API Assignment Questions

1. Explain the following:

A. PUT and PATCH methods

- PUT: Replaces the entire resource with the updated content. If the resource does not exist, it creates a new one.

- PATCH: Partially updates an existing resource. Only the specified fields are modified.

B. Headers and Cookies

- Headers: Provide metadata for HTTP requests and responses, such as content type, authorization, and caching directives.

- Cookies: Store data on the client side to maintain session state and track user activities across requests.

C. Endpoint and Base URL

- Endpoint: A specific URL that performs a particular action in an API.

- Base URL: The root URL for all API endpoints of a service.

D. Query Parameters and Path Parameters

- Query Parameters: Passed in the URL after `?` and provide filtering or sorting data (e.g., `?id=1&sort=asc`).

- Path Parameters: Part of the URL path and identify specific resources (e.g., `/users/1`).

E. What are error codes? Explain all series of error codes.

Error codes are HTTP status codes indicating the result of an API request.

- 1xx (Informational): Request received, continuing process.

- 2xx (Success): Request successfully processed (e.g., 200 OK, 201 Created).

- 3xx (Redirection): Further action required to complete request (e.g., 301 Moved Permanently).

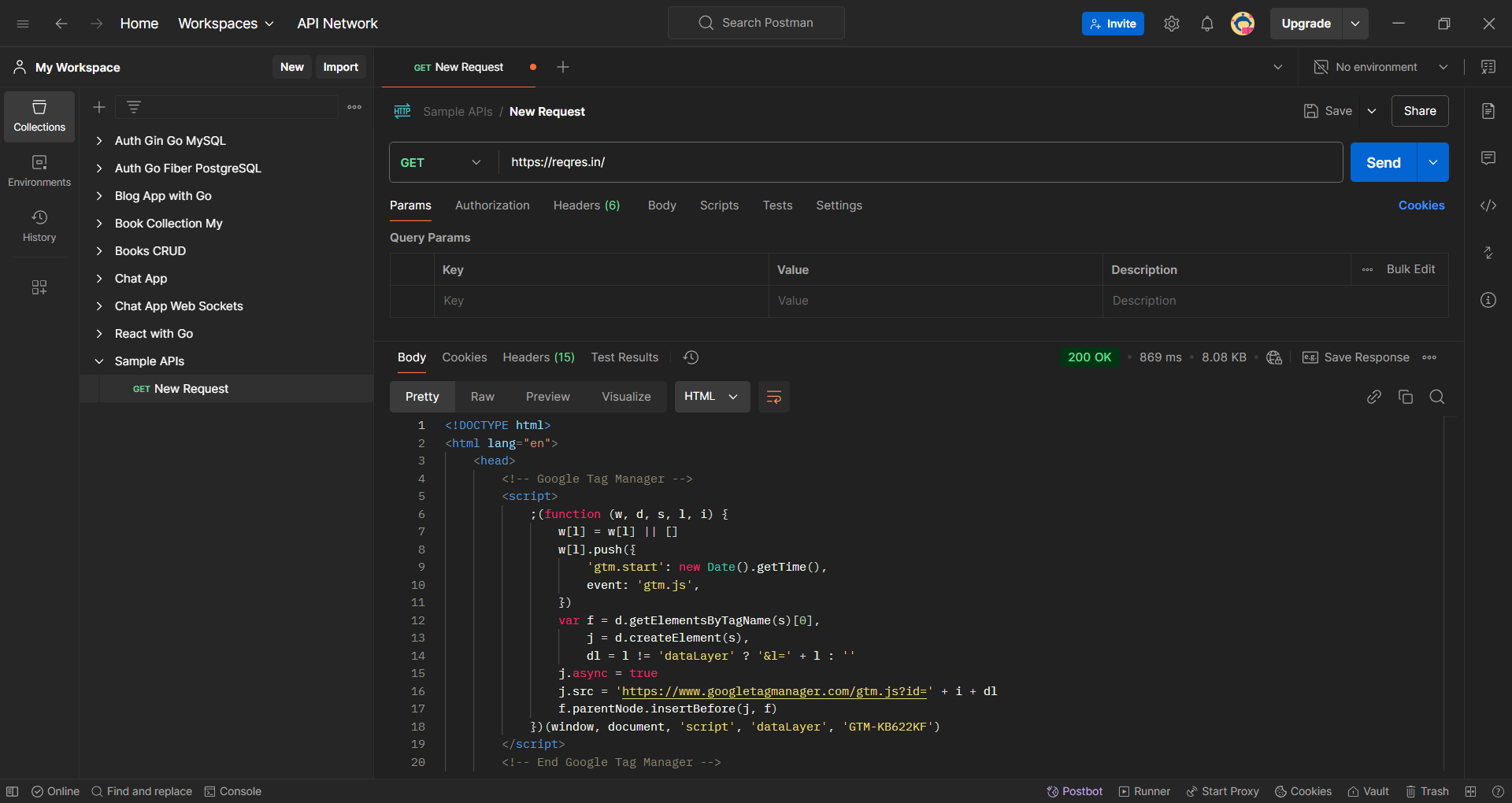
- 4xx (Client Error): Request contains bad syntax or cannot be fulfilled (e.g., 400 Bad Request, 401 Unauthorized).

- 5xx (Server Error): Server failed to process a valid request (e.g., 500 Internal Server Error).

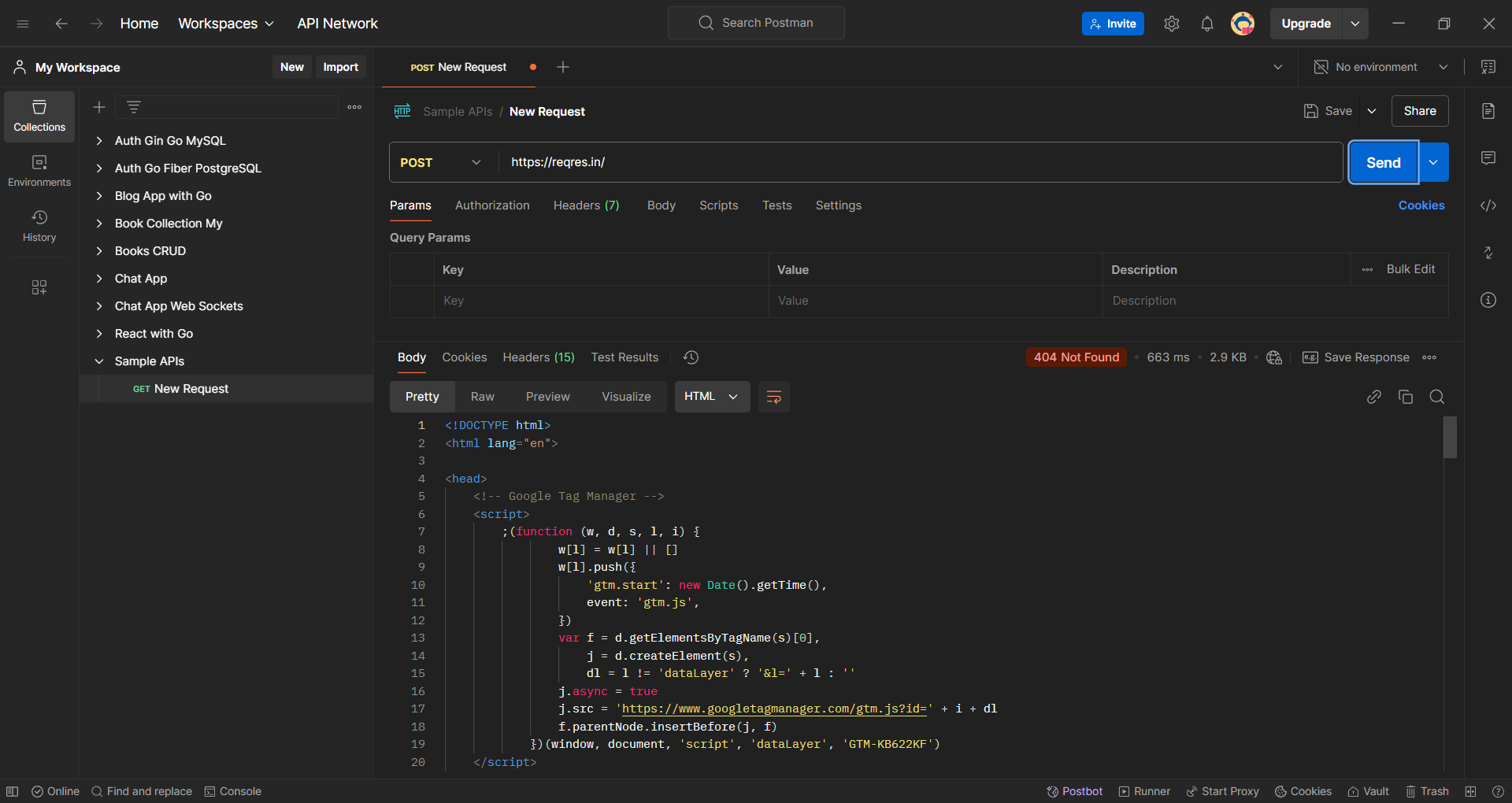
1. Create a new collection in Postman named "Sample APIs." Use https://reqres.in/ and implement GET, POST, PUT,

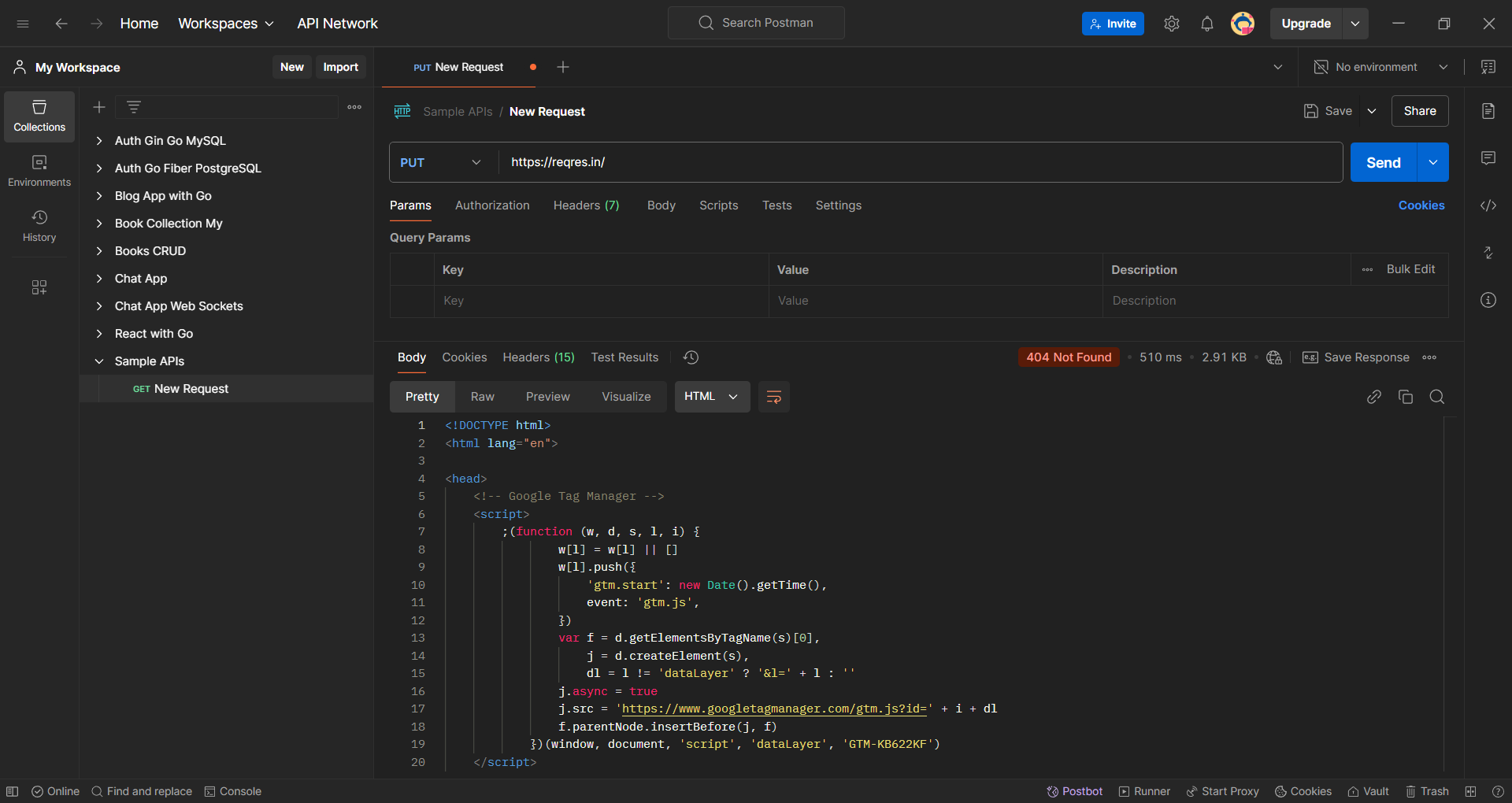
PATCH, and DELETE operations.

Get

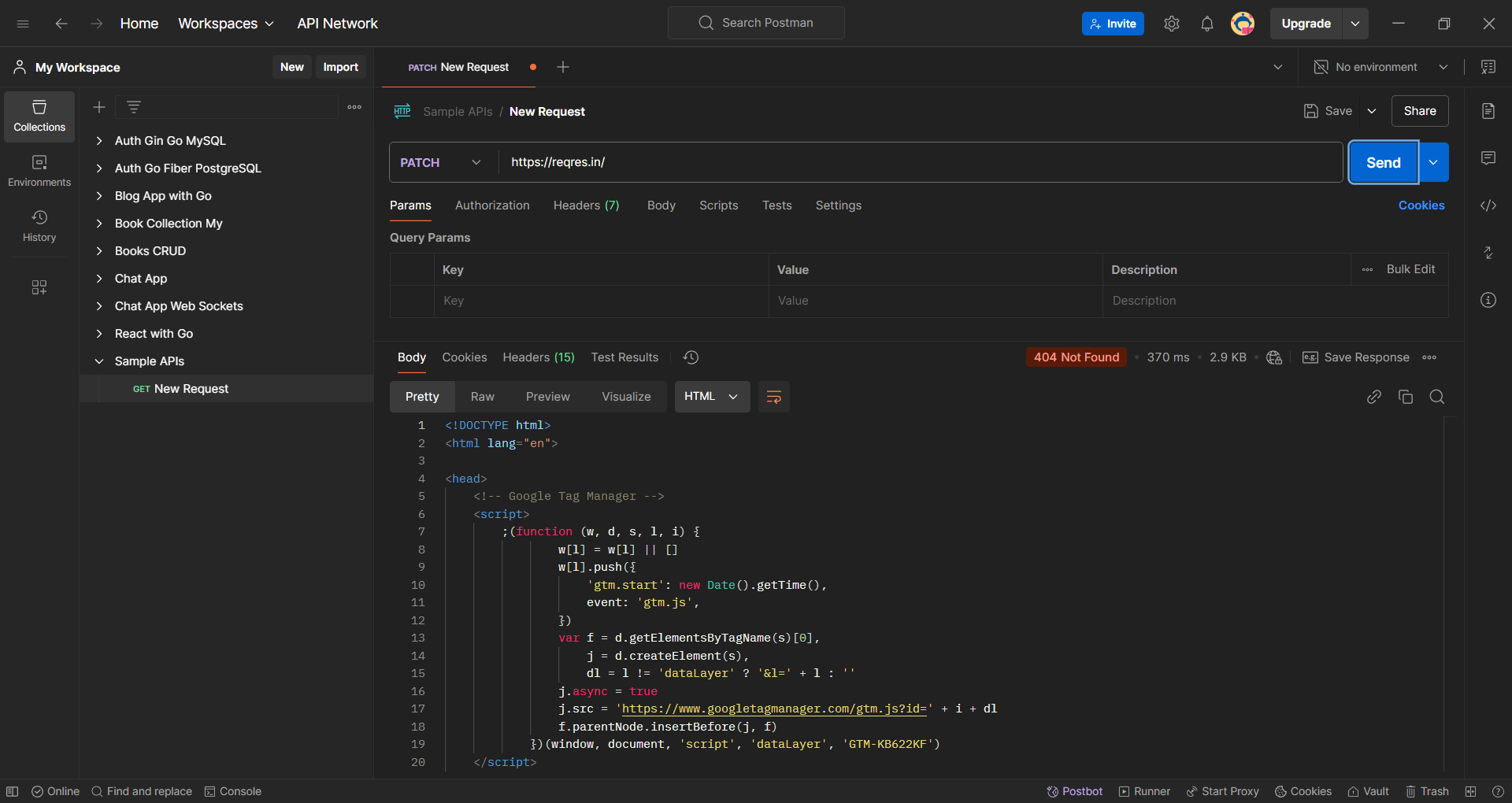


Post

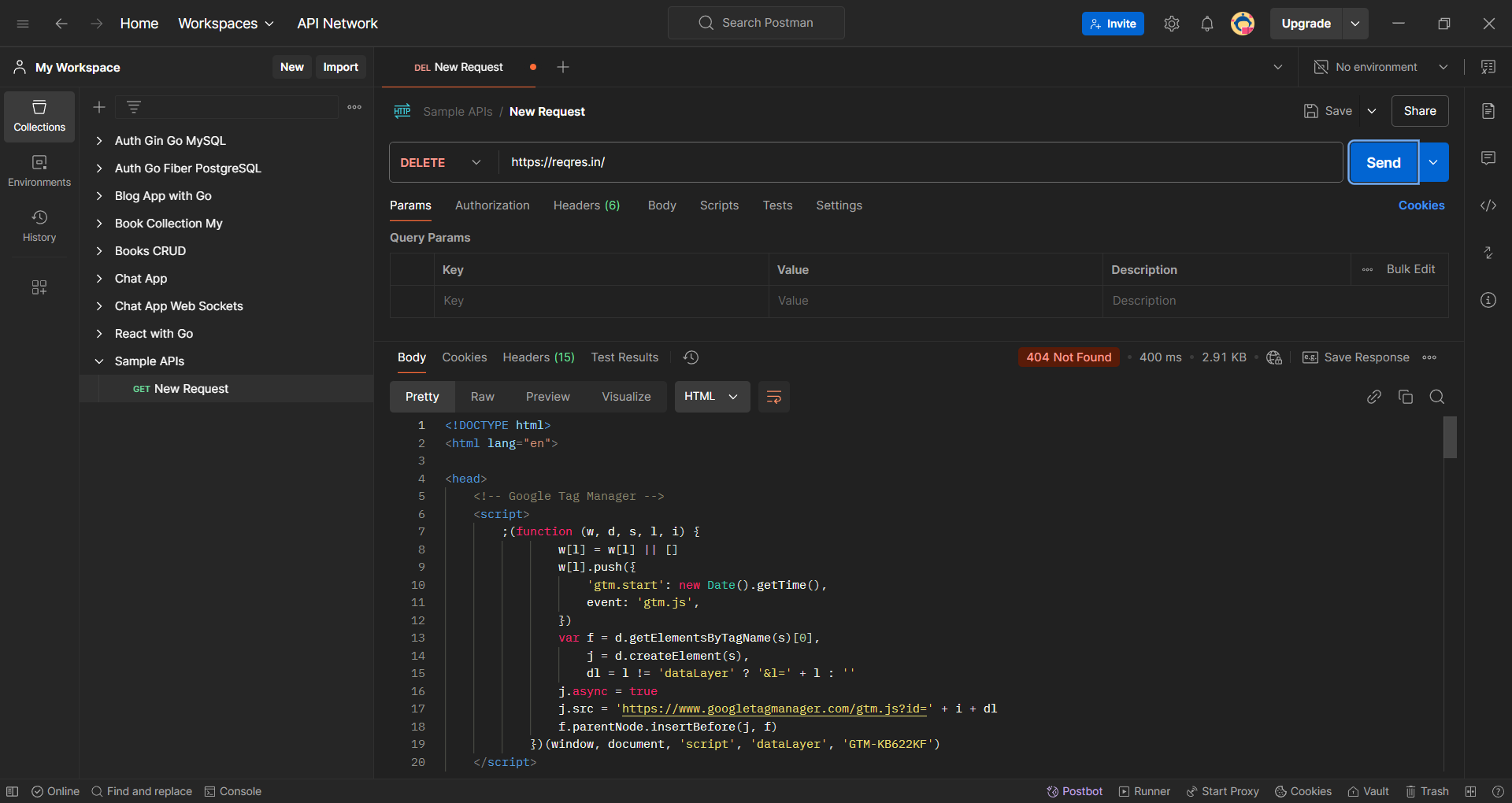


Put

Patch



Delete



Q3. What is the difference between given(), when(), and then() in Rest Assured?

- given(): Sets up the request specification, such as headers, parameters, and body.

- when(): Specifies the HTTP method (e.g., GET, POST) and the endpoint to be called.

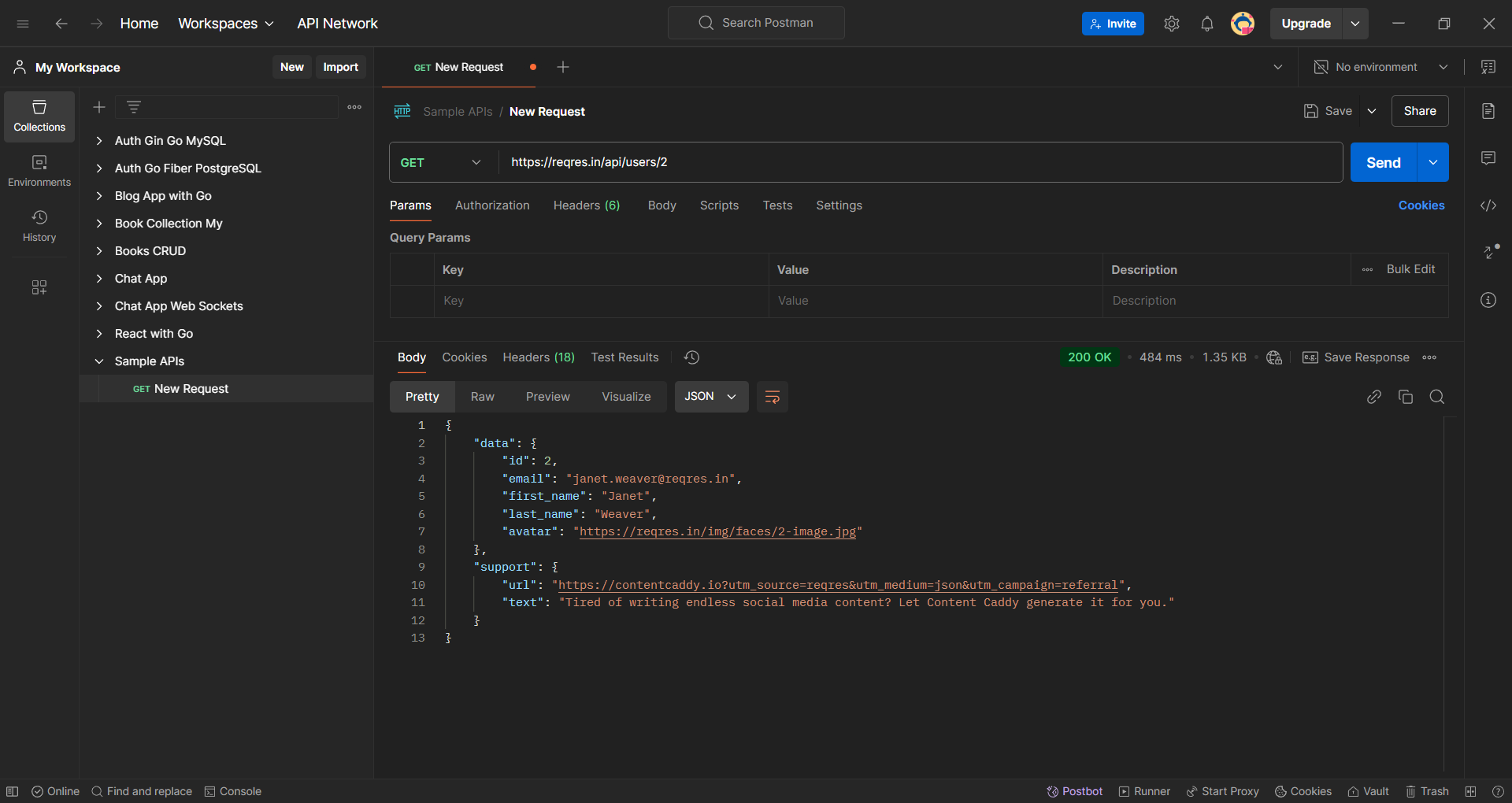
- then(): Verifies the response with assertions, such as status codes or response body content.

Q4. Perform the operations with Rest Assured (use https://reqres.in/):

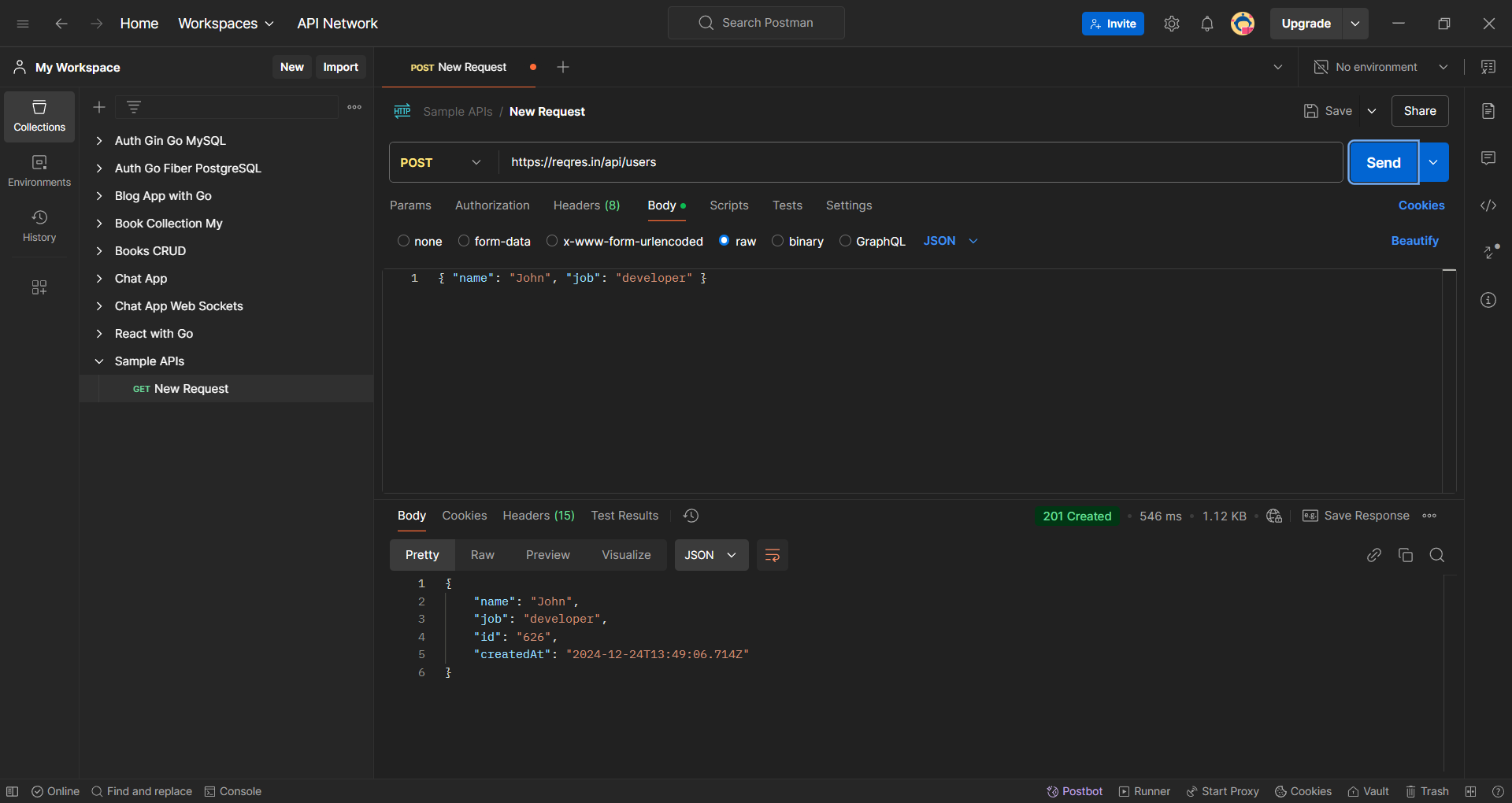
A. Create GET, POST, PUT, PATCH, and DELETE

- Use Rest Assured to perform all CRUD operations:

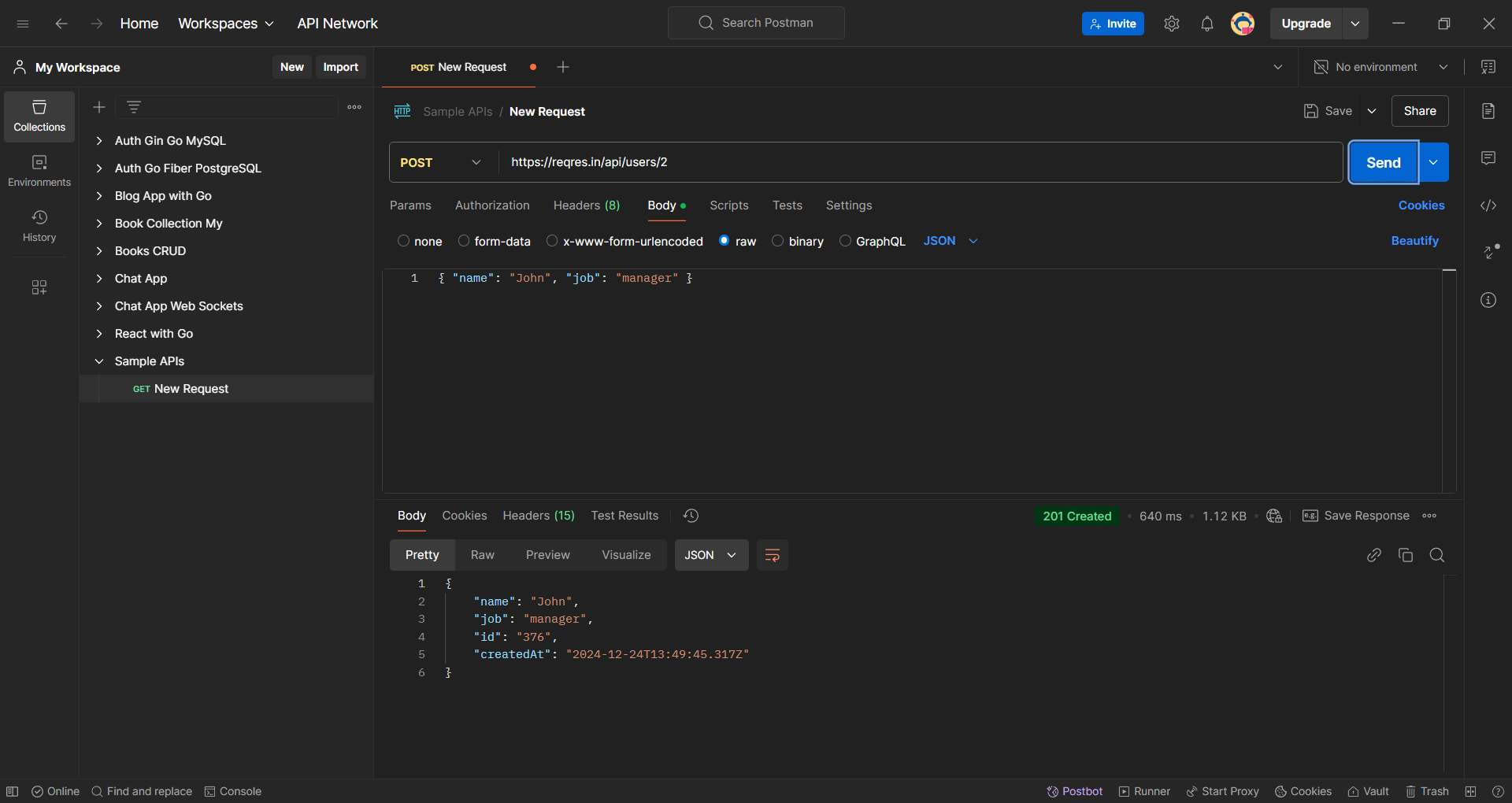
- GET: `GET /api/users/2`



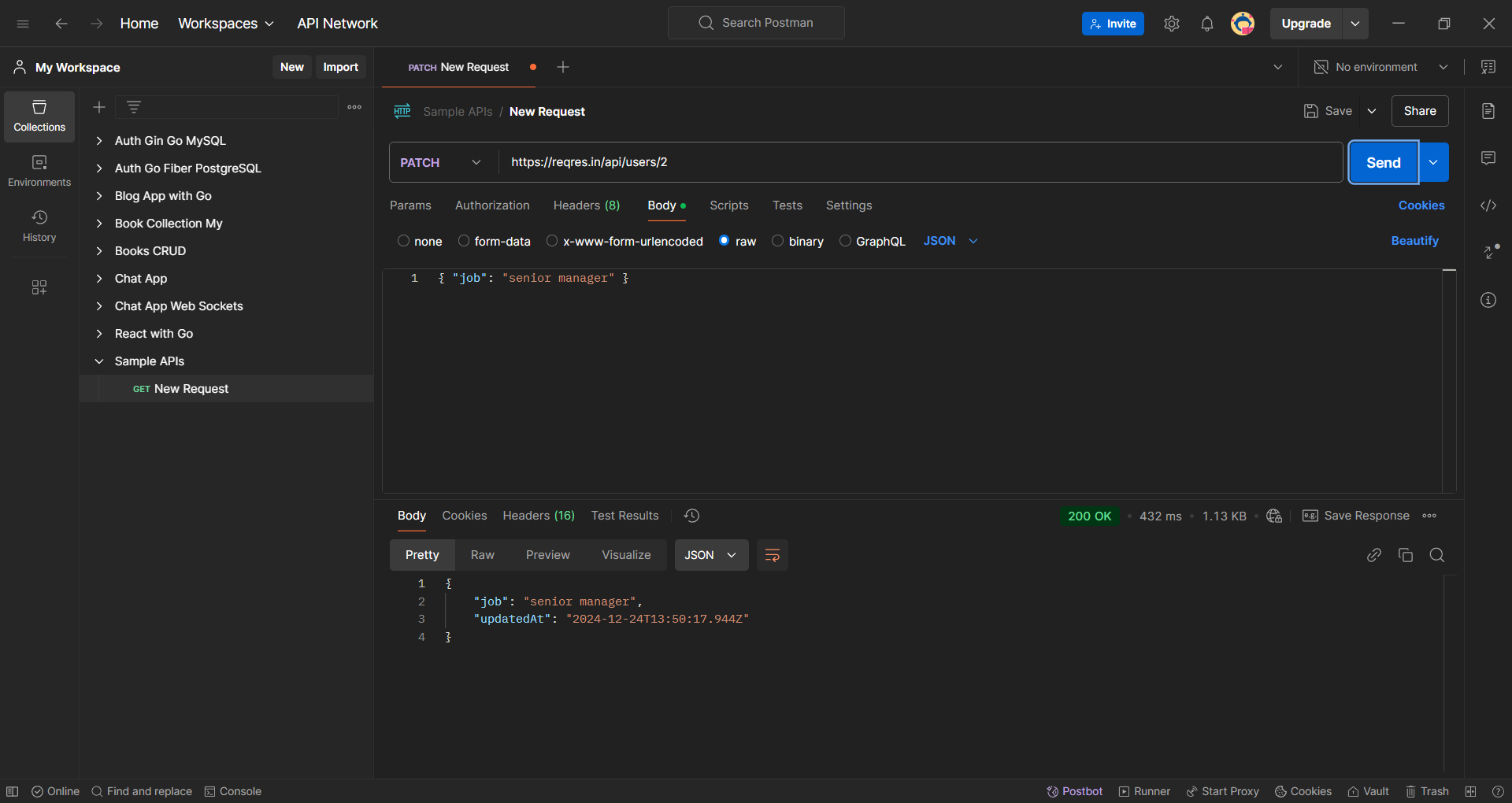
- POST: `POST /api/users` with payload `{ "name": "John", "job": "developer" }`



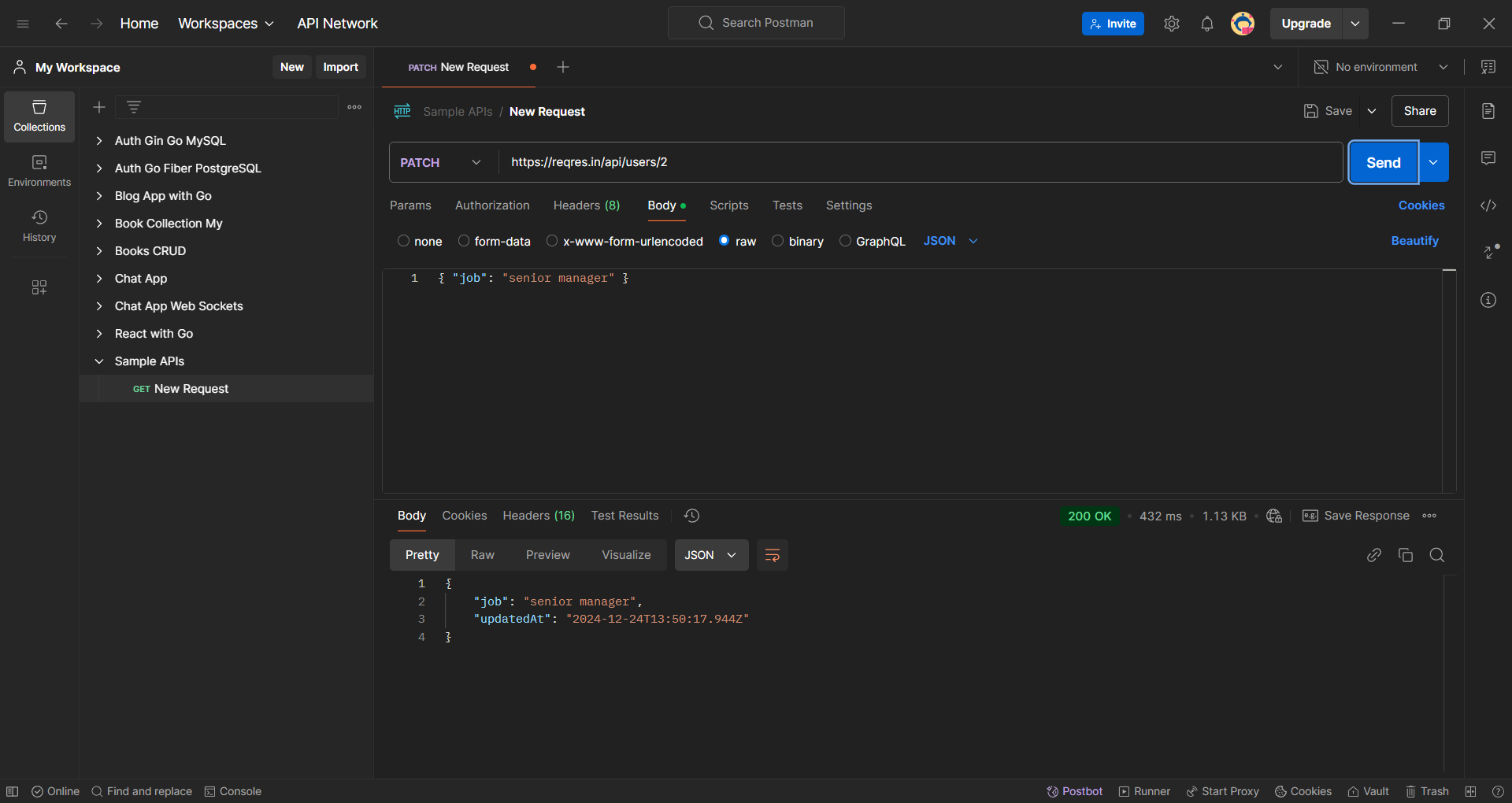
- PUT: `PUT /api/users/2` with payload `{ "name": "John", "job": "manager" }`



- PATCH: `PATCH /api/users/2` with payload `{ "job": "senior manager" }`



- DELETE: `DELETE /api/users/2`



B. Assert for 200/201 response

- Use `then().statusCode(200)` for GET, PUT, PATCH, DELETE.

- Use `then().statusCode(201)` for POST.

C. Verify “first\_name” key value from GET response

- Extract the value of the `first\_name` field and assert it matches the expected value.

D. Extract “email” from GET request and add it to the subsequent POST request payload and hit the request

- Retrieve the `email` field from GET response.

- Include it in the POST request payload, e.g., `{ "name": "John", "job": "developer", "email": "user@example.com" }`.

E. Write a function to verify presence of Message in response body

- Create a function that checks if the response body contains a specific message and returns a boolean value.