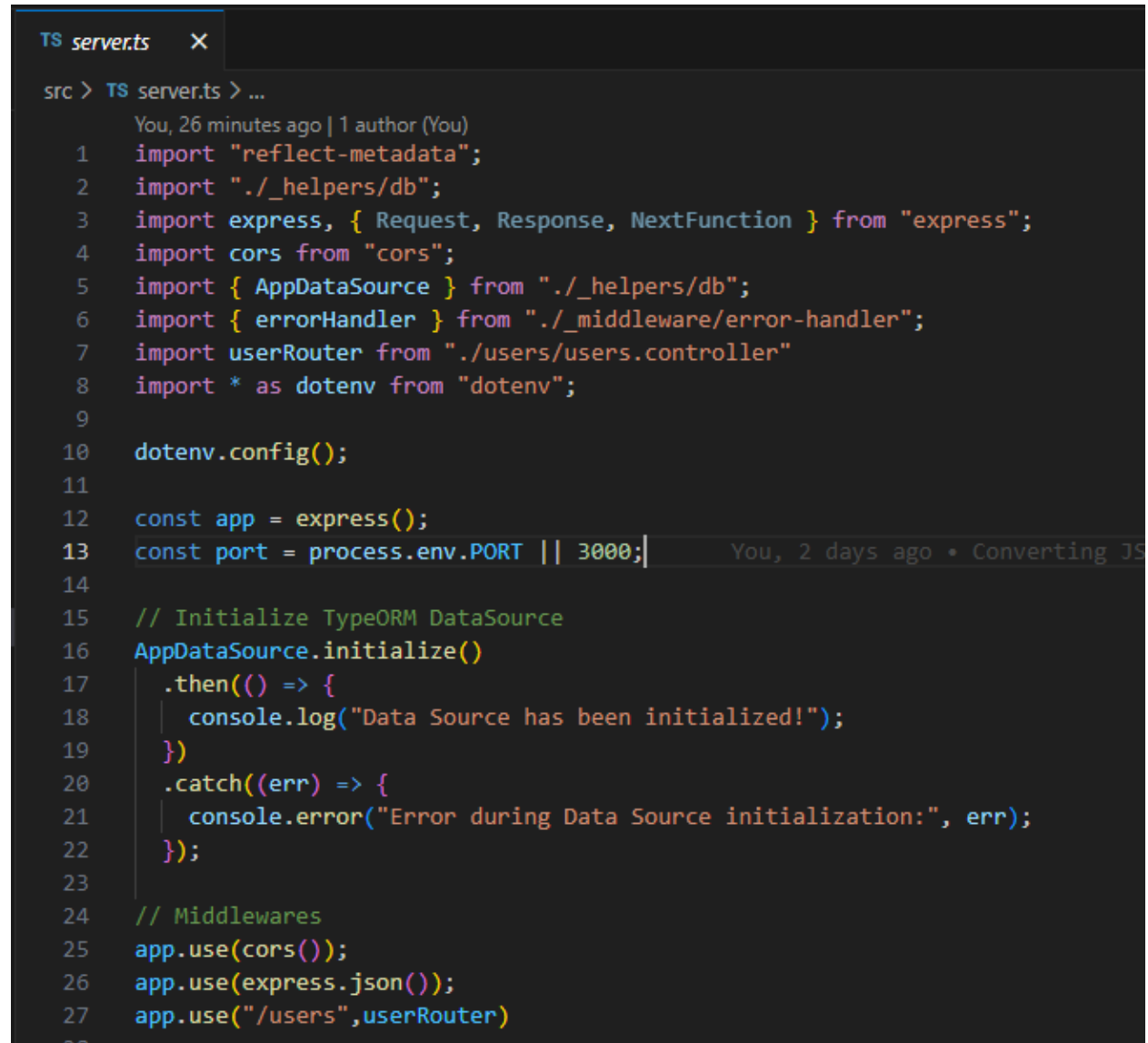


Updating the server.ts :

Importing the `userRouter` object from the file located at `./users/users.controller`



```
TS server.ts X
src > TS server.ts > ...
  You, 26 minutes ago | 1 author (You)
1  import "reflect-metadata";
2  import "../_helpers/db";
3  import express, { Request, Response, NextFunction } from "express";
4  import cors from "cors";
5  import { AppDataSource } from "../_helpers/db";
6  import { errorHandler } from "../_middleware/error-handler";
7  import userRouter from "../users/users.controller";
8  import * as dotenv from "dotenv";
9
10 dotenv.config();
11
12 const app = express();
13 const port = process.env.PORT || 3000;
14
15 // Initialize TypeORM DataSource
16 AppDataSource.initialize()
17   .then(() => {
18     console.log("Data Source has been initialized!");
19   })
20   .catch((err) => {
21     console.error("Error during Data Source initialization:", err);
22   });
23
24 // Middlewares
25 app.use(cors());
26 app.use(express.json());
27 app.use("/users", userRouter);
28
```

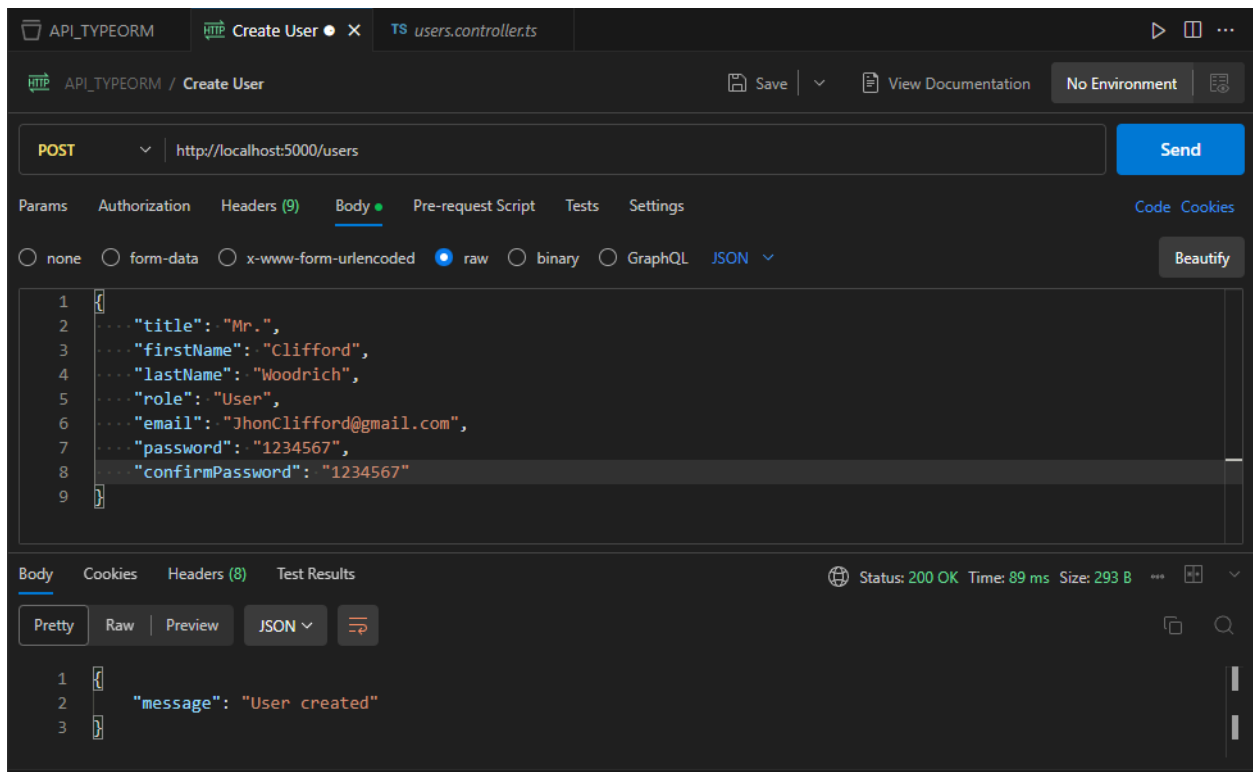
Updating the db.ts

Importing the `User` entity (model) from the file located at `../users/user.model`.

```
TS db.ts x
src > _helpers > TS db.ts > ...
You, 31 minutes ago | 1 author (You)
1 import "reflect-metadata";
2 import { DataSource } from "typeorm";
3 import { User } from "../users/user.model";
4 import mysql from "mysql2/promise";
5 import dotenv from "dotenv";
6
```

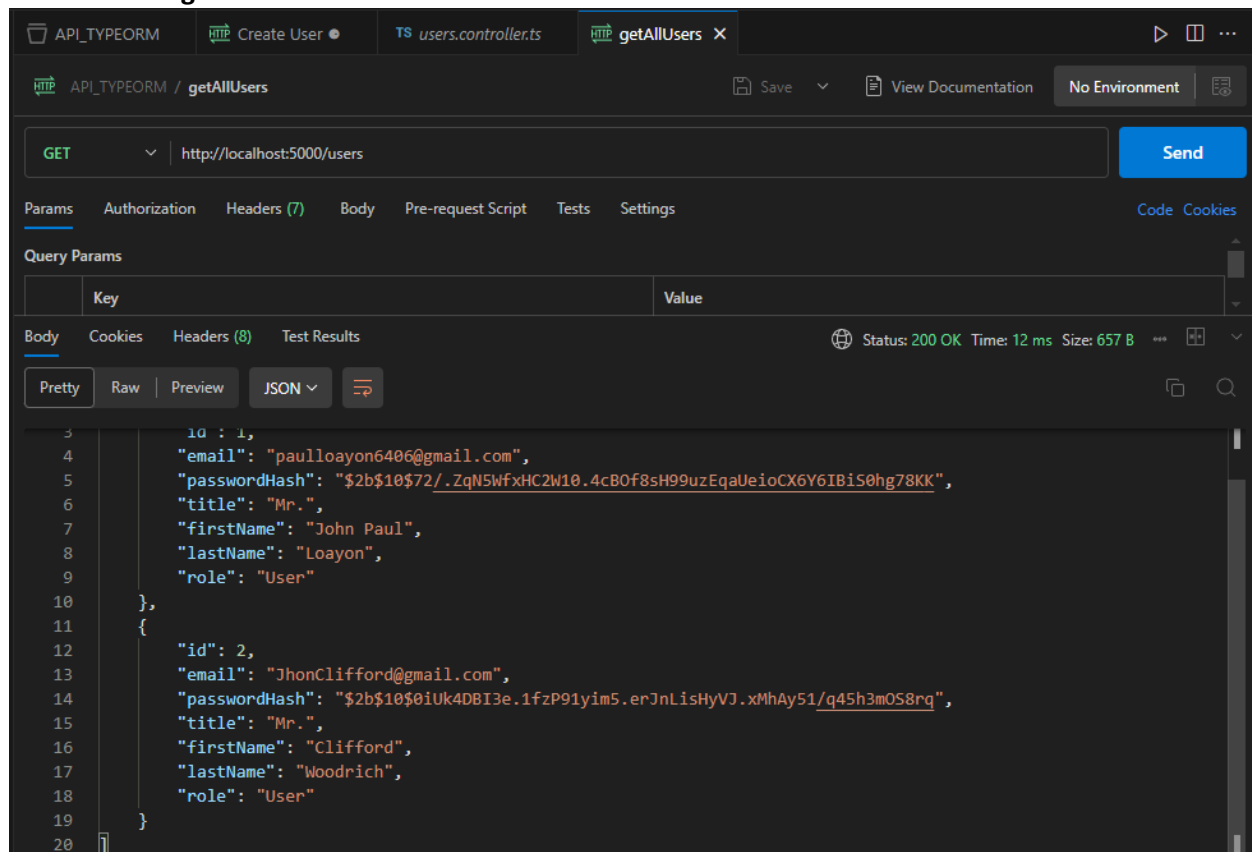
Creating a New user using postman:

In this photo I created a new user to test in postman.



Getting All Users Created:

Testing if all the users I created existed.



The screenshot shows a REST client interface with the following details:

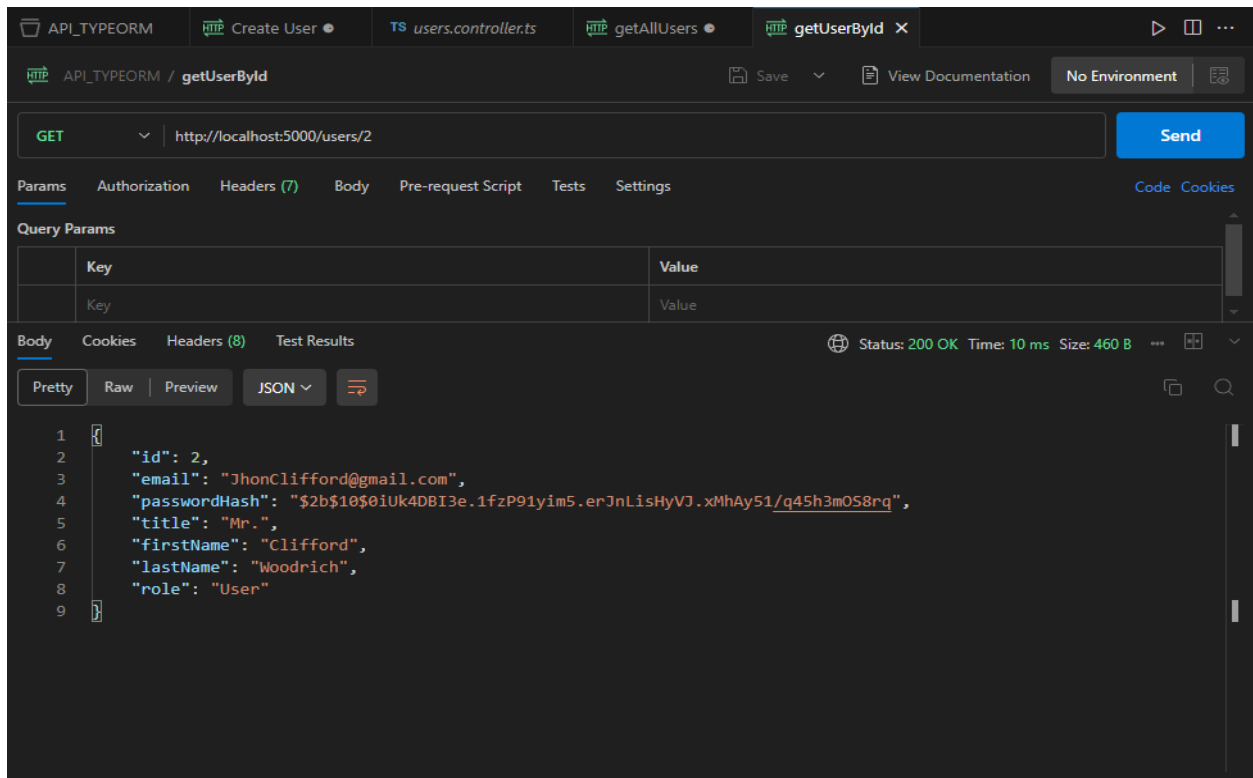
- API TypeORM / getAllUsers**
- Method:** GET
- URL:** http://localhost:5000/users
- Status:** 200 OK
- Time:** 12 ms
- Size:** 657 B

The response body is a JSON array of two user objects:

```
3  {
4    "id": 1,
5    "email": "paulloayon6406@gmail.com",
6    "passwordHash": "$2b$10$72/.ZqN5WfxHC2W10.4cB0f8sH99uzEqaUeioCX6Y6IBiS0hg78KK",
7    "title": "Mr.",
8    "firstName": "John Paul",
9    "lastName": "Loayon",
10   "role": "User"
11 },
12 {
13   "id": 2,
14   "email": "JhonClifford@gmail.com",
15   "passwordHash": "$2b$10$0iUk4DBI3e.1fzP91yim5.erJnLisHyVJ.xMhAy51/q45h3mOS8rq",
16   "title": "Mr.",
17   "firstName": "Clifford",
18   "lastName": "Woodrich",
19   "role": "User"
20 }
```

Getting User by ID:

In this photo, I checked if the function `getUserById` is working using postman. The user id I check is number 2.



Updating User:

I change something and updated the user with the ID of number 1.

The screenshot shows a REST client interface with a PUT request to `http://localhost:5000/users/1`. The request body is a JSON object with the following fields: `title` (Ms.), `firstName` (Gwenchana), `lastName` (Sayo), `role` (Admin), `email` (Gwentenining@gmail.com), `password` (12345679), and `confirmPassword` (12345679). The response is a JSON object with a `message` field set to "User updated".

API_TYPEORM / UpdateUser

PUT `http://localhost:5000/users/1` Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Code Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   ... "title": "Ms.",
3   ... "firstName": "Gwenchana",
4   ... "lastName": "Sayo",
5   ... "role": "Admin",
6   ... "email": "Gwentenining@gmail.com",
7   ... "password": "12345679",
8   ... "confirmPassword": "12345679"
9 }
```

Body Cookies Headers (8) Test Results Status: 200 OK Time: 125 ms Size: 293 B

Pretty Raw Preview JSON

```
1 {
2   "message": "User updated"
3 }
```

Checking in the database if the data existed.

```
information_schema |
mysql              |
node-mysql-crud-api |
performance_schema |
phpmyadmin         |
sysarch            |
test               |
+-----+
rows in set (0.028 sec)

ariaDB [(none)]> use `node-mysql-crud-api`;
Database changed
ariaDB [node-mysql-crud-api]> show tables;
+-----+
Tables_in_node-mysql-crud-api |
+-----+
user                          |
+-----+
row in set (0.001 sec)

ariaDB [node-mysql-crud-api]> select * from user;
+-----+-----+-----+-----+-----+-----+-----+
id | email | passwordHash | title | firstName | lastName | role |
+-----+-----+-----+-----+-----+-----+-----+
1 | Gwentenining@gmail.com | $2b$10$iJfW3JzeK8kMgHjS3Nj7DuUPZMLTjPBgF0WpFhLFp04Ihc1d0e0k. | Ms. | Gwenchana | Sayo | Admin |
2 | JhonClifford@gmail.com | $2b$10$0iUk4DBI3e.1fzP9iyim5.erJnLisHyVJ.xMhAy51/q45h3m0S8rq | Mr. | Clifford | Woodrich | User |
+-----+-----+-----+-----+-----+-----+-----+
rows in set (0.000 sec)

ariaDB [node-mysql-crud-api]> _
```

Deleting the User:

in this photo I deleted the user with the ID number of 2.

The screenshot shows a REST client interface with the following details:

- API Type:** API_TYPEORM
- Method:** DELETE
- URL:** http://localhost:5000/users/2
- Status:** 200 OK
- Time:** 40 ms
- Size:** 293 B
- Response:** {"message": "User deleted"}

Checking the database is updated after the deletion of user with id of number 2

```
information_schema |
mysql              |
node-mysql-crud-api |
performance_schema |
phpmyadmin         |
sysarch            |
test               |
+-----+
7 rows in set (0.028 sec)

MariaDB [(none)]> use `node-mysql-crud-api`;
Database changed
MariaDB [node-mysql-crud-api]> show tables;
+-----+
| Tables_in_node-mysql-crud-api |
+-----+
| user                           |
+-----+
1 row in set (0.001 sec)

MariaDB [node-mysql-crud-api]> select * from user;
+-----+-----+-----+-----+-----+-----+
| id | email | passwordHash | title | firstName | lastName | role |
+-----+-----+-----+-----+-----+-----+
| 1 | Gwenenining@gmail.com | $2b$10$iJfW3JzeK8kMgHjS3Nj7DuUPZMLTjPBgF0WpfhLFp04Ihc1d0e0k. | Ms. | Gwenchana | Sayo | Admin |
| 2 | JhonClifford@gmail.com | $2b$10$0iUK4DBI3e.1fzP9iyim5.erJnLiShyVJ.xMhAy51/q45h3m0S8nq | Mr. | Clifford | Woodrich | User |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.000 sec)

MariaDB [node-mysql-crud-api]> select * from user;
+-----+-----+-----+-----+-----+-----+
| id | email | passwordHash | title | firstName | lastName | role |
+-----+-----+-----+-----+-----+-----+
| 1 | Gwenenining@gmail.com | $2b$10$iJfW3JzeK8kMgHjS3Nj7DuUPZMLTjPBgF0WpfhLFp04Ihc1d0e0k. | Ms. | Gwenchana | Sayo | Admin |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.001 sec)

MariaDB [node-mysql-crud-api]> _
```