

Postman for API Testing – Beginner to Pro

Subtitle: Learn to test, debug, and manage APIs efficiently using Postman, without writing backend code.

Website Name: haas.dev

Website Link: <https://dev-roast-app.vercel.app>

Introduction

APIs are the backbone of modern web and mobile apps. Postman allows developers to **test requests, view responses, and debug APIs** without setting up a frontend or backend. This guide teaches beginners how to use Postman like a pro.

Step 1: Install Postman

- Download: <https://www.postman.com/downloads/>
 - Available for Windows, Mac, Linux
-

Step 2: Understanding Requests

- **GET:** Retrieve data from server
- **POST:** Send data to server
- **PUT / PATCH:** Update existing data
- **DELETE:** Remove data

Mini example:

GET <https://jsonplaceholder.typicode.com/todos/1> → Returns a sample task

Step 3: Sending Requests

1. Open Postman → New Request
 2. Enter URL and select HTTP method
 3. Click **Send** → View response
 4. Check **Status code**, **Response Body**, and **Headers**
-

Step 4: Working with Parameters & Body

- **Query parameters:** ?id=1
- **Body:** Used in POST/PUT requests
- **JSON format example:**

```
{  
  "title": "Learn Postman",  
  "completed": false  
}
```

Step 5: Collections & Environments

- **Collections:** Save multiple requests in one folder
- **Environments:** Store variables like baseUrl, apiKey
- Use {{variableName}} in request URLs for dynamic testing

Step 6: Testing & Automation

- Add **Tests** in Postman using JS syntax

```
pm.test("Status is 200", function () {  
  pm.response.to.have.status(200);  
});
```

- Run **Collection Runner** to test multiple requests at once

Step 7: Mini Project Idea

- Test a free API (e.g., JSONPlaceholder)
- Create a collection with:
 - GET all todos
 - POST a new todo
 - PUT an existing todo
 - DELETE a todo
- Add tests for each request and run collection

Step 8: Best Practices

- Always validate API responses before integrating
 - Use environments for multiple projects
 - Organize requests in collections for clarity
 - Add descriptive names and documentation for teammates
-

Step 9: Key Takeaways

- Postman simplifies API testing and debugging
 - Learn GET, POST, PUT, DELETE requests thoroughly
 - Collections & Environments improve workflow
 - Practice with real APIs to build confidence
-

Visit **haas.dev** for step-by-step Postman tutorials, API projects, and beginner-friendly web/mobile dev resources.

Website Name: haas.dev

Website Link: <https://dev-roast-app.vercel.app>
