**✅ TEST CASES FOR GATORREST INN BACKEND**

**🔐 1. User Registration**

**Purpose:** Register a new user with a valid email and password

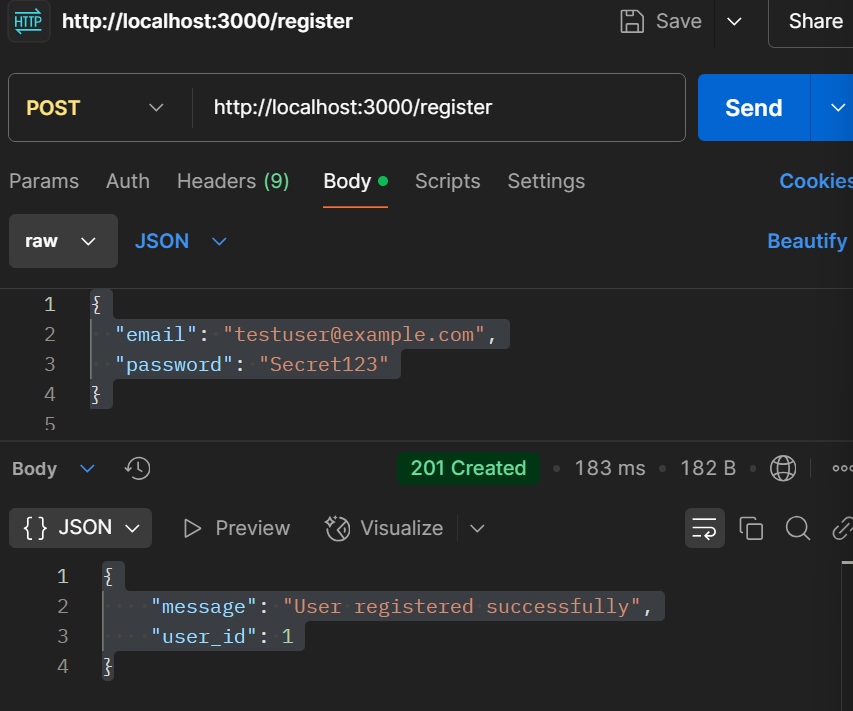
* **Method:** POST
* **URL:** http://localhost:3000/register
* **Headers:**
  + Content-Type: application/json
* **Body:**

{

"email": "testuser@example.com",

"password": "Secret123"

}



**🔐 2. User Login**

**Purpose:** Log in and receive a JWT token

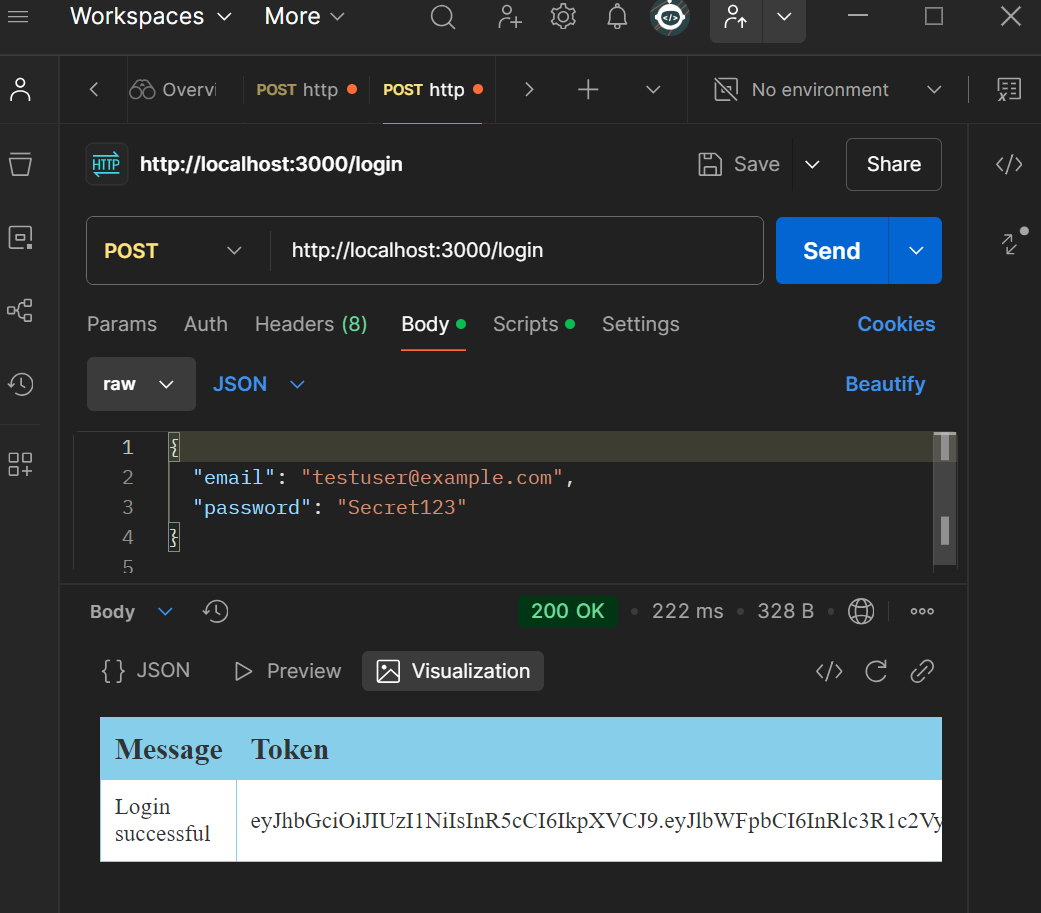
* **Method:** POST
* **URL:** http://localhost:3000/login
* **Headers:**
  + Content-Type: application/json
* **Body:**

{

"email": "testuser@example.com",

"password": "Secret123"

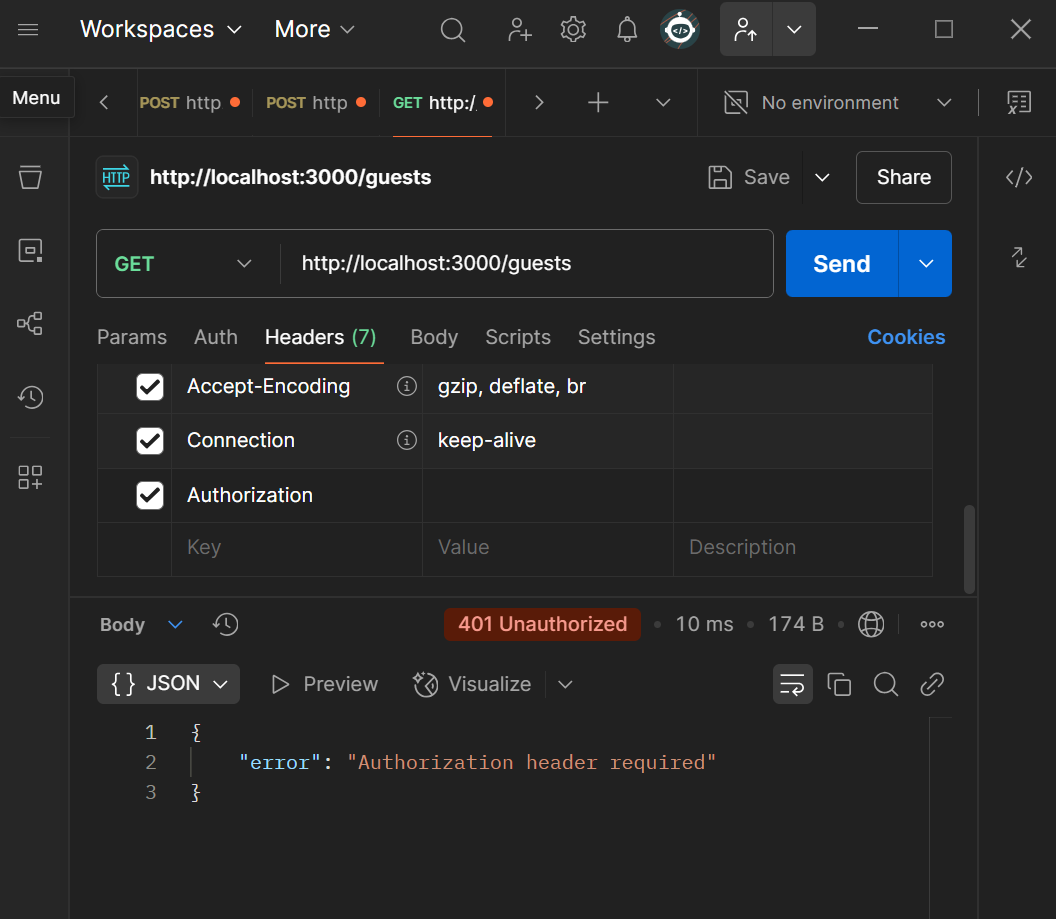
}



**🔐 3. Access Protected Route Without Token**

**Purpose:** Ensure that unauthorized access is blocked

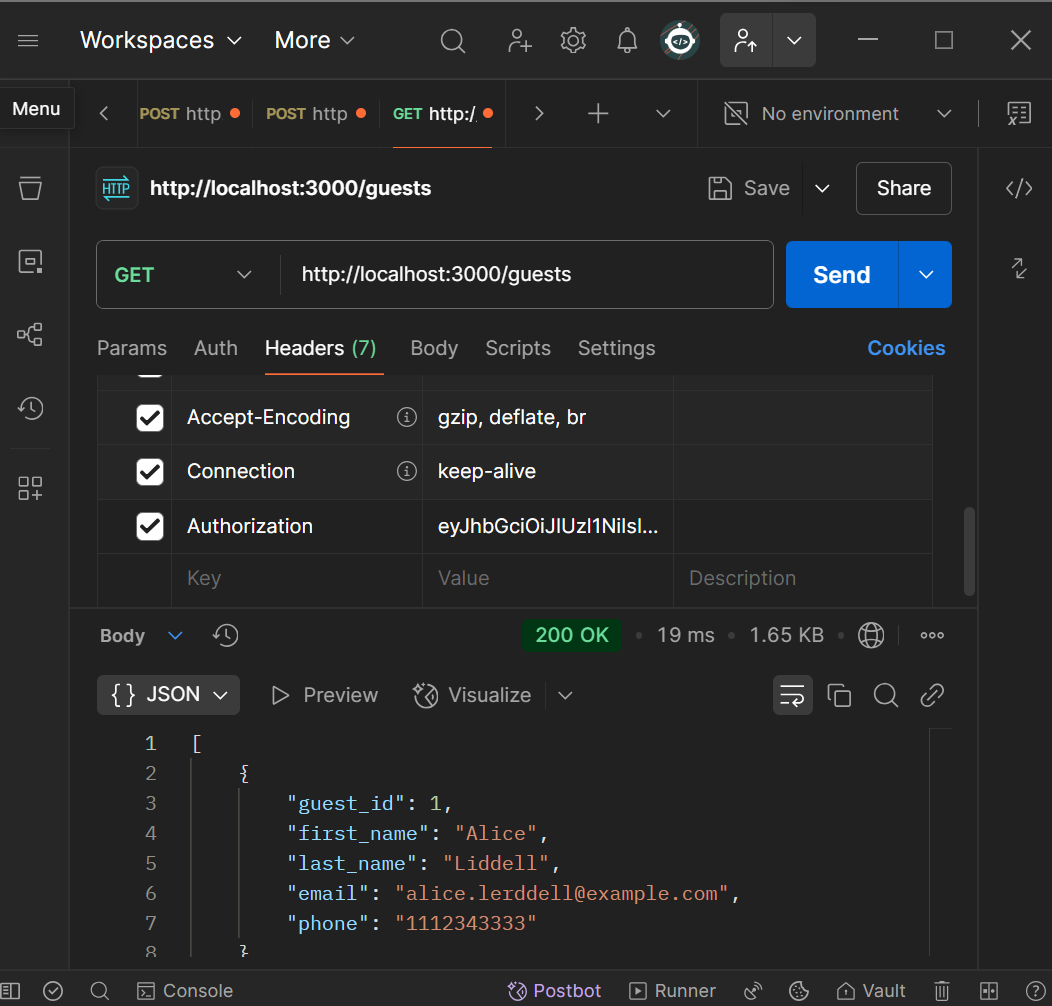
* **Method:** GET
* **URL:** http://localhost:3000/guests
* **Headers:** *(No Authorization header)*



**✅ 4. Get All Guests (With Token)**

**Purpose:** Fetch all guests linked to the authenticated user

* **Method:** GET
* **URL:** http://localhost:3000/guests
* **Headers:**
  + Authorization: <JWT token>



**✅ 5. Create a Guest**

**Purpose:** Add a guest linked to the authenticated user

* **Method:** POST
* **URL:** http://localhost:3000/guests
* **Headers:**
  + Authorization: <JWT token>
  + Content-Type: application/json
* **Body:**

{

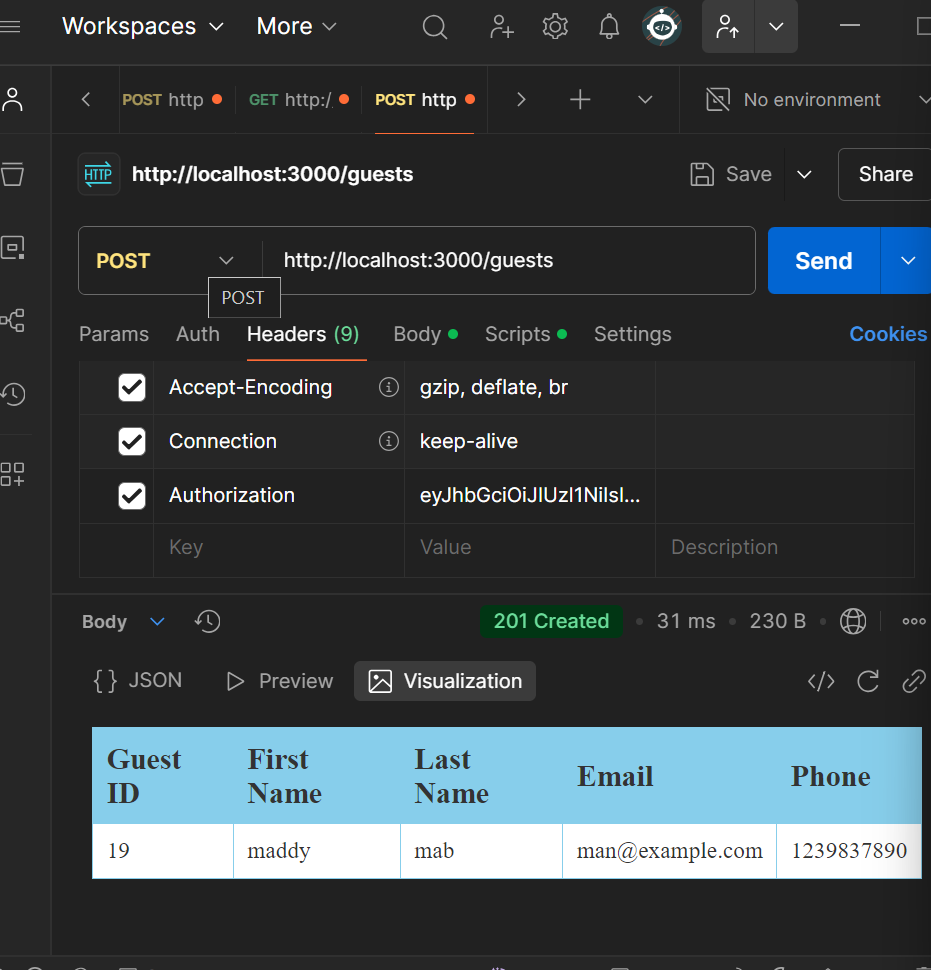
"first\_name": "maddy",

"last\_name": "mab",

"email": "man@example.com",

"phone": "1239837890"

}



**✅ 6. Create a Reservation**

**Purpose:** Reserve a room for a guest and associate it with the logged-in user

* **Method:** POST
* **URL:** http://localhost:3000/reservations
* **Headers:**
  + Authorization: <JWT token>
  + Content-Type: application/json
* **Body:**

{

"guest\_id": 1,

"room\_id": 2,

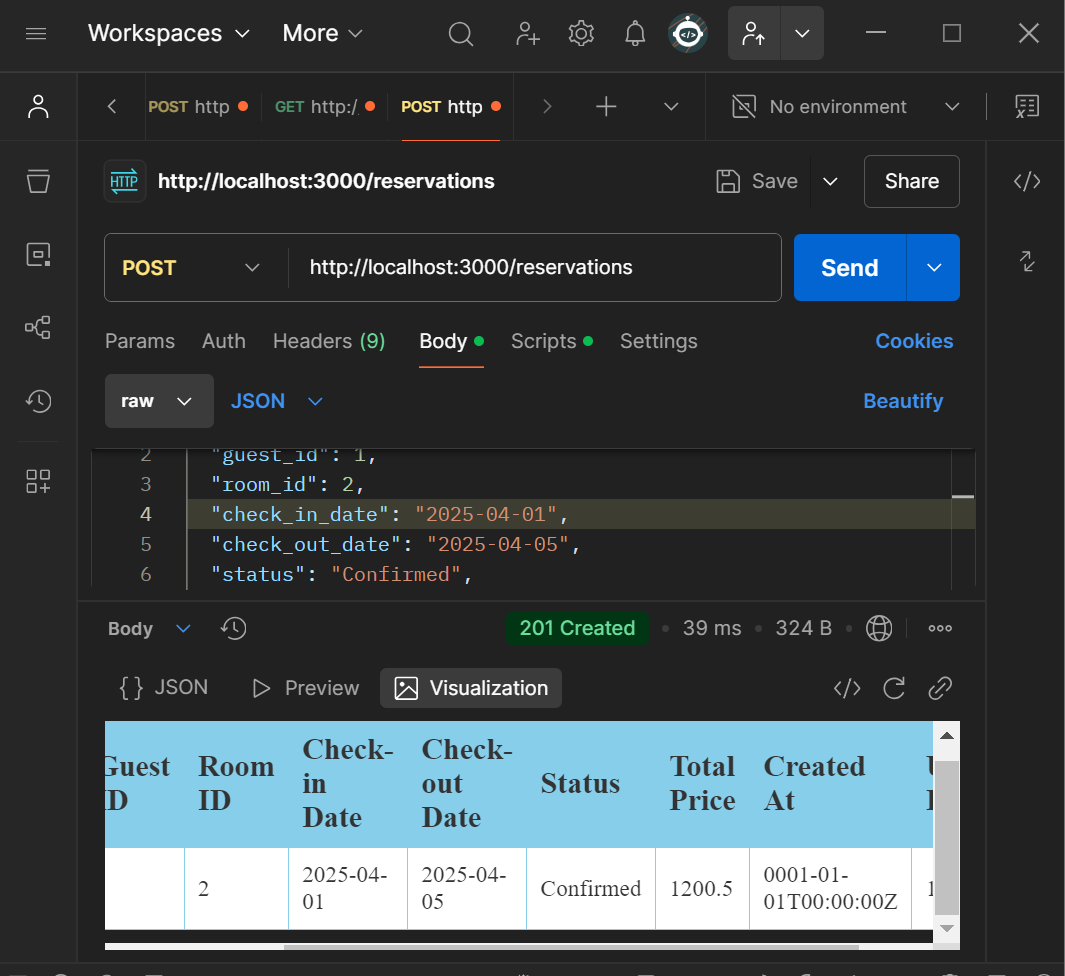
"check\_in\_date": "2025-04-01",

"check\_out\_date": "2025-04-05",

"status": "Confirmed",

"total\_price": 1200.50

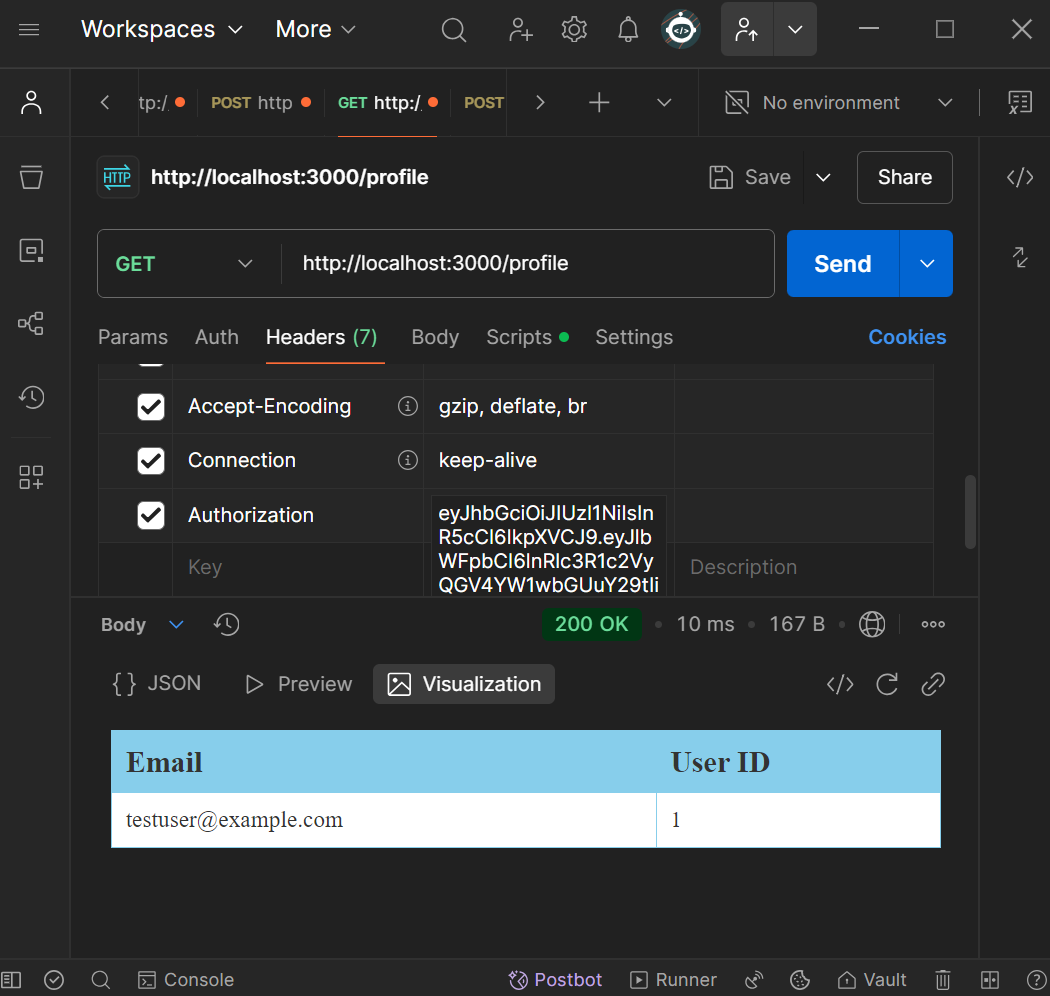
}



**✅ 7. Get User Profile**

**Purpose:** Get the authenticated user’s email and ID

* **Method:** GET
* **URL:** http://localhost:3000/profile
* **Headers:**
  + Authorization: <JWT token>



Created a file with all previous testcases implemented in sprint 2 .

Imported this on postman to run all the test cases at a time.



Results are given in the screenshot below.

