

Table of Content

[Check Status Code 404](#)

[Use Snippets](#)

[Check Status Code 201 - Submit an Order](#)

[Check Status Code 200 - Get an Order](#)

[Check Status Code 204 - Delete an Order](#)

[Submit an Order](#)

Check Status Code 404

Add script to the request, change Body to Test Result

The screenshot shows the Postman application interface. On the left, there's a sidebar with a search bar and a 'Request Collection' section containing several API endpoints:

- GET Status
- GET List of Books
- GET List of Books by Query
- GET Book by ID
- POST Register API Client
- POST Submit Order
- PATCH PATCH Update Order
- GET GET Order** (highlighted with a red arrow)
- DEL DELETE Order

The main panel shows a 'Request Collection / GET Order' with a 'GET' method and a URL template: {{BaseUrl}}/orders/:orderId. The 'Scripts' tab is selected, showing a 'Post-response' script block:

```
1 pm.test("Status code is 404", function () {  
2 | pm.response.to.have.status(404);  
3});
```

Below the script, the 'Body' dropdown is open, showing a JSON response:

```
{ } JSON ▾
```

The response body is:

```
1 {  
2 | "error": "No order with id cuYLSe0ZIAEqMrNB3090r."  
3}
```

At the bottom of the interface, there are status indicators: 404 Not Found, 698 ms, 798 B, and Save Response.

HTTP Request Collection / GET Order

Save Share

GET {{BaseUrl}}/orders/:orderId

Send

Docs Params Authorization Headers (8) Body Scripts Settings Cookies

Pre-request:

```
1 pm.test("Status code is 404", function () {  
2 | pm.response.to.have.status(404);  
3});
```

Post-response:

Packages Snippets

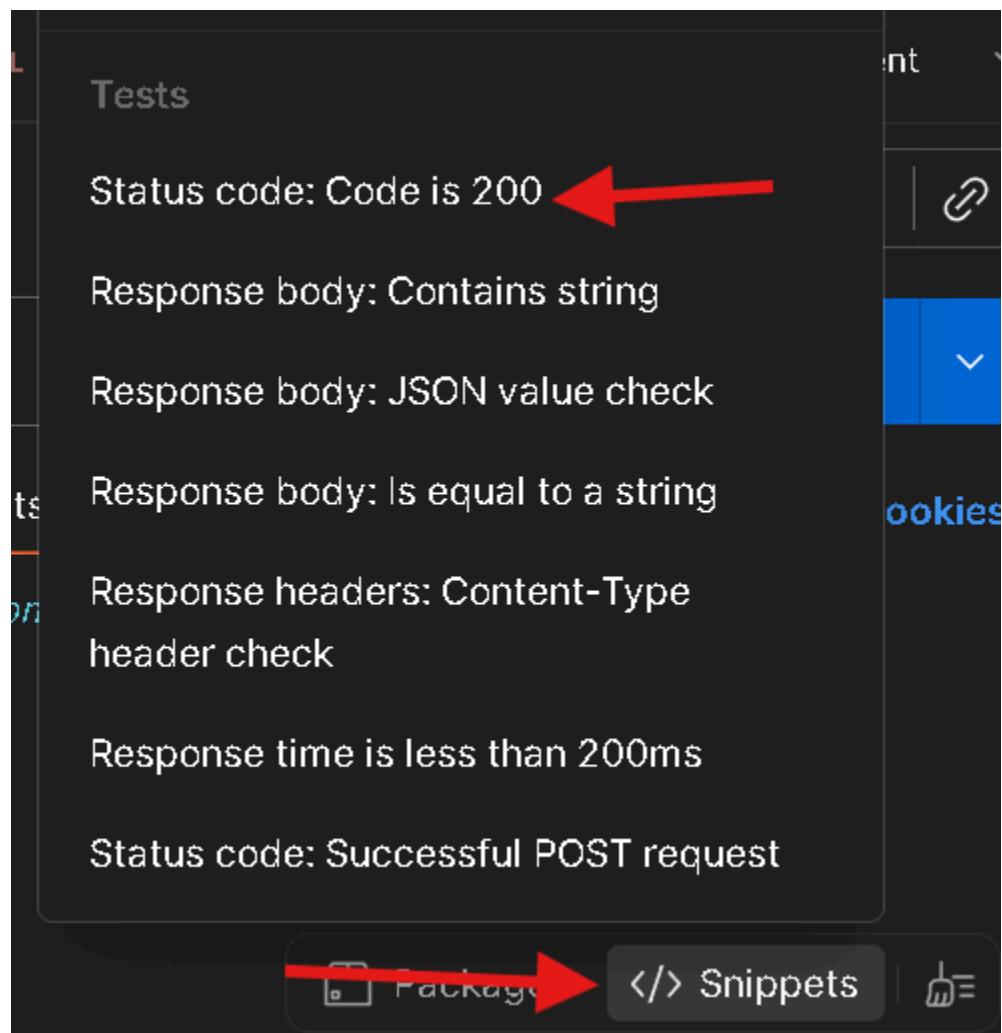
Test Results

404 Not Found • 698 ms • 798 B • Save Response

PASSED Status code is 404

The screenshot shows the Postman interface for a 'GET Order' request. The URL is set to {{BaseUrl}}/orders/:orderId. A red arrow points to the 'Send' button. The 'Test Results' dropdown is highlighted with a red box. The results show a single test case: 'Status code is 404' which passed. The status code is 404 Not Found, and the response time was 698 ms.

Use Snippets



Check Status Code 201 - Submit an Order

The screenshot shows the Postman interface. On the left, the 'Request Collection' sidebar lists various API endpoints. The 'POST POST Submit Order' endpoint is selected and highlighted with a red arrow. The main workspace displays a POST request to `"/orders"`. In the 'Scripts' tab, there is a pre-request script:

```
pm.test("Status code is 201", function () {  
    pm.response.to.have.status(201);  
});
```

Below the request, the 'Test Results' section shows a green 'PASSED' status with the message 'Status code is 201'. A red arrow points from the 'Test Results' box to the 'PASSED' status. The bottom right corner of the results area also has a red border.

The screenshot shows the 'Body' tab of the Postman interface. It displays the JSON response from the previous request. The response is:

```
1 {  
2   "created": true,  
3   "orderId": "7rFD_-9o_Lo0mwzBxwsV"  
4 }
```

The 'orderId' field is highlighted with a red box. Below the body, the text 'Add orderId to update' is visible.

The screenshot shows the Postman interface again. The 'PATCH PATCH Update Order' endpoint is selected and highlighted with a red arrow. The main workspace displays a PATCH request to `"/orders/:orderId"`. In the 'Params' tab, there is a table with one row:

Key	Value	Description
orderId	7rFD_-9o_Lo0mwzBxwsV	Description

The 'orderId' value is highlighted with a red box. Below the request, the 'Test Results' section shows a green 'PASSED' status with the message 'Status code is 201'. The bottom right corner of the results area also has a red border.

HTTP Request Collection / PATCH Update Order

PATCH {{BaseUrl}} /orders/:orderId

Send

Docs Params Authorization Headers (10) Body Scripts Settings Cookies

Pre-request

```
1 pm.test("Status code is 204", function () {
2     pm.response.to.have.status(204);
3 });
```

Post-response

Packages Snippets

Send

Docs Params Authorization Headers (10) Body Scripts Settings Cookies

Pre-request

```
1 pm.test("Status code is 204", function () {
2     pm.response.to.have.status(204);
3 });
```

Post-response

Packages Snippets

Test Results 204 No Content 313 ms 746 B Save Response

PASSED Status code is 204

*After updating an Order -> Get an Order
Add order ID*

Search collections

Request Collection

- GET GET Status
- GET GET List of Books
- GET GET List of Books by Query
- GET GET Book by ID
- POST POST Register API Client
- POST POST Submit Order
- PATCH PATCH Update Order
- GET GET Order

HTTP Request Collection / GET Order

GET {{BaseUrl}} /orders/:orderId

Send

Docs Params Authorization Headers (8) Body Scripts Settings Cookies

Key	Value	Description
orderId	7rFD-9o__Lo0mwzBxwsV	Description

Path Variables

Key	Value	Description	Bulk Edit
orderId	7rFD-9o__Lo0mwzBxwsV	Description	...

Check Status Code 200 - Get an Order

The screenshot shows two instances of the Postman application interface. The top instance displays a 'Request Collection / GET Order' with a successful response. The bottom instance shows the raw JSON response body.

Top Instance (Request Collection / GET Order):

- Request:** GET {{BaseUrl}}/orders/:orderId
- Script (Post-response):**

```
pm.test("Status code is 200", function () {
    pm.response.to.have.status(200);
});
```
- Test Results:** PASSED Status code is 200
- Response Headers:** 200 OK, 708 ms, 852 B

Bottom Instance (Raw Response Body):

```
{  
  "id": "7rFD_-9o__Lo0mwzBxwsV",  
  "bookId": 1,  
  "customerName": "new PostmanAPI",  
  "quantity": 1,  
  "timestamp": 1768396468373  
}
```

Check Status Code 204 - Delete an Order

Submit an Order

The screenshot shows the Postman interface with a request collection named "Request Collection". A POST request titled "POST Submit Order" is selected. The URL is {{BaseUrl}}/orders. The "Body" tab is active, showing raw JSON data:

```
1 {
2   "bookId": 1,
3   "customerName": "PostmanAPI"
4 }
```

The response status is 201 Created, and the response body is:

```
1 {
2   "created": true,
3   "orderId": "hsW7VNeMy0tmTV-XnmbOY"
4 }
```

Get orderId and apply in DELETE

The screenshot shows the Postman interface with a request collection named "Request Collection". A DELETE request titled "DELETE Order" is selected. The URL is {{BaseUrl}}/orders/:orderId. The "Params" tab is active, showing a table with one row:

Key	Value	Description
orderId	hsW7VNeMy0tmTV-XnmbOY	Description

A red arrow points to the "Value" column of the orderId row. Below the table, under "Path Variables", there is another table:

Key	Value	Description
orderId	hsW7VNeMy0tmTV-XnmbOY	Description

At the bottom, the "Scripts" tab is active, containing a pre-request script:1 pm.test("status code is 204", function () {
2 pm.response.to.have.status(204);
3});

HTTP Request Collection / **DELETE Order**

Save Share ⌂

DELETE {{BaseUrl}} /orders/:orderId **Send** ↗

Docs Params • Authorization • Headers (8) Body Scripts • Settings Cookies

Pre-request

```
1 pm.test("Status code is 204", function () {  
2     pm.response.to.have.status(204);  
3});
```

Post-response • ↗

Test Results ↗

Filter Results ↗

204 No Content • 326 ms • 746 B • Save Response ⌂

PASSED Status code is 204

```
pm.test("Status code is 204", function () {  
    pm.response.to.have.status(204);  
});
```