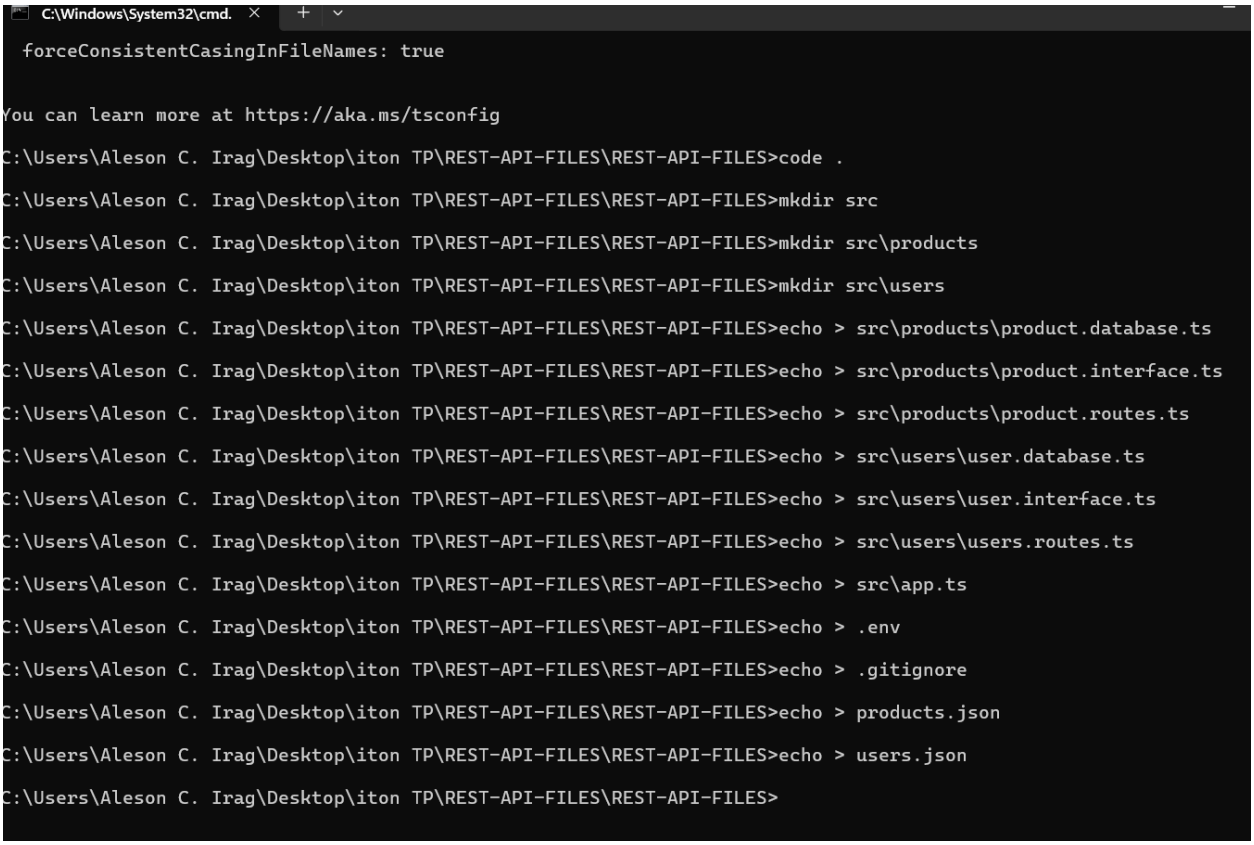
Aleson C. Irag

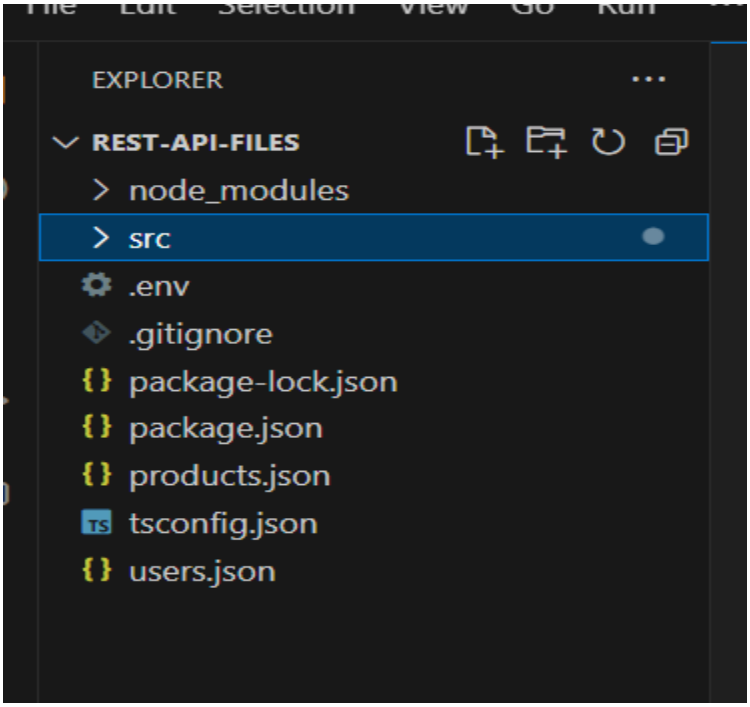
```
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES>mkdir REST-API-FILES && cd REST-API-FILES

C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>npm init -y
Wrote to C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES\package.json:

{
  "name": "rest-api-files",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": ""
}
```

```
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>npm install express dotenv

added 70 packages, and audited 71 packages in 4s

15 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

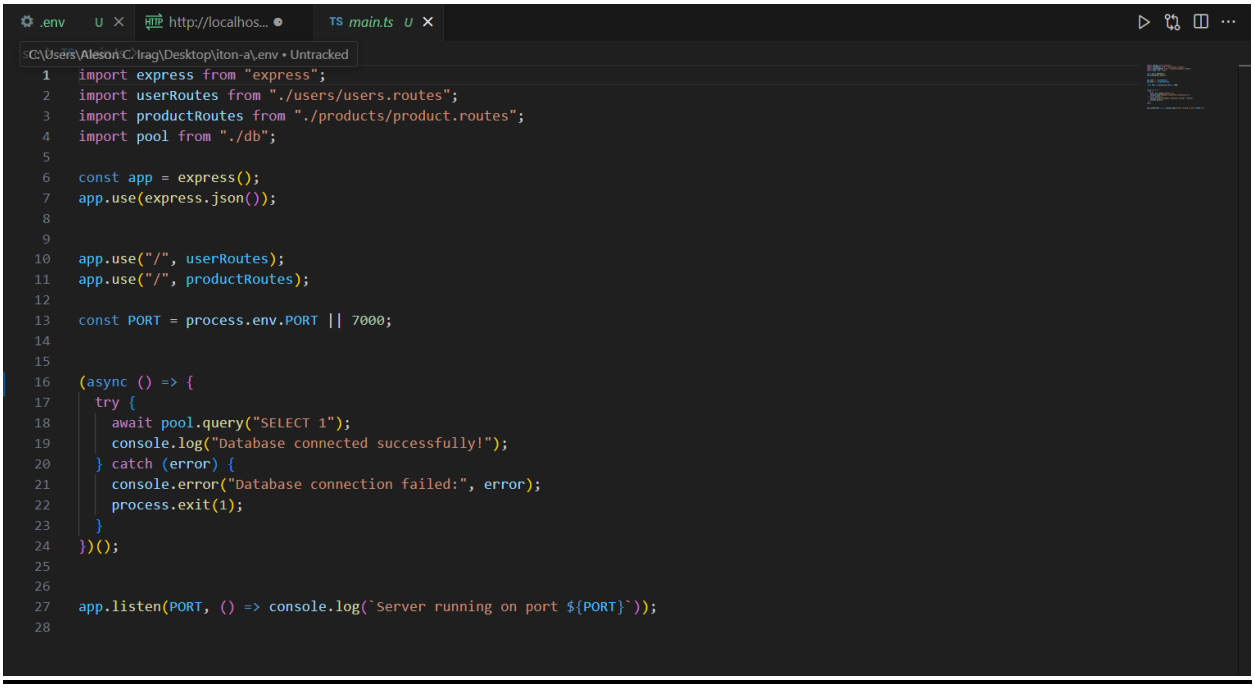
```
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>npm install --save-dev typescript @types/node @typ
es/express ts-node nodemon

added 59 packages, and audited 130 packages in 11s

19 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

```
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>npx tsc --init

Created a new tsconfig.json with:

  target: es2016
  module: commonjs
  strict: true
  esModuleInterop: true
  skipLibCheck: true
  forceConsistentCasingInFileNames: true


You can learn more at https://aka.ms/tsconfig

C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>
```

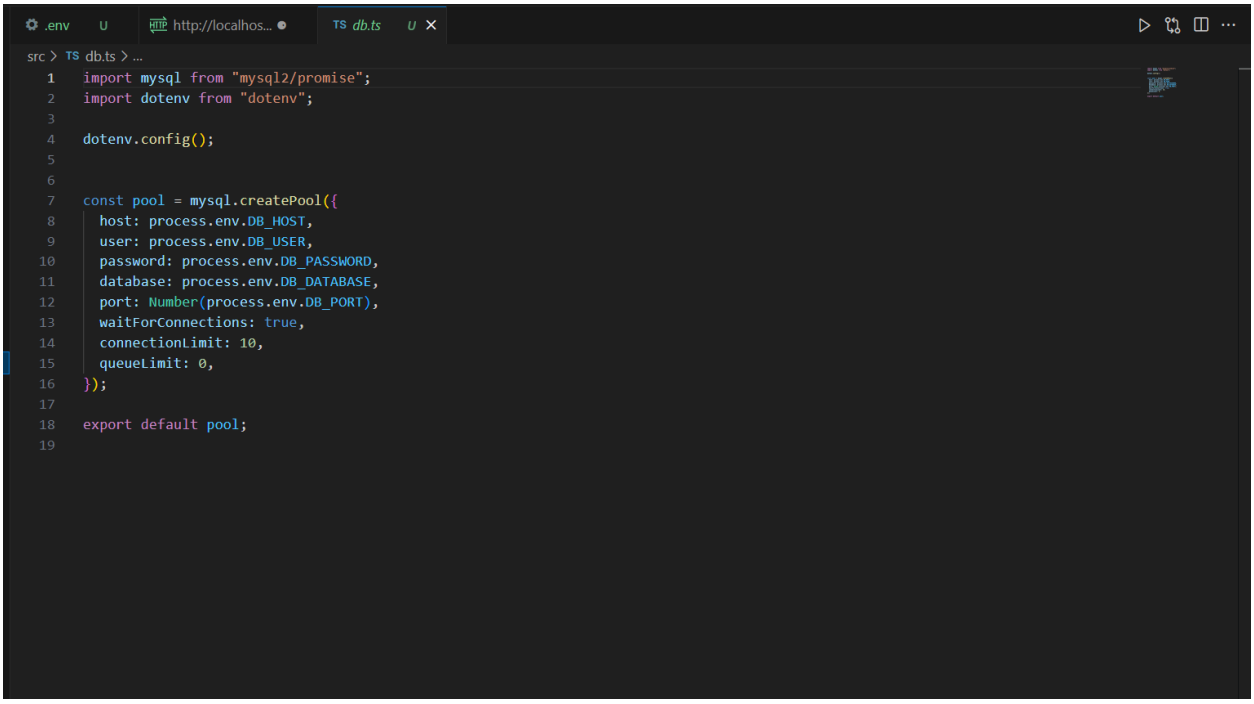

C:\Windows\System32\cmd.
forceConsistentCasingInFileNames: true
You can learn more at https://aka.ms/tsconfig
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>code .
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>mkdir src
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>mkdir src\products
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>mkdir src\users
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > src\products\product.database.ts
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > src\products\product.interface.ts
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > src\products\product.routes.ts
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > src\users\user.database.ts
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > src\users\user.interface.ts
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > src\users\users.routes.ts
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > src\app.ts
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > .env
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > .gitignore
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > products.json
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>echo > users.json
C:\Users\Aleson C. Irag\Desktop\iton TP\REST-API-FILES\REST-API-FILES>


EXPLORER
REST-API-FILES
node_modules
src
.env
.gitignore
package-lock.json
package.json
products.json
tsconfig.json
users.json

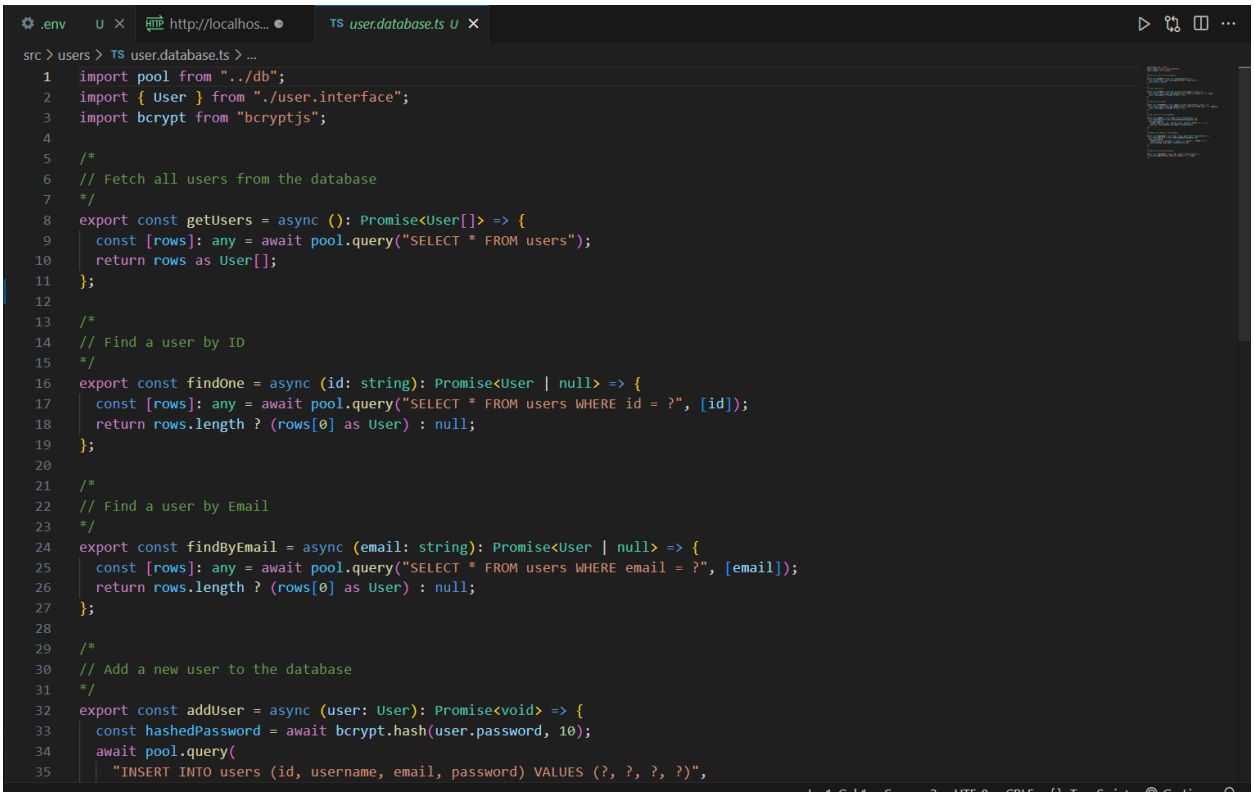
MAIN.TS


.env U
http://localhos...
main.ts U
C:\Users\Alesson C. Iraq\Desktop\iton-a\.env • Untracked
import express from "express";
import userRoutes from "./users/users.routes";
import productRoutes from "./products/product.routes";
import pool from "./db";
const app = express();
app.use(express.json());
app.use("/", userRoutes);
app.use("/", productRoutes);
const PORT = process.env.PORT || 7000;
(async () => {
try {
await pool.query("SELECT 1");
console.log("Database connected successfully!");
} catch (error) {
console.error("Database connection failed:", error);
process.exit(1);
}
})();
app.listen(PORT, () => console.log(`Server running on port ${PORT}`));

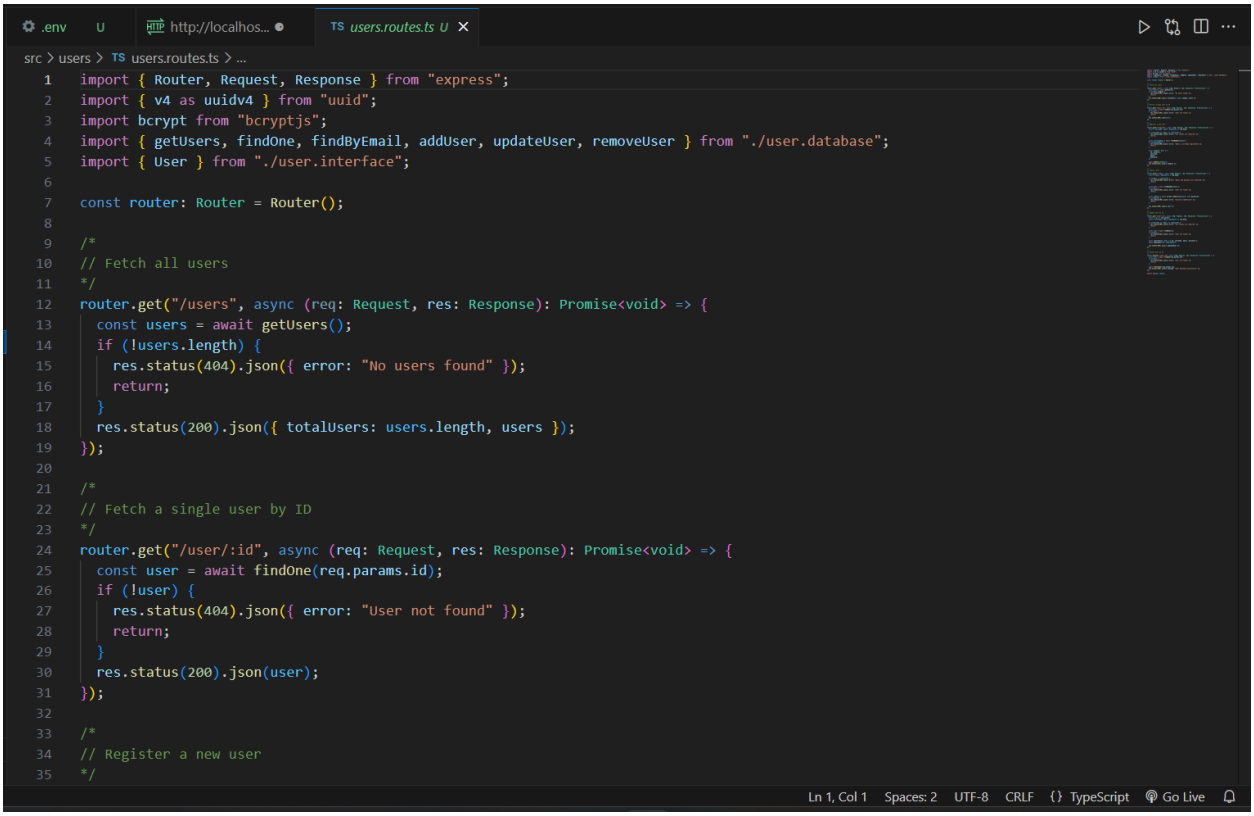
DB.TS

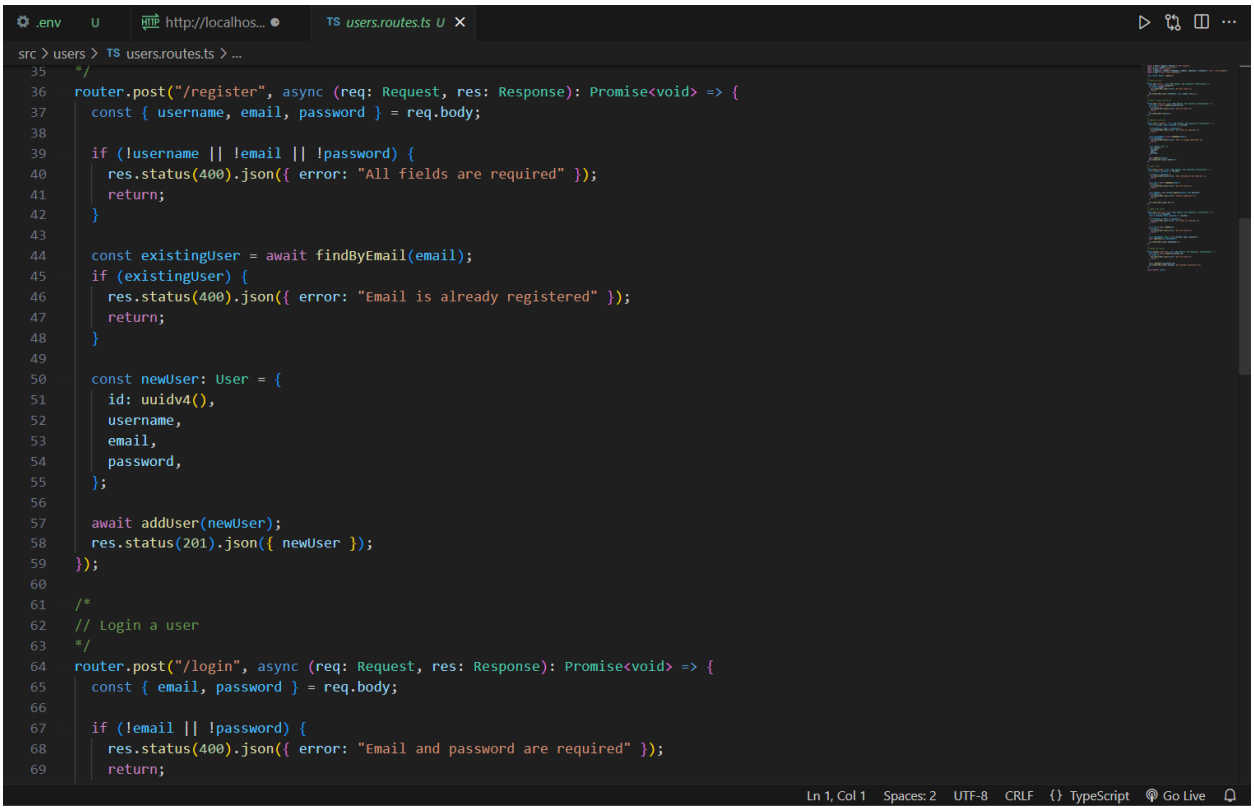


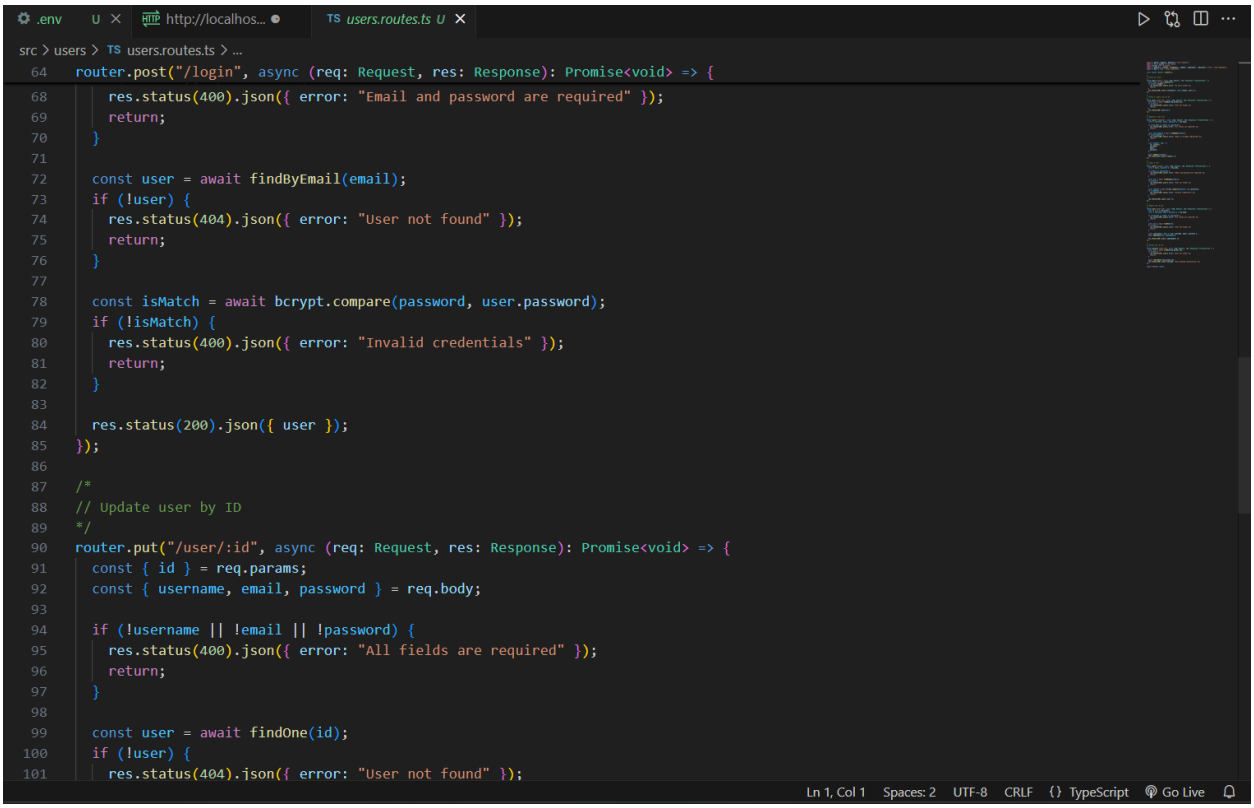
USER.DATABASE.TS

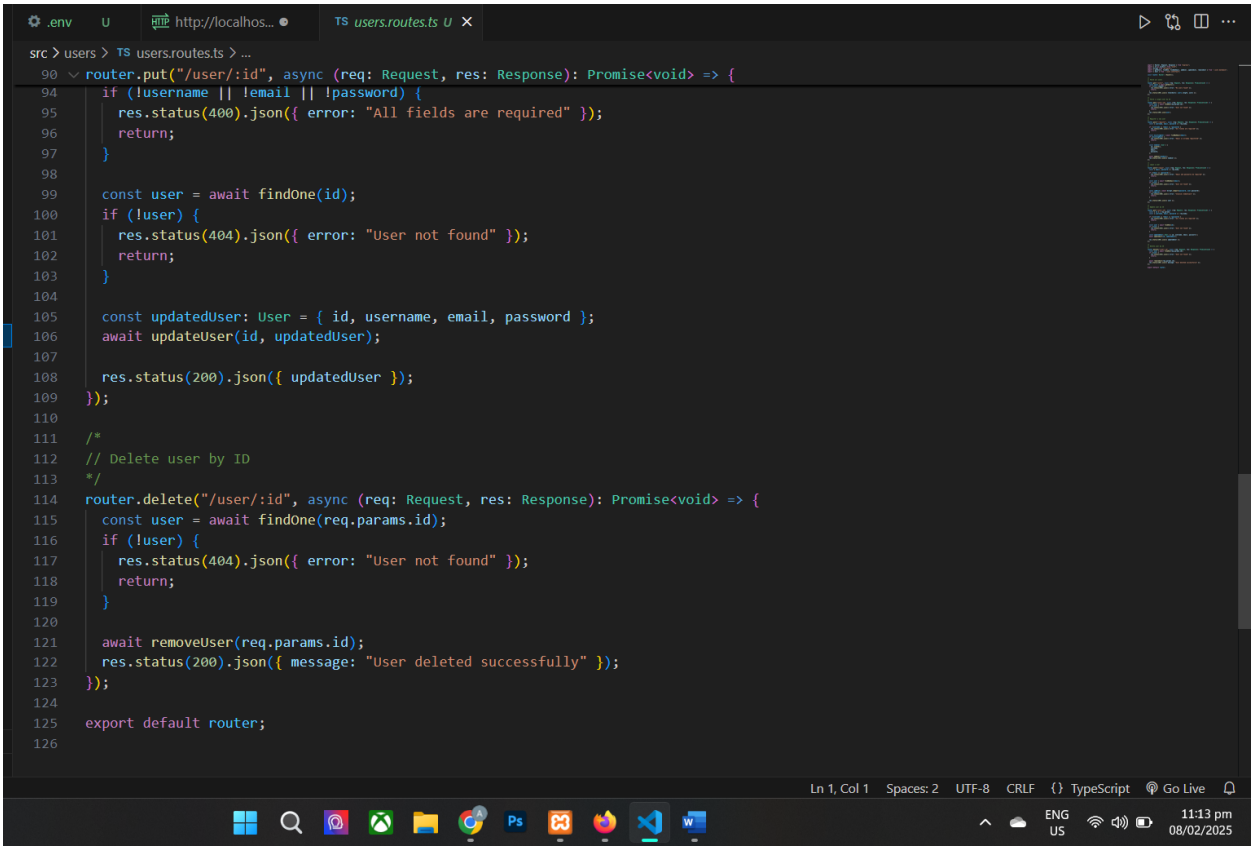


USER.ROUTES.TS




.env U
http://localhos...
users.routes.ts U
src > users > users.routes.ts > ...
*/
router.post("/register", async (req: Request, res: Response): Promise<void> => {
const { username, email, password } = req.body;
if (!username || !email || !password) {
res.status(400).json({ error: "All fields are required" });
return;
}
const existingUser = await findByEmail(email);
if (existingUser) {
res.status(400).json({ error: "Email is already registered" });
return;
}
const newUser: User = {
id: uuidv4(),
username,
email,
password,
};
await addUser(newUser);
res.status(201).json({ newUser });
});
/*
// Login a user
*/
router.post("/login", async (req: Request, res: Response): Promise<void> => {
const { email, password } = req.body;
if (!email || !password) {
res.status(400).json({ error: "Email and password are required" });
return;
Ln 1, Col 1 Spaces: 2 UTF-8 CRLF {} TypeScript Go Live


.env U
http://localhos...
users.routes.ts U
src > users > users.routes.ts > ...
router.post("/login", async (req: Request, res: Response): Promise<void> => {
res.status(400).json({ error: "Email and password are required" });
return;
}
const user = await findByEmail(email);
if (!user) {
res.status(404).json({ error: "User not found" });
return;
}
const isMatch = await bcrypt.compare(password, user.password);
if (!isMatch) {
res.status(400).json({ error: "Invalid credentials" });
return;
}
res.status(200).json({ user });
});
/*
// Update user by ID
*/
router.put("/user/:id", async (req: Request, res: Response): Promise<void> => {
const { id } = req.params;
const { username, email, password } = req.body;
if (!username || !email || !password) {
res.status(400).json({ error: "All fields are required" });
return;
}
const user = await findOne(id);
if (!user) {
res.status(404).json({ error: "User not found" });
Ln 1, Col 1 Spaces: 2 UTF-8 CRLF {} TypeScript Go Live

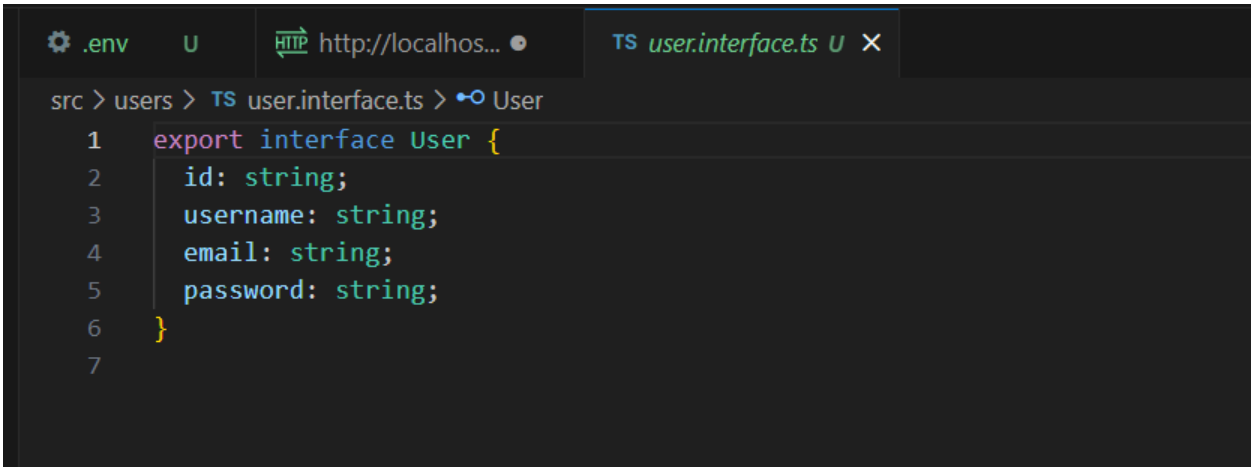


```
.env U | http://localhos... | users.routes.ts U

src > users > TS users.routes.ts > ...

router.put("/user/:id", async (req: Request, res: Response): Promise<void> => {
  if (!username || !email || !password) {
    res.status(400).json({ error: "All fields are required" });
    return;
  }

  const user = await findOne(id);
  if (!user) {
    res.status(404).json({ error: "User not found" });
    return;
  }

  const updatedUser: User = { id, username, email, password };
  await updateUser(id, updatedUser);

  res.status(200).json({ updatedUser });
});

/*
// Delete user by ID
*/
router.delete("/user/:id", async (req: Request, res: Response): Promise<void> => {
  const user = await findOne(req.params.id);
  if (!user) {
    res.status(404).json({ error: "User not found" });
    return;
  }

  await removeUser(req.params.id);
  res.status(200).json({ message: "User deleted successfully" });
});

export default router;
```

Ln 1, Col 1 Spaces: 2 UTF-8 CRLF {} TypeScript Go Live

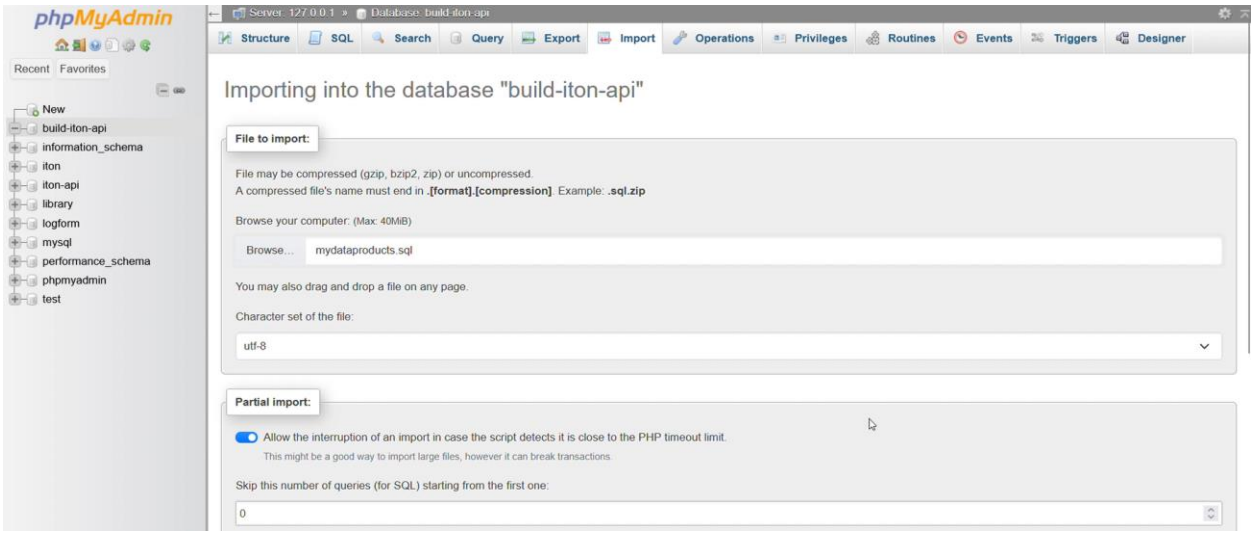
ENG US 11:13 pm 08/02/2025

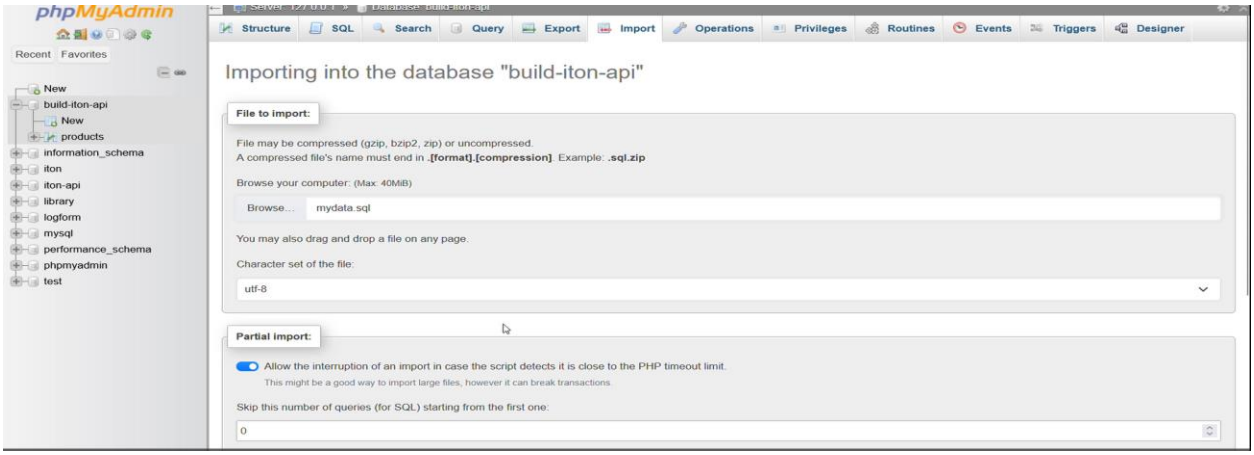
USER.INTERFACE.TS

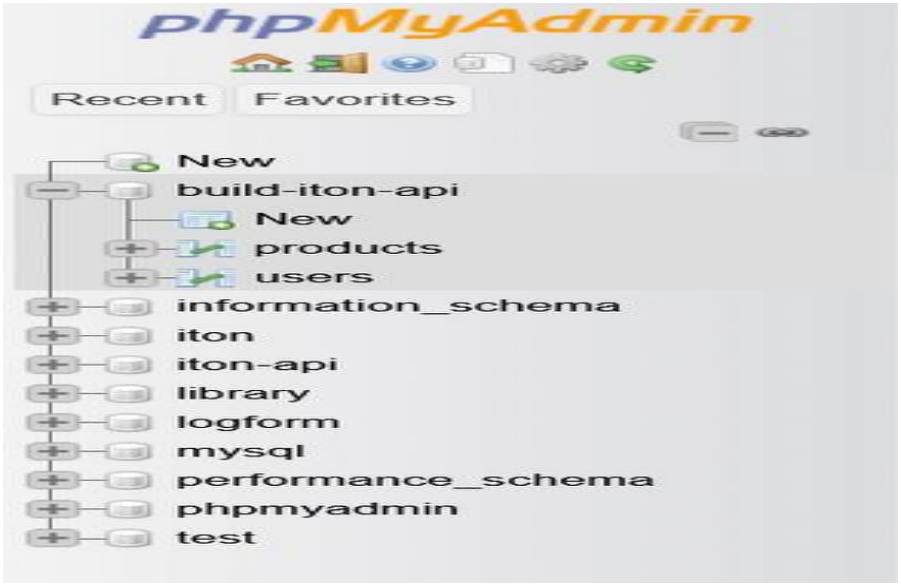


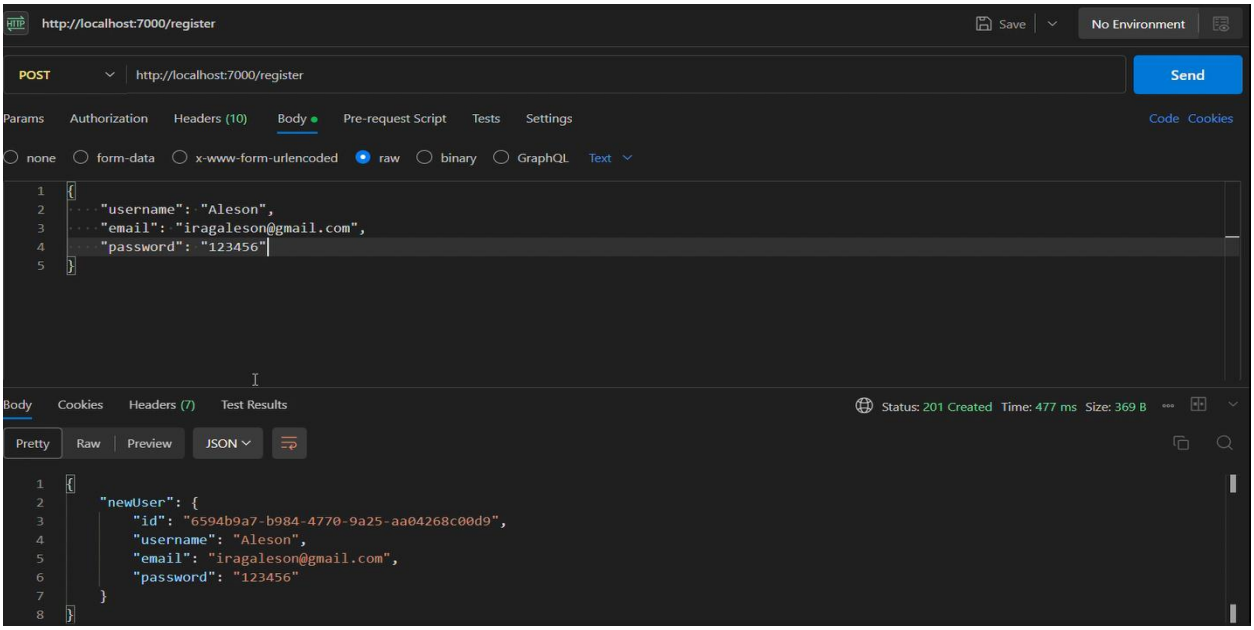
```
.env U | http://localhos... | user.interface.ts U

src > users > TS user.interface.ts > User

export interface User {
  id: string;
  username: string;
  email: string;
  password: string;
}
```

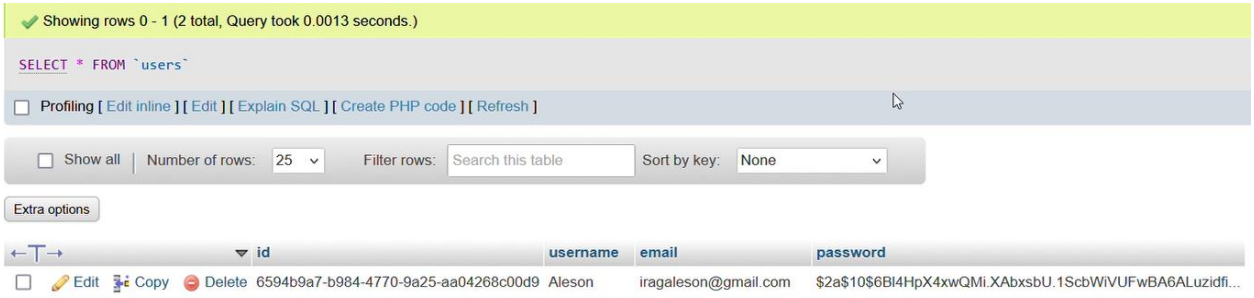
```
Run `npm audit` for details.

C:\Users\Aleson C. Irag\Desktop\ITON TP>npm run dev

> typescript-nodejs@1.0.0 dev
> ts-node-dev --pretty --respawn ./src/app.ts

[INFO] 17:01:36 ts-node-dev ver. 2.0.0 (using ts-node ver. 10.9.2, typescript ver. 5.7.3)
Server is listening on port ${PORT}
```

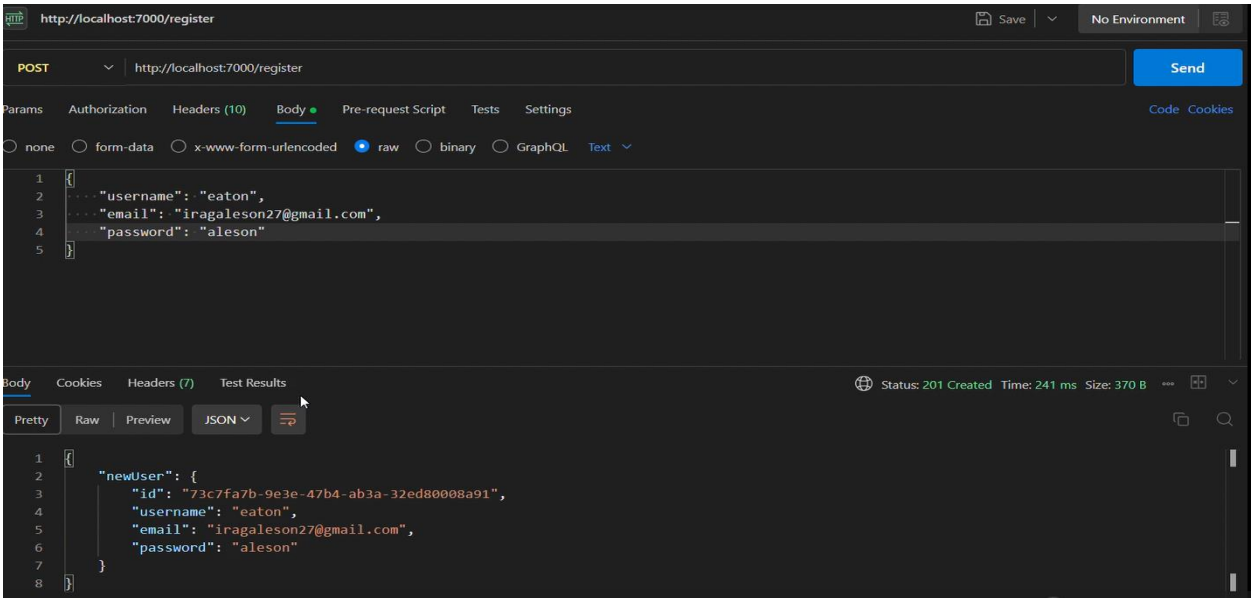

phpMyAdmin
Recent Favorites
New
build-iton-api
information_schema
iton
iton-api
library
logform
mysql
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: build-iton-api
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Designer
Importing into the database "build-iton-api"
File to import:
File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in .[format].[compression]. Example: .sql.zip
Browse your computer: (Max: 40MiB)
Browse... mydataproducts.sql
You may also drag and drop a file on any page.
Character set of the file:
utf-8
Partial import:
Allow the interruption of an import in case the script detects it is close to the PHP timeout limit.
This might be a good way to import large files, however it can break transactions.
Skip this number of queries (for SQL) starting from the first one:
0

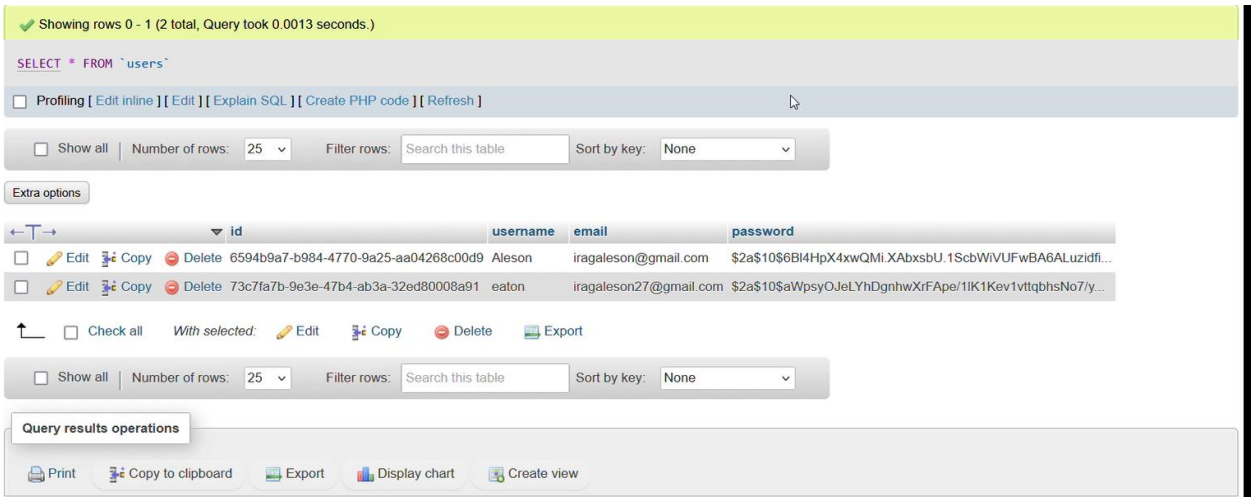

phpMyAdmin
Recent Favorites
New
build-iton-api
New
products
information_schema
iton
iton-api
library
logform
mysql
performance_schema
phpmyadmin
test
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Designer
Importing into the database "build-iton-api"
File to import:
File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in .[format].[compression]. Example: .sql.zip
Browse your computer: (Max: 40MiB)
Browse... mydata.sql
You may also drag and drop a file on any page.
Character set of the file:
utf-8
Partial import:
Allow the interruption of an import in case the script detects it is close to the PHP timeout limit.
This might be a good way to import large files, however it can break transactions.
Skip this number of queries (for SQL) starting from the first one:
0


phpMyAdmin
Recent
Favorites
New
build-iton-api
New
products
users
information_schema
iton
iton-api
library
logform
mysql
performance_schema
phpmyadmin
test


http://localhost:7000/register
Save
No Environment
POST
http://localhost:7000/register
Send
Params
Authorization
Headers (10)
Body
Pre-request Script
Tests
Settings
Code
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
Text
{
"username": "Aleson",
"email": "iragaleson@gmail.com",
"password": "123456"
}
Body
Cookies
Headers (7)
Test Results
Status: 201 Created
Time: 477 ms
Size: 369 B
Pretty
Raw
Preview
JSON
{
"newUser": {
"id": "6594b9a7-b984-4770-9a25-aa04268c00d9",
"username": "Aleson",
"email": "iragaleson@gmail.com",
"password": "123456"
}
}


Showing rows 0 - 1 (2 total, Query took 0.0013 seconds.)
SELECT * FROM `users`
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all
Number of rows:
25
Filter rows:
Search this table
Sort by key:
None
Extra options
id
username
email
password
Edit
Copy
Delete
6594b9a7-b984-4770-9a25-aa04268c00d9
Aleson
iragaleson@gmail.com
$2a$10$6Bl4HpX4xwQMi.XAbxsbU.1ScbWiVUFwBA6ALuzidfi...


http://localhost:7000/register
Save
No Environment
POST
http://localhost:7000/register
Send
Params
Authorization
Headers (10)
Body
Pre-request Script
Tests
Settings
Code
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
Text
{
"username": "eaton",
"email": "iragaleson27@gmail.com",
"password": "aleson"
}
Body
Cookies
Headers (7)
Test Results
Status: 201 Created
Time: 241 ms
Size: 370 B
Pretty
Raw
Preview
JSON
{
"newUser": {
"id": "73c7fa7b-9e3e-47b4-ab3a-32ed80008a91",
"username": "eaton",
"email": "iragaleson27@gmail.com",
"password": "aleson"
}
}


Showing rows 0 - 1 (2 total, Query took 0.0013 seconds.)
SELECT * FROM `users`
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options
id username email password
Edit Copy Delete 6594b9a7-b984-4770-9a25-aa04268c00d9 Aleson iragaleson@gmail.com $2a$10$6Bl4HpX4xwQMi.XAbxsbU.1ScbWiVUFwBA6ALuzidfi...
Edit Copy Delete 73c7fa7b-9e3e-47b4-ab3a-32ed80008a91 eaton iragaleson27@gmail.com $2a$10$aWpsyOJeLYhDgnhwXrFApe/1lK1Kev1vttqbhsNo7/y...
Check all With selected: Edit Copy Delete Export
Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Query results operations
Print Copy to clipboard Export Display chart Create view