Name: Deep Patel

Stu ID: 101415117

GitHub Repository Link: https://github.com/Deeppatel91/COMP3133_101415117_Assignment1

COMP 3133 Assignment-1

Test It Locally: http://localhost:5000/graphql

Sample User for Testing:

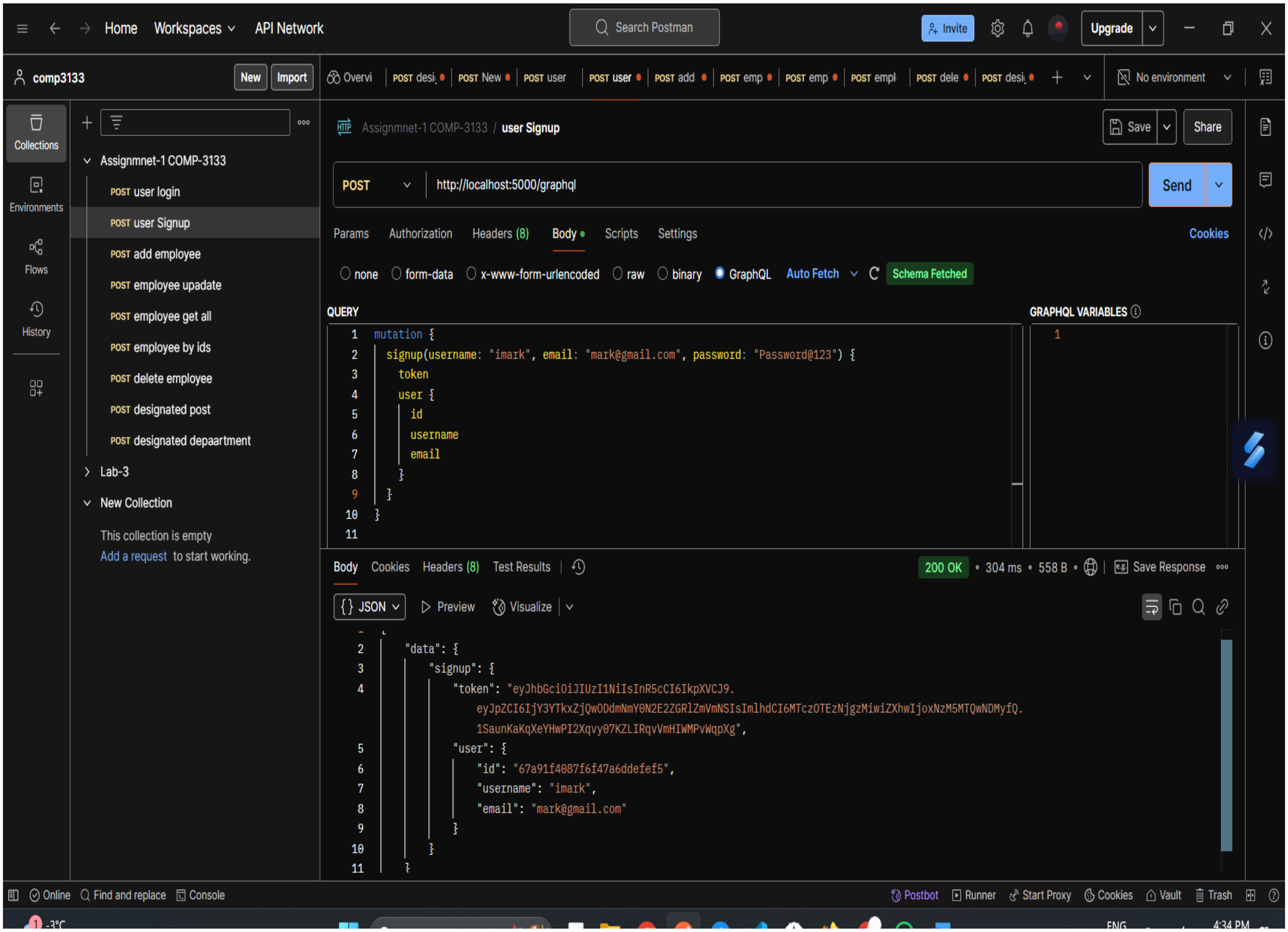
Username: ico

Email: ico@example.com

Password: Password@123

Screen Shots:

1) **User Sign-Up Screenshots: Mutation**



POST http://localhost:5000/graphql

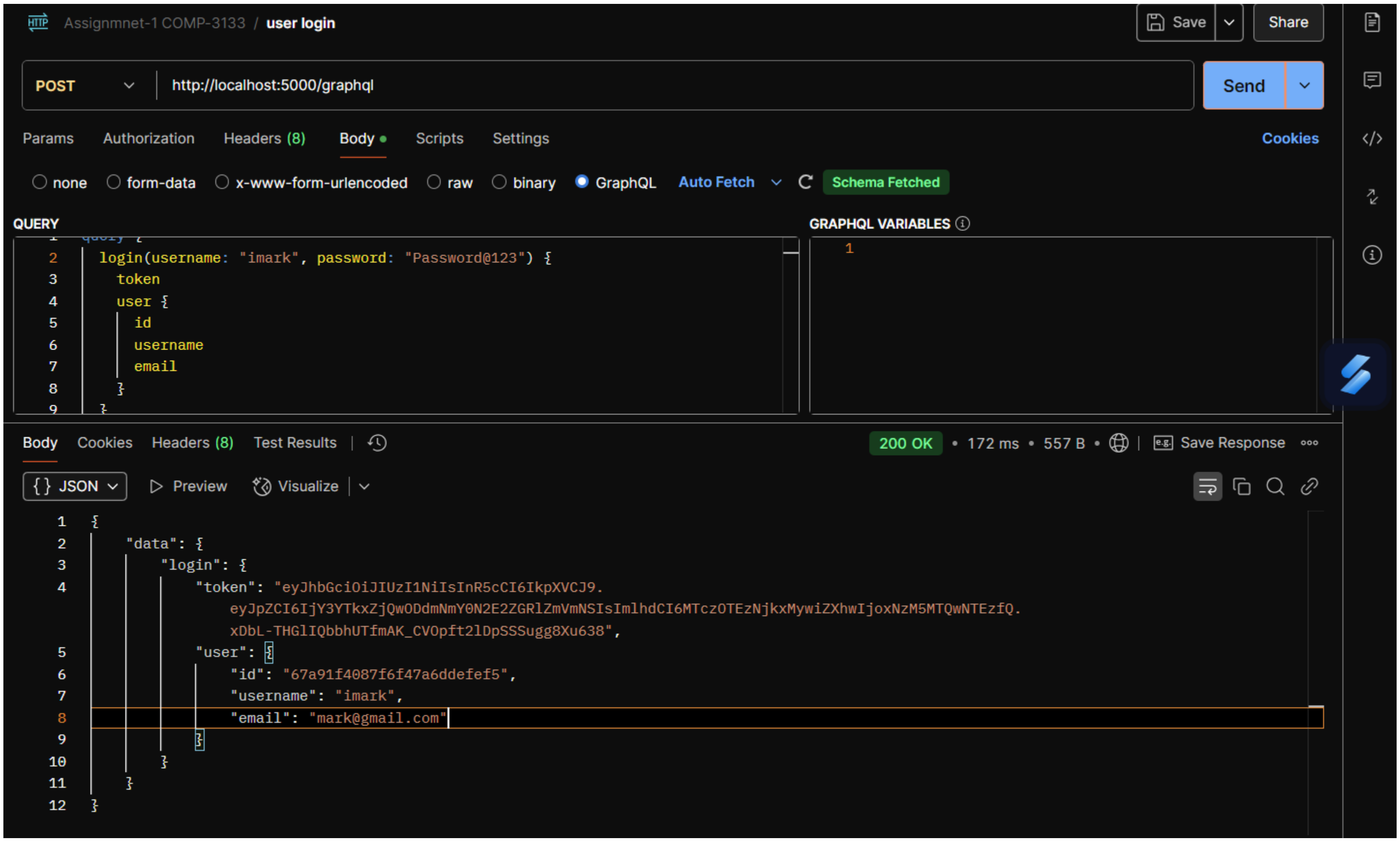
QUERY

```graphql
mutation {
  signup(username: "imark", email: "mark@gmail.com", password: "Password@123") {
    token
    user {
      id
      username
      email
    }
  }
}
```

Response:

```json
{
    "data": {
        "signup": {
            "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
                eyJpZCI6IjY3YTkxZjQwODdmNmY0N2E2ZGRlZmVmNSIsImlhdCI6MTczOTEzNjgzMiwiZXhwIjoxNzM5MTQwNDMyfQ.
                1SaunKaKqXeYHwPI2Xqvy07KZLIRqvVmHIWMPvWqpXg",
            "user": {
                "id": "67a91f4087f6f47a6ddefef5",
                "username": "imark",
                "email": "mark@gmail.com"
            }
        }
    }
}
```

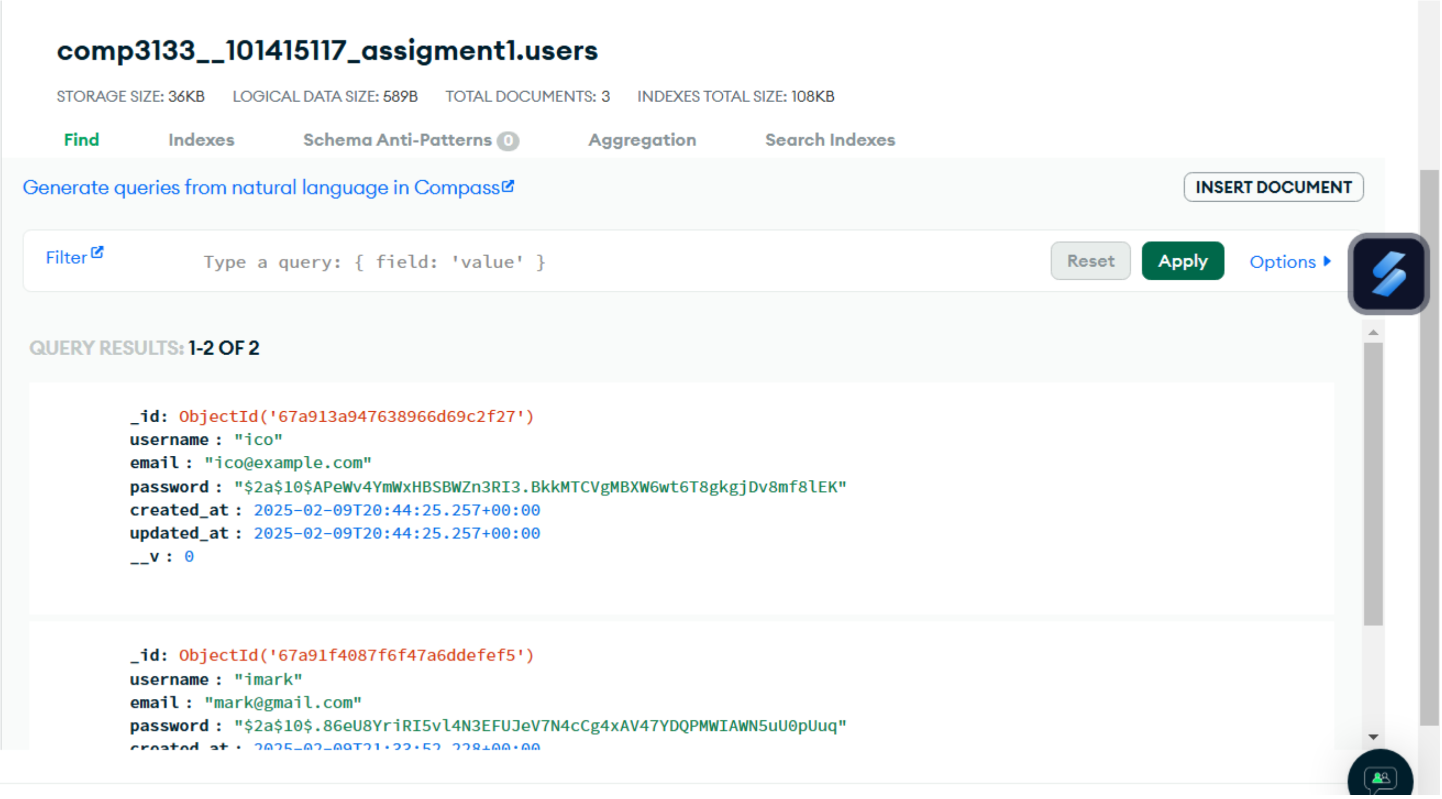
2) User Login: Query


Assignmnet-1 COMP-3133 / user login
Save
Share
POST
http://localhost:5000/graphql
Send
Params
Authorization
Headers (8)
Body
Scripts
Settings
Cookies
○ none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ● GraphQL
Auto Fetch
Schema Fetched
QUERY
GRAPHQL VARIABLES

```
query {
  login(username: "imark", password: "Password@123") {
    token
    user {
      id
      username
      email
    }
  }
```

```
{
    "data": {
        "login": {
            "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
                eyJpZCI6IjY3YTkxZjQwODdmNmY0N2E2ZGRlZmVmNSIsImlhdCI6MTczOTEzNjkxMywiZXhwIjoxNzM5MTQwNTEzfQ.
                xDbL-THGlIQbbhUTfmAK_CVOpft2lDpSSSugg8Xu638",
            "user": {
                "id": "67a91f4087f6f47a6ddefef5",
                "username": "imark",
                "email": "mark@gmail.com"
            }
        }
    }
}
```

3) User Database:



comp3133__101415117_assigment1.users

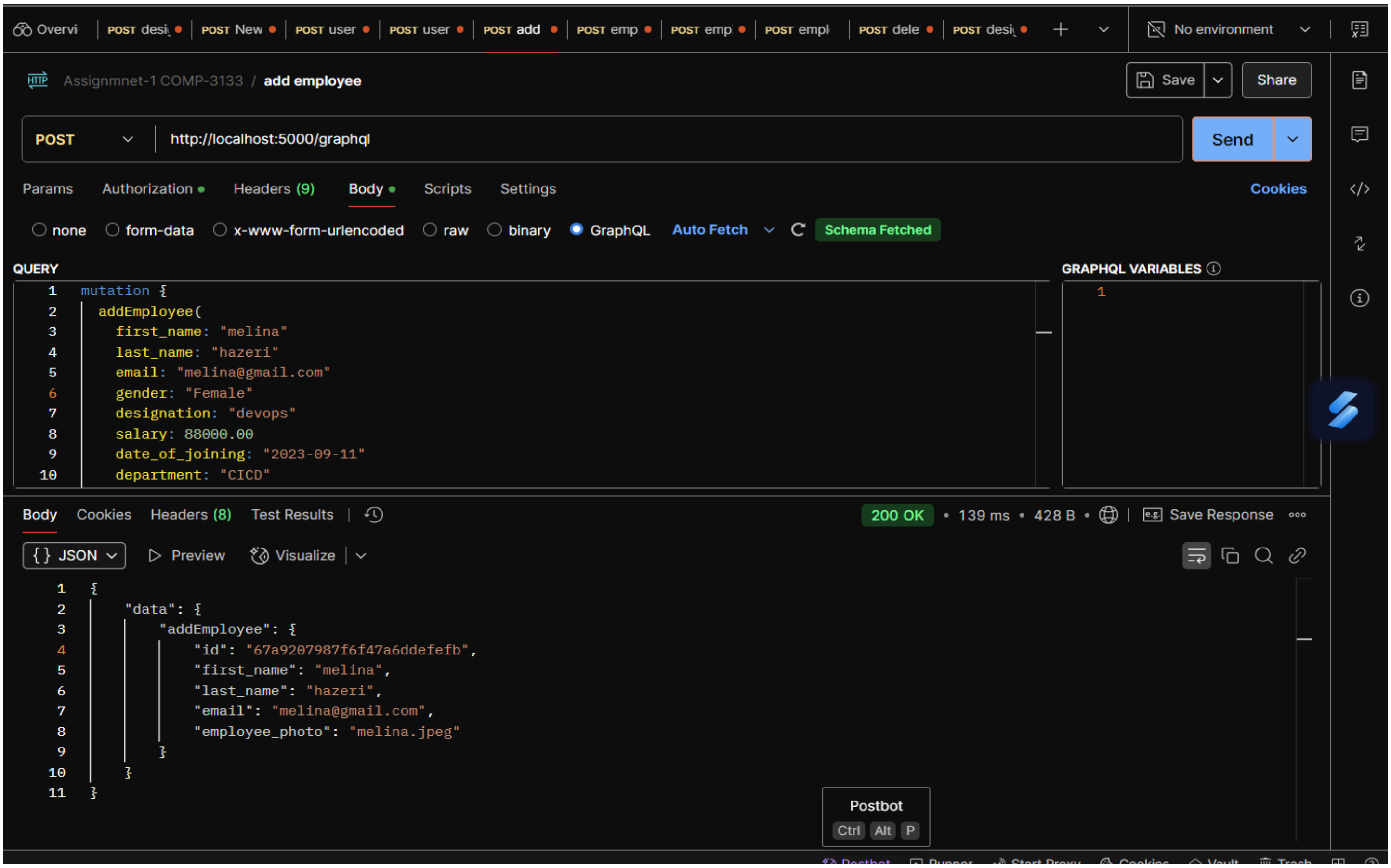
STORAGE SIZE: 36KB LOGICAL DATA SIZE: 589B TOTAL DOCUMENTS: 3 INDEXES TOTAL SIZE: 108KB

QUERY RESULTS: 1-2 OF 2

```
_id: ObjectId('67a913a947638966d69c2f27')
username : "ico"
email : "ico@example.com"
password : "$2a$10$APeWv4YmWxHBSBWZn3RI3.BkkMTCVgMBXW6wt6T8gkgjDv8mf8lEK"
created_at : 2025-02-09T20:44:25.257+00:00
updated_at : 2025-02-09T20:44:25.257+00:00
__v : 0
```

```
_id: ObjectId('67a91f4087f6f47a6ddefef5')
username : "imark"
email : "mark@gmail.com"
password : "$2a$10$.86eU8YriRI5vl4N3EFUJeV7N4cCg4xAV47YDQPMWIAWN5uU0pUuq"
created_at : 2025-02-09T21:33:52.228+00:00
```

4) Add Employee Protected JWT Token Required Authorized -Mutation



POST http://localhost:5000/graphql

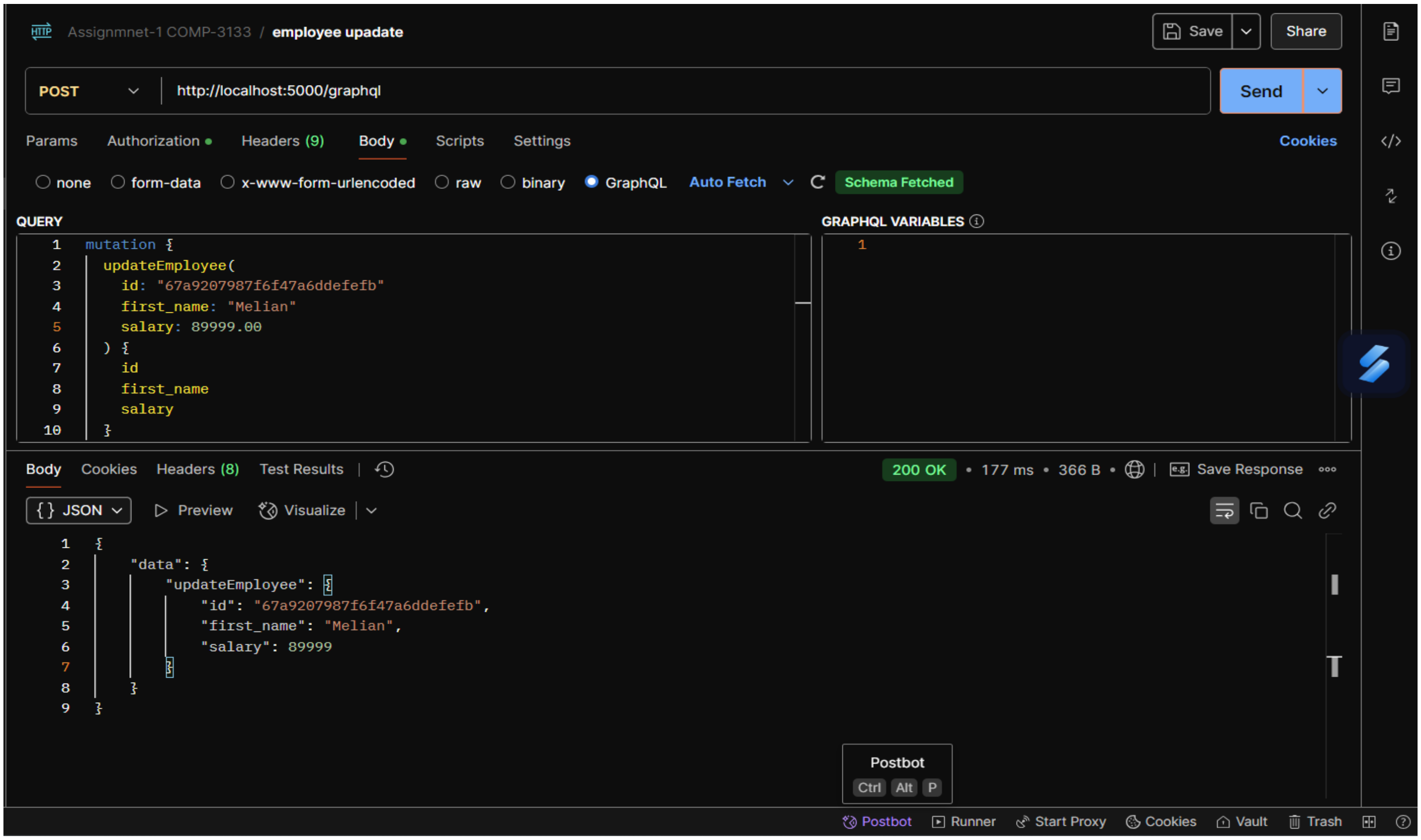
QUERY

```
mutation {
  addEmployee(
    first_name: "melina"
    last_name: "hazeri"
    email: "melina@gmail.com"
    gender: "Female"
    designation: "devops"
    salary: 88000.00
    date_of_joining: "2023-09-11"
    department: "CICD"
```

200 OK • 139 ms • 428 B

```
{
    "data": {
        "addEmployee": {
            "id": "67a9207987f6f47a6ddefefb",
            "first_name": "melina",
            "last_name": "hazeri",
            "email": "melina@gmail.com",
            "employee_photo": "melina.jpeg"
        }
    }
}
```

5) Update Employee Protected JWT Token Required Authorized - Mutation



POST http://localhost:5000/graphql

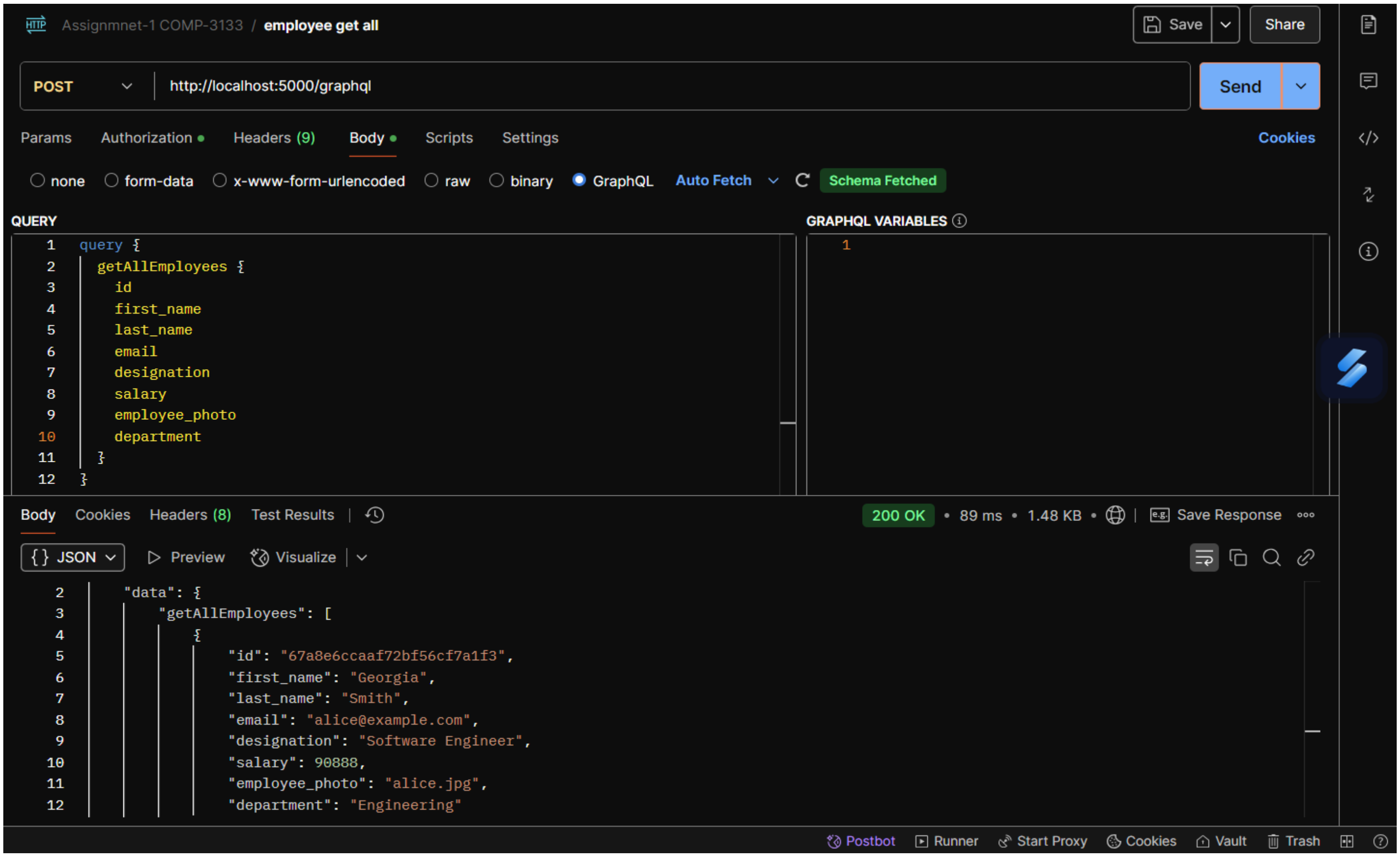
QUERY

```
mutation {
  updateEmployee(
    id: "67a9207987f6f47a6ddefefb"
    first_name: "Melian"
    salary: 89999.00
  ) {
    id
    first_name
    salary
  }
```

200 OK • 177 ms • 366 B

```
{
    "data": {
        "updateEmployee": {
            "id": "67a9207987f6f47a6ddefefb",
            "first_name": "Melian",
            "salary": 89999
        }
    }
}
```

6) Get All Employee Protected JWT Token Required Authorized – Query



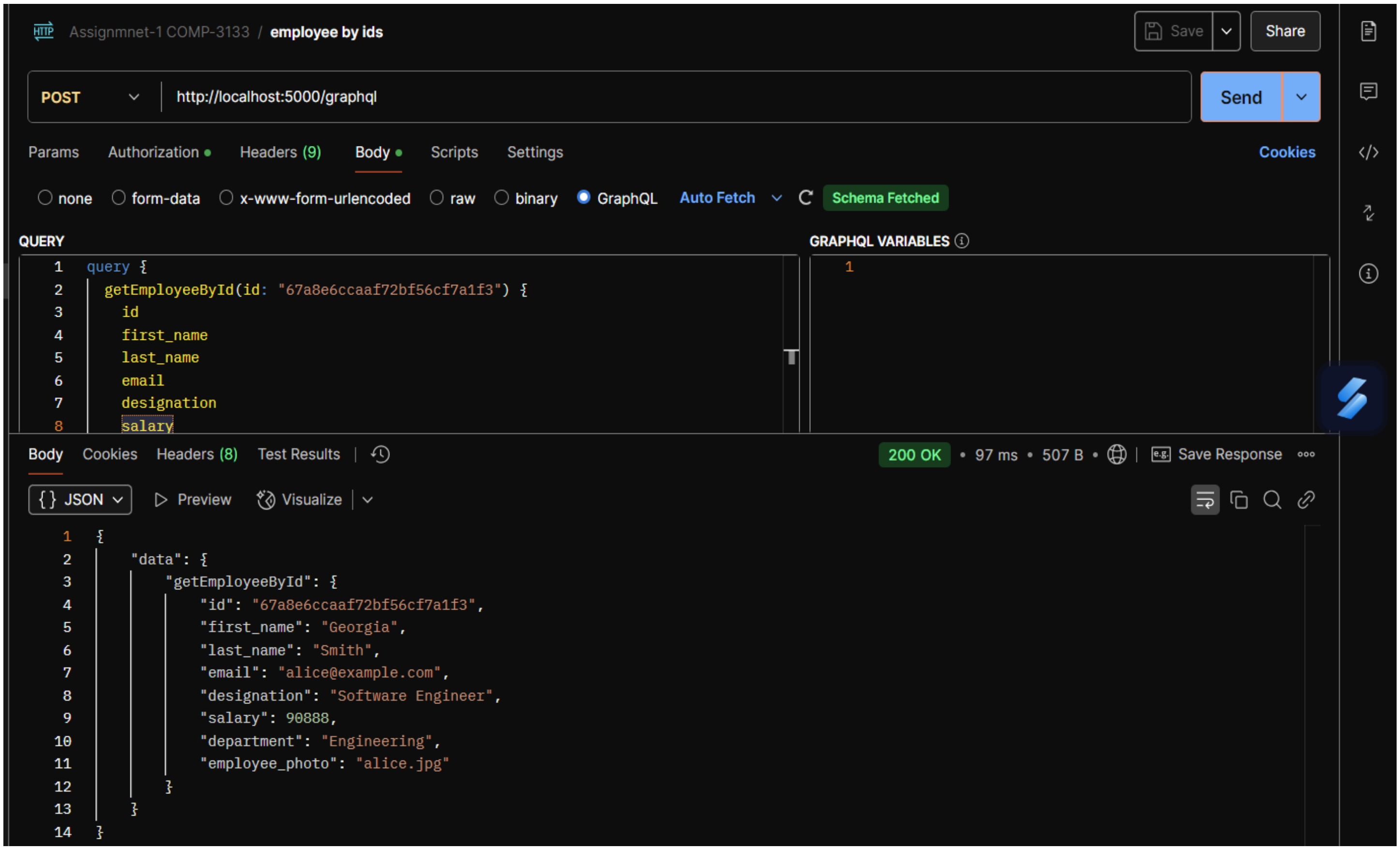
POST http://localhost:5000/graphql

```
query {
  getAllEmployees {
    id
    first_name
    last_name
    email
    designation
    salary
    employee_photo
    department
  }
}
```

200 OK • 89 ms • 1.48 KB

```
    "data": {
        "getAllEmployees": [
            {
                "id": "67a8e6ccaaf72bf56cf7a1f3",
                "first_name": "Georgia",
                "last_name": "Smith",
                "email": "alice@example.com",
                "designation": "Software Engineer",
                "salary": 90888,
                "employee_photo": "alice.jpg",
                "department": "Engineering"
```

7) Employee By ID Protected JWT Token Required Authorized – Query



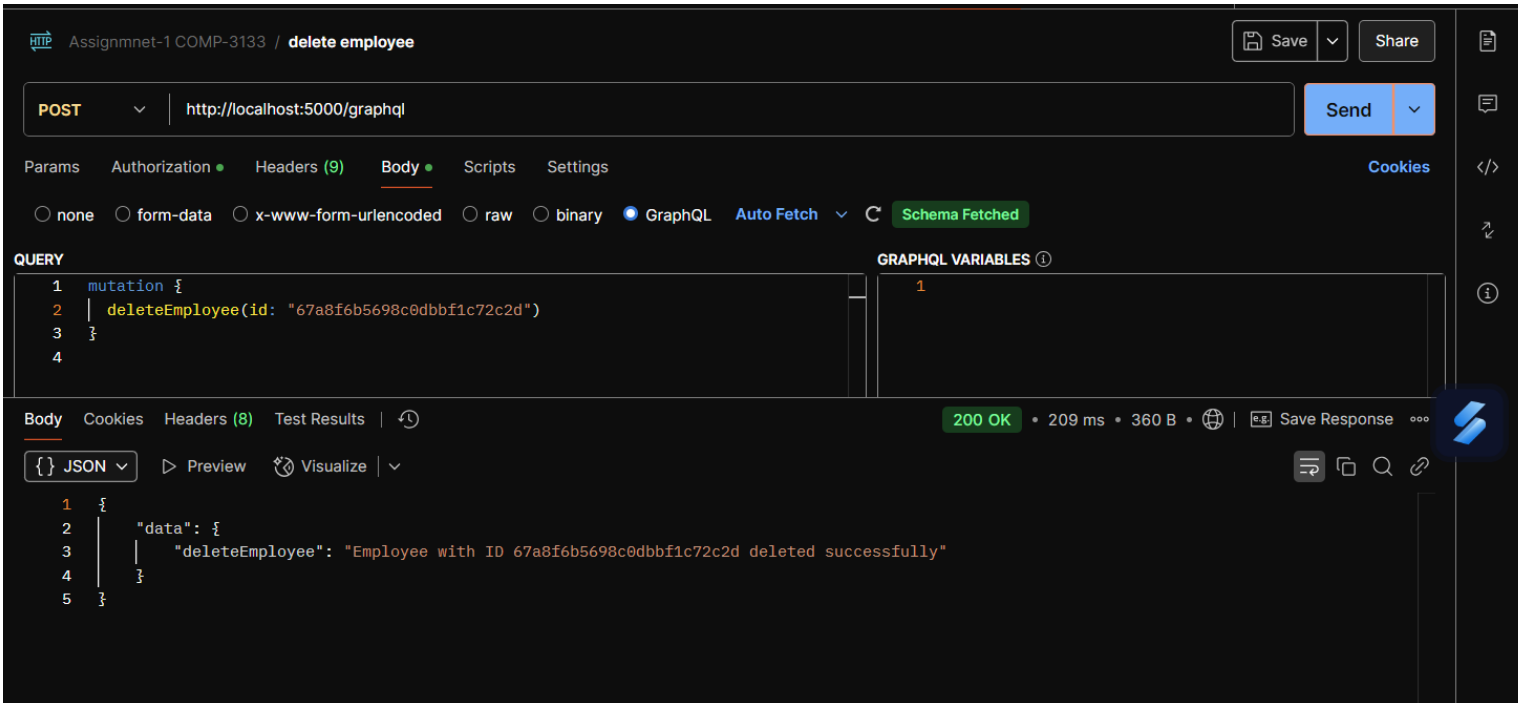
POST http://localhost:5000/graphql

```
query {
  getEmployeeById(id: "67a8e6ccaaf72bf56cf7a1f3") {
    id
    first_name
    last_name
    email
    designation
    salary
```

200 OK • 97 ms • 507 B

```
{
    "data": {
        "getEmployeeById": {
            "id": "67a8e6ccaaf72bf56cf7a1f3",
            "first_name": "Georgia",
            "last_name": "Smith",
            "email": "alice@example.com",
            "designation": "Software Engineer",
            "salary": 90888,
            "department": "Engineering",
            "employee_photo": "alice.jpg"
        }
    }
}
```

8) Delete Employee Protected JWT Token Required Authorized – Mutation



POST http://localhost:5000/graphql

QUERY

```
mutation {
  deleteEmployee(id: "67a8f6b5698c0dbbf1c72c2d")
}

```

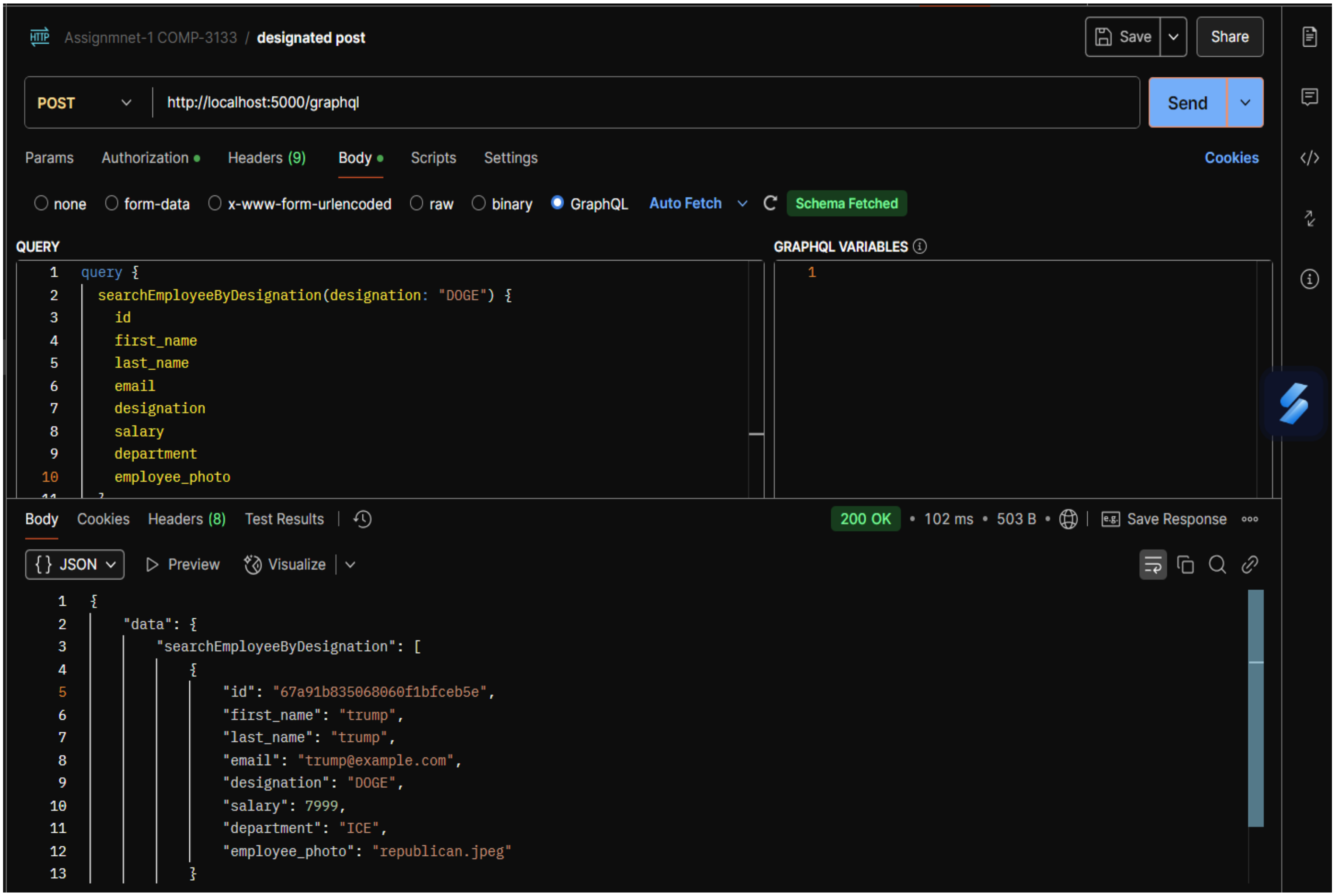
GRAPHQL VARIABLES

```
1
```

200 OK • 209 ms • 360 B

```
{
    "data": {
        "deleteEmployee": "Employee with ID 67a8f6b5698c0dbbf1c72c2d deleted successfully"
    }
}
```

9) Search Designation Employee Protected JWT Token Required Authorized – Query



POST http://localhost:5000/graphql

QUERY

```
query {
  searchEmployeeByDesignation(designation: "DOGE") {
    id
    first_name
    last_name
    email
    designation
    salary
    department
    employee_photo
  }
```

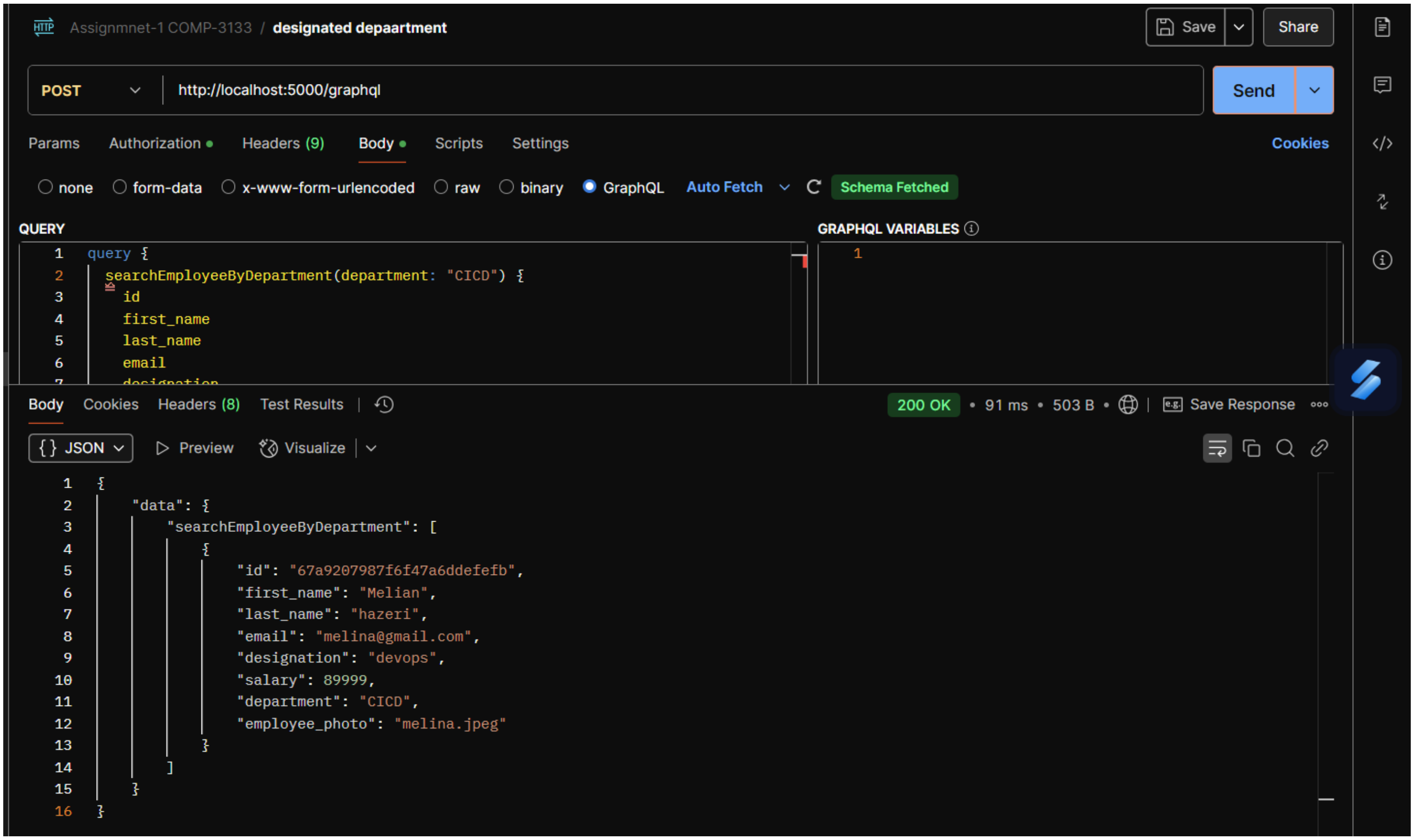
GRAPHQL VARIABLES

```
1
```

200 OK • 102 ms • 503 B

```
{
    "data": {
        "searchEmployeeByDesignation": [
            {
                "id": "67a91b835068060f1bfceb5e",
                "first_name": "trump",
                "last_name": "trump",
                "email": "trump@example.com",
                "designation": "DOGE",
                "salary": 7999,
                "department": "ICE",
                "employee_photo": "republican.jpeg"
            }
```

10) Search Department Employee Protected JWT Token Required Authorized – Query



POST http://localhost:5000/graphql

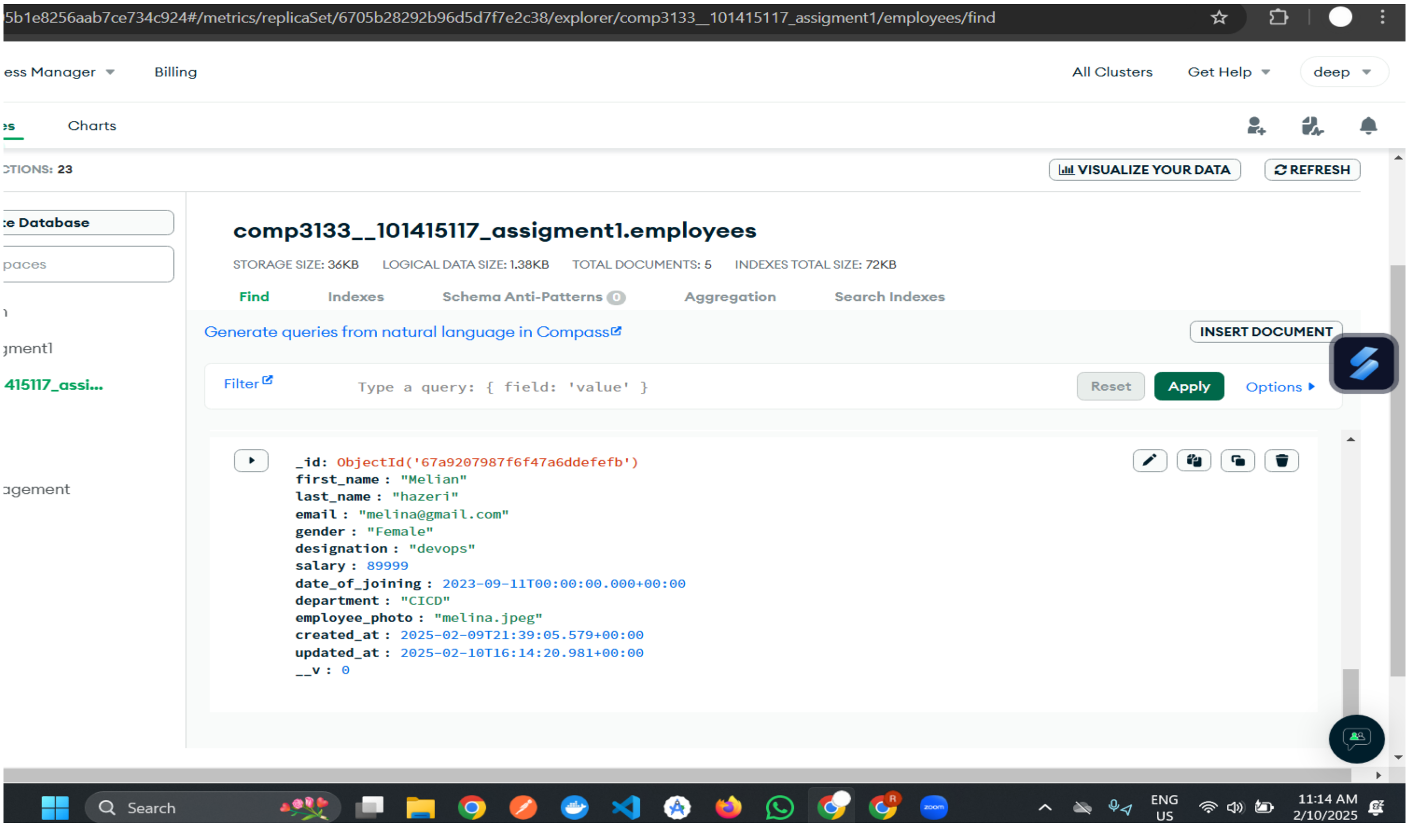
Send

```
query {
  searchEmployeeByDepartment(department: "CICD") {
    id
    first_name
    last_name
    email
    designation
```

200 OK • 91 ms • 503 B

```
{
    "data": {
        "searchEmployeeByDepartment": [
            {
                "id": "67a9207987f6f47a6ddefefb",
                "first_name": "Melian",
                "last_name": "hazeri",
                "email": "melina@gmail.com",
                "designation": "devops",
                "salary": 89999,
                "department": "CICD",
                "employee_photo": "melina.jpeg"
            }
        ]
    }
}
```

11) Employee Database Documentation:



comp3133__101415117_assigment1.employees

STORAGE SIZE: 36KB LOGICAL DATA SIZE: 1.38KB TOTAL DOCUMENTS: 5 INDEXES TOTAL SIZE: 72KB

```
_id: ObjectId('67a9207987f6f47a6ddefefb')
first_name : "Melian"
last_name : "hazeri"
email : "melina@gmail.com"
gender : "Female"
designation : "devops"
salary : 89999
date_of_joining : 2023-09-11T00:00:00.000+00:00
department : "CICD"
employee_photo : "melina.jpeg"
created_at : 2025-02-09T21:39:05.579+00:00
updated_at : 2025-02-10T16:14:20.981+00:00
__v : 0
```

- Passwords are hashed before saving.

- JWT is required for protected routes (except login/signup).

- GraphQL API tested using Postman.

- Postman collection is attached for easy testing.