Automation Testing

Introduction to API Testing

A Day in the Life of an Automation Testing Engineer

John works at SunPro Infosystems as an Automation Testing Engineer and is familiar with functional testing of the company's systems. He is now assigned to test the APIs of the new application in development.

After completing this session, John will be able to use Postman to set up, configure, and test APIs.

Learning Objectives

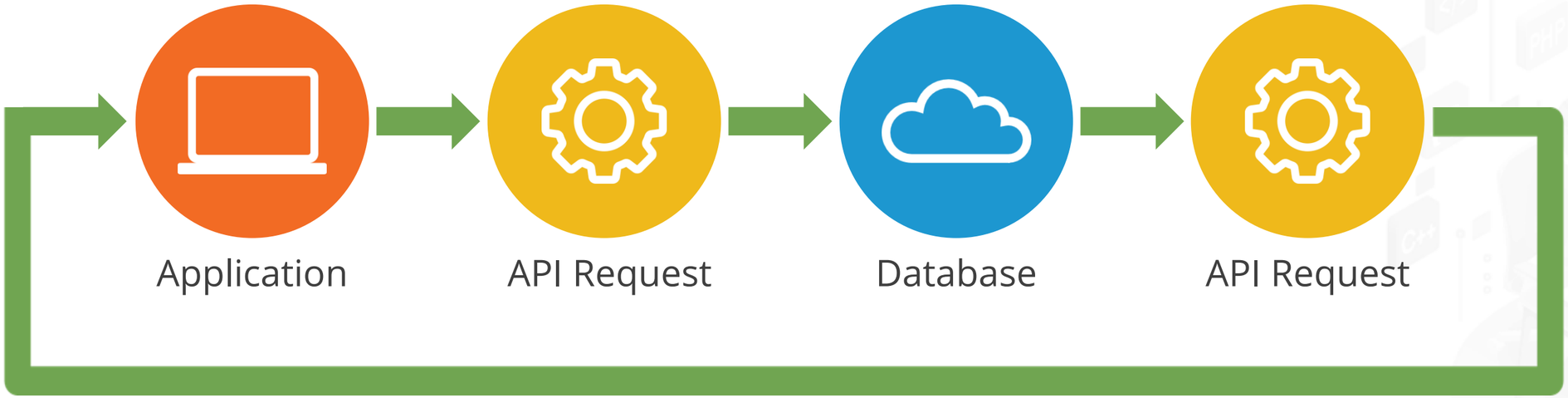
By the end of this lesson, you will be able to:

- Identify API Testing
- List various tools used for API Testing
- Explain the use of Postman
- State Postman GUI
- List the steps to create different API requests in Postman
- Explain how SOAP works with Postman

What Is API Testing?

What Is API Testing?

API stands for Application Programming Interface.
API is the middle layer between the application and the database.



API testing evaluates the APIs for security, reliability, and accuracy.

What Is API Testing?

A data layer, a service layer (the API layer), and a presentation layer (user interface layer) are the three layers that most applications contain.

API testing examines the business logic and the security of the application and data replies.

The API layer contains the application's business logic. More precisely, it describes how users can interact with the app's services, functions, and data.

An API test is typically carried out by sending requests to one or more API endpoints and comparing the results to what is expected.

How to Approach API Testing?

API testing should start with a clearly defined program scope and a thorough understanding of how the API performs.

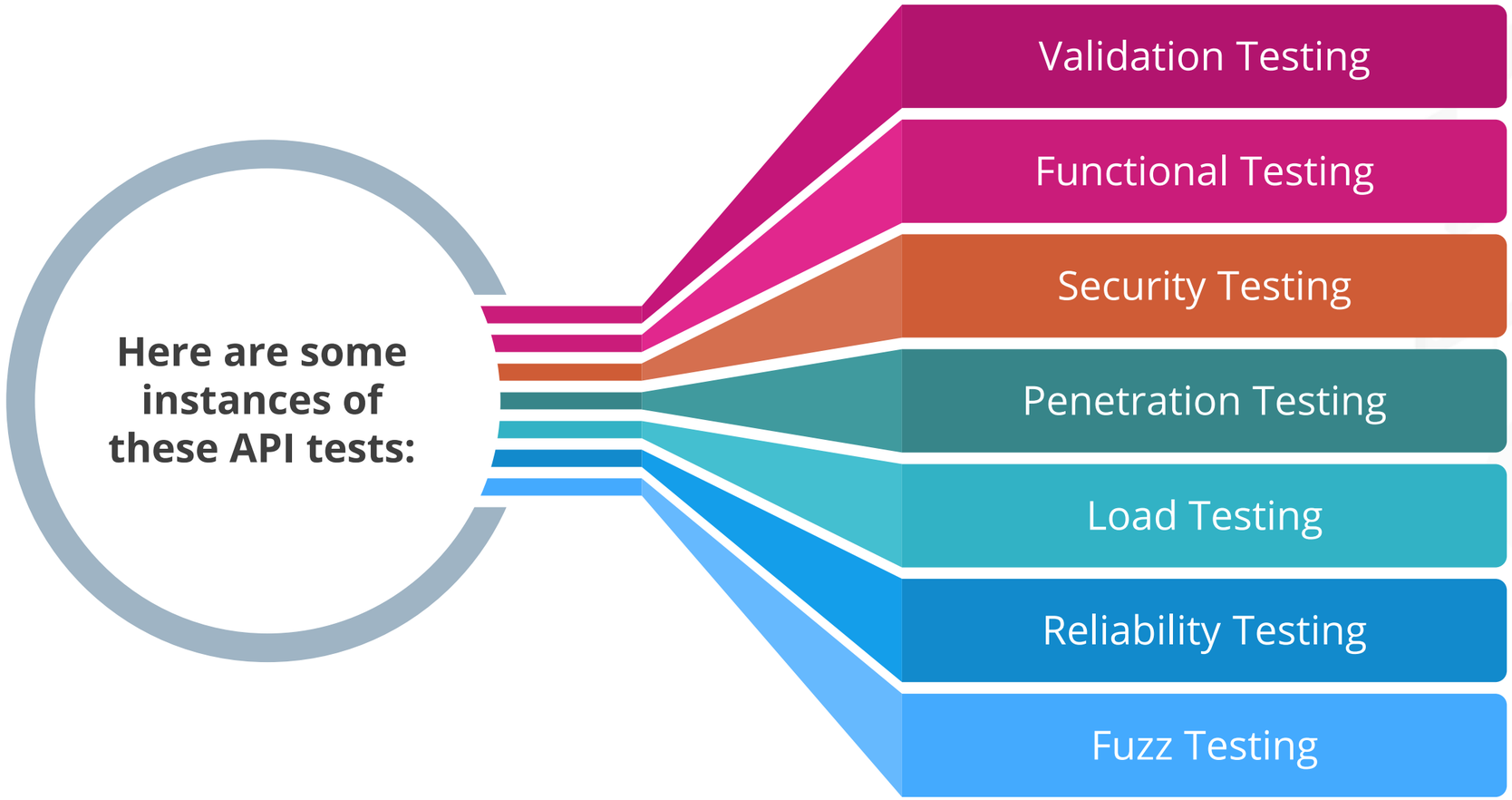
The following are some questions that testers should think about:

- What testing endpoints are available?
- What are the anticipated response codes for successful requests?
- What are the response codes for unsuccessful requests?
- What error message pops up in the body of a failed request?

Types of API Testing

Types of API Testing

Users can perform several API tests to confirm that the API is functioning correctly. These tests range from broad to detailed examination of the software.



Validation Testing

Validation testing consists of a few simple queries covering the project's entirety.

- Was the right product created?
- Is the developed API the best solution for the problem it is attempting to solve?
- Was there any severe code bloat throughout development that would make the API unsustainable?

- Is the correct data accessed the way it should be?
- Is excessive amount of data accessed?
- Is the data stored correctly through the API after considering the data set's specific integrity and confidentiality needs?

- Is this API the most efficient and precise way to complete a task?
- Is it possible to change or remove any codebase to improve overall service and eliminate impairments?

Functional Testing and Load Testing

Functional Testing

Functional testing guarantees that the API does what it is designed to do.

This test examines certain functions in the codebase to ensure that the API operates within its expected parameters and can handle errors if the results fall outside of those parameters.

Load Testing

Load testing is a technique for determining how many requests an API can handle.

This test is frequently carried out after a specific unit, or the entire codebase has been completed to see if the theoretical answer can also operate as a practical solution when working under a specified load.

Reliability Testing and Security Testing

Reliability Testing

Reliability testing guarantees that the API produces consistent results and maintains a constant connection between platforms.

Security Testing

In the more extensive security auditing process, security testing is frequently lumped in with penetration and fuzz testing.

Security testing combines penetration and fuzz testing elements, as well as attempts to check the API's encryption mechanisms and access control design.

The validation of permission checks for resource access and user rights management is part of security testing.

Penetration Testing and Fuzz Testing

Penetration Testing

Penetration testing is a step forward from security testing.

An individual with insufficient knowledge of the API attacks the API in this test. It allows testers to look at the attack vector from a different angle. Penetration testing attacks might be limited to certain aspects of the API, or they can target the API as a whole.

Fuzz Testing

Fuzz testing involves forcing large amounts of random data into a system, often known as noise or fuzz, to induce undesired behavior such as a caused crash or overflow.

API Testing Best Practices

Best Practices of API Testing

- Group test cases into categories while defining them
- Include the parameters chosen in the test case
- Create test cases for each API input combination for test coverage
- Reuse and repeat test cases to monitor the API during production

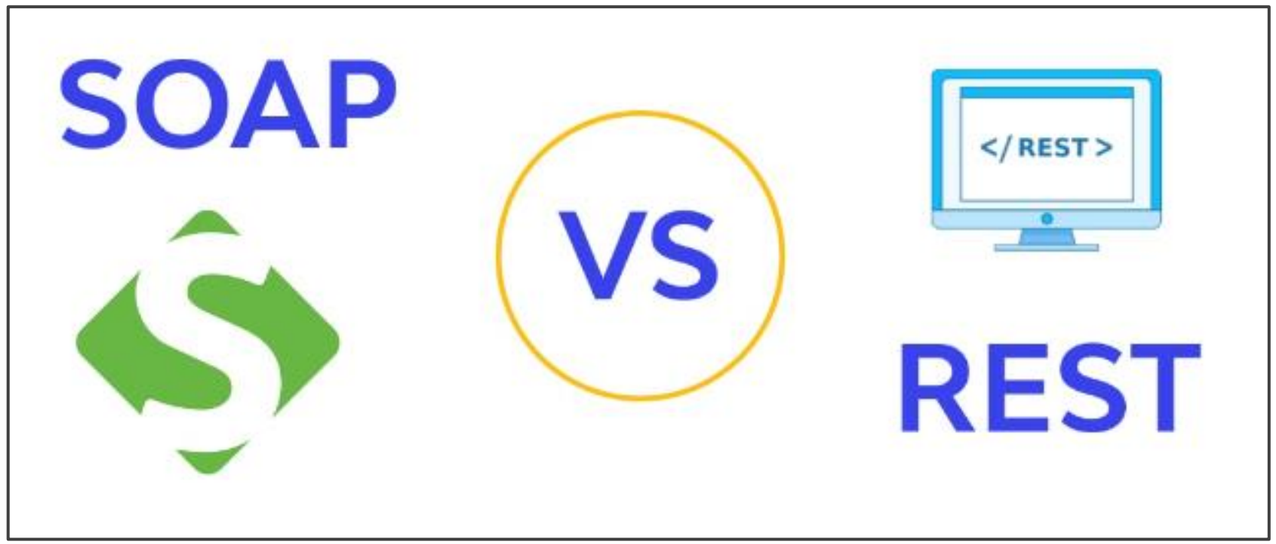
- Sequence calls according to a well-thought-out strategy
- Run the tests until a failure output is obtained
- Conduct API load tests to test the stress on the system
- Note regular and irregular activities during API testing

- Combine manual and automated testing to achieve reliable findings

Rest and Soap API

What Is REST and SOAP API?

REpresentational State Transfer (REST) and Simple Object Access Protocol (SOAP) are different types of API architecture. Both the APIs serve different purposes.



REST requests are written in JSON, whereas SOAP requests are in XML.

API Testing Tools

Postman

What Is Postman?

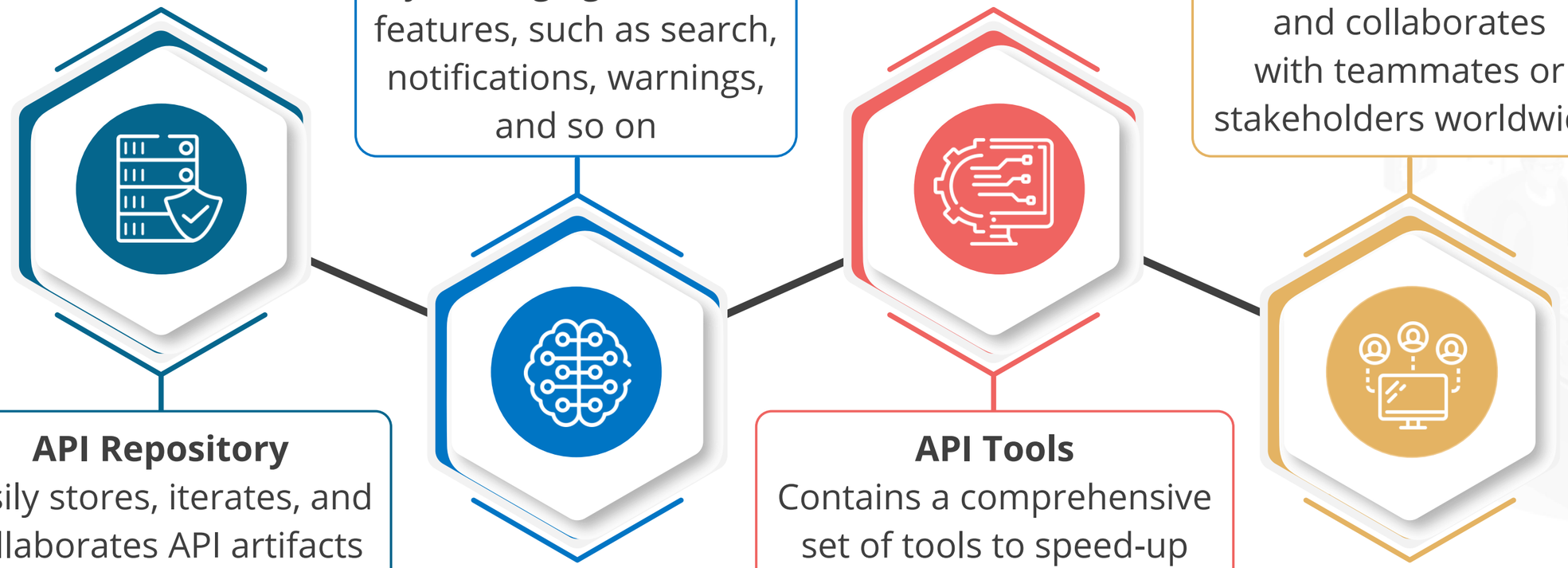
Postman is a tool used to build and test APIs. It simplifies each step of the API lifecycle and streamlines collaboration so that users can create efficient APIs faster.

POSTMAN

Why Use Postman?

Postman is a comprehensive tool for working with APIs. It is an HTTP client that uses a graphical user interface to test HTTP queries, allowing users to get various forms of responses that need to be validated.

Why Use Postman?



API Repository
Easily stores, iterates, and collaborates API artifacts on one central platform

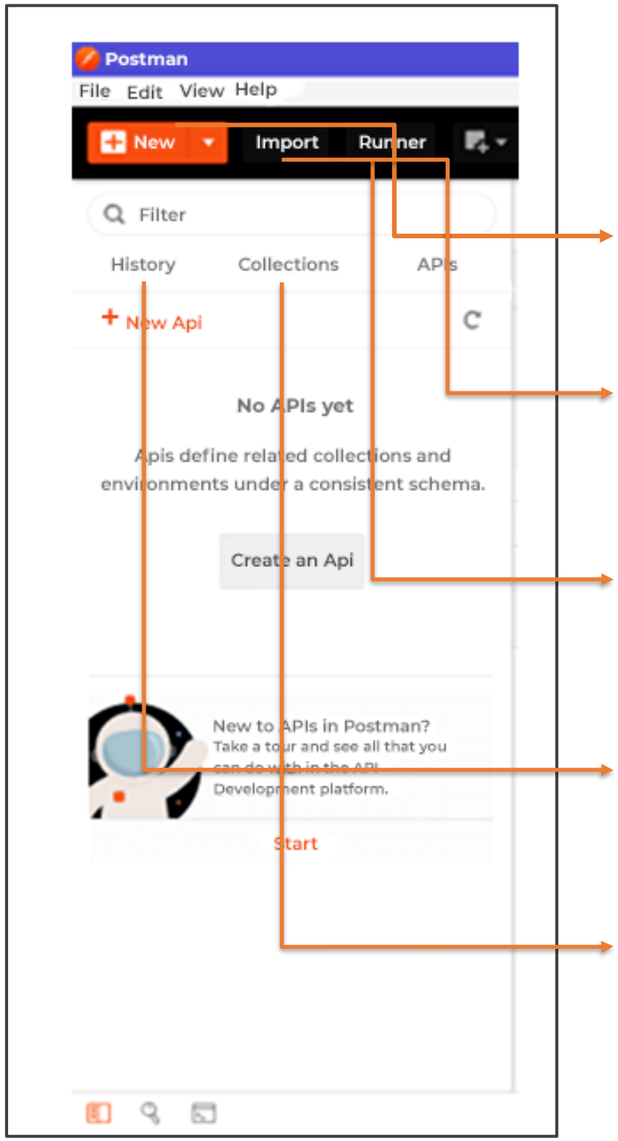
Intelligence
Improves API operations by leveraging advanced features, such as search, notifications, warnings, and so on

API Tools
Contains a comprehensive set of tools to speed-up API lifecycle

Workspaces
Organizes API work and collaborates with teammates or stakeholders worldwide

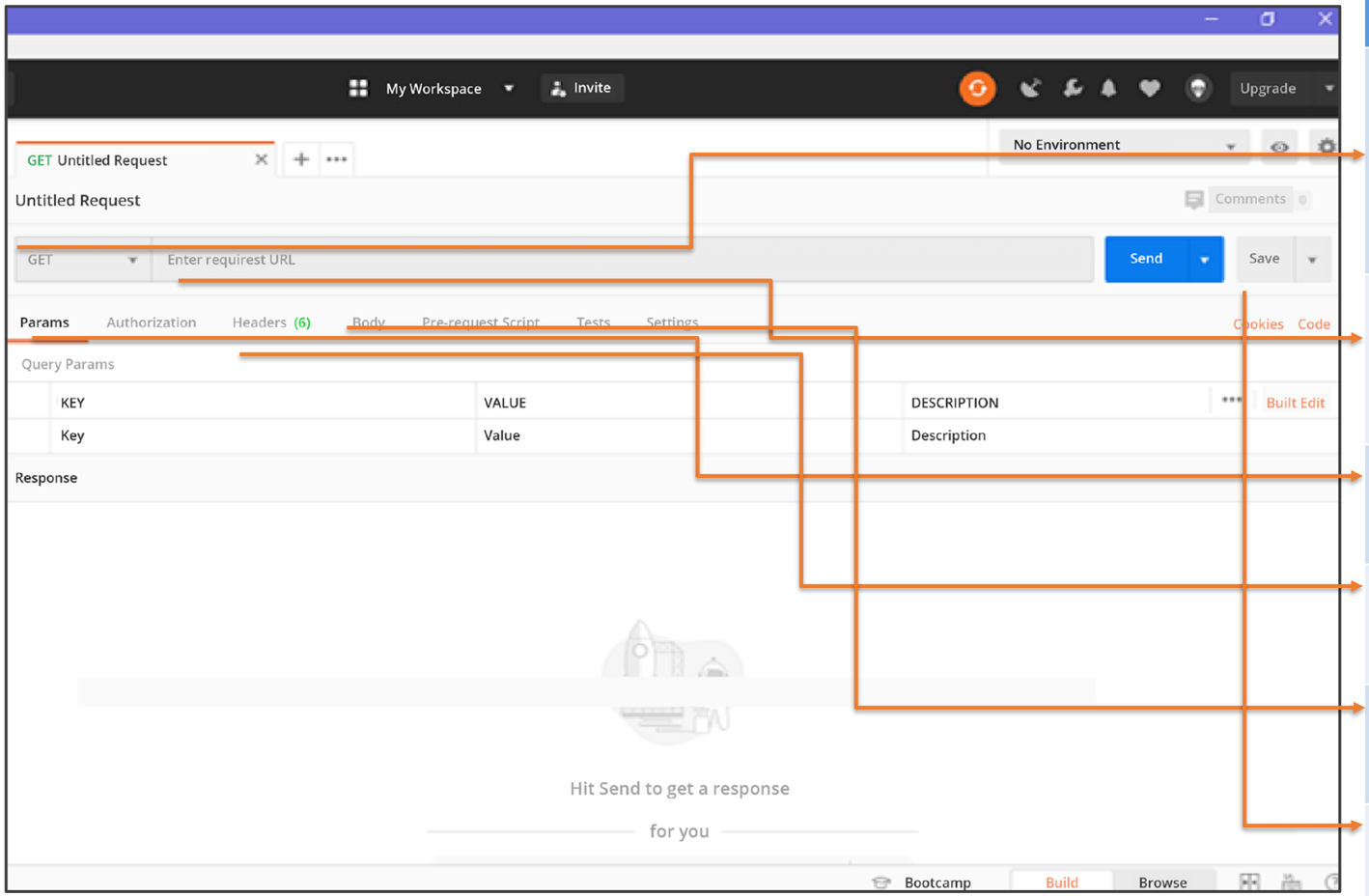
Understanding Postman GUI

Understand Postman GUI: Project Explorer



Element	Use
New	Creates a new request, collection, or environment
Import	• Imports from a file, folder, or link and paste the raw data • Brings a 'collection' or an 'environment' into the program
Runner	Uses Collection Runner to run automation tests
History	Displays API requests sent in the past
Collections	• Creates Collections to organize a test suite • Creates subfolders for Collections • Accommodates many requests in each collection • Duplicates a request or a folder

Understand Postman GUI: Editor



Element	Use
HTTP request option	Lists menu for different requests, such as Get, POST, and so on
Enter request URL	Allows users to enter the endpoints (URL)
Params	Allows users to enter key values that an API needs
Header	Allows users to enter request headers
Body	Allows users to add details in a request
Save	Allows users to save changes to a request

Basics Before Creating the First API Request

Basics Before Creating the First API Request

Here are some basic terms to know before hitting the first API:

Term	Meaning
API	Application Programming Interface (API) is a software middleman that allows two applications to communicate with one another. Users can utilize an API every time they use an app like Facebook, send an instant message, or check the weather on their phone.
Request	When users send a command, the server accepts it and reciprocates with some information. This command is known as a request.
Response	A response is a message that a client sends to the server in response to a user's request.

Basics Before Creating the First API Request

Why do we need API?

A mobile phone application connects to the Internet and sends data to a server when users use it. The server then retrieves the data, interprets it, and delivers it back to their phone with the relevant actions. The software then analyzes the data and displays the information they had requested legibly. All this happens through API.

Why do we need Postman?

Postman is an API client for developers that makes it simple to create, share, test, and document APIs. To do this, users can construct and save simple and sophisticated HTTP/s queries and view their responses. As a result, work is more efficient and less tiresome.

Get, POST and Parameterized Request

Get, POST, and Parameterized Requests

A GET Request

A GET request retrieves the required information from the server and does not alter the data on the server.

A POST Request

POST is an HTTP method like GET. When users submit a POST request, the server updates, removes, or inserts data.

Get, POST, and Parameterized Requests

Users can parameterize the GET and POST requests by adding data in the request body or the URL. There are two types of parameters:

Types	Description	Example
Query parameters	They are appended to the end of the request URL. After the '?' syntax, they are listed as key-value pairs separated by the '&' syntax.	?id=1&type=new
Path parameters	They are placeholders in the request URL before the colon ':'.	/customer/:id

How SOAP Works with Postman?

SOAP

Simple Object Access Protocol, or SOAP is a messaging protocol for computer communication based on XML.

Few noteworthy points:

1. It extends HTTP for XML Messaging
2. It is language & platform independent
3. A message can be broadcast using SOAP
4. Other frameworks, such as CORBA, DCOM, and Java RMI, offer comparable features to SOAP, but since SOAP messages are exclusively expressed in XML, they are completely platform- and language-independent

Key Takeaways

- Postman is an all-in-one API development platform with a variety of built-in features.
- Postman uses GUI to test HTTP queries.
- GET and POST are the most commonly used HTTP methods.
- Query parameter and path parameter are the two types of parameterized requests.
- REST and SOAP are two different types of API architecture.