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

NAME: DIVVELA NAGA PRVALLIKA DEVI

ROLL NO: 208X1A4213

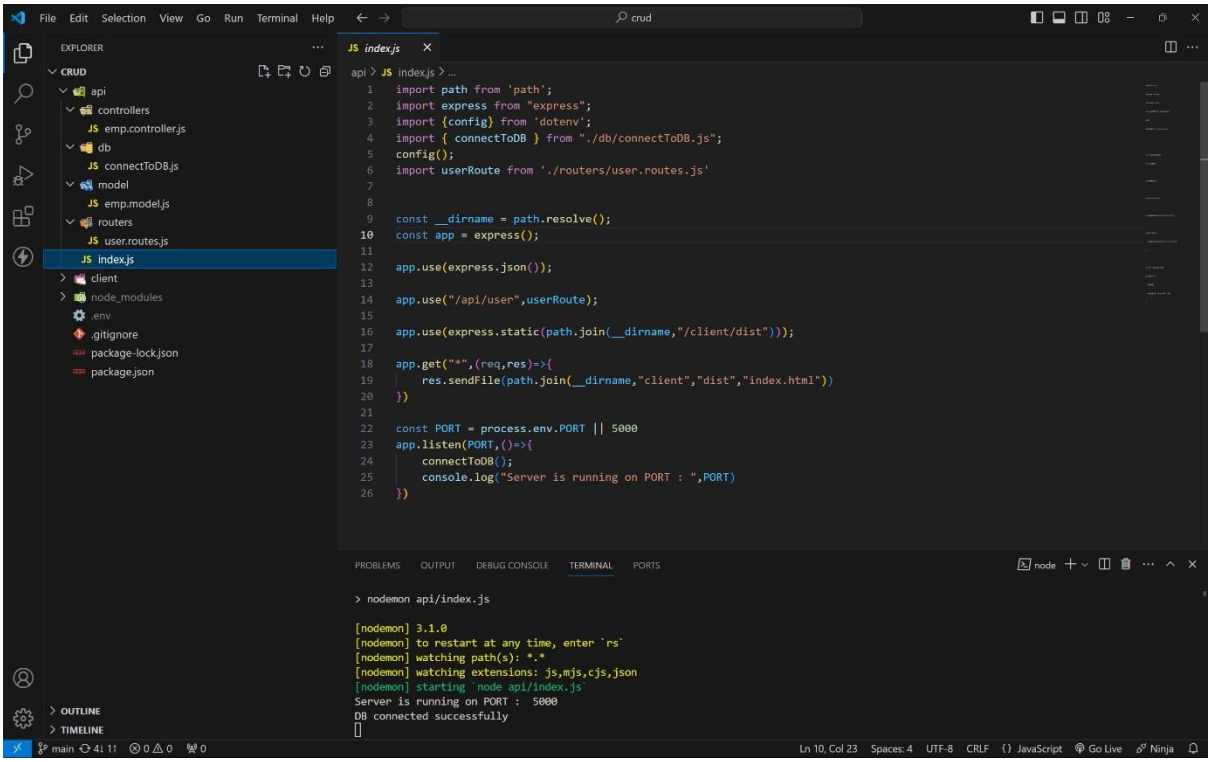
MAIL ID: 208x1a4213@khitguntur.ac.in

PHONE NO: 7386866925

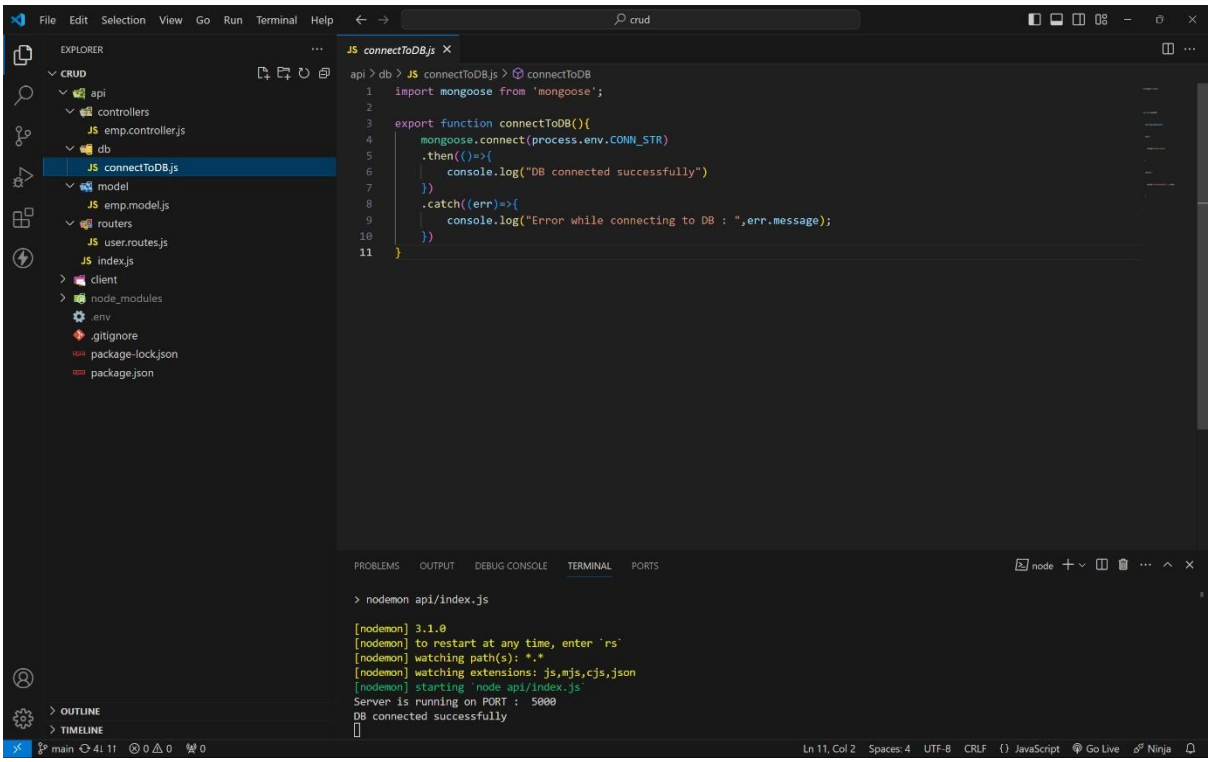
COLLEGE NAME: KALLAM HARANADHAREDDY INSTITUTE OF TECHNOLOGY

source code :

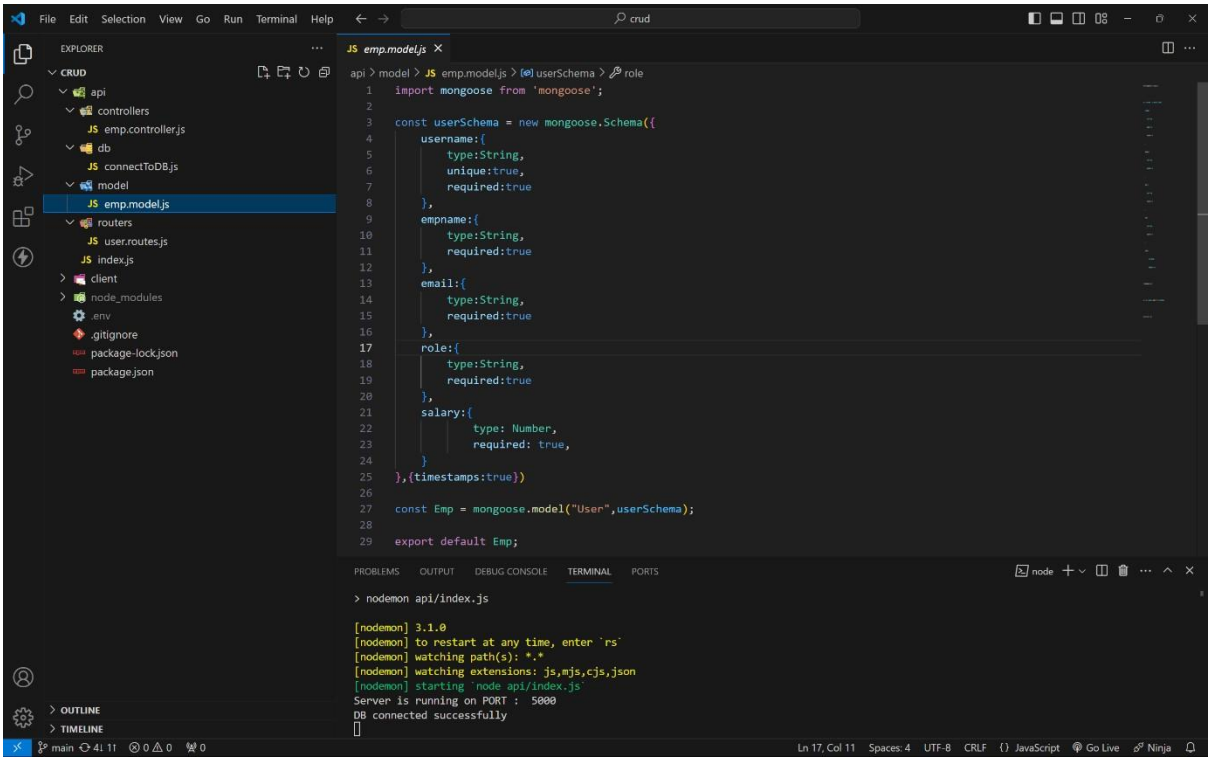
index.js file :



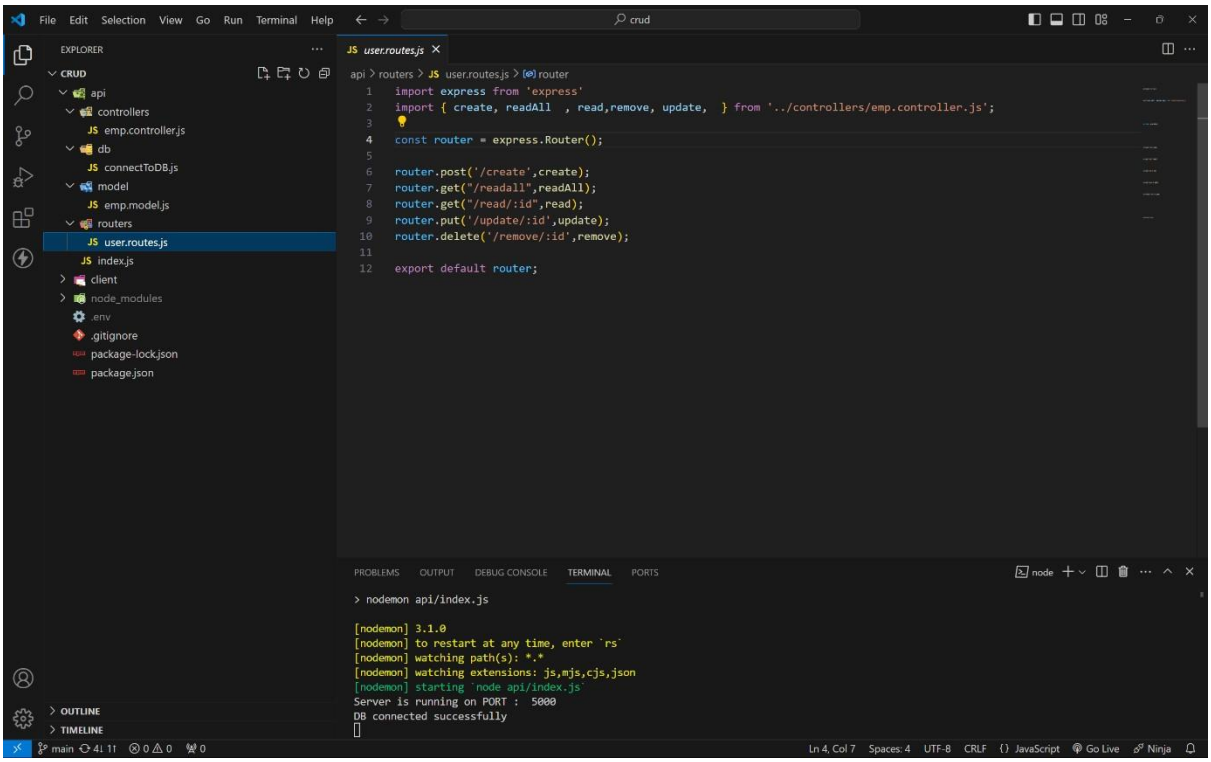
MONGODB CONNECTION :



MODEL :

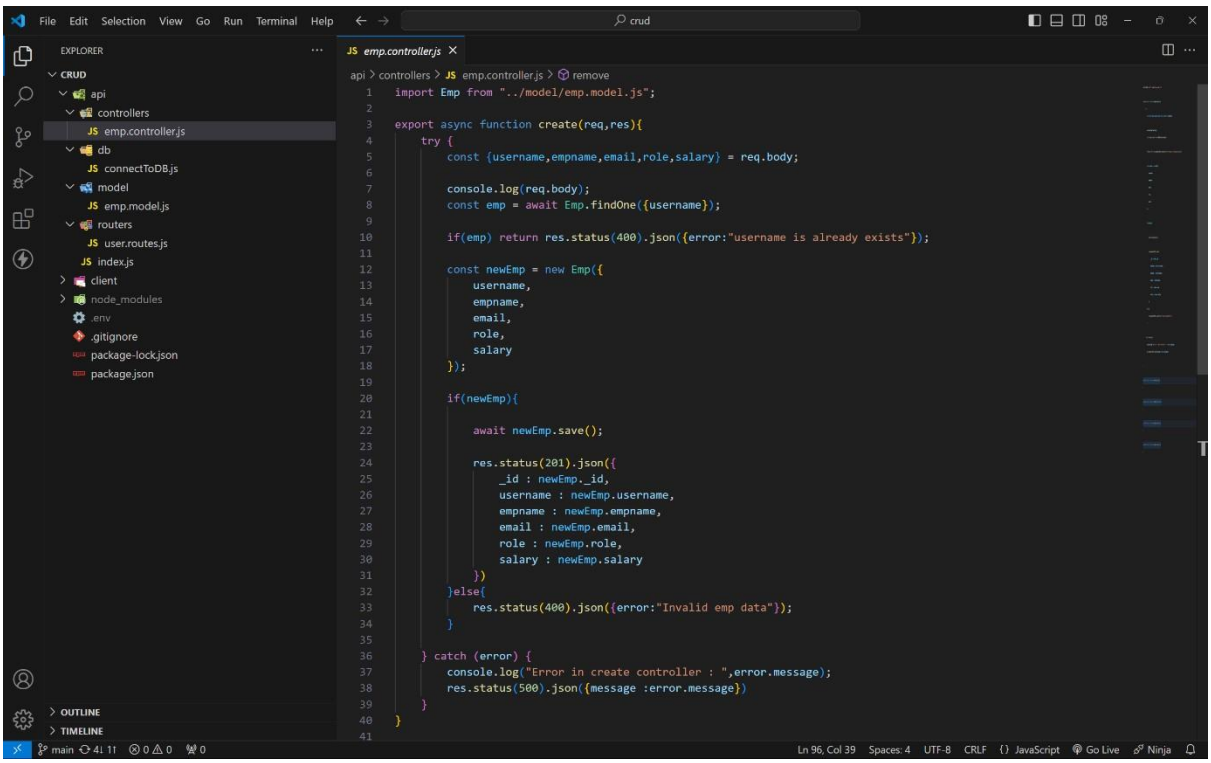


ROUTES:

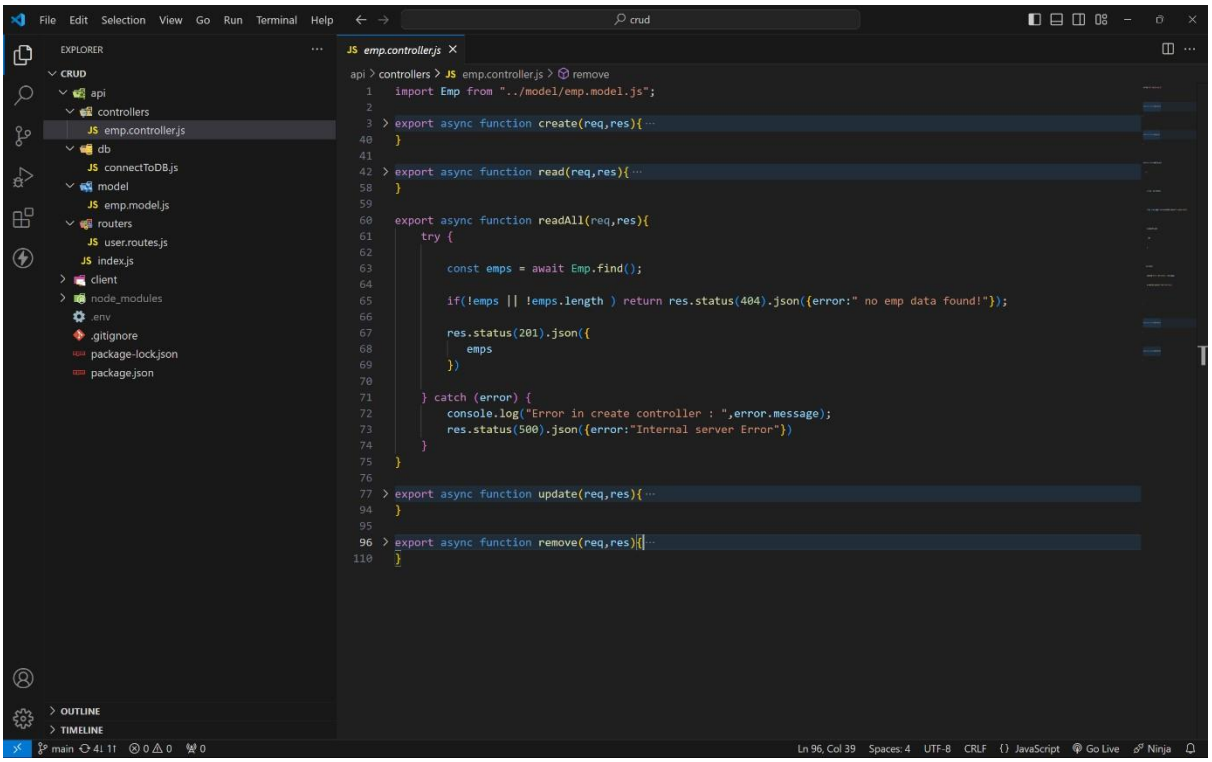


CONTROLLERS :

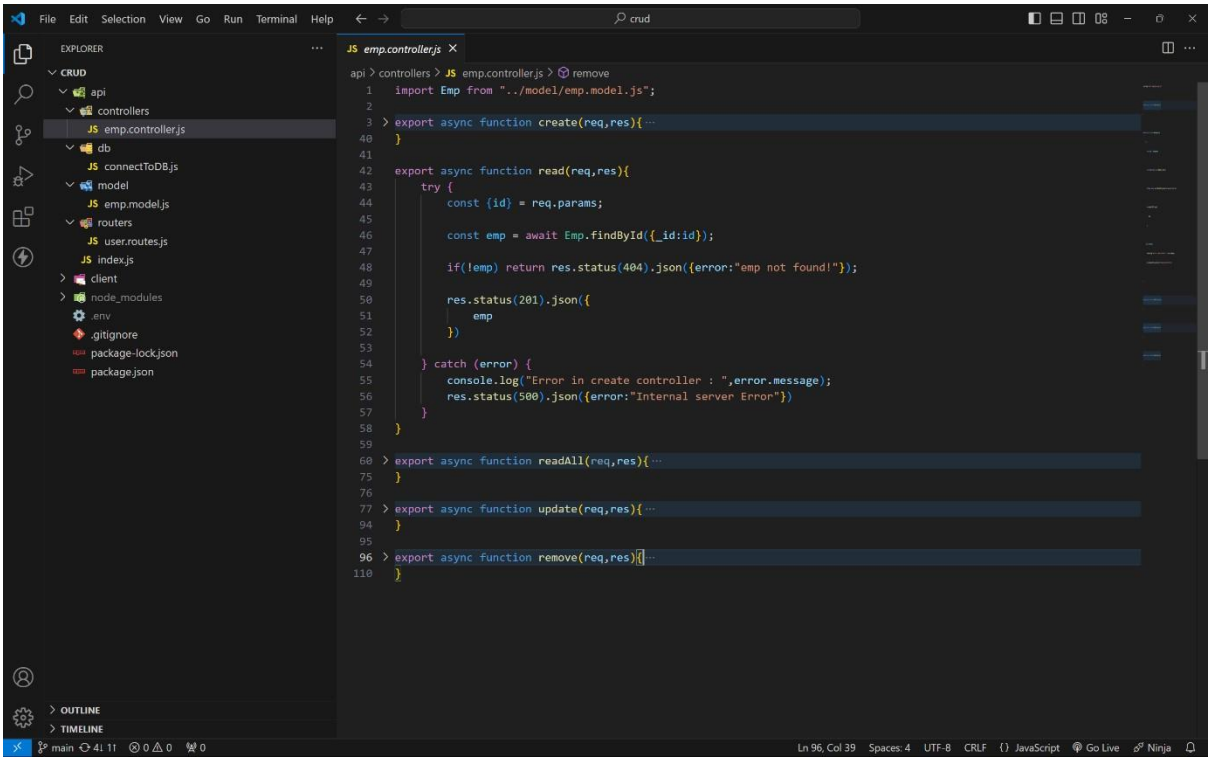
CREATE :



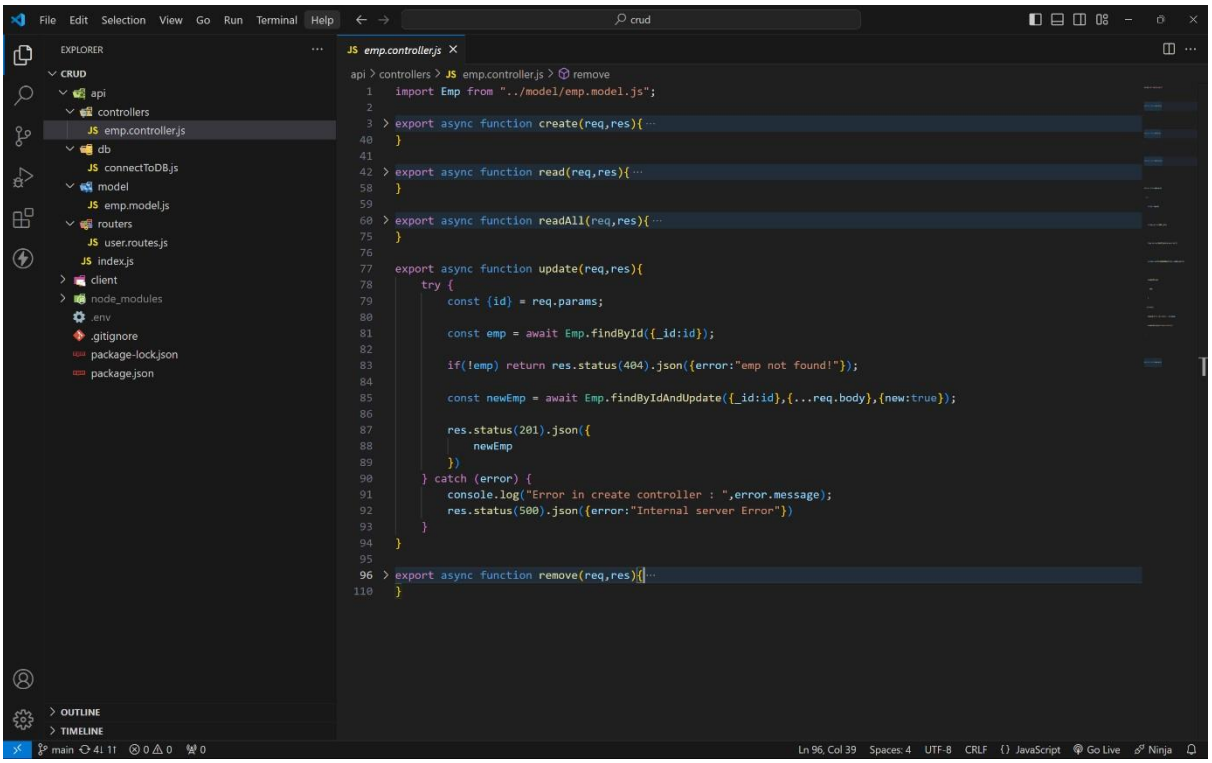
READALL:



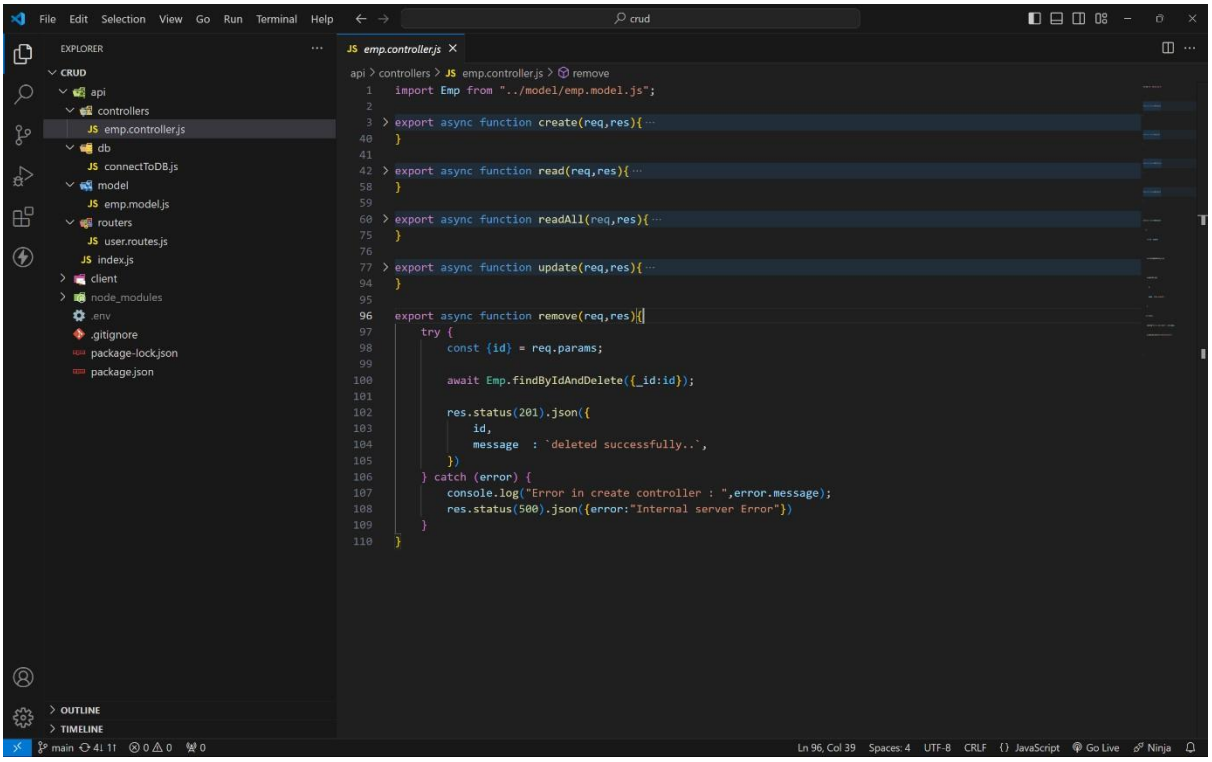
READONE :

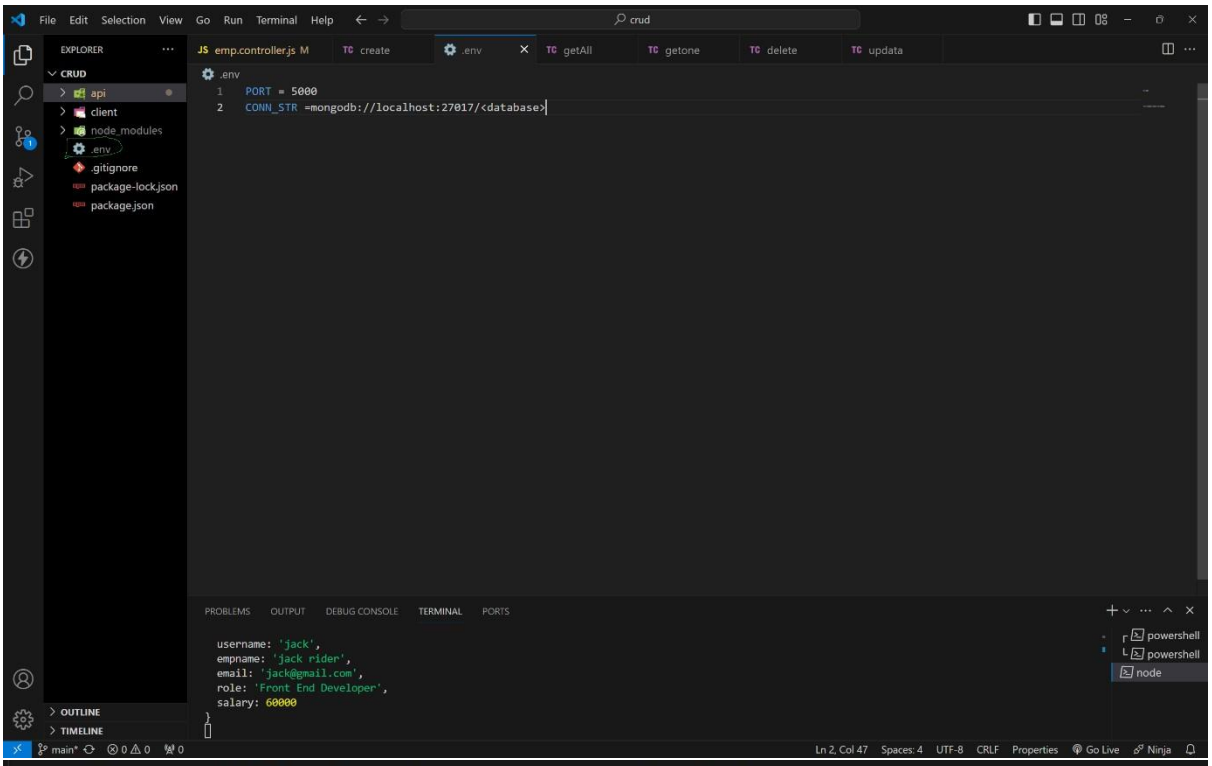


UPDATE :


File Edit Selection View Go Run Terminal Help
crud
EXPLORER
CRUD
api
controllers
emp.controller.js
db
connectToDB.js
model
emp.model.js
routers
user.routes.js
index.js
client
node_modules
.env
.gitignore
package-lock.json
package.json
emp.controller.js
api > controllers > emp.controller.js > remove
import Emp from "../model/emp.model.js";
export async function create(req,res){
export async function read(req,res){
export async function readAll(req,res){
export async function update(req,res){
try {
const {id} = req.params;
const emp = await Emp.findById({_id:id});
if(!emp) return res.status(404).json({error:"emp not found!"});
const newEmp = await Emp.findByIdAndUpdate({_id:id},{...req.body},{new:true});
res.status(201).json({
newEmp
})
} catch (error) {
console.log("Error in create controller : ",error.message);
res.status(500).json({error:"Internal server Error"})
}
}
export async function remove(req,res){
OUTLINE
TIMELINE
main
Ln 96, Col 39
Spaces: 4
UTF-8
CRLF
JavaScript
Go Live
Ninja

DELETE :


File Edit Selection View Go Run Terminal Help
crud
EXPLORER
CRUD
api
controllers
emp.controller.js
db
connectToDB.js
model
emp.model.js
routers
user.routes.js
index.js
client
node_modules
.env
.gitignore
package-lock.json
package.json
emp.controller.js
api > controllers > emp.controller.js > remove
import Emp from "../model/emp.model.js";
export async function create(req,res){
export async function read(req,res){
export async function readAll(req,res){
export async function update(req,res){
export async function remove(req,res){
try {
const {id} = req.params;
await Emp.findByIdAndDelete({_id:id});
res.status(201).json({
id,
message : `deleted successfully..`,
})
} catch (error) {
console.log("Error in create controller : ",error.message);
res.status(500).json({error:"Internal server Error"})
}
}
OUTLINE
TIMELINE
main
Ln 96, Col 39
Spaces: 4
UTF-8
CRLF
JavaScript
Go Live
Ninja




```
PS C:\Users\4727y\OneDrive\Desktop\internshala\crud> npm run dev

> crud@1.0.0 dev
> nodemon api/index.js

[nodemon] 3.1.0
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node api/index.js`
Server is running on PORT :  5000
DB connected 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

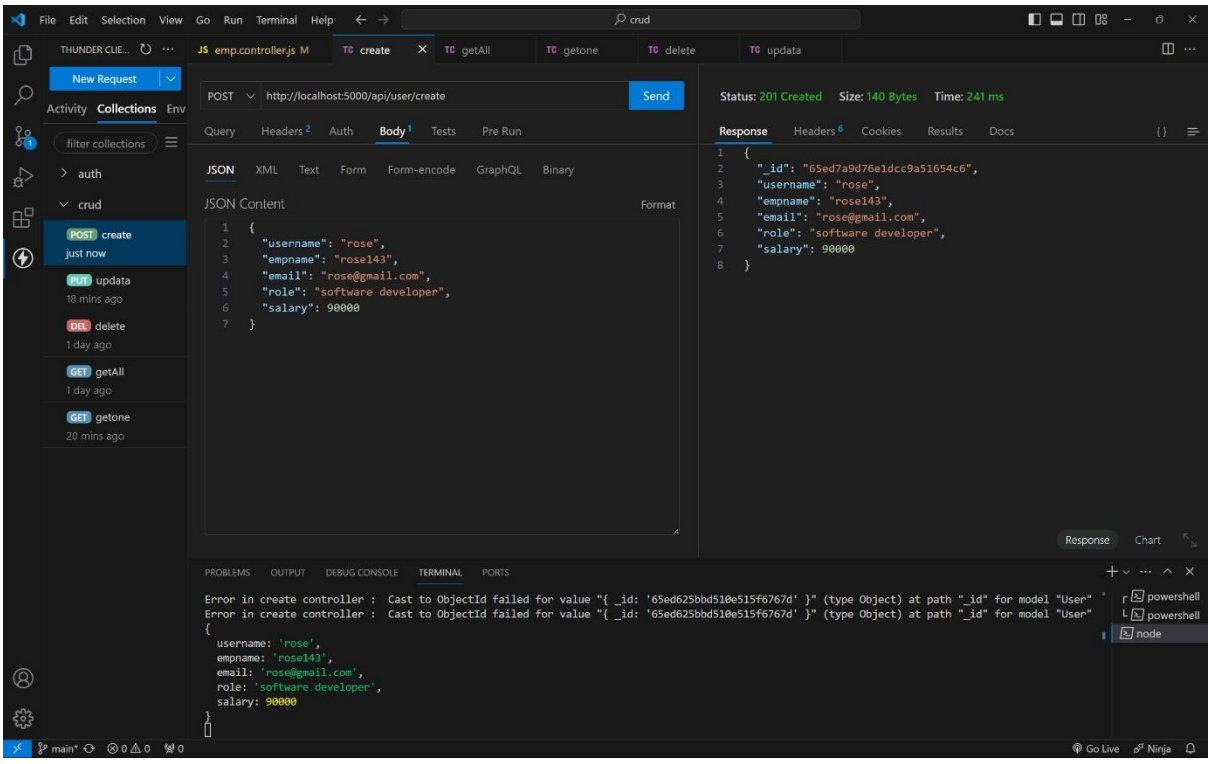
as delete

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

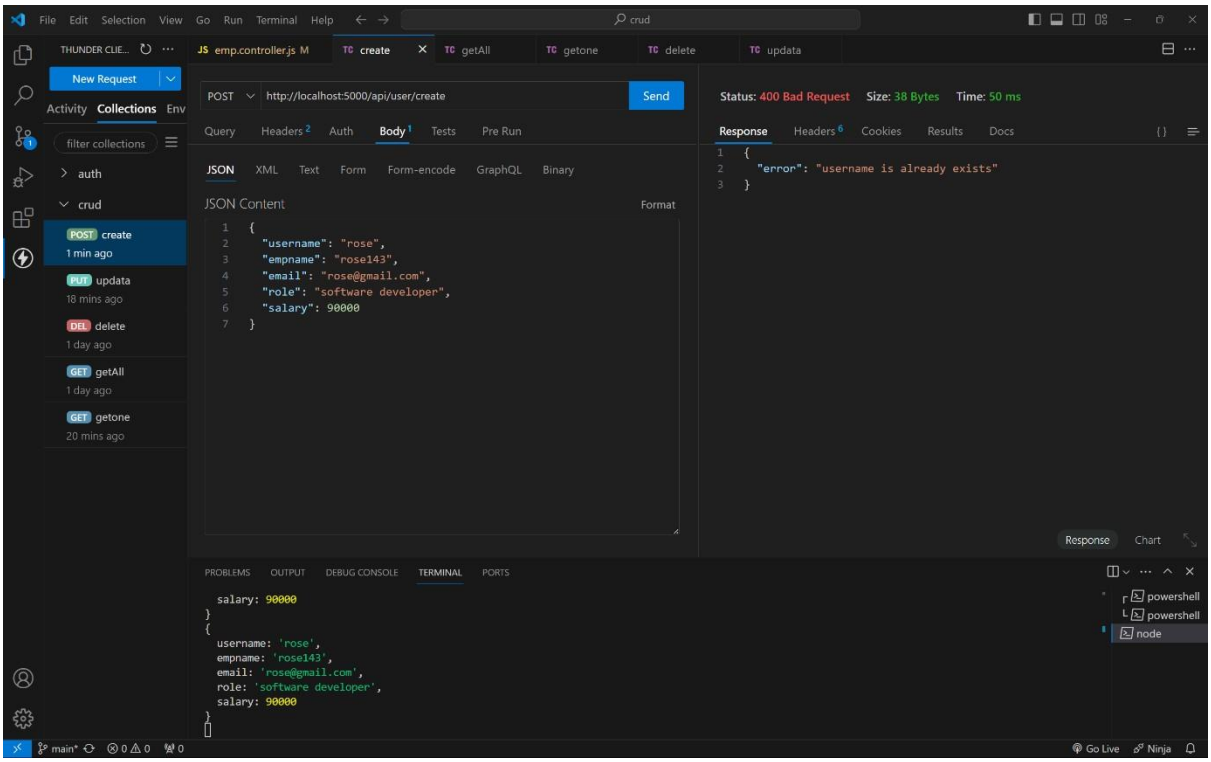
2 . After sending you will deleted successfully as response.

OUTPUT :

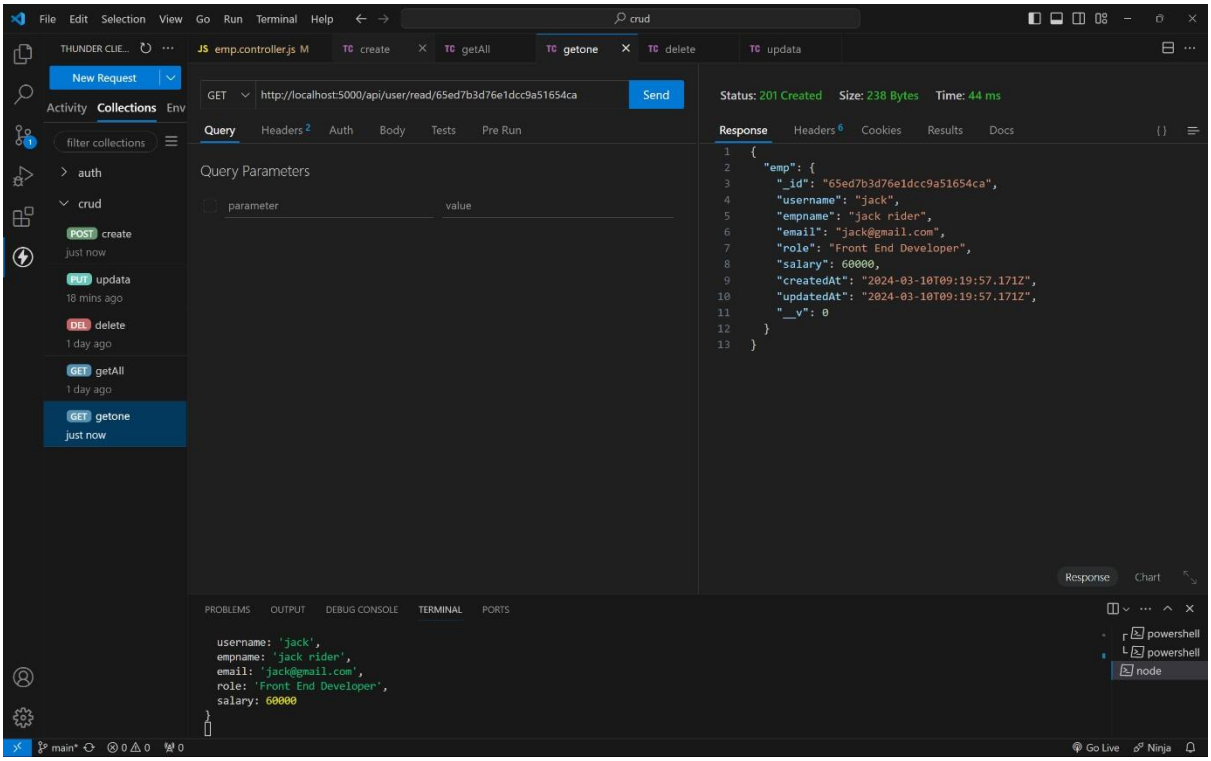
CREATE A NEW USER :



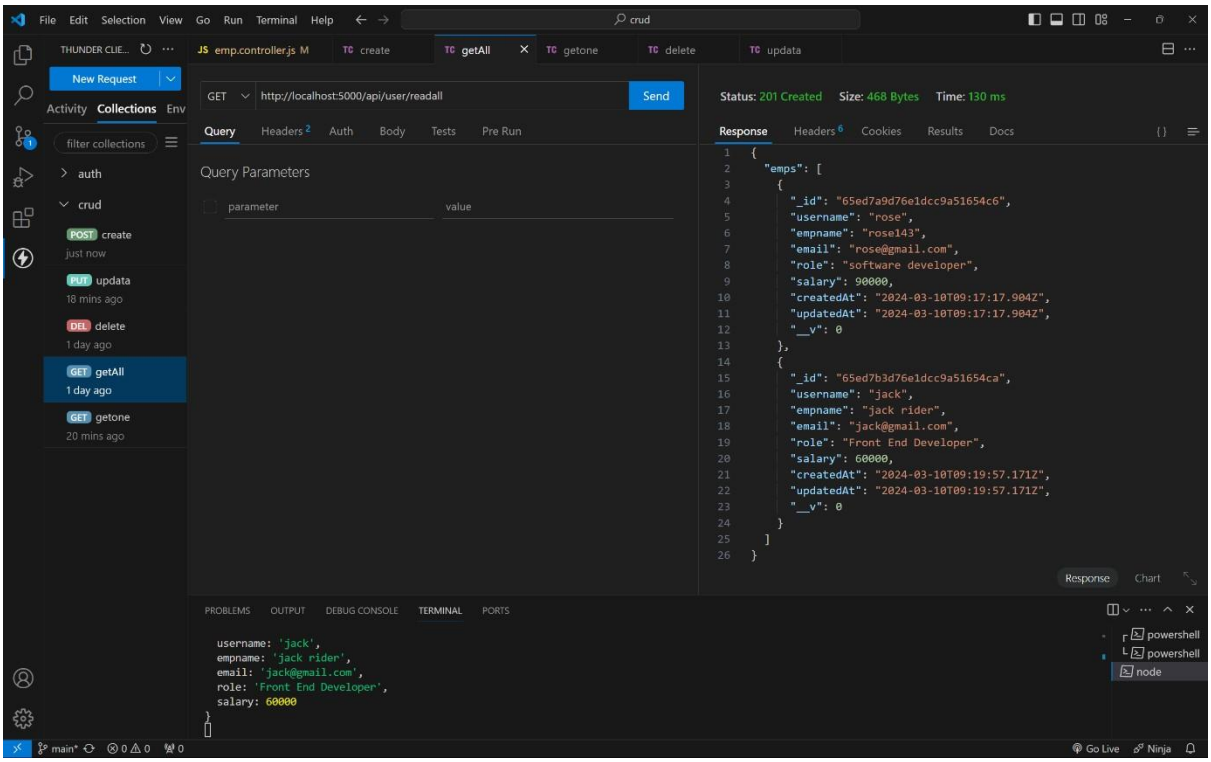
CREATING USER WITH EXISTING USERNAEM :


File Edit Selection View Go Run Terminal Help
crud
THUNDER CLIE...
New Request
Activity Collections Env
filter collections
auth
crud
POST create
1 min ago
PUT updata
18 mins ago
DEL delete
1 day ago
GET getAll
1 day ago
GET getone
20 mins ago
JS emp.controller.js M
TC create
TC getAll
TC getone
TC delete
TC updata
POST
http://localhost:5000/api/user/create
Send
Query Headers 2 Auth Body 1 Tests Pre Run
JSON XML Text Form Form-encode GraphQL Binary
JSON Content
Format
{
"username": "rose",
"empname": "rose143",
"email": "rose@gmail.com",
"role": "software developer",
"salary": 90000
}
Status: 400 Bad Request Size: 38 Bytes Time: 50 ms
Response Headers 6 Cookies Results Docs
{
"error": "username is already exists"
}
Response Chart
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
salary: 90000
}
{
username: 'rose',
empname: 'rose143',
email: 'rose@gmail.com',
role: 'software developer',
salary: 90000
}
powershell
powershell
node
main*
Go Live
Ninja

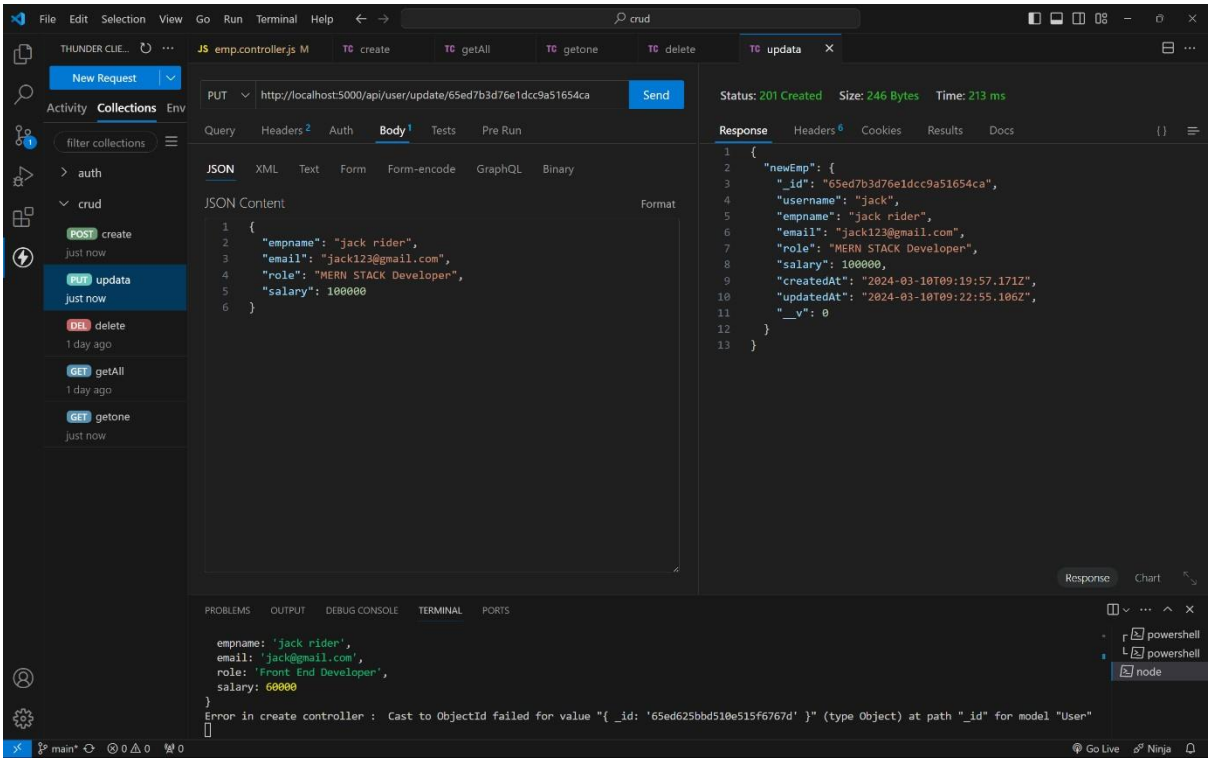
READONE :


File Edit Selection View Go Run Terminal Help
crud
THUNDER CLIE...
New Request
Activity Collections Env
filter collections
auth
crud
POST create
just now
PUT updata
18 mins ago
DEL delete
1 day ago
GET getAll
1 day ago
GET getone
just now
JS emp.controller.js M
TC create
TC getAll
TC getone
TC delete
TC updata
GET
http://localhost:5000/api/user/read/65ed7b3d76e1dcc9a51654ca
Send
Query Headers 2 Auth Body Tests Pre Run
Query Parameters
parameter
value
Status: 201 Created Size: 238 Bytes Time: 44 ms
Response Headers 6 Cookies Results Docs
{
"emp": {
"_id": "65ed7b3d76e1dcc9a51654ca",
"username": "jack",
"empname": "jack rider",
"email": "jack@gmail.com",
"role": "Front End Developer",
"salary": 60000,
"createdAt": "2024-03-10T09:19:57.171Z",
"updatedAt": "2024-03-10T09:19:57.171Z",
"__v": 0
}
}
Response Chart
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
username: 'jack',
empname: 'jack rider',
email: 'jack@gmail.com',
role: 'Front End Developer',
salary: 60000
}
powershell
powershell
node
main*
Go Live
Ninja

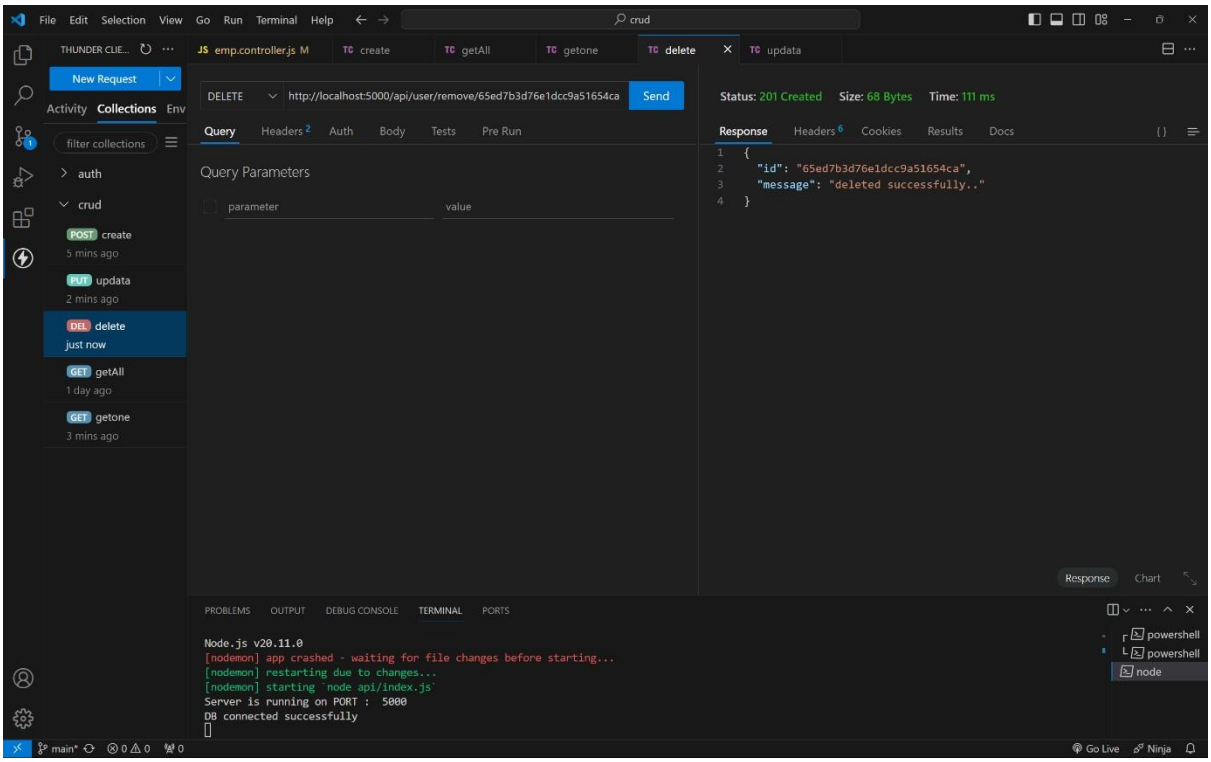
READ ALL :



UPDATE :



DELETE :


File Edit Selection View Go Run Terminal Help
crud
THUNDER CLIE...
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
JS emp.controller.js M
TC create
TC getAll
TC getone
TC delete
TC updata
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