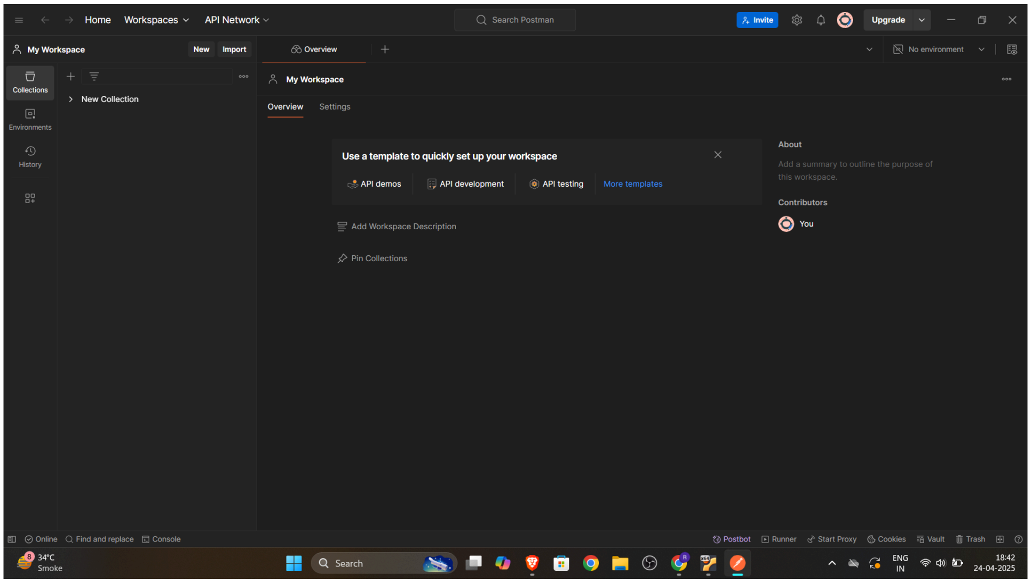
Ritesh Jha | SY-IT(B3) | 16010423076 | Cyber Honors

Compensation-3 | Date : 24 - April - 2025

API Testing

For this Simulation we will be using :

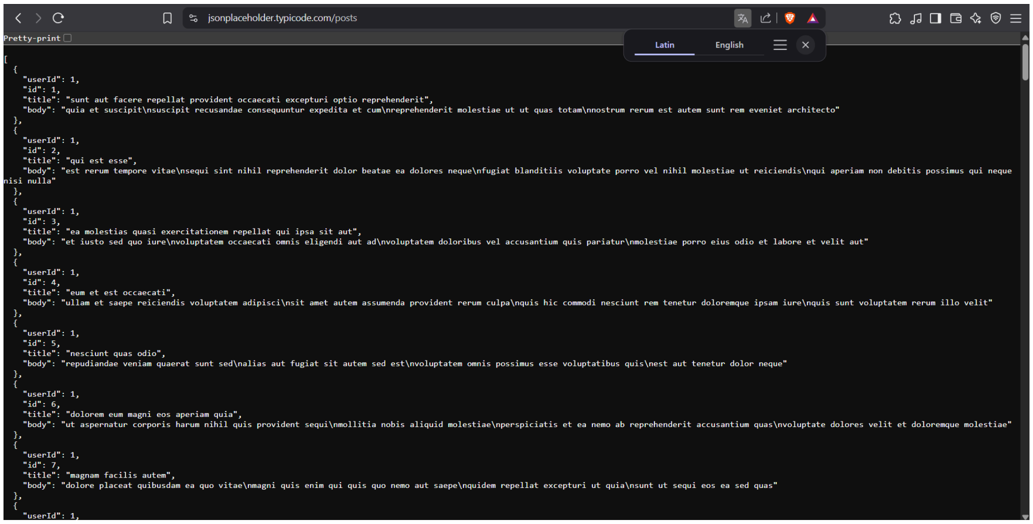
- Postman



Use a template to quickly set up your workspace

Add a summary to outline the purpose of this workspace.

- A free dummy API: https://jsonplaceholder.typicode.com/posts

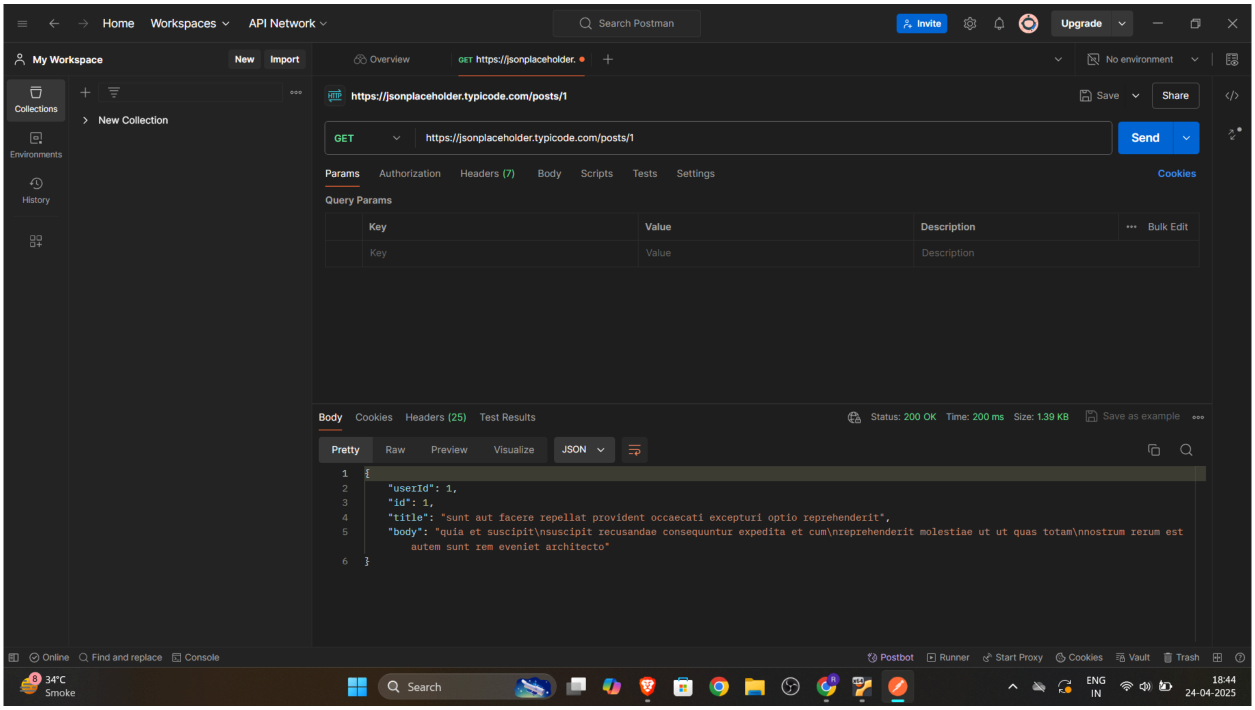


Pretty-print □

```
[
  {
    "userId": 1,
    "id": 1,
    "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
    "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto"
  },
  {
    "userId": 1,
    "id": 2,
    "title": "qui est esse",
    "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque
nisi nulla"
  },
  {
    "userId": 1,
    "id": 3,
    "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut",
    "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut"
  },
  {
    "userId": 1,
    "id": 4,
    "title": "eum et est occaecati",
    "body": "ullam et saepe reiciendis voluptatem adipisci\nsit amet autem assumenda provident rerum culpa\nquis hic commodi nesciunt rem tenetur doloremque ipsam iure\nquis sunt voluptatem rerum illo velit"
  },
  {
    "userId": 1,
    "id": 5,
    "title": "nesciunt quas odio",
    "body": "repudiandae veniam quaerat sunt sed\nalias aut fugiat sit autem sed est\nvoluptatem omnis possimus esse voluptatibus quis\nest aut tenetur dolor neque"
  },
  {
    "userId": 1,
    "id": 6,
    "title": "dolorem eum magni eos aperiam quia",
    "body": "ut aspernatur corporis harum nihil quis provident sequi\nmollitia nobis aliquid molestiae\nperspiciatis et ea nemo ab reprehenderit accusantium quas\nvoluptate dolores velit et doloremque molestiae"
  },
  {
    "userId": 1,
    "id": 7,
    "title": "magnam facilis autem",
    "body": "dolore placeat quibusdam ea quo vitae\nmagni quis enim qui quis quo nemo aut saepe\nquidem repellat excepturi ut quia\nsunt ut sequi eos ea sed quas"
  },
  {
    "userId": 1,
```

1. GET Request – View Data

- **Process:**
  - I first launched Postman and set up a new HTTP request with the GET method.
  - The URL I used was https://jsonplaceholder.typicode.com/posts/1, which returns data for a specific post.
  - After hitting "Send," I got a JSON response with information about a post, like its userId, id, title, and body.



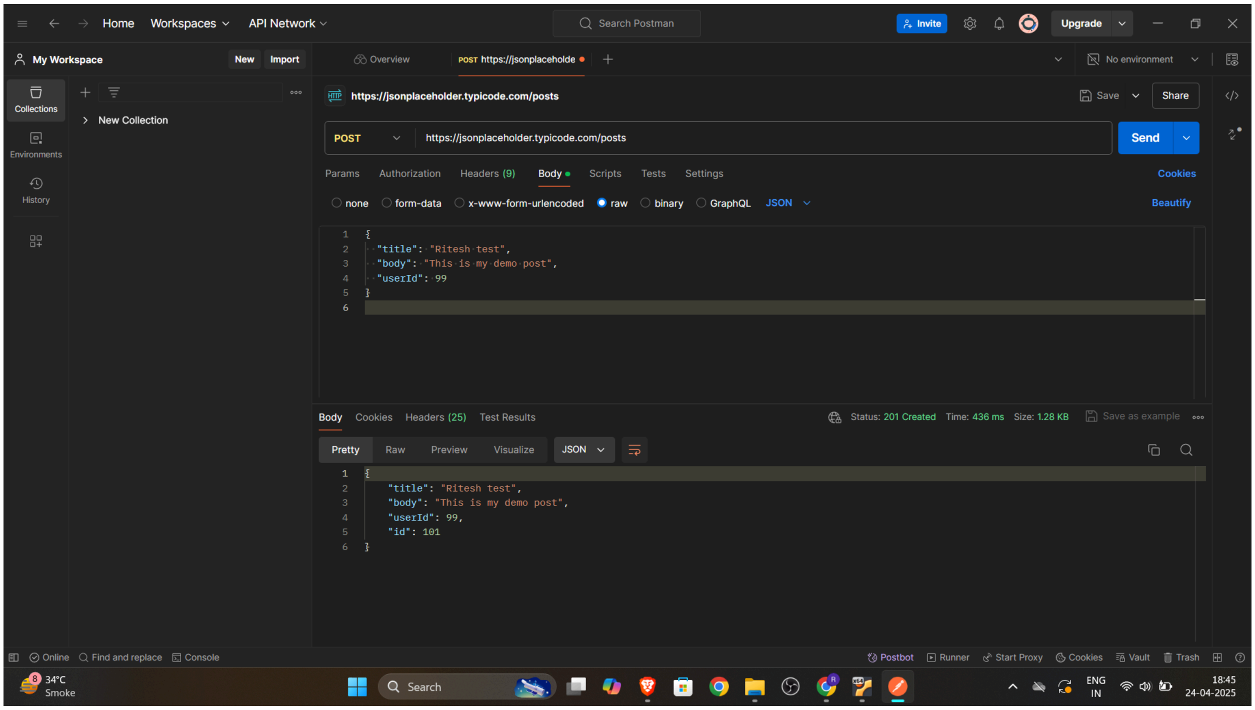
GET https://jsonplaceholder.typicode.com/posts/1

Status: 200 OK Time: 200 ms Size: 1.39 KB

```
{
    "userId": 1,
    "id": 1,
    "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
    "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est
        autem sunt rem eveniet architecto"
}
```

2. POST Request – Send Data

- **Process:**
    - I switched the method to POST and set the URL to https://jsonplaceholder.typicode.com/posts.
    - In the “Body” tab, I selected “raw” and chose JSON format.
      I entered a JSON object to create a new post, containing the title, body, and userId.
    - When I clicked “Send,” I got a response indicating that the post was created, including a new id (in this case, 101).



https://jsonplaceholder.typicode.com/posts

POST https://jsonplaceholder.typicode.com/posts

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
  "title": "Ritesh test",
  "body": "This is my demo post",
  "userId": 99
}
```

Body Cookies Headers (25) Test Results — Status: 201 Created Time: 436 ms Size: 1.28 KB

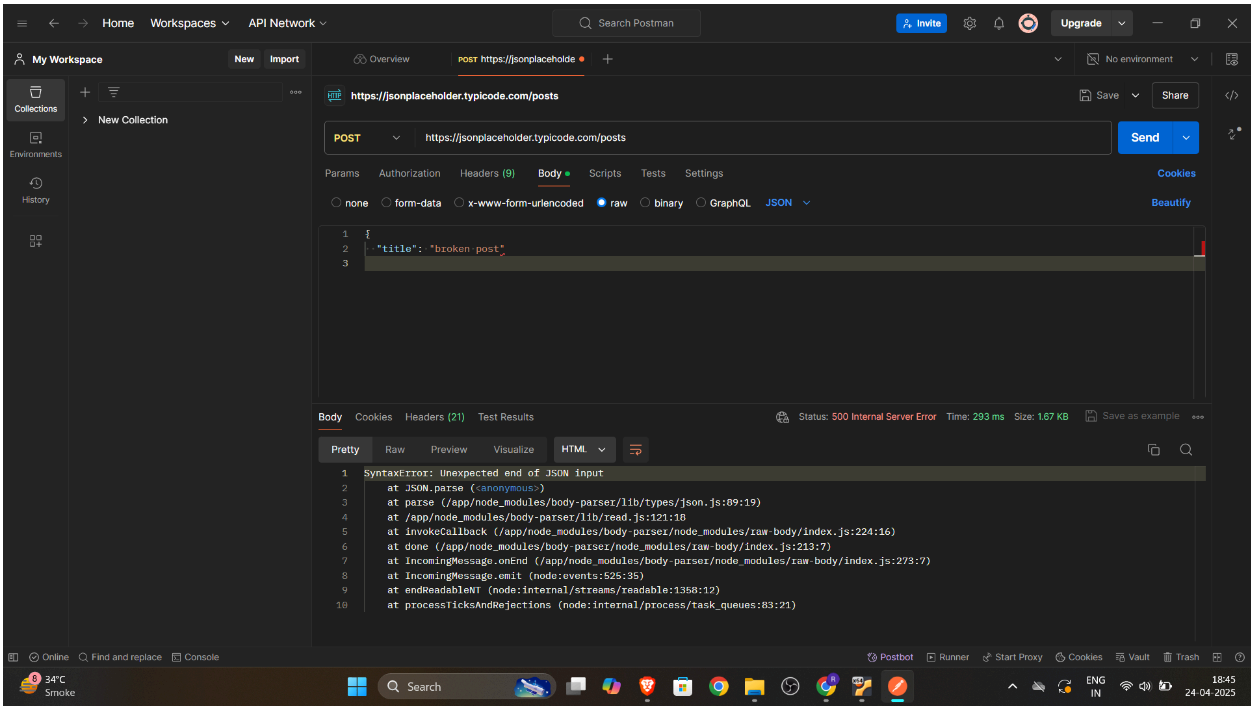
JSON

```
{
    "title": "Ritesh test",
    "body": "This is my demo post",
    "userId": 99,
    "id": 101
}
```

---

3. Negative Test – Broken POST

- **Process:**
    - I sent a POST request with a broken JSON object (missing a body field).
    - The request was made with the data { "title": "broken post" }.



POST https://jsonplaceholder.typicode.com/posts

Body: ● raw JSON

```
{
  "title": "broken post"

```

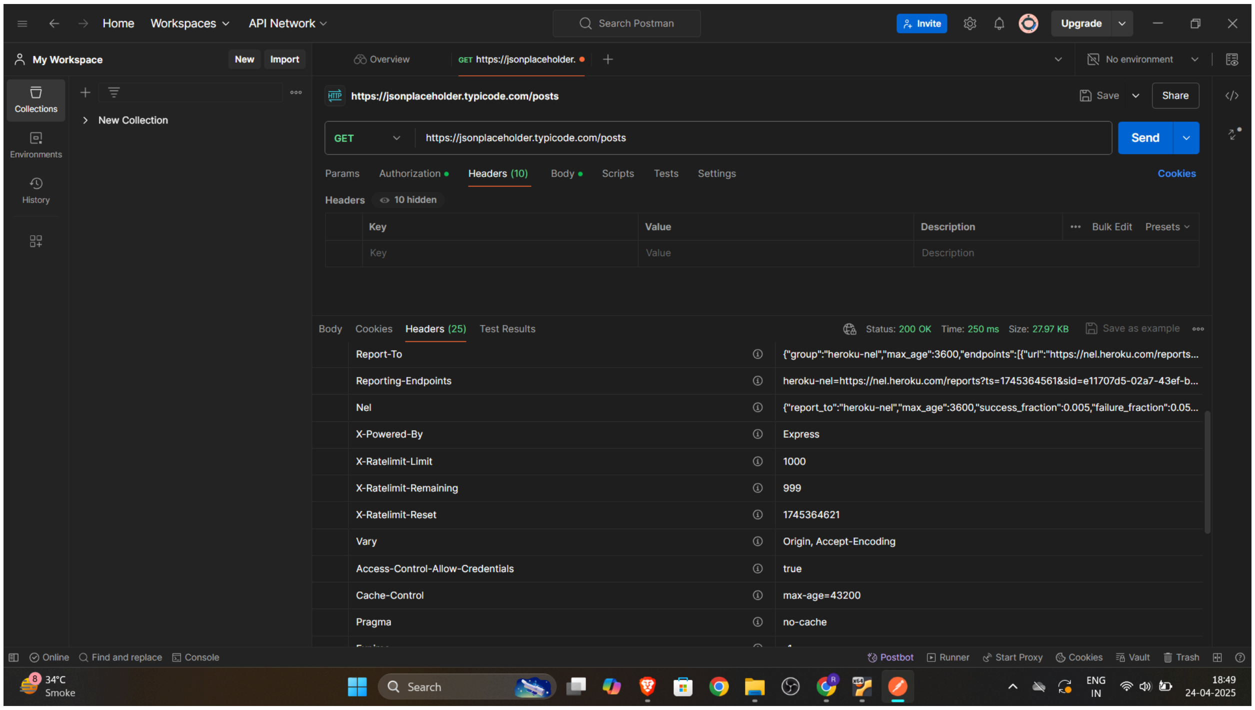
Status: 500 Internal Server Error Time: 293 ms Size: 1.67 KB

```
SyntaxError: Unexpected end of JSON input
    at JSON.parse (<anonymous>)
    at parse (/app/node_modules/body-parser/lib/types/json.js:89:19)
    at /app/node_modules/body-parser/lib/read.js:121:18
    at invokeCallback (/app/node_modules/body-parser/node_modules/raw-body/index.js:224:16)
    at done (/app/node_modules/body-parser/node_modules/raw-body/index.js:213:7)
    at IncomingMessage.onEnd (/app/node_modules/body-parser/node_modules/raw-body/index.js:273:7)
    at IncomingMessage.emit (node:events:525:35)
    at endReadableNT (node:internal/streams/readable:1358:12)
    at processTicksAndRejections (node:internal/process/task_queues:83:21)
```

---

4. Rate Limiting & Response Headers

- **Process:**
  - I repeatedly sent GET requests to the API in quick succession to simulate a burst of requests.
  - I checked the response headers in Postman for rate-limiting information, specifically looking for X-RateLimit-Limit, X-RateLimit-Remaining, and X-RateLimit-Reset.



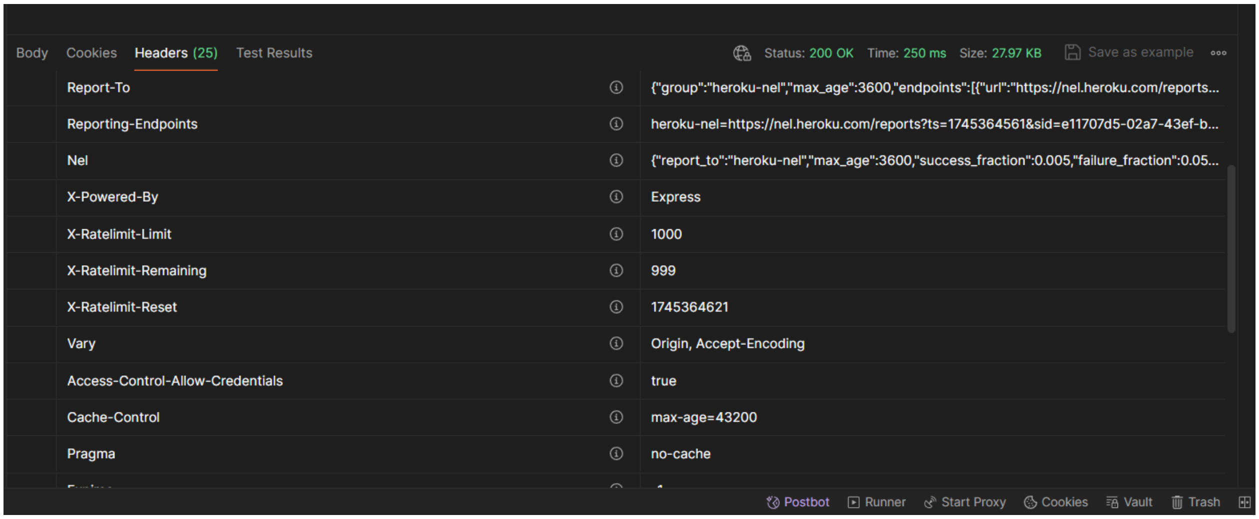
https://jsonplaceholder.typicode.com/posts

GET https://jsonplaceholder.typicode.com/posts

Body Cookies Headers (25) Test Results

Status: 200 OK Time: 250 ms Size: 27.97 KB Save as example

Key	Value
Report-To	{"group":"heroku-nel","max_age":3600,"endpoints":[{"url":"https://nel.heroku.com/reports...
Reporting-Endpoints	heroku-nel=https://nel.heroku.com/reports?ts=1745364561&sid=e11707d5-02a7-43ef-b...
Nel	{"report_to":"heroku-nel","max_age":3600,"success_fraction":0.005,"failure_fraction":0.05...
X-Powered-By	Express
X-Ratelimit-Limit	1000
X-Ratelimit-Remaining	999
X-Ratelimit-Reset	1745364621
Vary	Origin, Accept-Encoding
Access-Control-Allow-Credentials	true
Cache-Control	max-age=43200
Pragma	no-cache



Body Cookies Headers (25) Test Results

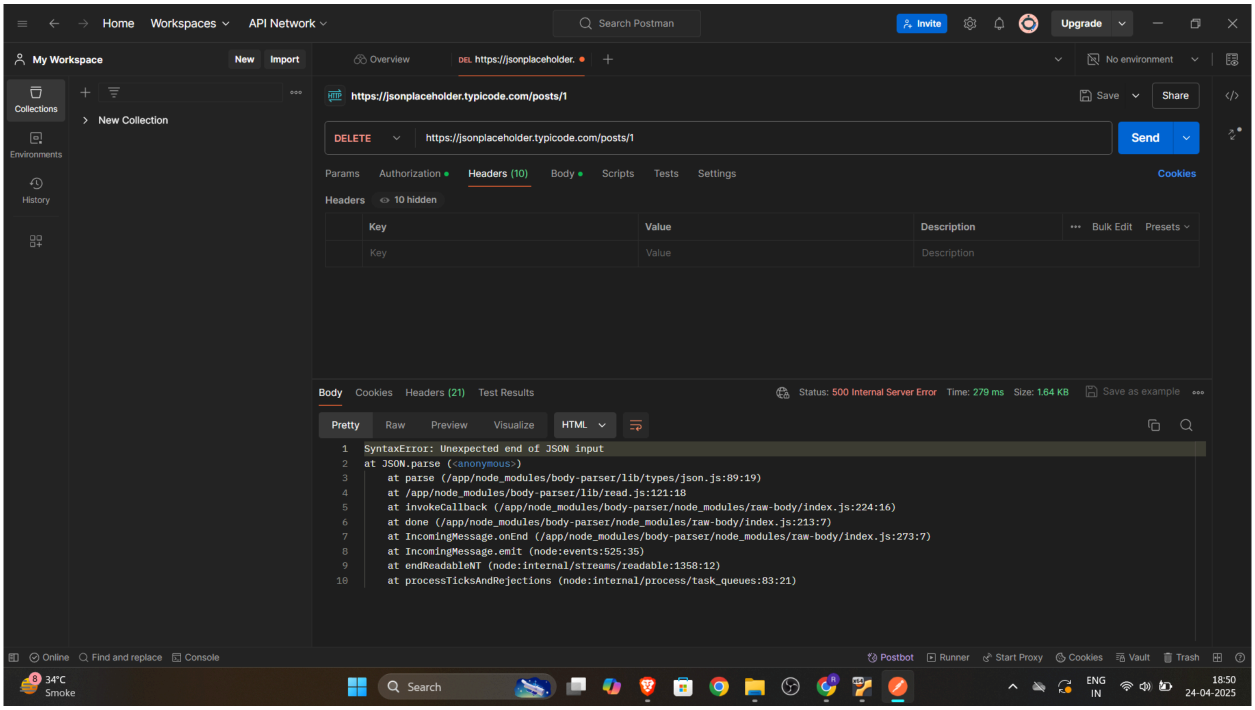
Status: 200 OK Time: 250 ms Size: 27.97 KB Save as example

Key	Value
Report-To	{"group":"heroku-nel","max_age":3600,"endpoints":[{"url":"https://nel.heroku.com/reports...
Reporting-Endpoints	heroku-nel=https://nel.heroku.com/reports?ts=1745364561&sid=e11707d5-02a7-43ef-b...
Nel	{"report_to":"heroku-nel","max_age":3600,"success_fraction":0.005,"failure_fraction":0.05...
X-Powered-By	Express
X-Ratelimit-Limit	1000
X-Ratelimit-Remaining	999
X-Ratelimit-Reset	1745364621
Vary	Origin, Accept-Encoding
Access-Control-Allow-Credentials	true
Cache-Control	max-age=43200
Pragma	no-cache

---

5. DELETE Request – Remove Data

- **Process:**
  - I set the method to DELETE and used the URL https://jsonplaceholder.typicode.com/posts/1 to delete the first post.
  - After sending the request, I expected to receive a response indicating that the API didn’t support deletion.



https://jsonplaceholder.typicode.com/posts/1

DELETE https://jsonplaceholder.typicode.com/posts/1

Headers 10 hidden

Key	Value	Description
Key	Value	Description

Body — Status: 500 Internal Server Error Time: 279 ms Size: 1.64 KB

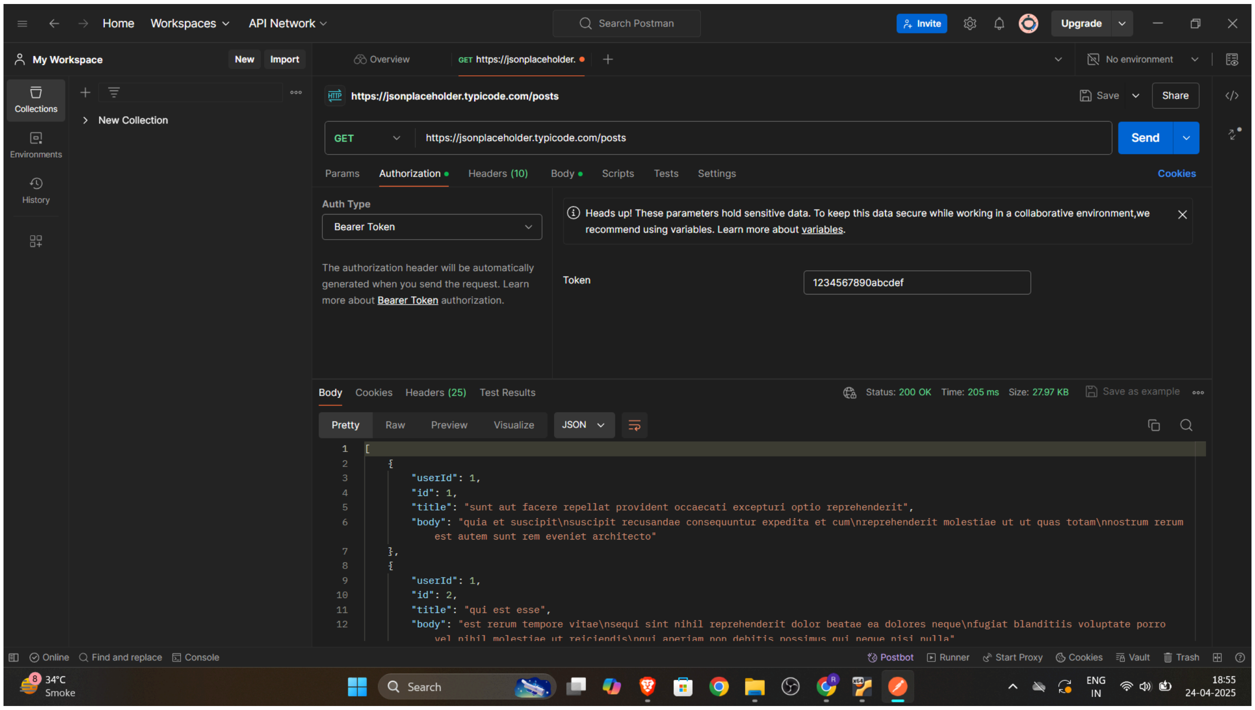
HTML

```
SyntaxError: Unexpected end of JSON input
at JSON.parse (<anonymous>)
    at parse (/app/node_modules/body-parser/lib/types/json.js:89:19)
    at /app/node_modules/body-parser/lib/read.js:121:18
    at invokeCallback (/app/node_modules/body-parser/node_modules/raw-body/index.js:224:16)
    at done (/app/node_modules/body-parser/node_modules/raw-body/index.js:213:7)
    at IncomingMessage.onEnd (/app/node_modules/body-parser/node_modules/raw-body/index.js:273:7)
    at IncomingMessage.emit (node:events:525:35)
    at endReadableNT (node:internal/streams/readable:1358:12)
    at processTicksAndRejections (node:internal/process/task_queues:83:21)
```

---

6. Authentication Testing (Simulating Bearer Token Authentication)

- **Process:**
  - I created a new GET request to https://jsonplaceholder.typicode.com/posts and went to the Authorization tab in Postman.
  - I selected the “Bearer Token” option and entered a fake token (1234567890abcdef).
  - Here, the status is OK because there was not authorization setup in the API.



https://jsonplaceholder.typicode.com/posts

GET https://jsonplaceholder.typicode.com/posts

Auth Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about Bearer Token authorization.

Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment,we recommend using variables. Learn more about variables.

Token: 1234567890abcdef

Status: 200 OK Time: 205 ms Size: 27.97 KB

```
[
    {
        "userId": 1,
        "id": 1,
        "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
        "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum
            est autem sunt rem eveniet architecto"
    },
    {
        "userId": 1,
        "id": 2,
        "title": "qui est esse",
        "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro
            vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque nisi nulla"
```