

MONGODB SCREENSHOT:

The screenshot shows the MongoDB Atlas web interface. The top navigation bar includes 'Atlas', 'Gokul's Org', 'Access Manager', 'Billing', 'All Clusters', 'Get Help', and a user profile 'Gokul'. The main header shows 'Project 0' and 'Data Services'. The left sidebar contains navigation links for Overview, DEPLOYMENT, Database, SERVICES, and SECURITY. The main content area displays 'MyCluster' with version 7.0.14 and region AZURE Toronto (canadacentral). The 'Collections' tab is active, showing a table of collections for the 'comp3123-assign1' database. The table has columns: Collection Name, Documents, Logical Data Size, Avg Document Size, Storage Size, Indexes, Index Size, and Avg Index Size. The collections listed are 'employees' and 'users'.

Collection Name	Documents	Logical Data Size	Avg Document Size	Storage Size	Indexes	Index Size	Avg Index Size
employees	2	489B	245B	36KB	1	36KB	36KB
users	3	597B	199B	36KB	1	36KB	36KB

POSTMAN SCREENSHOTS:

<http://localhost:8081/api/v1/user/signup>

The screenshot shows the Postman API client interface. The left sidebar displays the workspace 'My Workspace' with a collection 'comp3123-assign1'. The main panel shows a POST request to 'http://localhost:8081/api/v1/user/signup'. The request body is in JSON format with the following data: 'username': 'johndoe', 'email': 'johndoe@example.com', and 'password': 'password123'. The response is also in JSON format, showing a successful status: 'message': 'User created successfully.' and 'user_id': '670c64f4108b27ad89a3a968'. The status bar at the bottom indicates '201 Created' with a response time of 146 ms and a body size of 317 B.

```
POST http://localhost:8081/api/v1/user/signup

{
  "username": "johndoe",
  "email": "johndoe@example.com",
  "password": "password123"
}
```

```
{
  "message": "User created successfully.",
  "user_id": "670c64f4108b27ad89a3a968"
}
```

The screenshot shows the VS Code interface with the REST Client extension. The left sidebar has a 'My Workspace' section with a file explorer showing a project named 'comp3123-assign1'. Inside this project, there are several HTTP request files, including 'comp3123-assign1.http'. The main editor area displays a 'POST' request to 'http://localhost:8081/api/v1/user/login'. The request body is in JSON format: { 'email': 'johndoe@example.com', 'password': 'password123' }. The response section shows a '200 OK' status with a response time of 120 ms and a body containing a success message and a JWT token.

The screenshot displays the Postman application interface. On the left sidebar, under 'My Workspace', there is a collection named 'comp3123-assign1' which contains several API requests. The main panel shows a GET request to 'http://localhost:8081/api/v1/emp/employees'. The response status is '200 OK' with a response time of '38 ms' and a size of '869 B'. The response body is a JSON object representing an employee record. The bottom status bar shows the system clock as 8:37 PM and the weather as '25°C Mostly cloudy'.

My Workspace

- comp3123-assign1
 - POST http://localhost:8081/api/v1/us...
 - POST http://localhost:8081/api/v1/us...
 - GET http://localhost:8081/api/v1/e...
 - POST http://localhost:8081/api/v1/e...
 - GET http://localhost:8081/api/v1/e...
 - PUT http://localhost:8081/api/v1/e...
 - DEL http://localhost:8081/api/v1/e...
- comp3123-exec06
- lab5
- microservices

Request Details

Method: GET
URL: http://localhost:8081/api/v1/emp/employees

Response

Status: 200 OK
Time: 38 ms
Size: 869 B

Body

```

1  [
2    {
3      "_id": "67071f4a67380f778df75a08",
4      "first_name": "Babu",
5      "last_name": "Antony",
6      "email": "john.doe@example.com",
7      "position": "Senior Designer",
8      "Salary": 75000,
9      "date_of_joining": "2023-09-15T00:00:00.000Z",
10     "department": "Development",
11     "created_at": "2024-10-10T00:26:50.849Z",
12     "updated_at": "2024-10-10T00:26:50.849Z",
13     "__v": 0,
  
```

System Status

Postbot Runner Start Proxy Cookies Vault Trash 8:37 PM 25°C Mostly cloudy

<http://localhost:8081/api/v1/emp/employees:>

The screenshot shows the Postman interface with a workspace named "My Workspace". The left sidebar displays a collection named "comp3123-assign1" with several API requests. The main panel shows a POST request to "http://localhost:8081/api/v1/emp/employees". The request body is a JSON object with the following fields: "first_name": "Alice", "last_name": "Johnson", "email": "alice.johnson@example.com", "position": "Designer", "salary": 85000, "date_of_joining": "2023-08-10T00:00:00.000Z", and "department": "Design". The response is a 201 Created status with a response time of 42 ms and a body size of 325 B. The response body is a JSON object with "message": "Employee created successfully." and "employee_id": "670c67e2fc91e0bf29aae06d".

```
POST http://localhost:8081/api/v1/emp/employees
```

```
{  "first_name": "Alice",  "last_name": "Johnson",  "email": "alice.johnson@example.com",  "position": "Designer",  "salary": 85000,  "date_of_joining": "2023-08-10T00:00:00.000Z",  "department": "Design"}
```

```
{  "message": "Employee created successfully.",  "employee_id": "670c67e2fc91e0bf29aae06d"}
```

<http://localhost:8081/api/v1/emp/employees/670c67e2fc91e0bf29aae06d:>

The screenshot shows the Postman interface with a workspace named "My Workspace". The left sidebar displays a collection named "comp3123-assign1" with several API requests. The main panel shows a PUT request to "http://localhost:8081/api/v1/emp/employees/670c67e2fc91e0bf29aae06d". The request body is a JSON object with the following fields: "first_name": "Babu", "position": "Senior Designer", and "salary": 5000. The response is a 200 OK status with a response time of 30 ms and a body size of 547 B. The response body is a JSON object with "_id": "670c67e2fc91e0bf29aae06d", "first_name": "Babu", "last_name": "Johnson", "email": "alice.johnson@example.com", "position": "Senior Designer", "salary": 5000, and "date_of_joining": "2023-08-10T00:00:00.000Z".

```
PUT http://localhost:8081/api/v1/emp/employees/670c67e2fc91e0bf29aae06d
```

```
{  "first_name": "Babu",  "position": "Senior Designer",  "salary": 5000}
```

```
{  "_id": "670c67e2fc91e0bf29aae06d",  "first_name": "Babu",  "last_name": "Johnson",  "email": "alice.johnson@example.com",  "position": "Senior Designer",  "salary": 5000,  "date_of_joining": "2023-08-10T00:00:00.000Z"}
```

<http://localhost:8081/api/v1/emp/employees/670c67e2fc91e0bf29aae06d:>

The screenshot shows the Postman interface with a PUT request to the endpoint `http://localhost:8081/api/v1/emp/employees/670c67e2fc91e0bf29aae06d`. The request body is a JSON object:

```
{
  "first_name": "Babu",
  "position": "Senior Designer",
  "salary": 5000
}
```

The response is a 200 OK status with a JSON object:

```
{
  "_id": "670c67e2fc91e0bf29aae06d",
  "first_name": "Babu",
  "last_name": "Johnson",
  "email": "alice.johnson@example.com",
  "position": "Senior Designer",
  "salary": 5000,
  "date_of_joining": "2023-08-10T00:00:00.000Z"
}
```

<http://localhost:8081/api/v1/emp/employees?eid=670c67e2fc91e0bf29aae06d:>

The screenshot shows the Postman interface with a DELETE request to the endpoint `http://localhost:8081/api/v1/emp/employees?eid=670c67e2fc91e0bf29aae06d`. The response is a 204 No Content status.