

Assignment 3(b)

b. Create four API using Node.JS, ExpressJS and MongoDB for CRUD Operations on Assignment 2 C.

CODE:

index.js

```
const { MongoClient } = require("mongodb");
```

```
const url = "mongodb://localhost:27017";
```

```
const database = "student";
```

```
const client = new MongoClient(url);
```

```
const dbConnect = async () => {  
  const result = await client.connect();  
  const db = await result.db(database);  
  return db.collection("profile");  
};
```

```
module.exports = dbConnect;
```

mongodb.js

```
const { MongoClient } = require("mongodb");
```

```
const url =
```

```
"mongodb+srv://latesh:latesh@cluster0.aqhqpqm.mongodb.net/student?retryWrites=true&w=majority";
```

```
const database = "student";
```

```
const client = new MongoClient(url);
```

```
const dbConnect = async () => {  
  const result = await client.connect();  
  const db = await result.db(database);  
  return db.collection("profile");  
};
```

```
module.exports = dbConnect;
```

OUTPUT:

The screenshot shows the MongoDB Compass interface. On the left, the 'Connections' panel lists various databases, with 'student' expanded to show a 'profile' collection. The main area displays the 'profile' collection with 2 documents. The documents are:

- `{ "_id": ObjectId("6806879f6b048b22dac73bf8"), "name": "samruddhi", "age": 19, "city": "nashik" }`
- `{ "_id": ObjectId("680687b06b048b22dac73bf9"), "name": "harsha", "age": 21, "city": "pune" }`
- `{ "_id": ObjectId("68068c016b7e5231e06bee0a"), "name": "shubham", "age": 23, "city": "Satana" }`

The screenshot shows the Postman interface. A GET request is sent to `http://localhost:3000/getData`. The response is a 200 OK status with a JSON body containing three documents, matching the output from MongoDB Compass.

```
{
  "_id": "6806879f6b048b22dac73bf8",
  "name": "samruddhi",
  "age": 19,
  "city": "nashik",
  "_id": "680687b06b048b22dac73bf9",
  "name": "harsha",
  "age": 21,
  "city": "pune",
  "_id": "68068c016b7e5231e06bee0a",
  "name": "shubham",
  "age": 23,
  "city": "Satana"
}
```