

## Experiment 4

**Student Name:** Rahul Saxena

**UID:** 24MCI10204

**Branch:** MCA (AI-ML)

**Section/Group:** MAM - 3(B)

**Semester:** II

**Subject Name:** Software Testing [24CAH-654]

**Aim:** Use Rest API using POSTMAN for API testing and development.

**API Tested:** <https://jsonplaceholder.typicode.com/posts>

### Steps:

#### 1. Setup and Open Postman

- Download and install Postman
- Open the Postman UI.

#### 2. Testing a GET Request

- Select GET as the request method.
- Enter the URL: <https://jsonplaceholder.typicode.com/posts/1>
- Click Send.
- Observe the response data containing a post with ID 1.

#### 3. Testing a POST Request

- Select POST as the request method.
- Enter the URL: <https://jsonplaceholder.typicode.com/posts>
- Go to the Body tab and select raw, then choose JSON as the format.
- Enter the JSON data:
- Click Send.
- Observe the response containing a newly created post.

#### 4. Testing a PUT Request

- Select PUT as the request method.
- Enter the URL: <https://jsonplaceholder.typicode.com/posts/1>

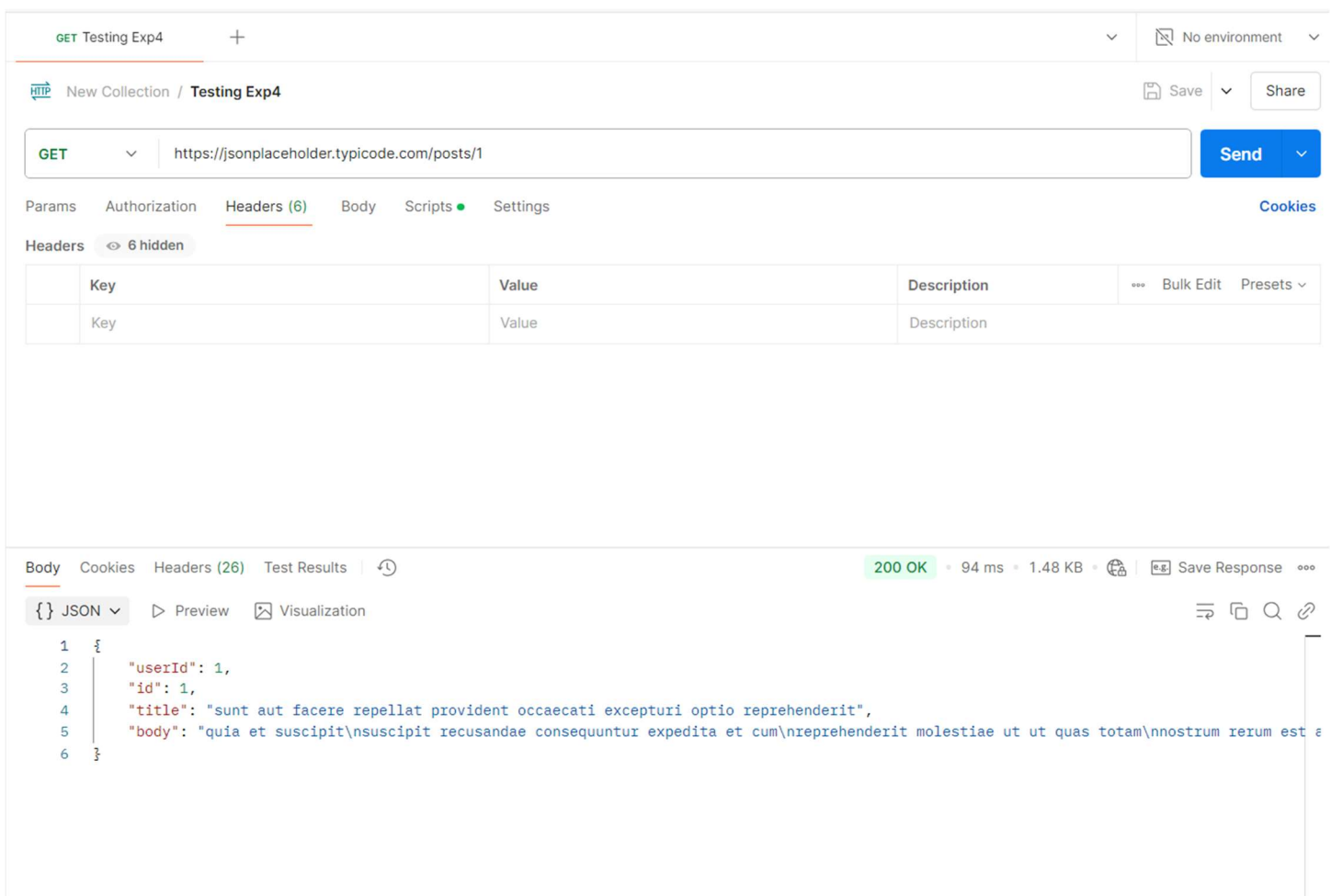
- In the Body tab, enter your data.
- Click Send.
- Observe the response containing the updated post.

## 5. Testing a DELETE Request

- Select DELETE as the request method.
- Enter the URL: <https://jsonplaceholder.typicode.com/posts/1>
- Click Send.
- Observe an empty response `{}` indicating successful deletion.

## Output:

### For Get Method:



The screenshot displays a REST client interface with a collection named "Testing Exp4". A GET request is configured for the URL `https://jsonplaceholder.typicode.com/posts/1`. The "Headers" tab is active, showing 6 hidden headers. The "Body" tab is selected, displaying the JSON response:

```

1 {
2   "userId": 1,
3   "id": 1,
4   "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
5   "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est e
6 }
  
```

The response status is **200 OK** with a response time of 94 ms and a size of 1.48 KB. The response is saved and can be viewed in Preview or Visualization mode.

## For Post Method:

POST Testing Exp4

New Collection / Testing Exp4

POST <https://jsonplaceholder.typicode.com/posts> Send

Params Authorization Headers (8) Body Scripts Settings Cookies Beautify

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "title": "Test Post",
3   "body": "This is a test post by Rahul Saxean",
4   "userId": 1
5 }
```

Body Cookies Headers (26) Test Results 201 Created • 259 ms • 1.48 KB Save Response

{ } JSON Preview Visualization

```
1 {
2   "title": "Test Post",
3   "body": "This is a test post by Rahul Saxean",
4   "userId": 1,
5   "id": 101
6 }
```

## For Put Method:

PUT Testing Exp4

New Collection / Testing Exp4

PUT <https://jsonplaceholder.typicode.com/posts/1> Send

Params Authorization Headers (8) Body Scripts Settings Cookies Beautify

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

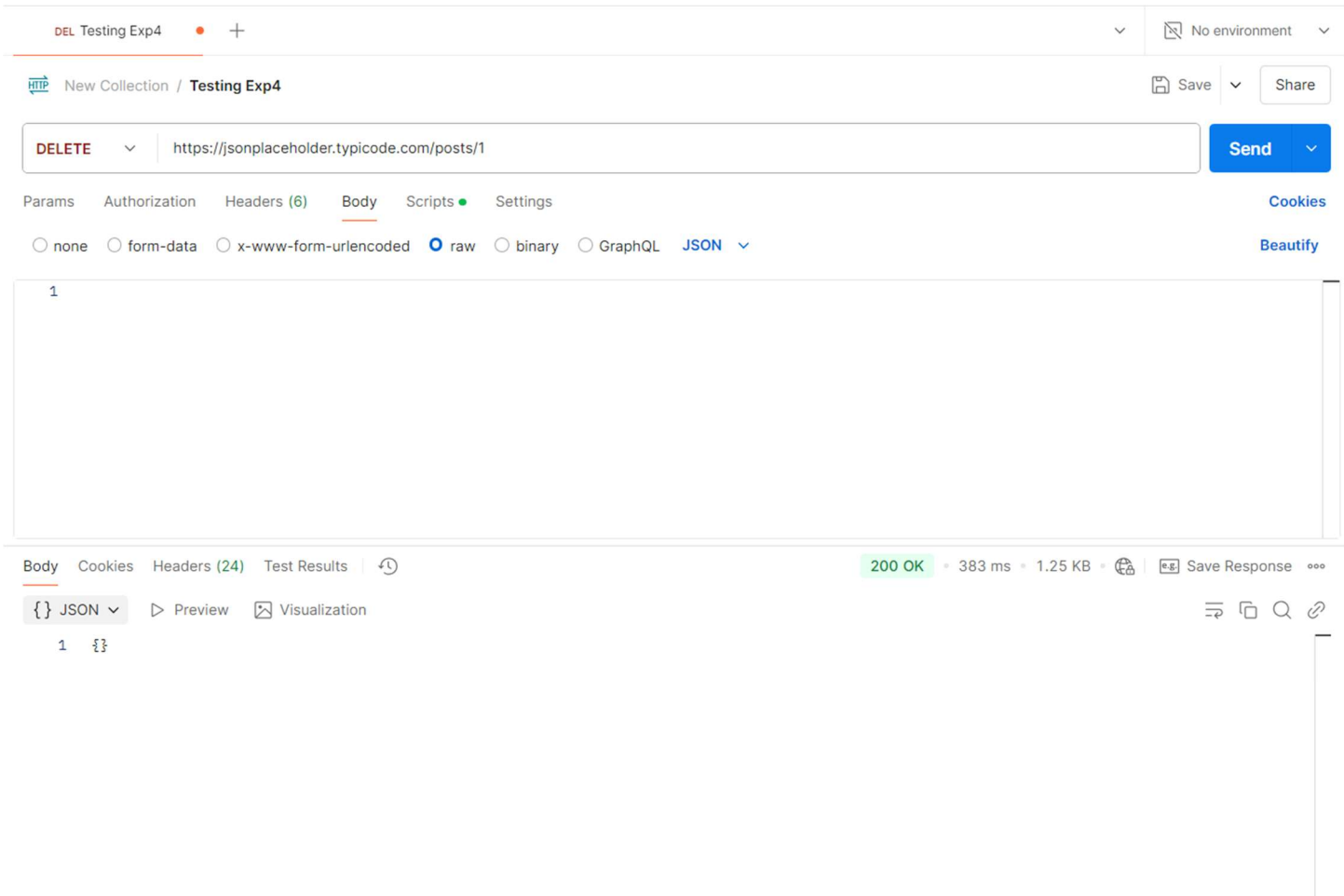
```
1 {
2   "id": 1,
3   "title": "Updated Title",
4   "body": "Updated content.",
5   "userId": 1
6 }
```

Body Cookies Headers (25) Test Results 200 OK • 923 ms • 1.35 KB Save Response

{ } JSON Preview Visualization

```
1 {
2   "id": 1,
3   "title": "Updated Title",
4   "body": "Updated content.",
5   "userId": 1
6 }
```

## For Delete Method:



The screenshot shows the Postman interface for a DELETE request. The top bar indicates the request is in a collection named 'Testing Exp4'. The URL is 'https://jsonplaceholder.typicode.com/posts/1'. The method is 'DELETE'. The 'Body' tab is selected, and the content is '1'. The response is a 200 OK status, with a response time of 383 ms and a size of 1.25 KB. The response is displayed in the 'Body' tab, showing the JSON object: { "id": 1, "title": "sunt aut facere repellat provident occaecat excepturi anim repellendus natus volupt accusantium ea delectat sed dolor ea sed ipsam quosam aut perferendis nisi velit eligendi vel quod voluptatem sit neque suscipit id est quasi." }.

## Learning Outcome:

- Understood the process of API testing using Postman.
- Learned to send GET, POST, PUT, and DELETE requests.
- Explored how APIs respond to various HTTP methods.
- Gained experience in interacting with RESTful web services.