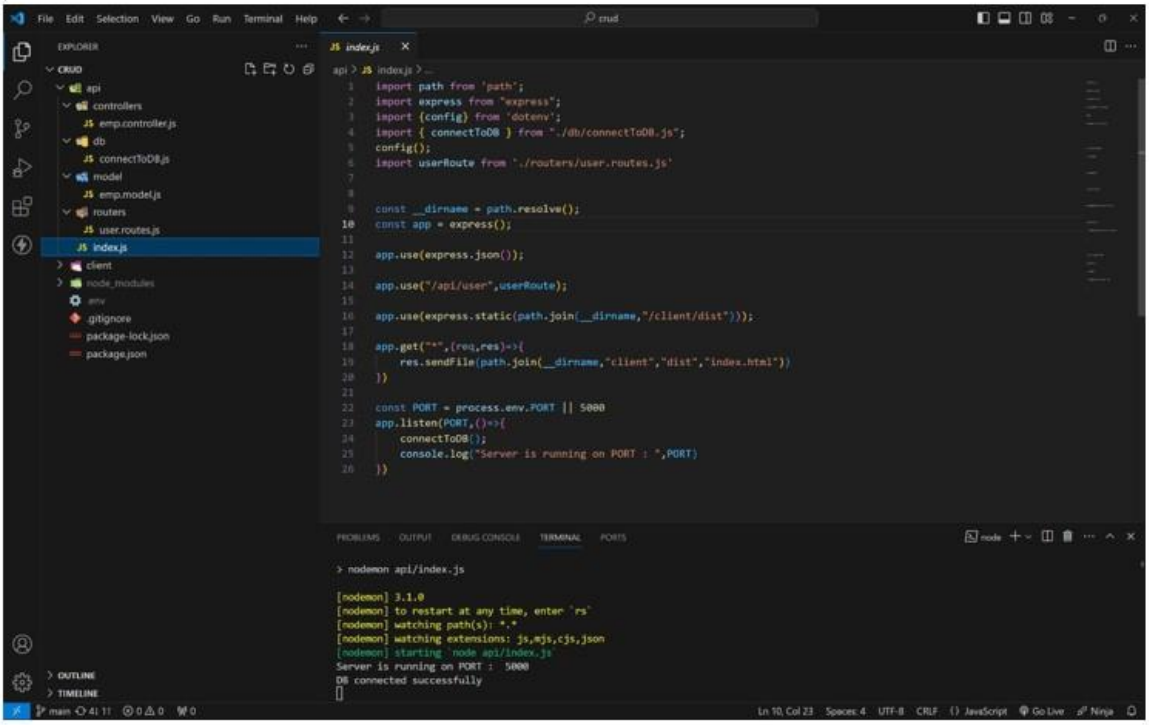
Creating a RESTful API usingexpress.js and creating a database and index in MongoDB.

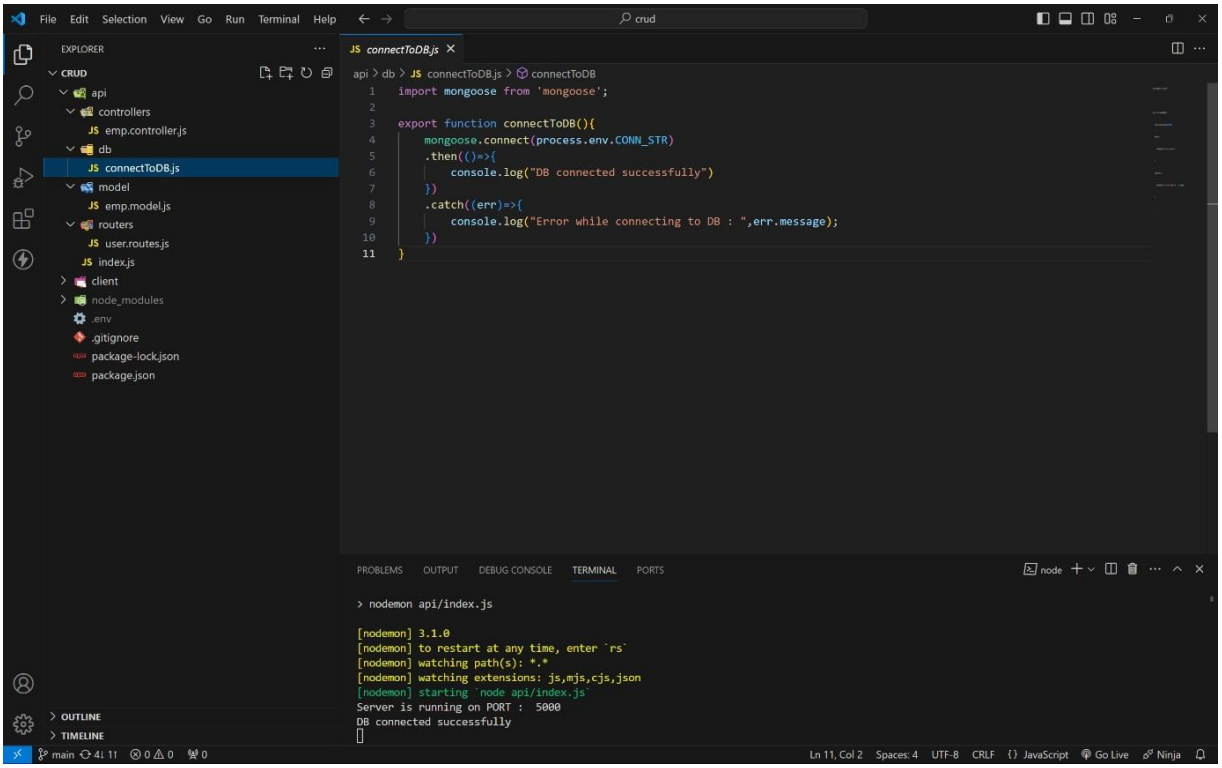
NAME:- THOTA DHARMA TEJA SRI
EMAIL ID: tejasrithota1234@gmail.com
PHONE NO:- 6301865779
ROLL NUMBER: 20ME1A0425(ECE)
COLLEGE NAME:- RAMACHANDRA COLLEGE OF ENGINEERING,ELURU

source code :

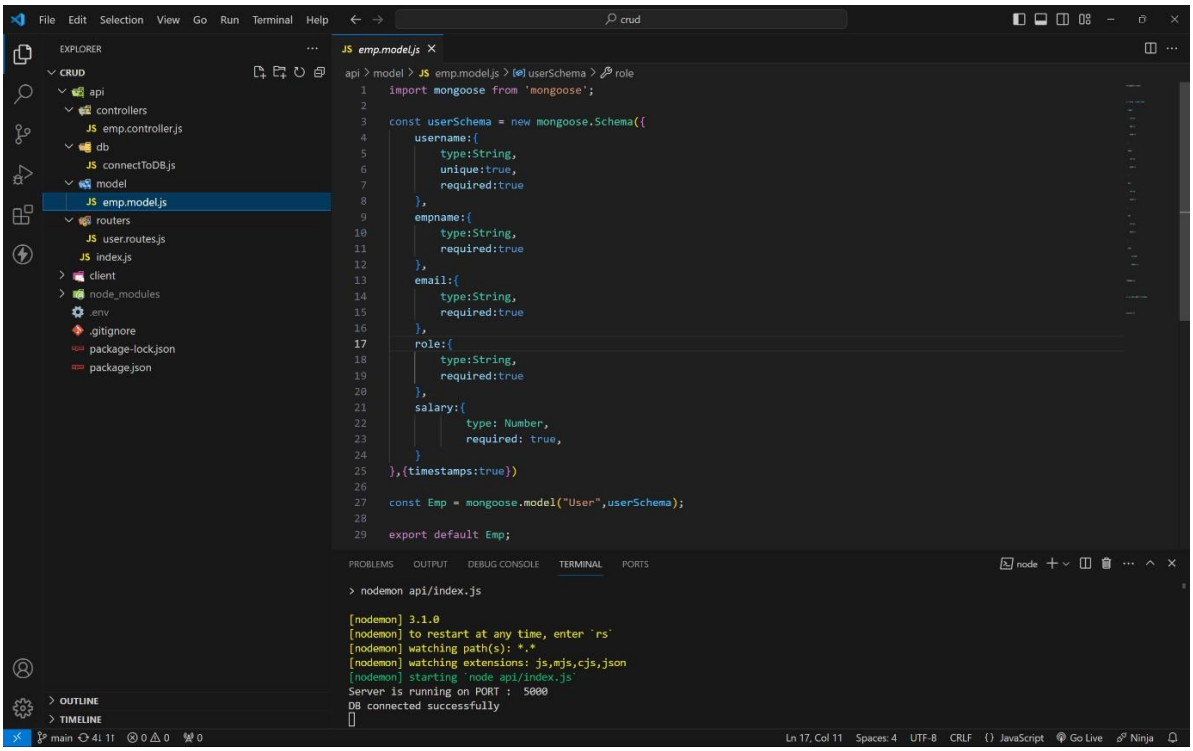
index.js file :



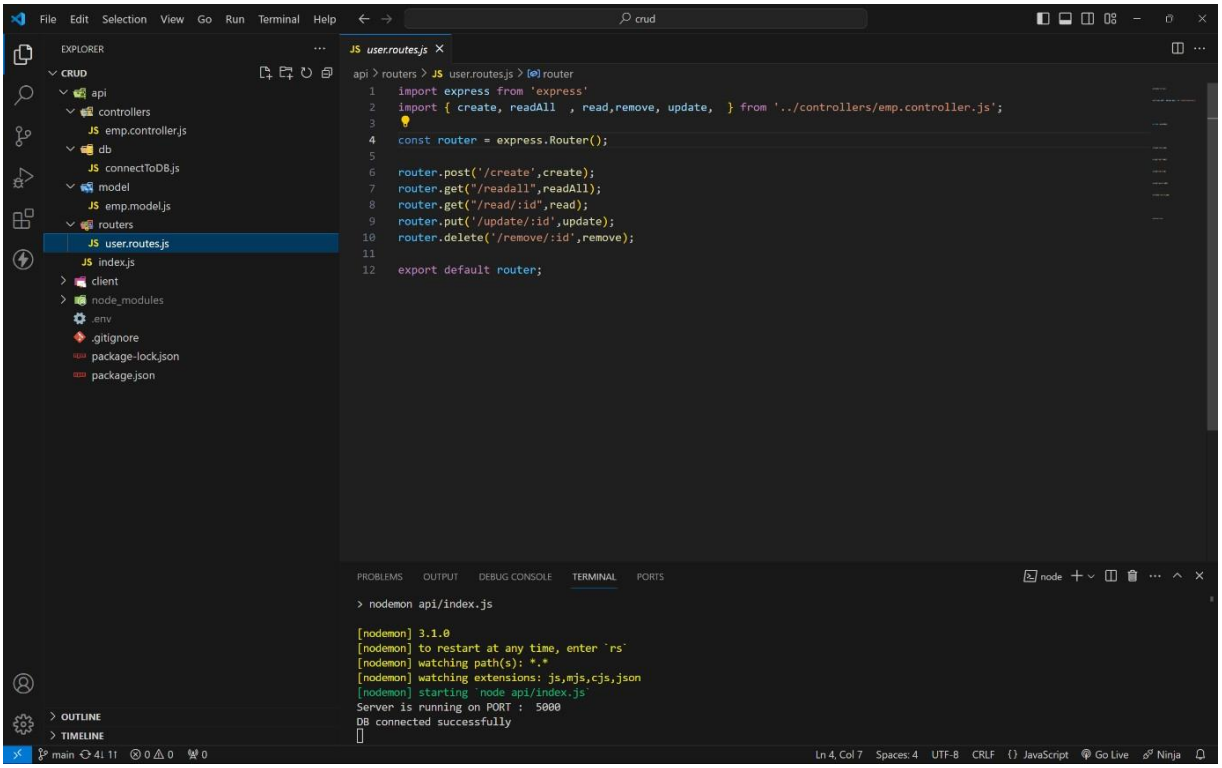
MONGODB CONNECTION :



MODEL :

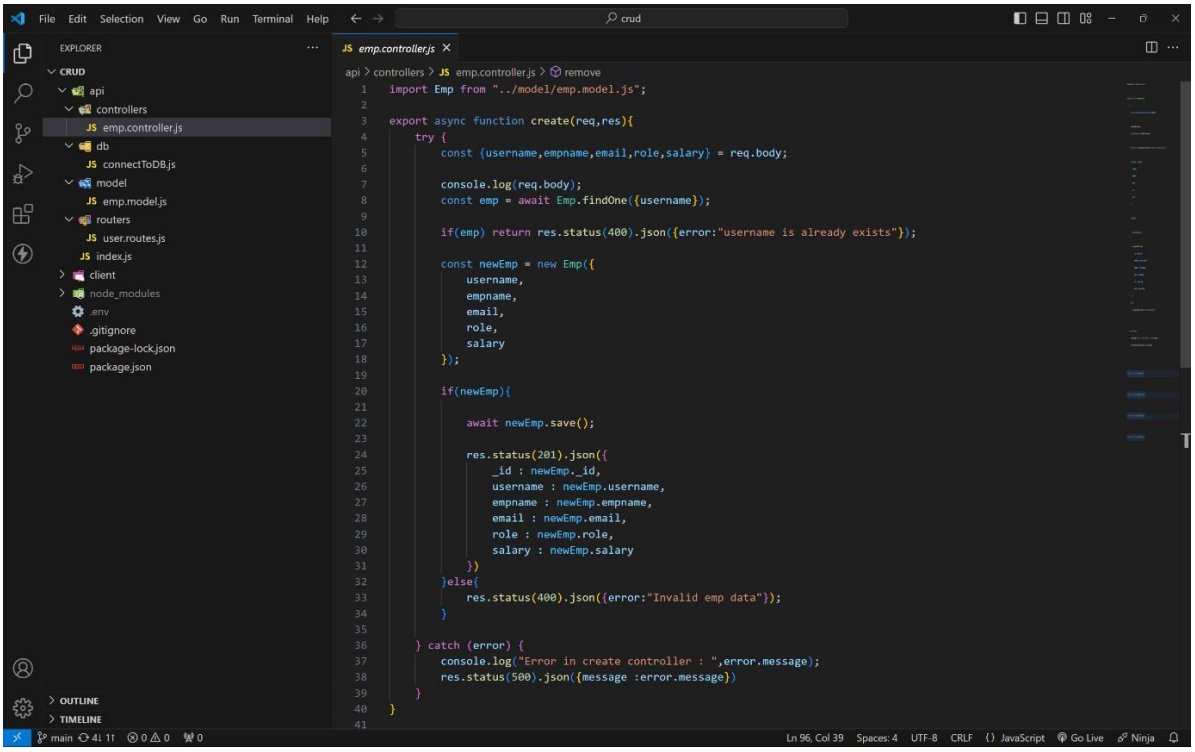


ROUTES:

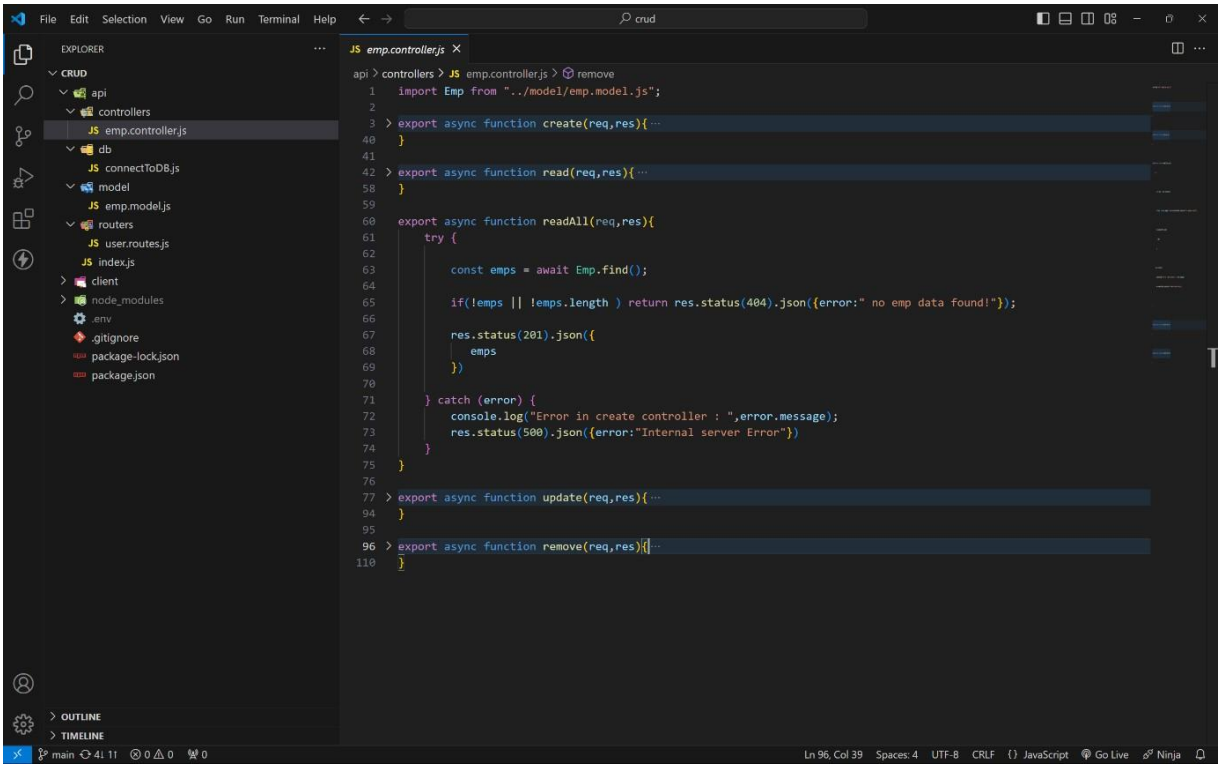


CONTROLLERS :

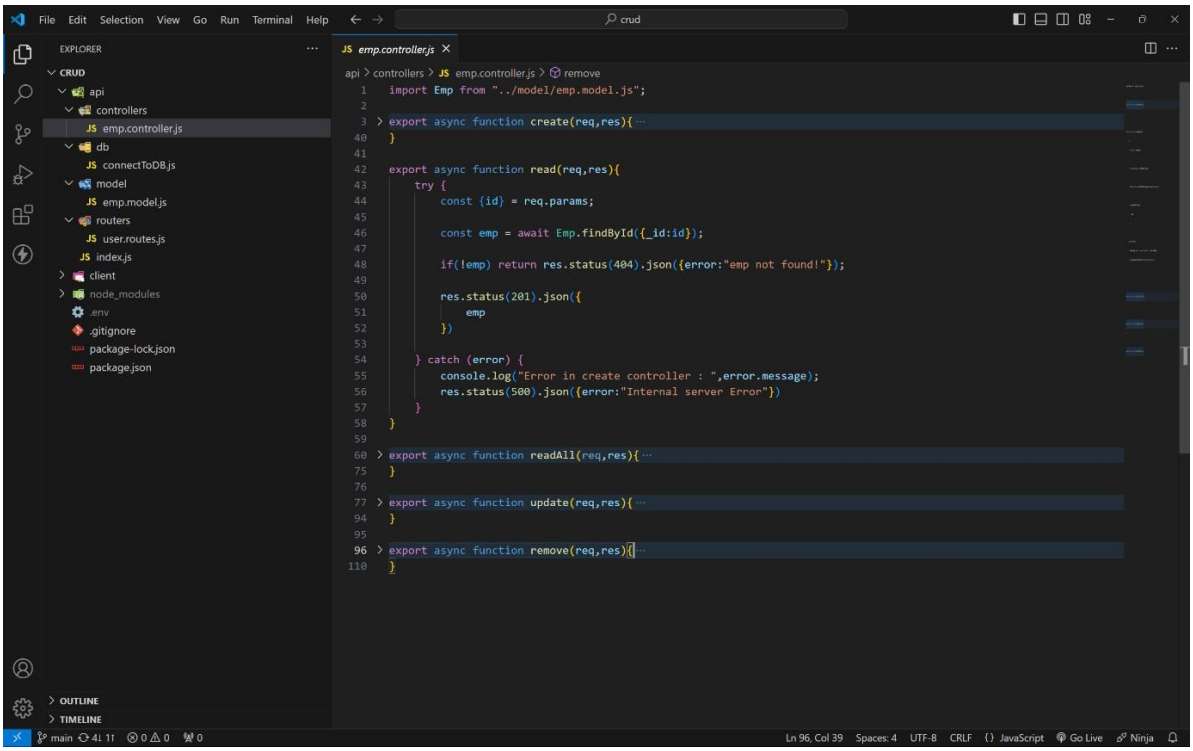
CREATE :



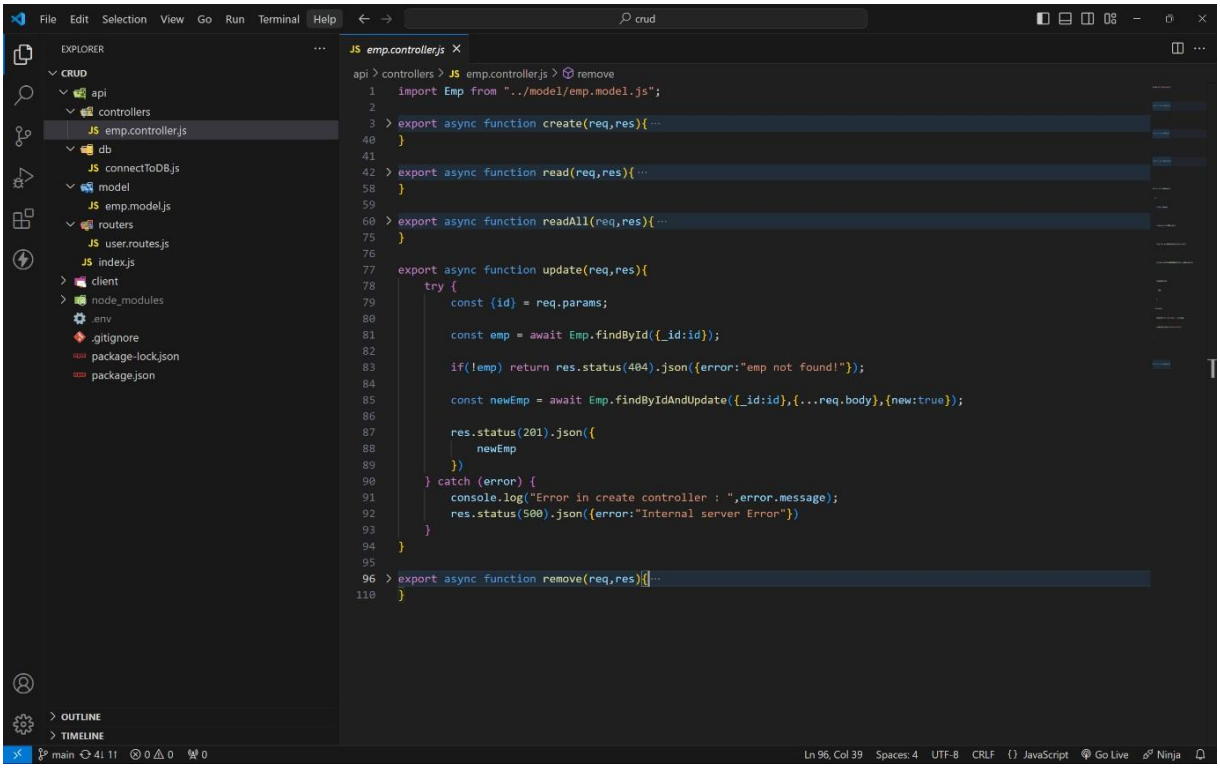
READALL:



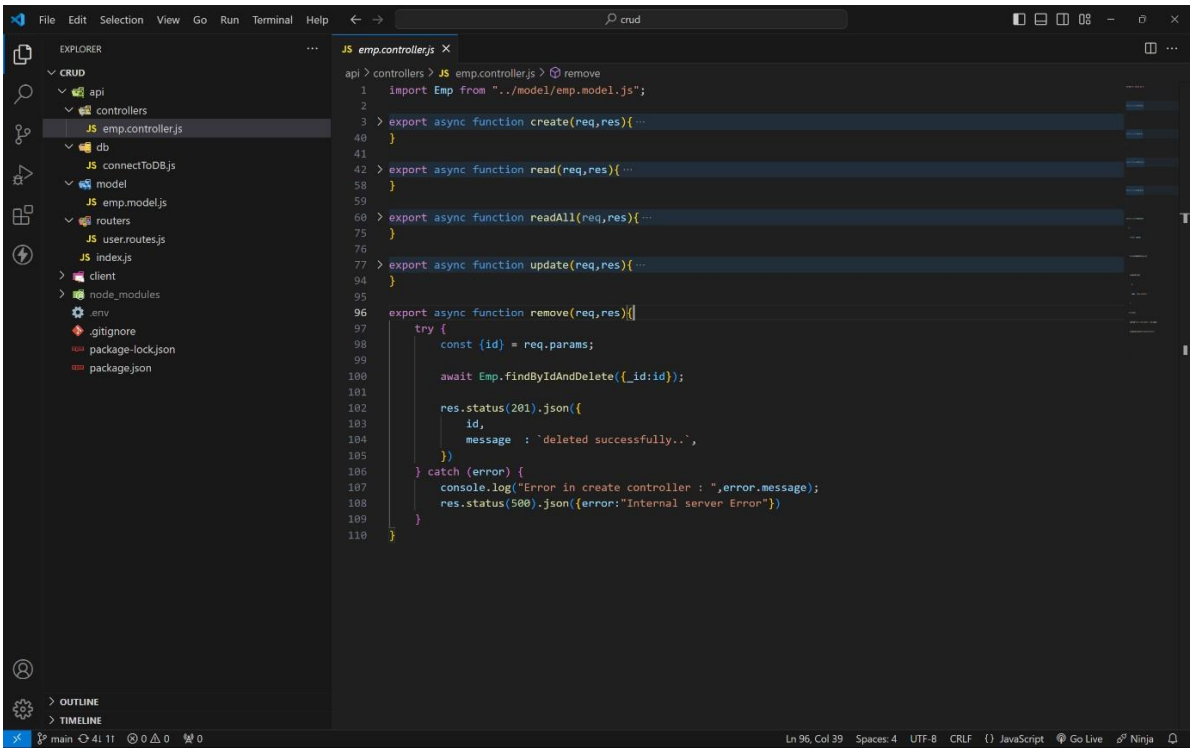
READONE :



UPDATE :



DELETE :



HOW TO RUN ON LOCALLY :

1 . Create a folder as any name.

2 . Open that folder in any code editor (vs code).

3 . Open terminal (ctrl + ~) on code editor.

4 . Type this code to get code locally.

git clone https://github.com/4727yesuraju/crud.git

5 . Now move to crud folder (cd crud in terminal)

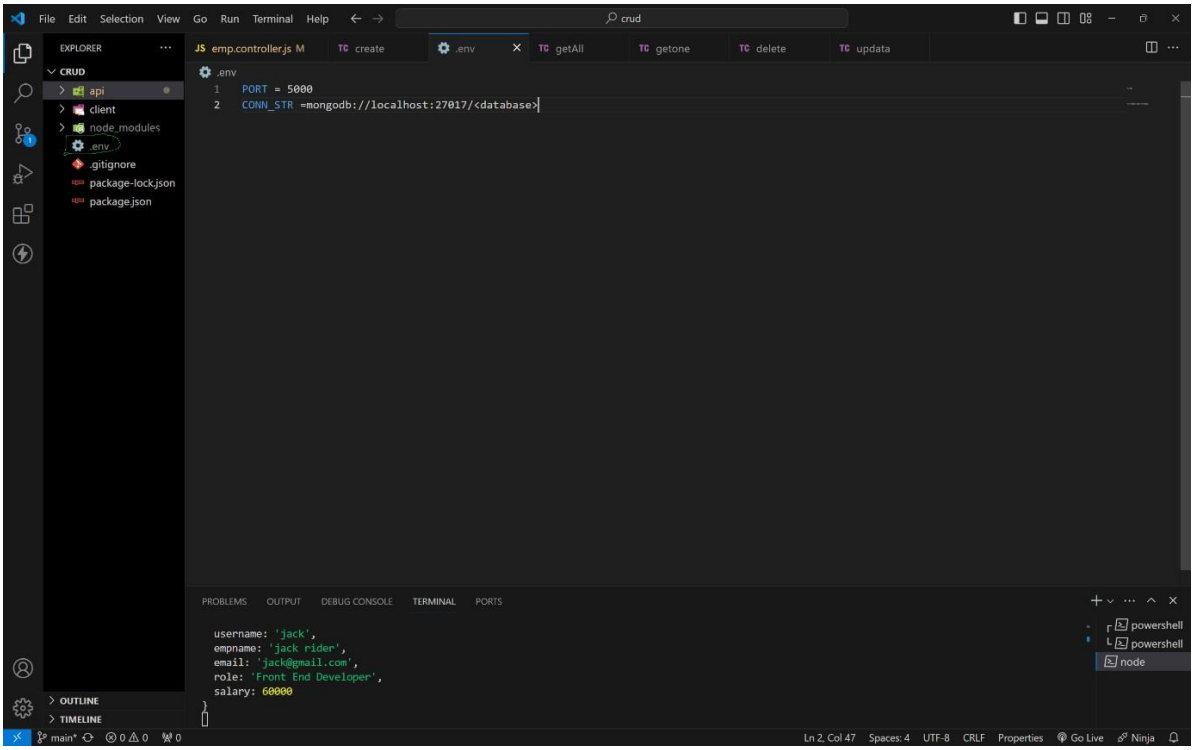
6 . Ignore client folder.

7 . Here crud is root folder.

8 . In root folder create a .env file and create a PORT and CONN_STR variables and assign value.

ex : PORT = 3000 (commonly any number between 3000 - 8080).

CONN_STR = your mongodb_connection_string.



--- trouble in above process ? :

simply paste this code in .env file .

PORT = 5000

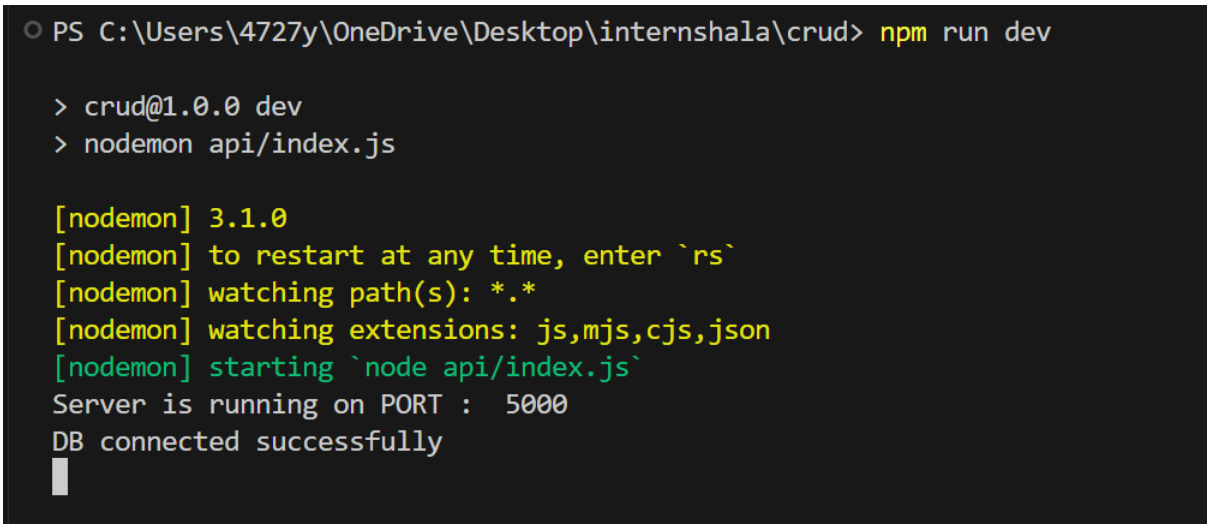
CONN_STR=mongodb+srv://4727yesuraju:rough@cluster0.wbclvtg.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0

9 . After in terminal (in crud folder as root folder) type this command to run server.

npm i (installing all dependencies)

npm run dev (to run server)

10 . if you get below message in terminal then your server will running successfully.



route and its functionality :

For this use any API using tools like Postman or Thunder Client.

i use THUNDER CLIENT.

CREATE ROUTE :

1 . This route is used to create a new employee in database with a below fields.

username, empname, email, role, salary

2 . in thunder client click on new request and select this options

method as post

url as http://localhost:5000/api/user/create

pass this json data as a body as your required value.

```
{
"username": "jack",
"empname": "jack rider",
"email": "jack@gmail.com",
"role": "Front End Developer",
"salary": 60000
}
```

3 . finally press send to insert data in mongodb data base and get a inserted

data as a response.

4 . If user is already in db it will return User is already exist as response.

for more details visit below output images...

READONE :

1 . This route is used to read specific user info by passing that user id as a param.

method as get

url as
http://localhost:5000/api/user/read/65ed7b3d76e1dcc9a51654ca

2 . After sending you will get that specific user details as response.

READALL :

1 . Read all route is used to get all the user data existing in the mongodb data base .

method as get

url as http://localhost:5000/api/user/readall

2 . After sending you will get that all user details as response.

UPDATE :

1 . This route is used to update specific user by passing that user id as a param.

method as put

url as
http://localhost:5000/api/user/update/65ed7b3d76e1dcc9a51654ca

2 . After sending you will get updated user details as response.

DELETE :

1 . This route is used to delete specific user by passing that user id as a param.

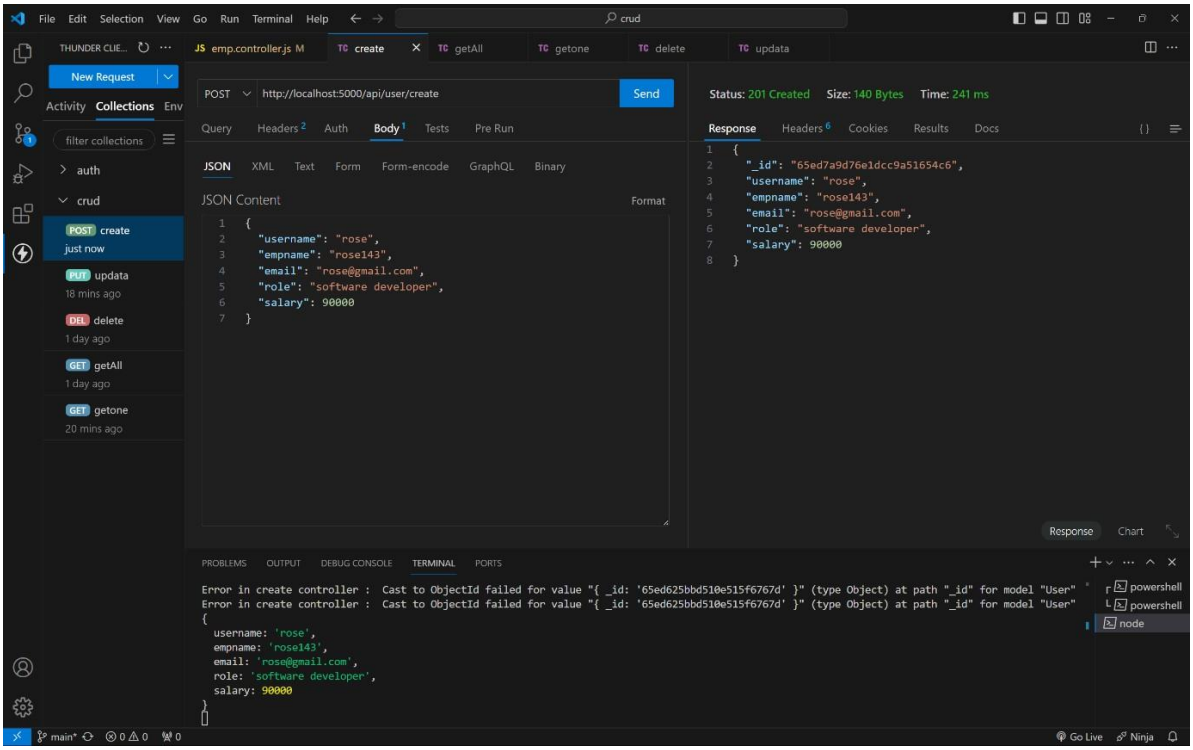
method as delete

url as
http://localhost:5000/api/user/delete/65ed7b3d76e1dcc9a51654ca

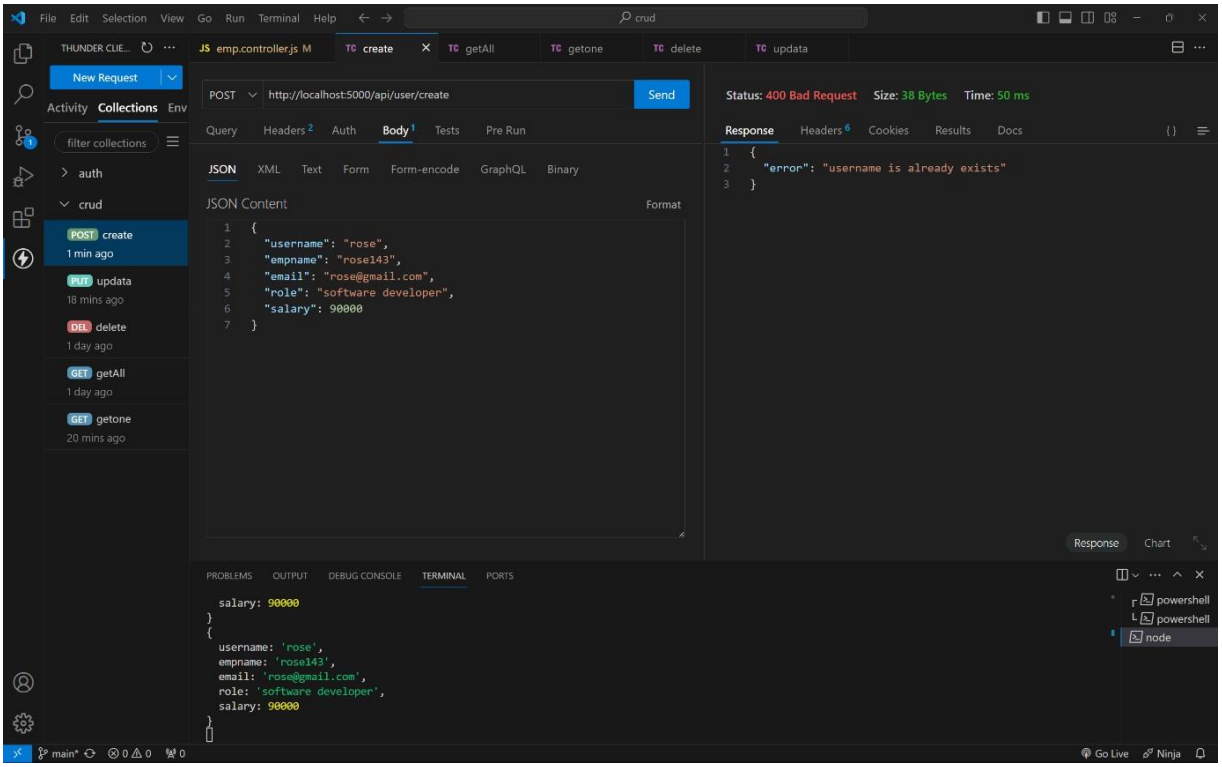
2 . After sending you will deleted successfully as response.

OUTPUT :

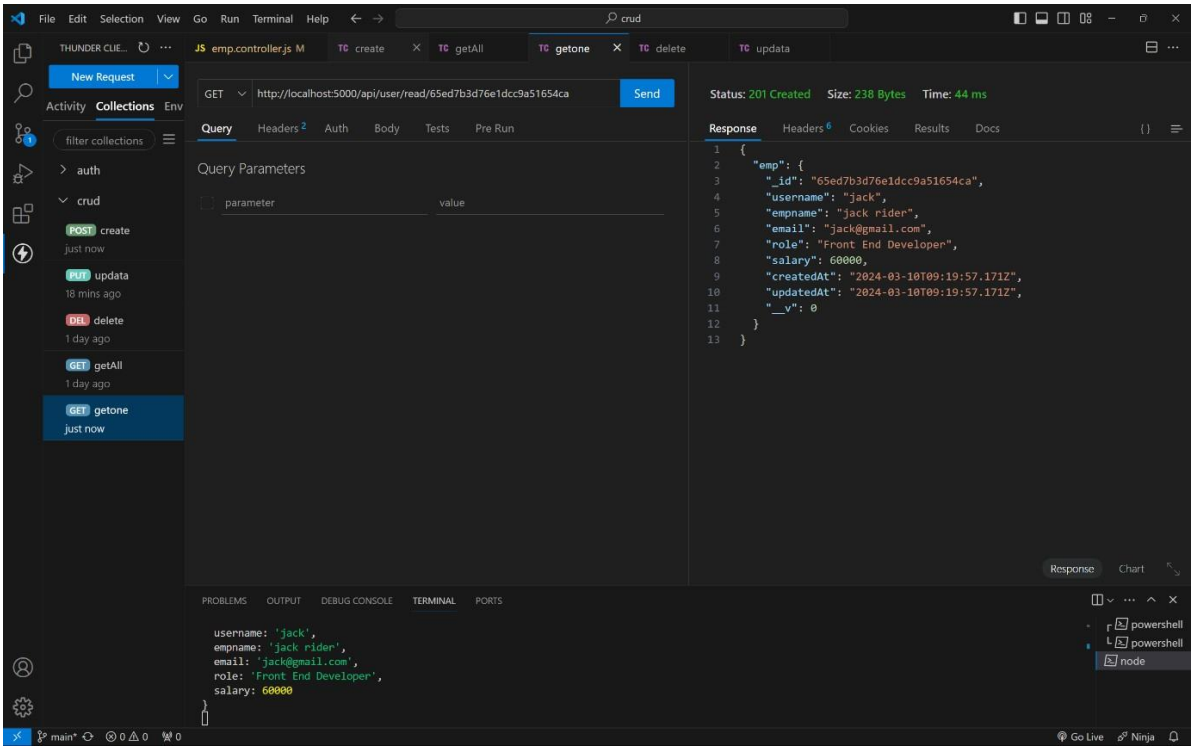
CREATE A NEW USER :



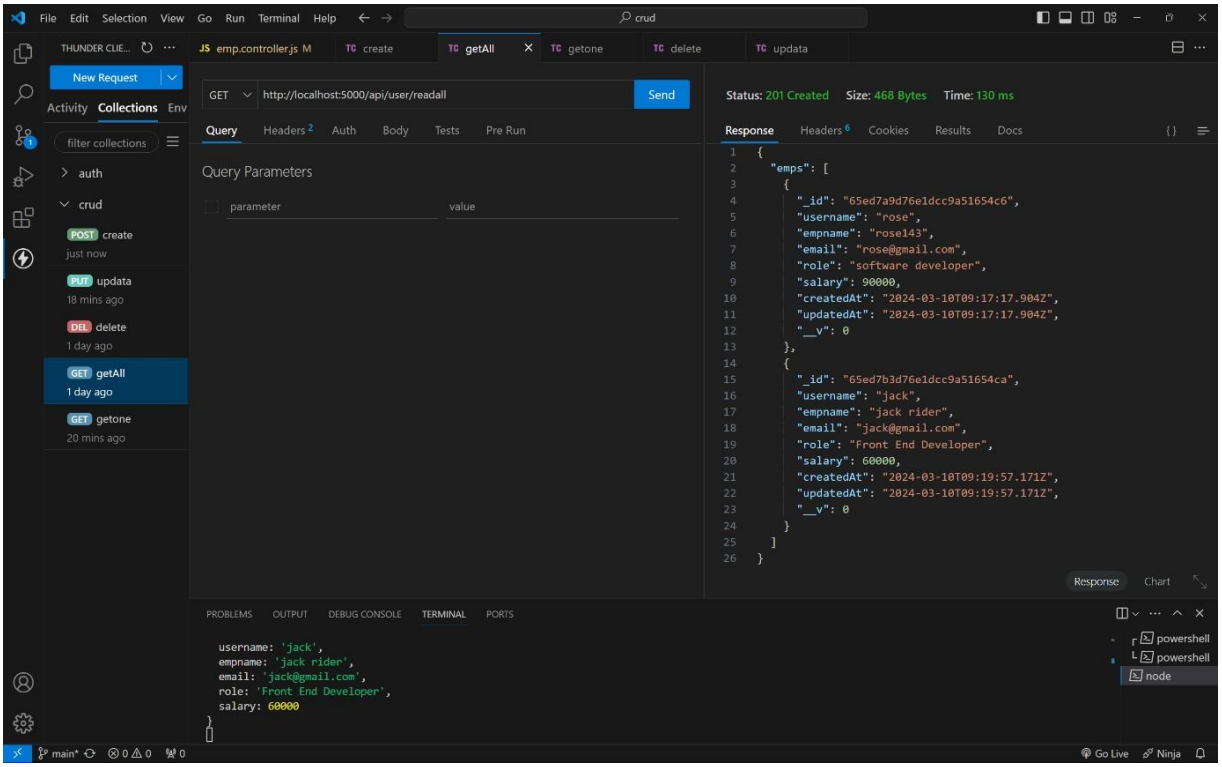
CREATING USER WITH EXISTING USERNAEM :



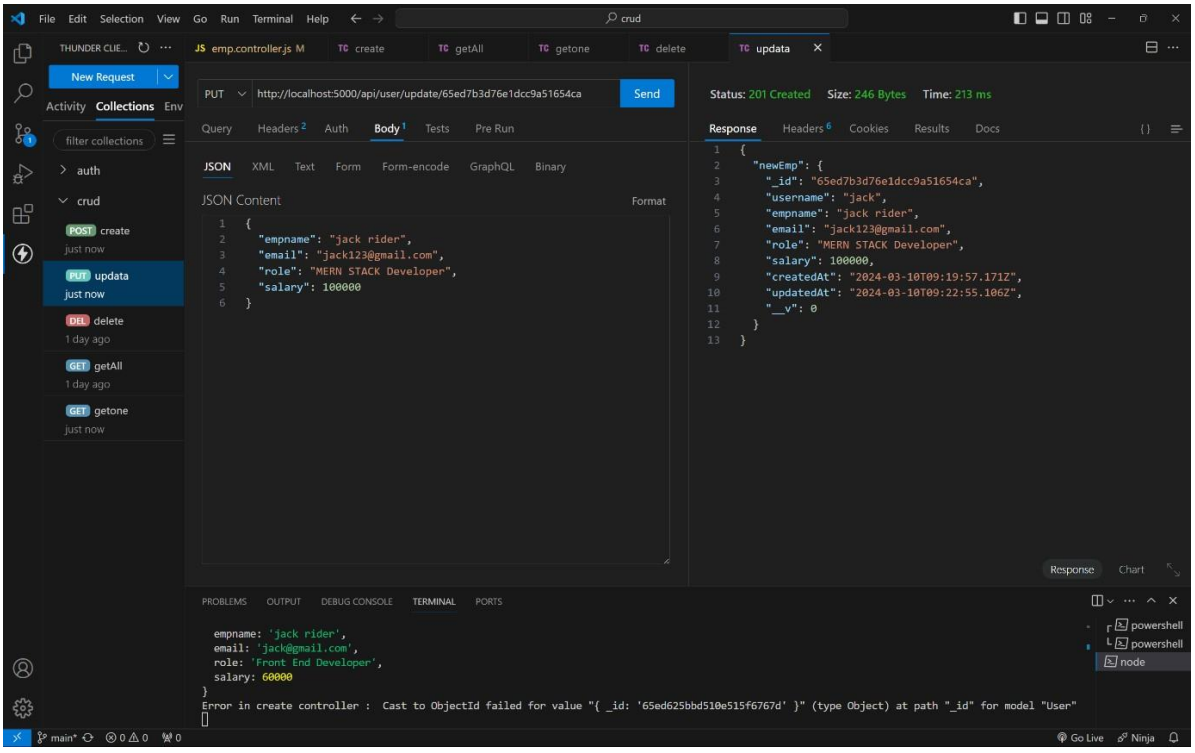
READONE :



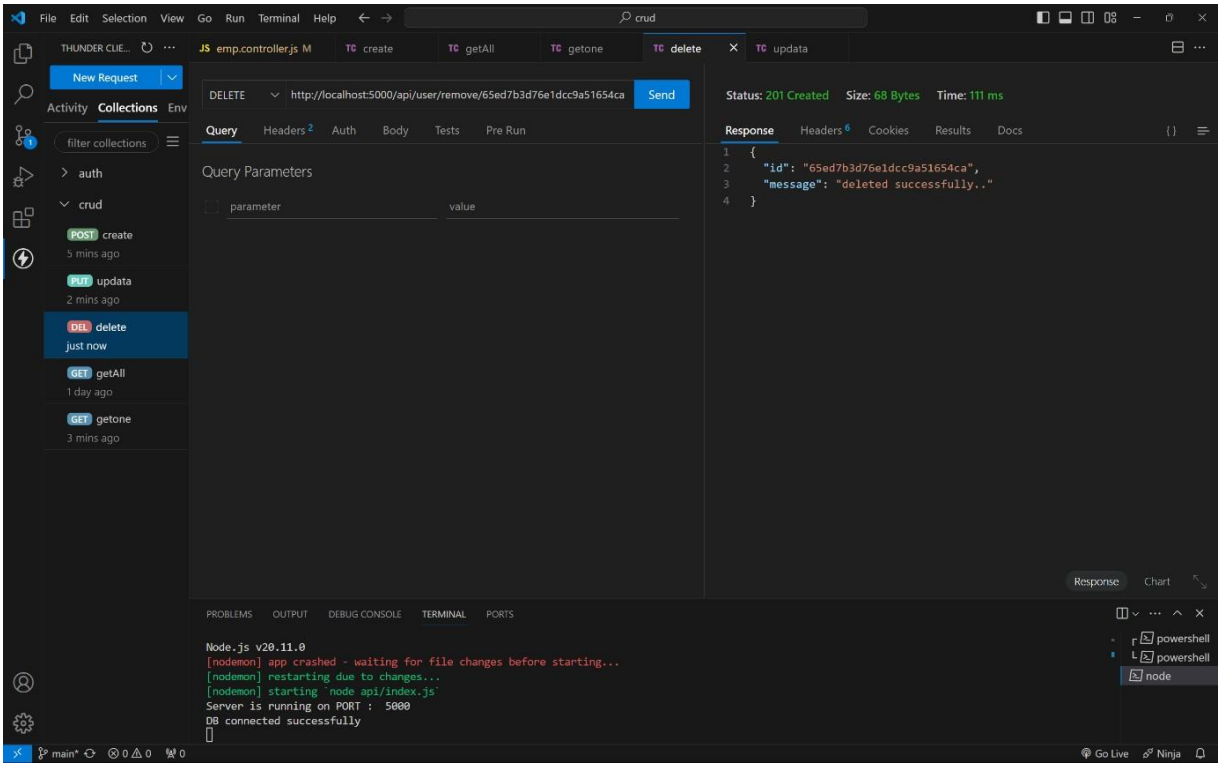
READ ALL :



UPDATE :



DELETE :


File Edit Selection View Go Run Terminal Help
crud
emp.controller.js M
create
getAll
getone
delete
updata
New Request
Activity Collections Env
filter collections
auth
crud
POST create
5 mins ago
PUT updata
2 mins ago
DEL delete
just now
GET getAll
1 day ago
GET getone
3 mins ago
DELETE
http://localhost:5000/api/user/remove/65ed7b3d76e1dcc9a51654ca
Send
Query
Headers 2
Auth
Body
Tests
Pre Run
Query Parameters
parameter
value
Status: 201 Created
Size: 68 Bytes
Time: 111 ms
Response
Headers 6
Cookies
Results
Docs
{
"id": "65ed7b3d76e1dcc9a51654ca",
"message": "deleted successfully.."
}
Response
Chart
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Node.js v20.11.0
[nodemon] app crashed - waiting for file changes before starting...
[nodemon] restarting due to changes...
[nodemon] starting `node api/index.js`
Server is running on PORT : 5000
DB connected successfully
powershell
powershell
node
main*
Go Live
Ninja