**04**

Set iterations for testing with different data.

**05**

Run

# Initial vs Current Values in Variables

## Initial Value

Default value shared when collaborating or exporting collections.

Usecase-
Urls or Consistent Input Data

**VS**

## Current Value

Value local to your Postman instance, not shared with others.

Usecase-
personal, sensitive data (e.g., API keys).

https://learning.postman.com/docs/sending-requests/variables/variables/
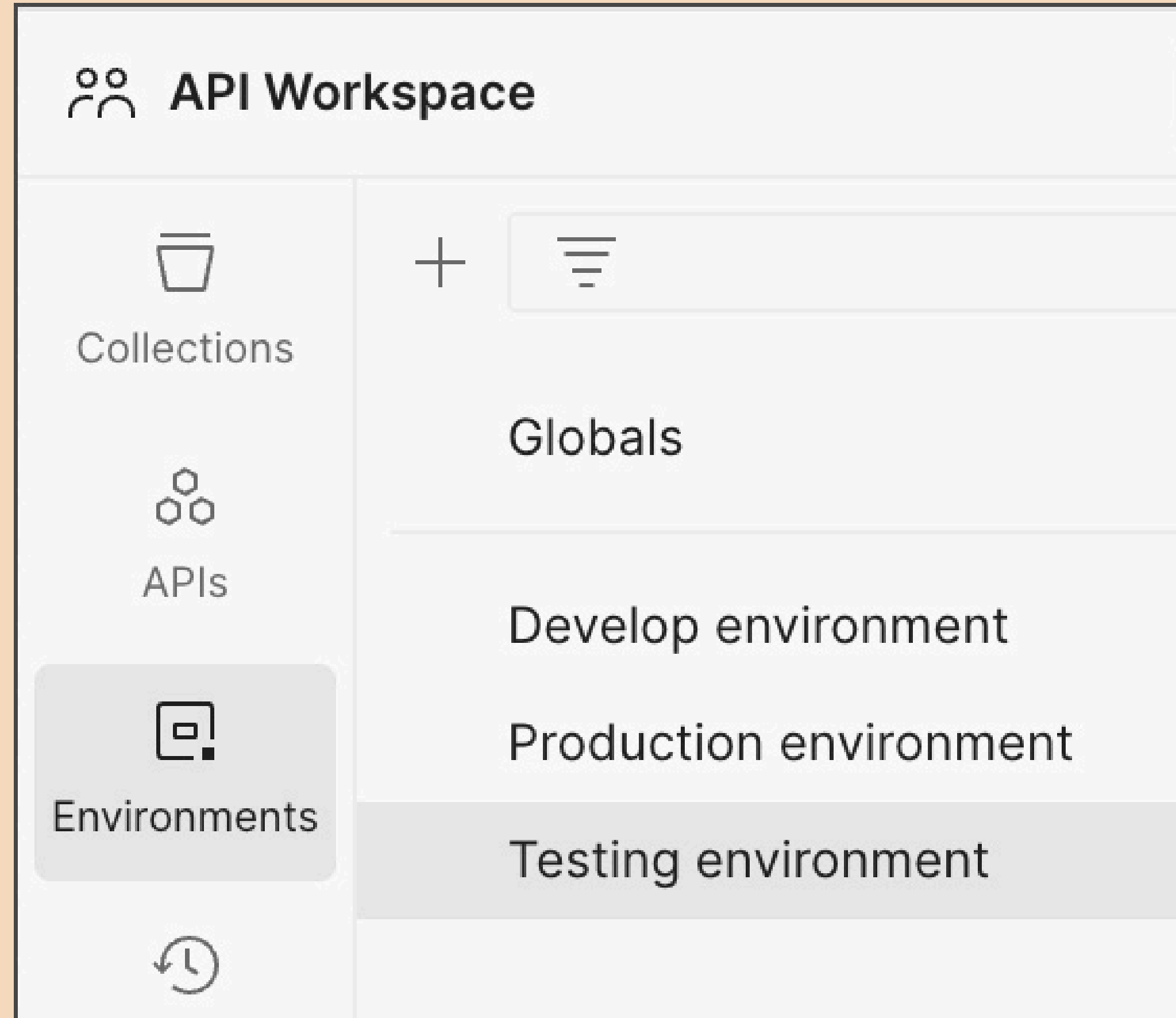
# Leveraging Environments for Flexibility

What Are Environments?

A collection of variables for specific setups (e.g., Development,

Staging, Production).

Benefits:

1. Switch between environments without changing requests.

2. Use variables like {{baseUrl}} for dynamic endpoints.

## API Workspace

Collections

APIs

Environments

Globals

Develop environment

Production environment

Testing environment

# Data-Driven Testing in Postman

**Why Use Data Files?**

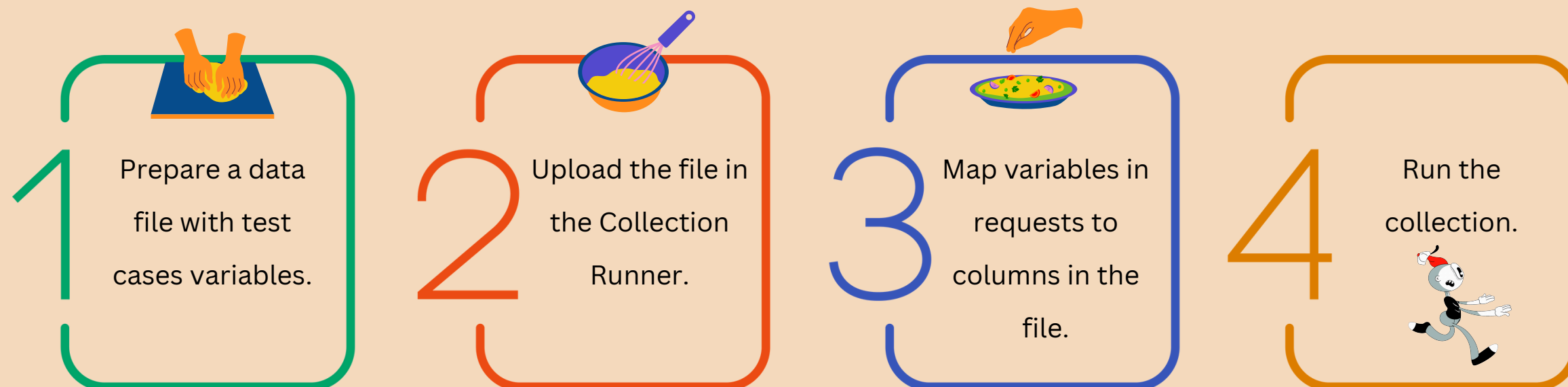Test multiple scenarios with varying inputs (e.g., different user credentials).
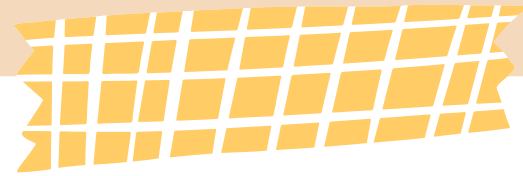
**Supported File Types:**

CSV, JSON.

| username | password |
|----------|----------|
| user1    | pass1    |
| user2    | pass2    |

```json
[
  {
    "username": "user1",
    "password": "pass1"
  },
  {
    "username": "user2",
    "password": "pass2"
  }
]
```

**1** Prepare a data file with test cases variables.

**2** Upload the file in the Collection Runner.

**3** Map variables in requests to columns in the file.

**4** Run the collection.

Lets see it 

See you in Next Video