**HackFSE2 eStock Market Application Implementation**

**Technologies Used:**

|  |  |  |
| --- | --- | --- |
| S.No | Phase | Technologies used |
| 1 | Backend | Java, Spring boot, Microservices, Eureka server, API Gateway, Swagger2 |
| 2 | Database | Mongo DB, MySql |
| 3 | Frontend | Angular 12, Bootstrap |
| 4 | Cloud | AWS ECS, AWS Fargate, MongoDB Atlas |

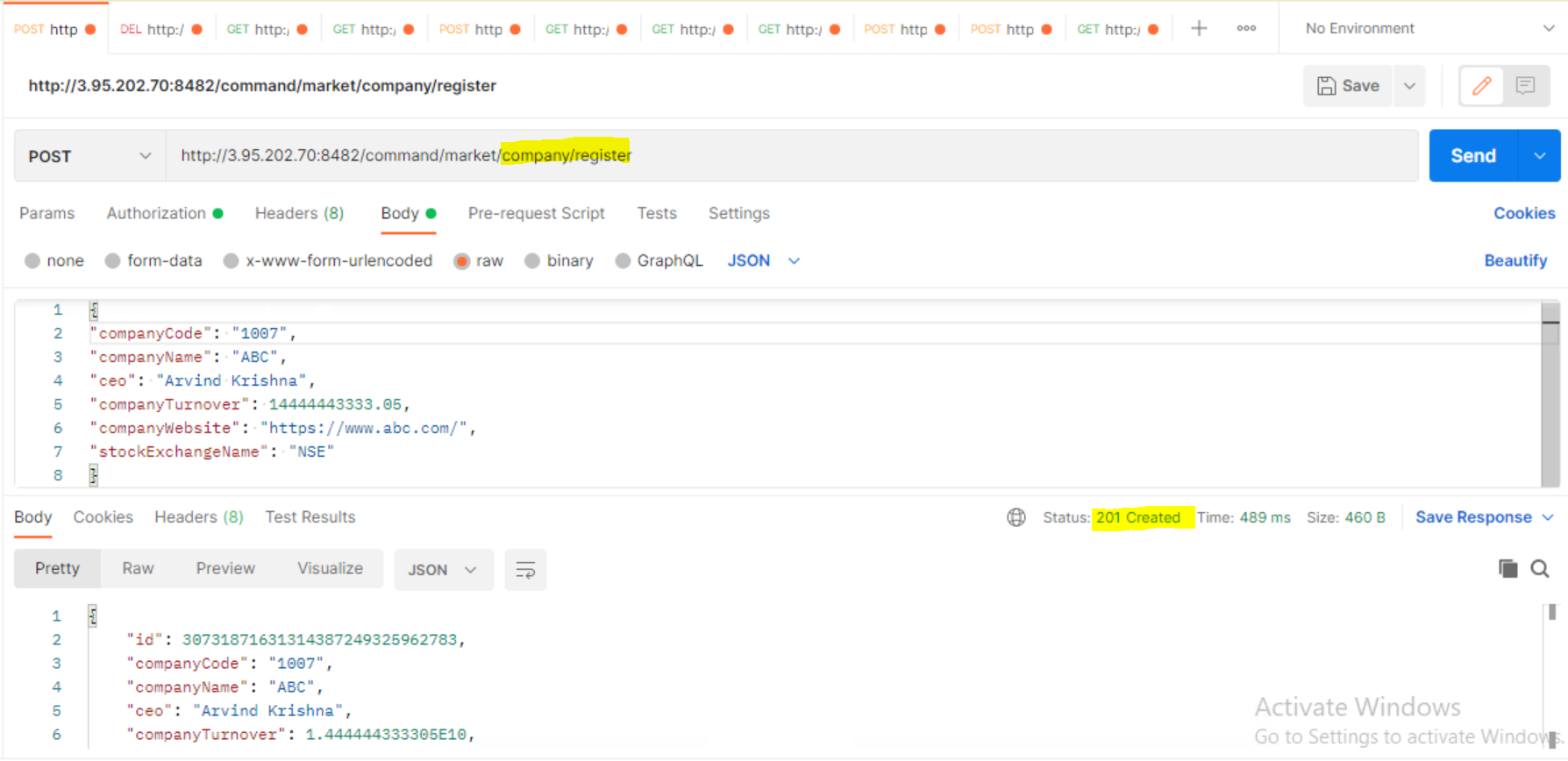
**REST API Endpoints used:**

|  |  |  |
| --- | --- | --- |
| POST | http://<host>:<port>/command/market/company/register | Register a new company |
| DELETE | http://<host>:<port>/command/market/company/delete/<companyCode> | Deletes a company |
| GET | http://<host>:<port>/query/market/company/getall | Fetches all the Company Details |
| GET | http://<host>:<port>/query/market/company/info/<companyCode> | Fetches the Company Details |
| POST | http://<host>:<port>/command/market/stock/add | Add new stock price |
| GET | http://<host>:<port>/query/market/stock/get/<companycode>/<startdate>/<enddate> | Fetches Stock Price List |
| GET | http://<host>:<port>/query/market/stock/aggregate/<companyCode> | Get the aggregate values |
| POST | http://<host>:<port>/user/signup | Register user |
| POST | http://<host>:<port>/user/login | Login user |

**Backend implementation**:

1. Register a new company

If company code is not present

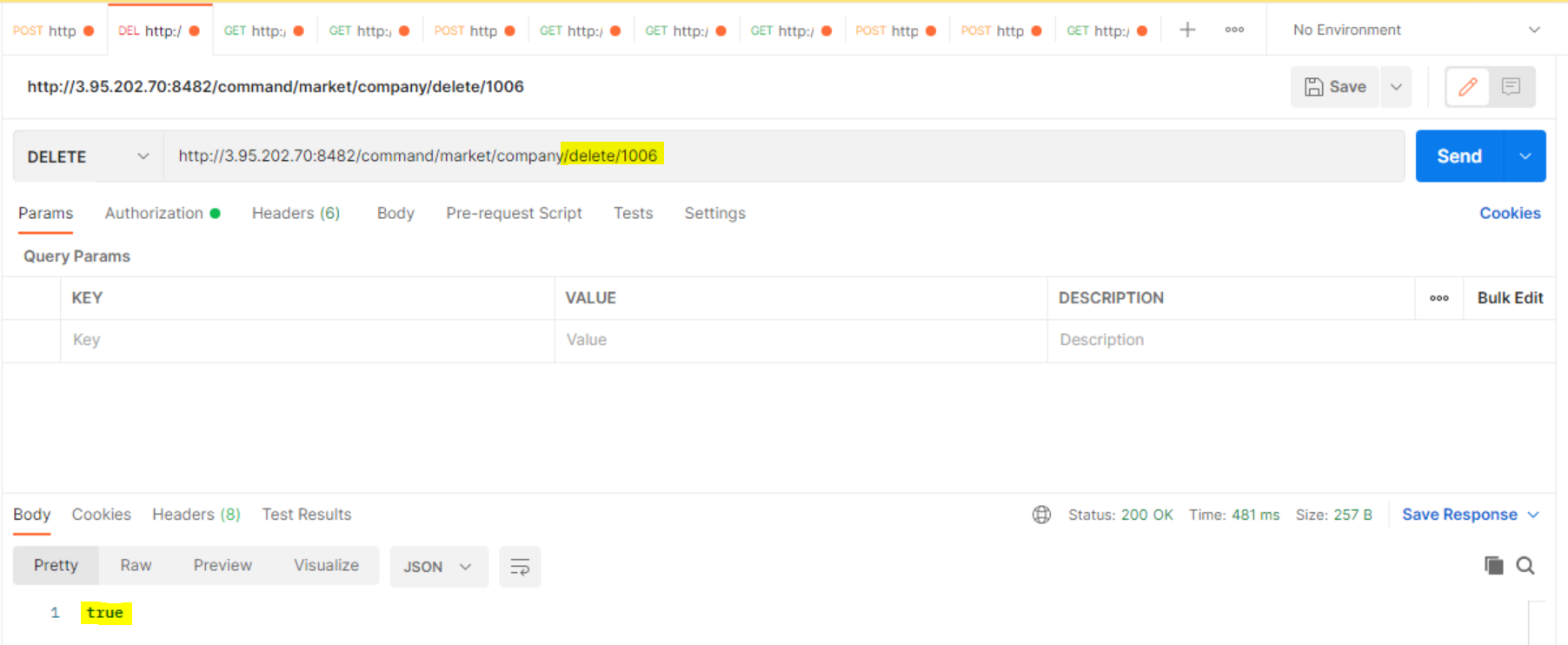


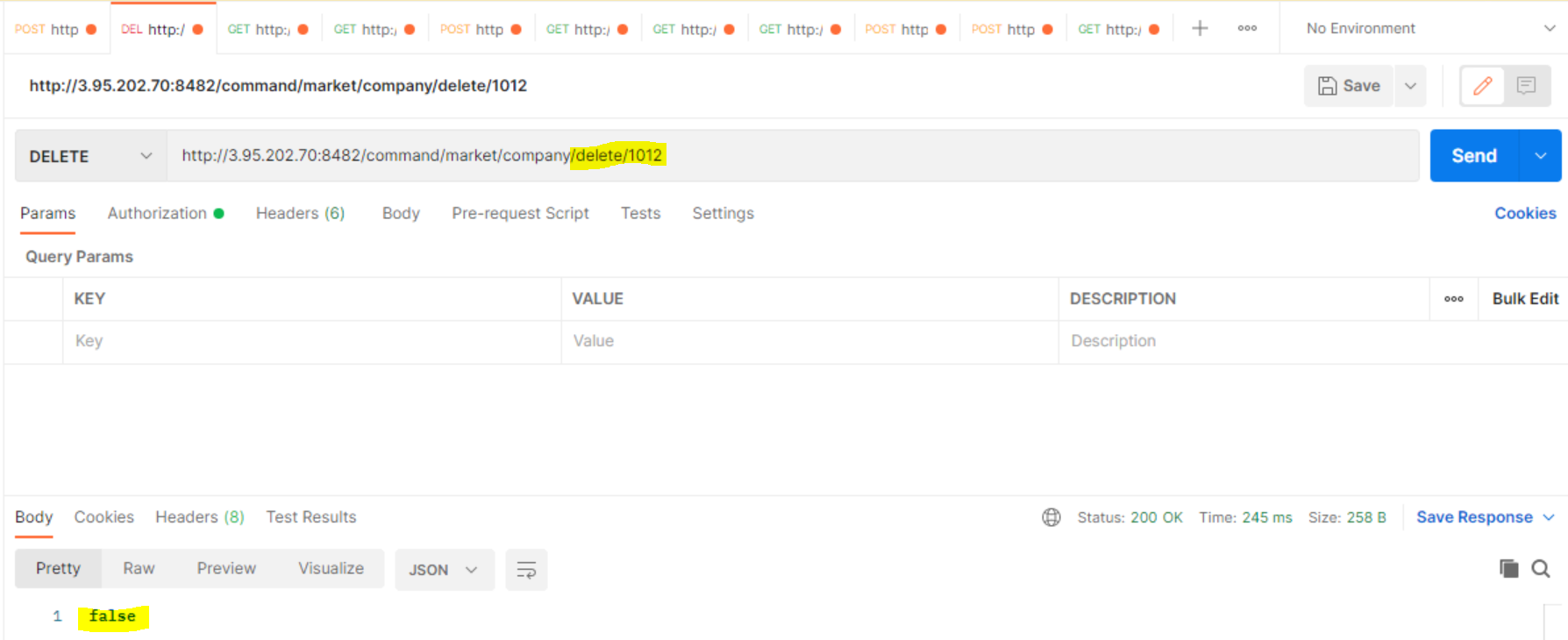
If already exists



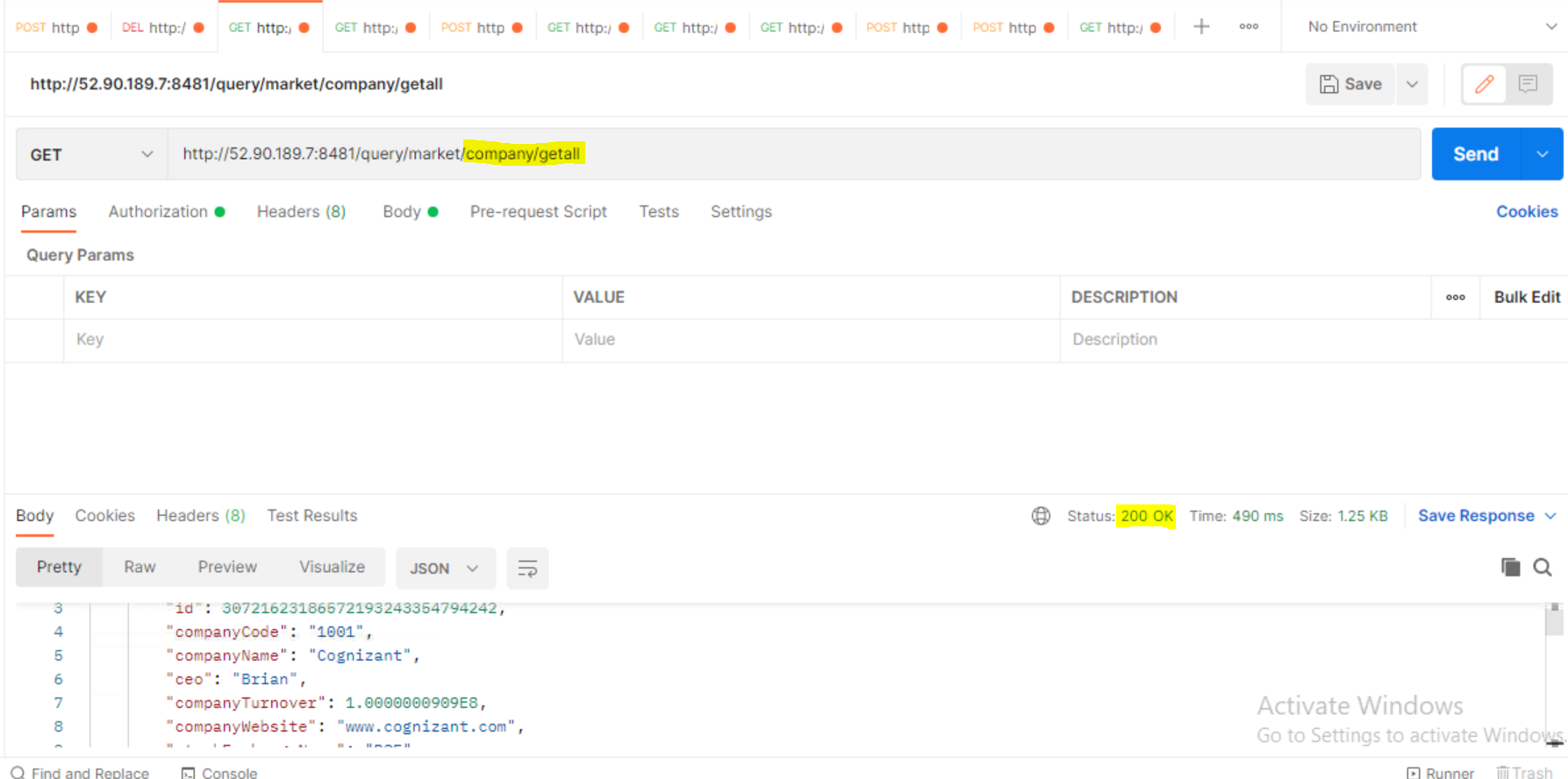
2. Delete a company

If company exists, return true after deleting. Otherwise, false.



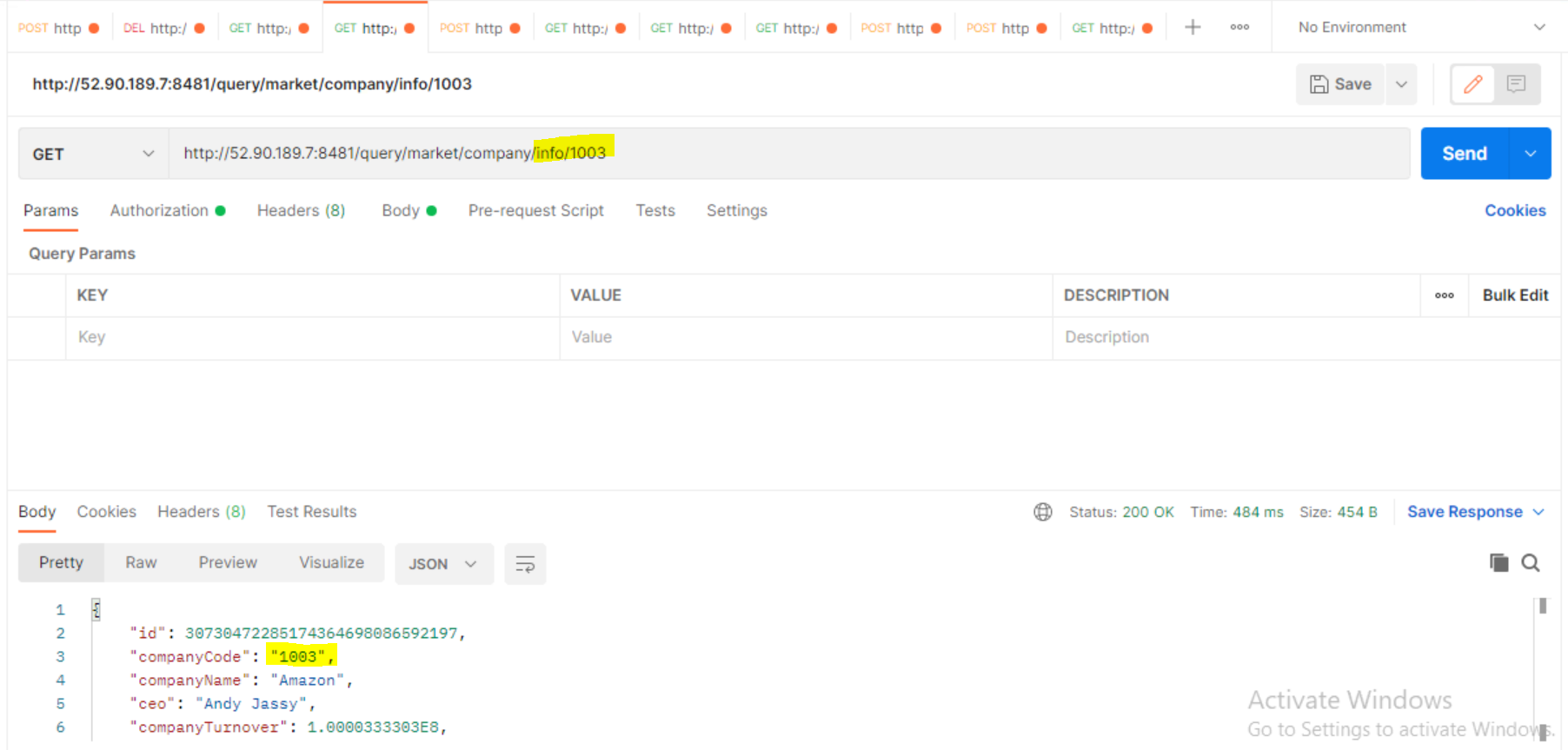


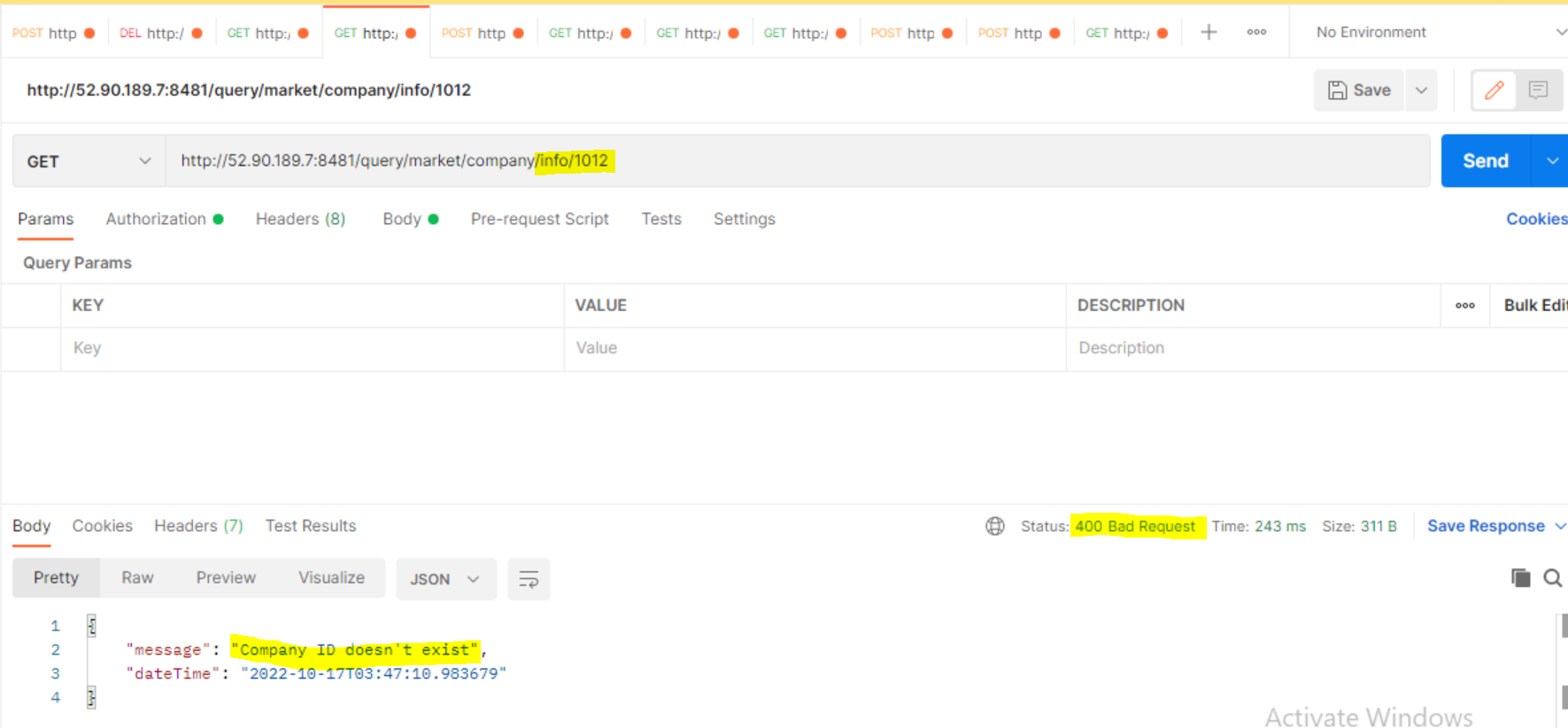
3. Fetch all the companies



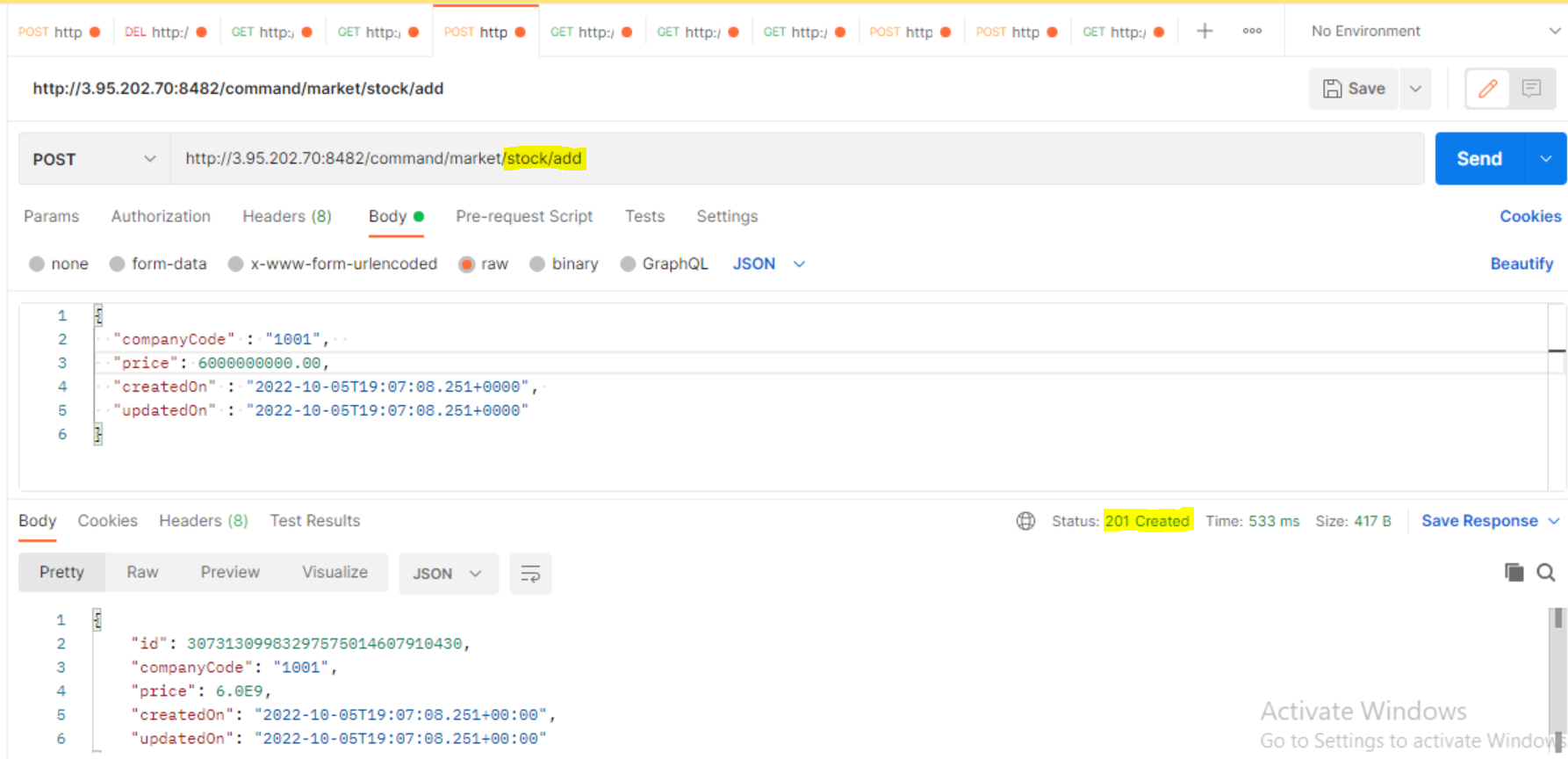
4. Fetch specific company details

If company present, returns the company details. If not, then gives error – Company ID doesn’t exist.

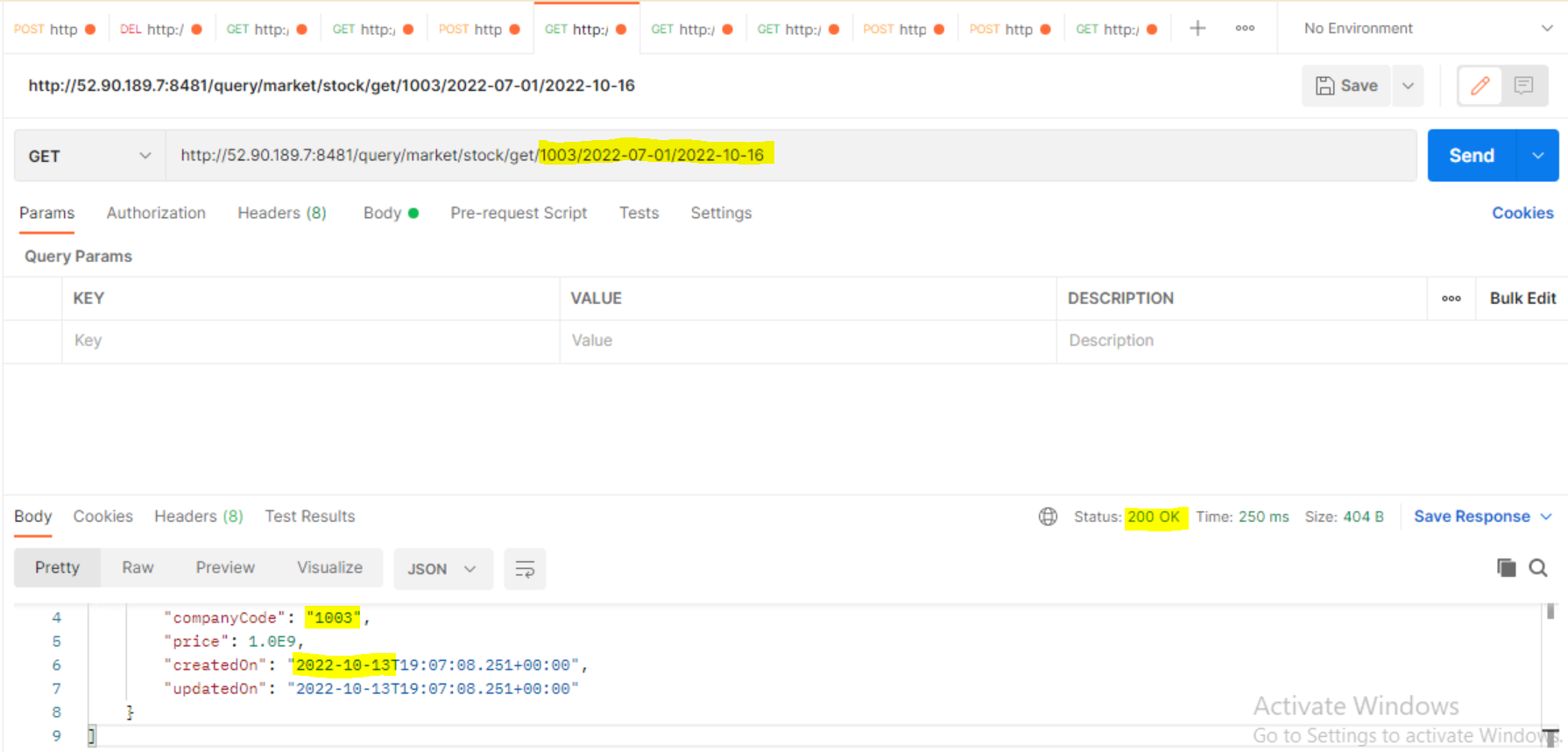




5. Add company stock details

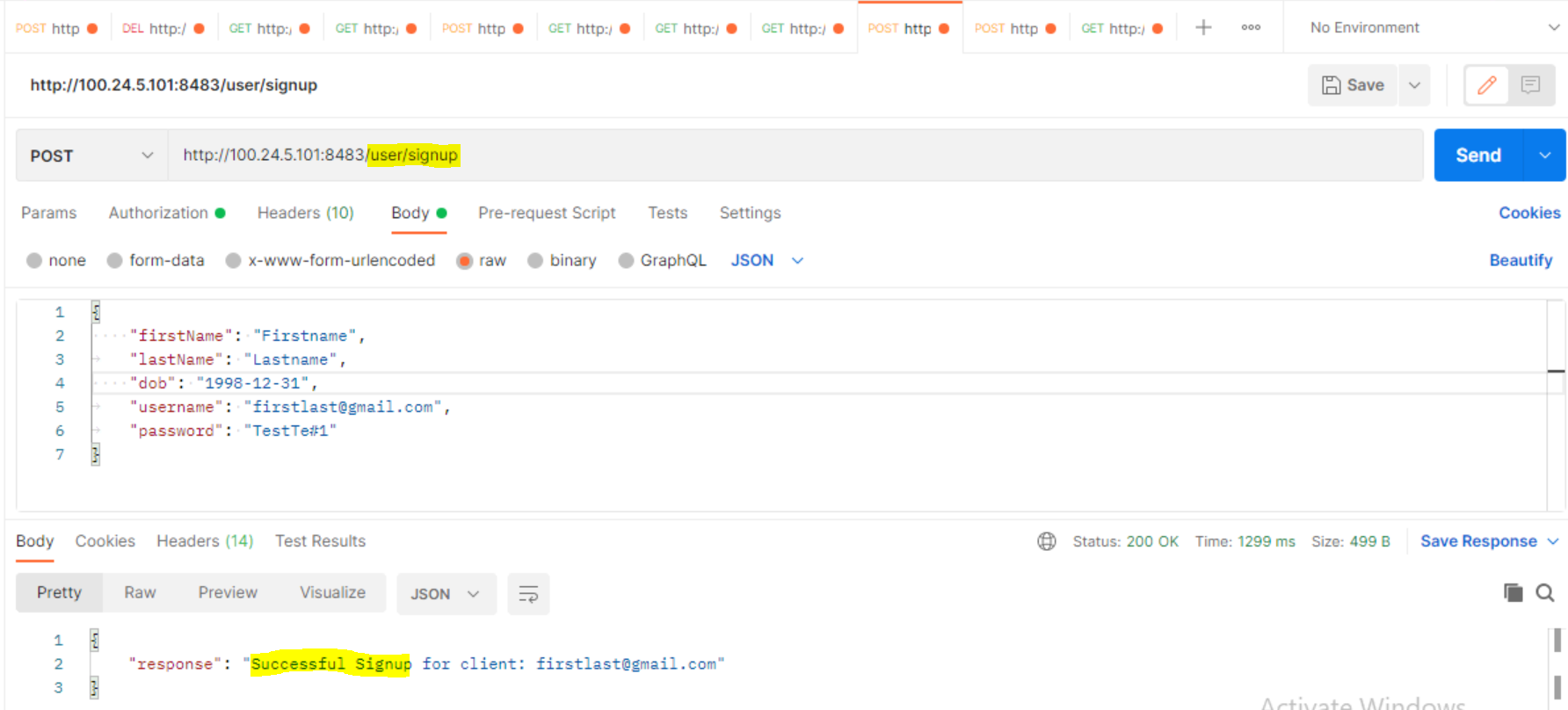


6. Fetch the stock details between the given dates

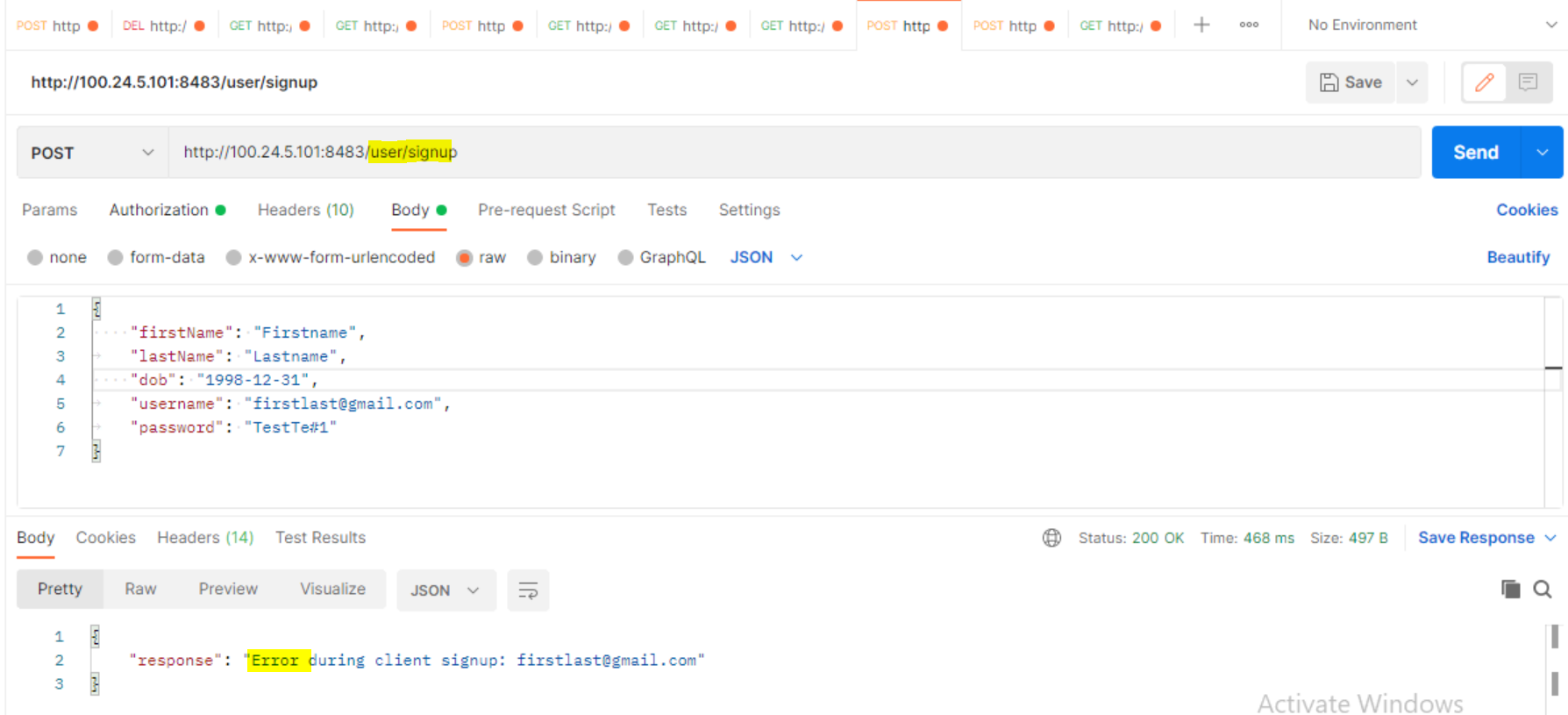


7. User signup

Successful register of the user:

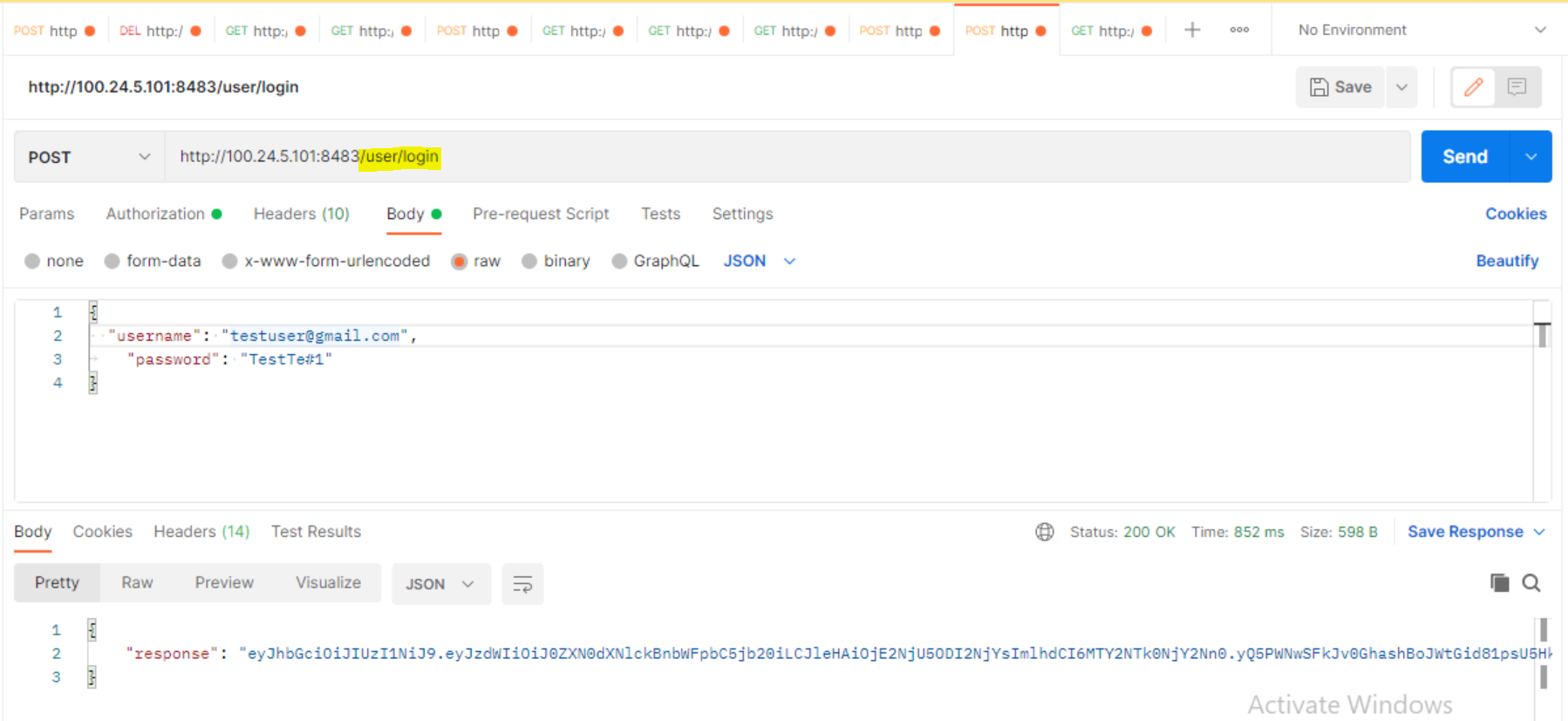


If user ID already exists, then throws an error

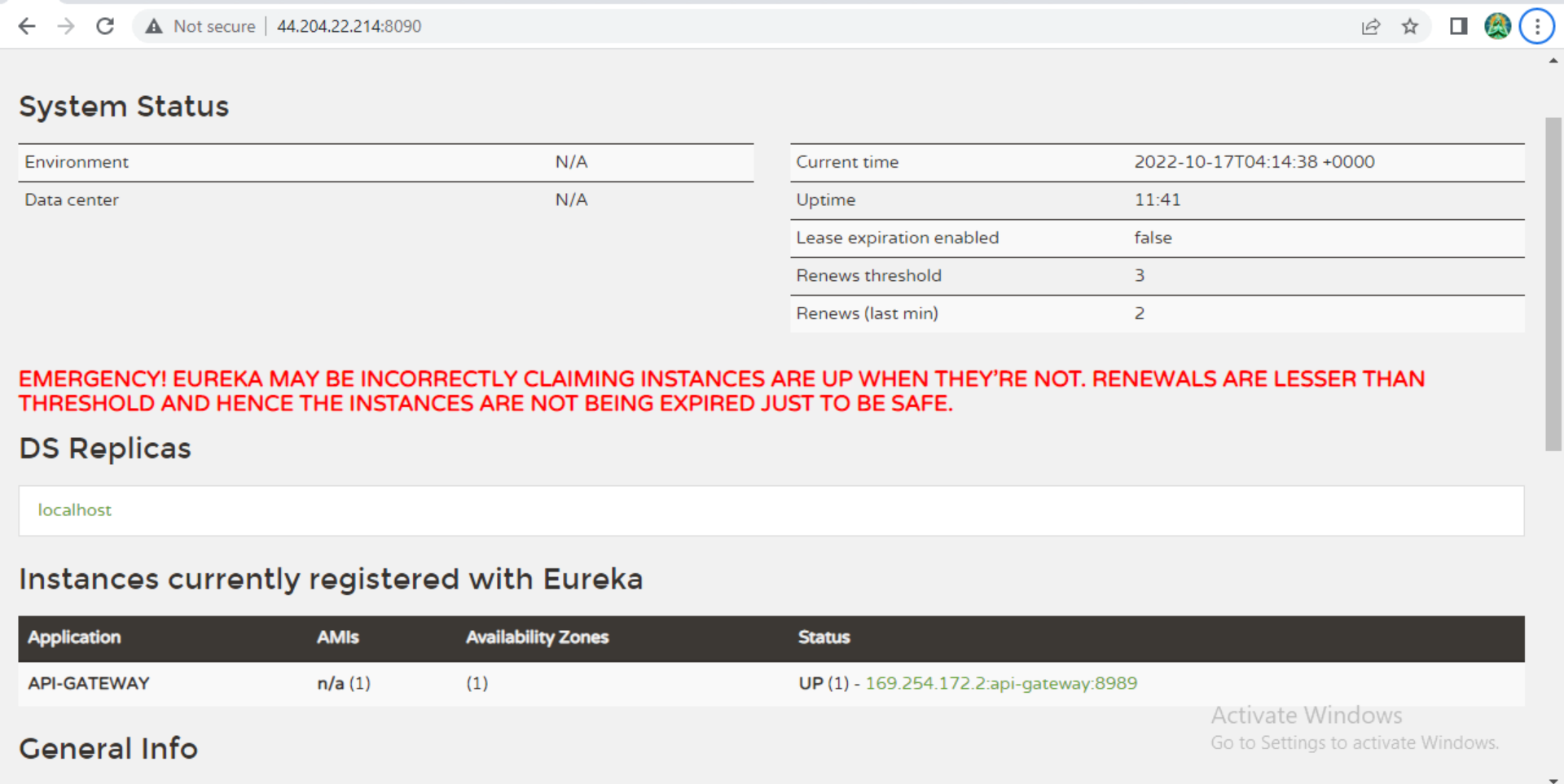


8. User Login

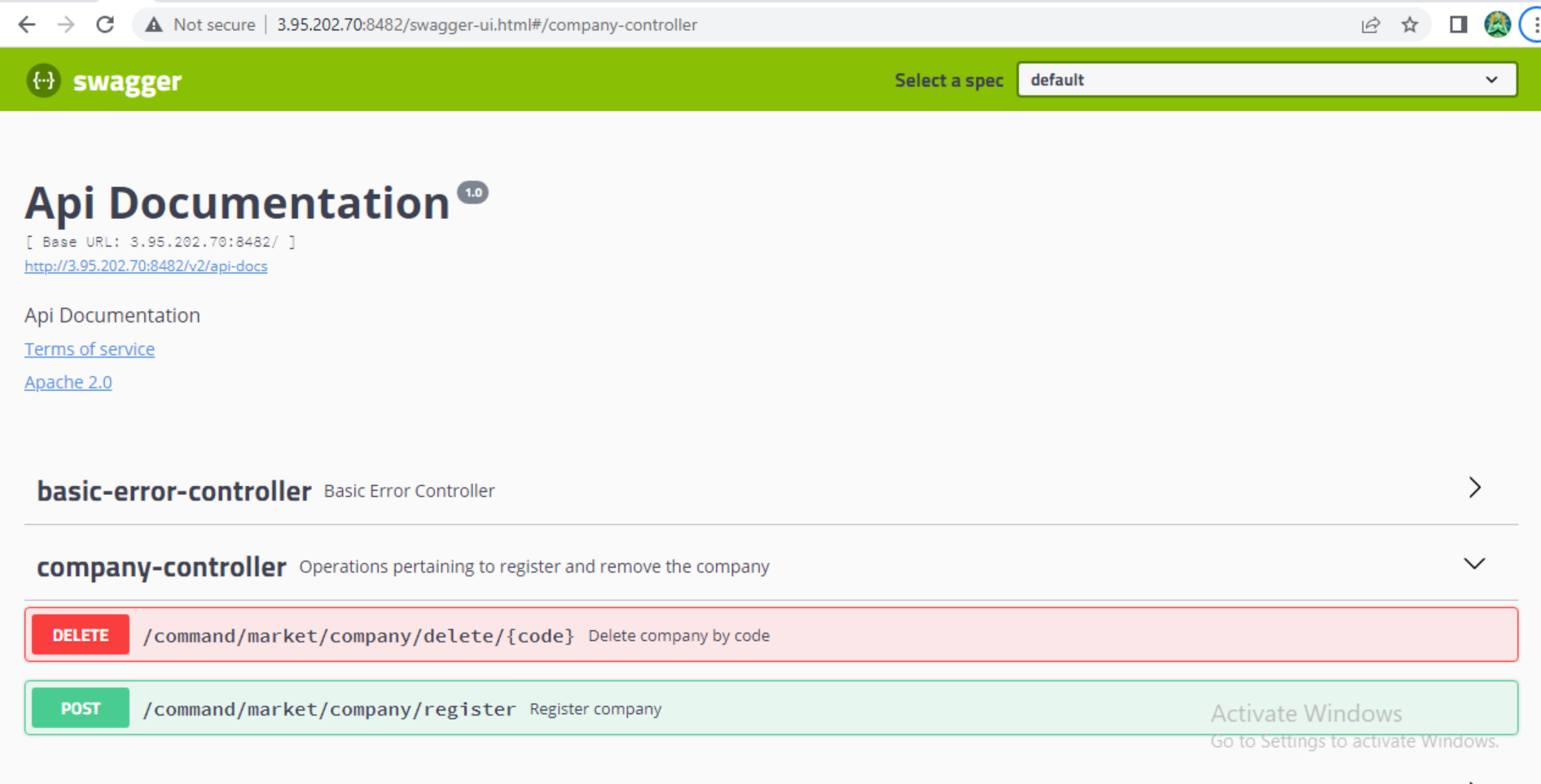
Returns token when user login is successful.



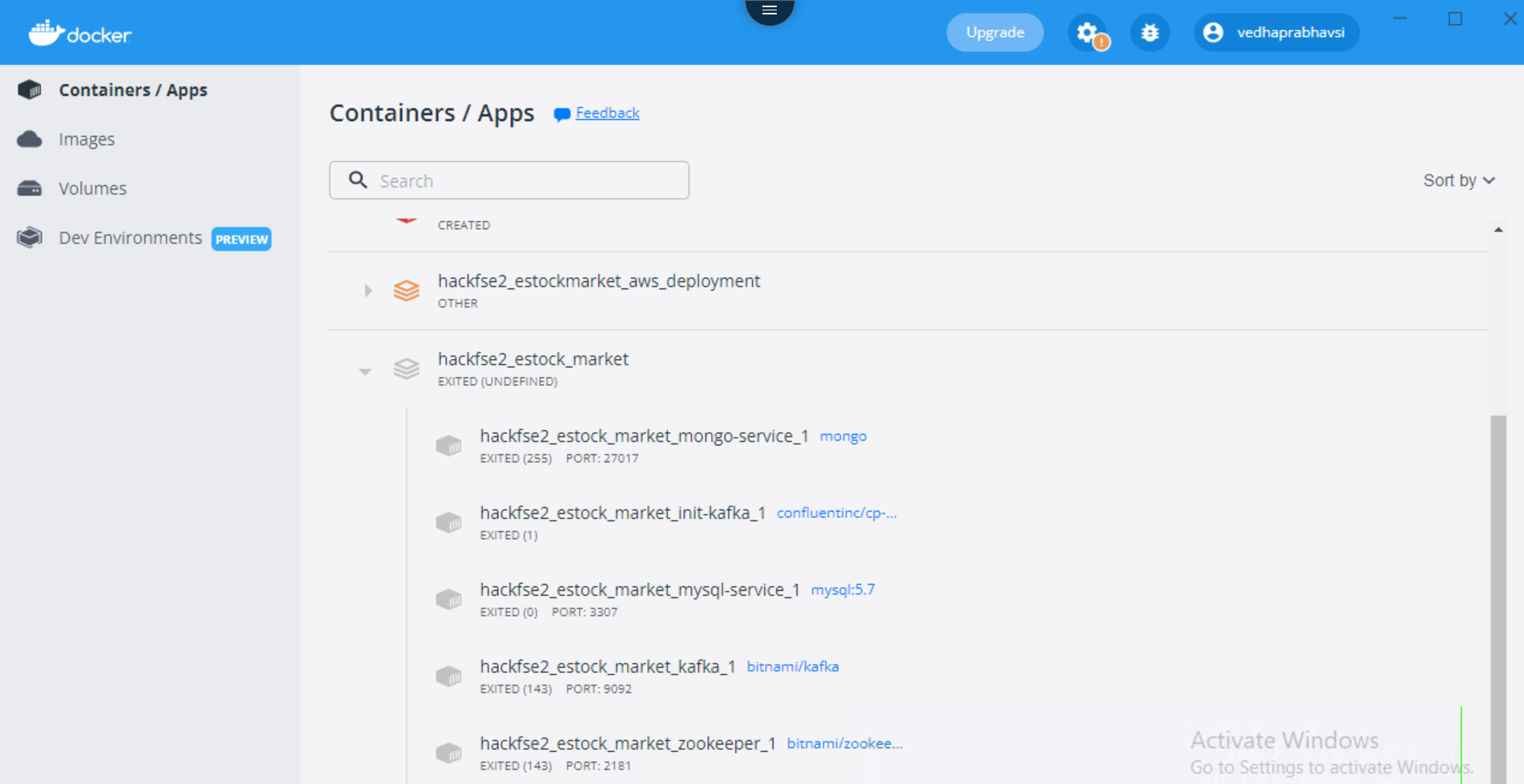
9. Eureka Server



10. Swagger Doc

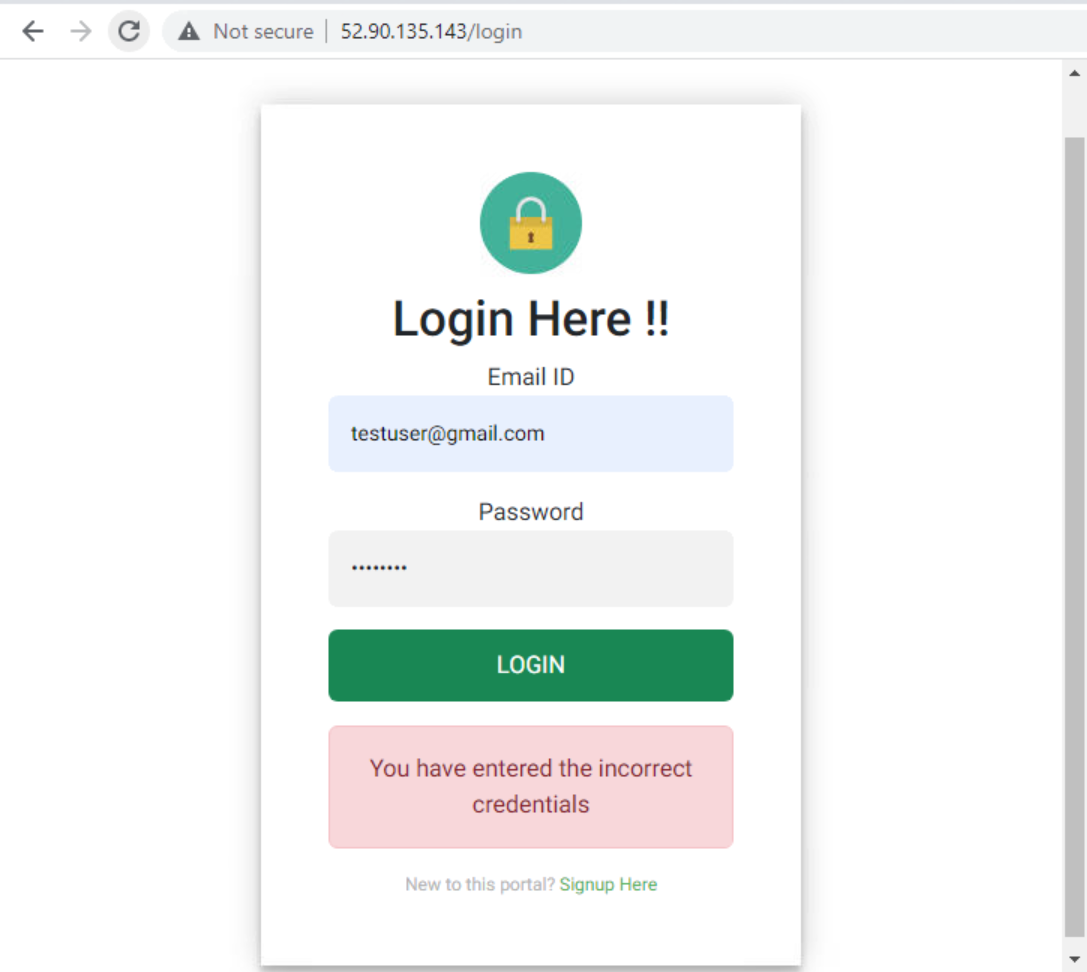


11. Docker Container (Docker-compose.yml)

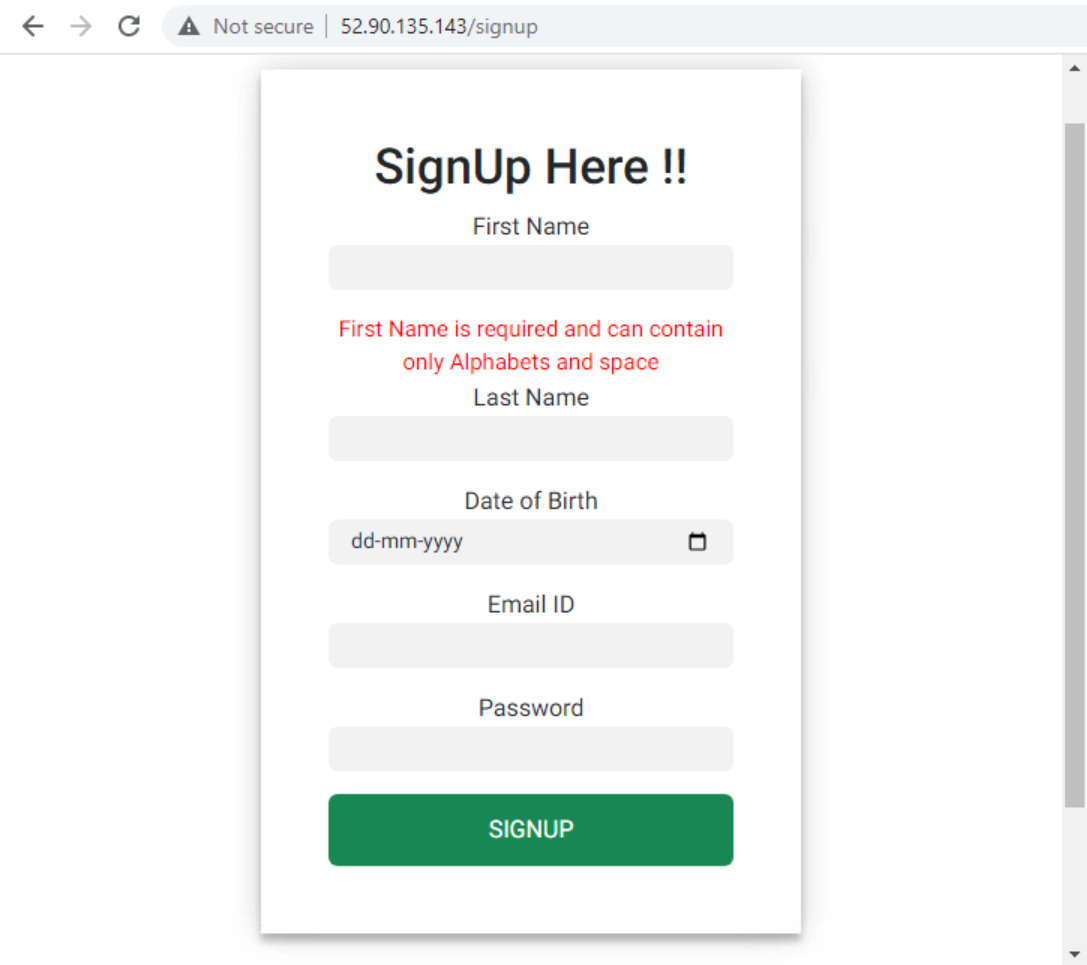


**Frontend implementation:**

1. User Login

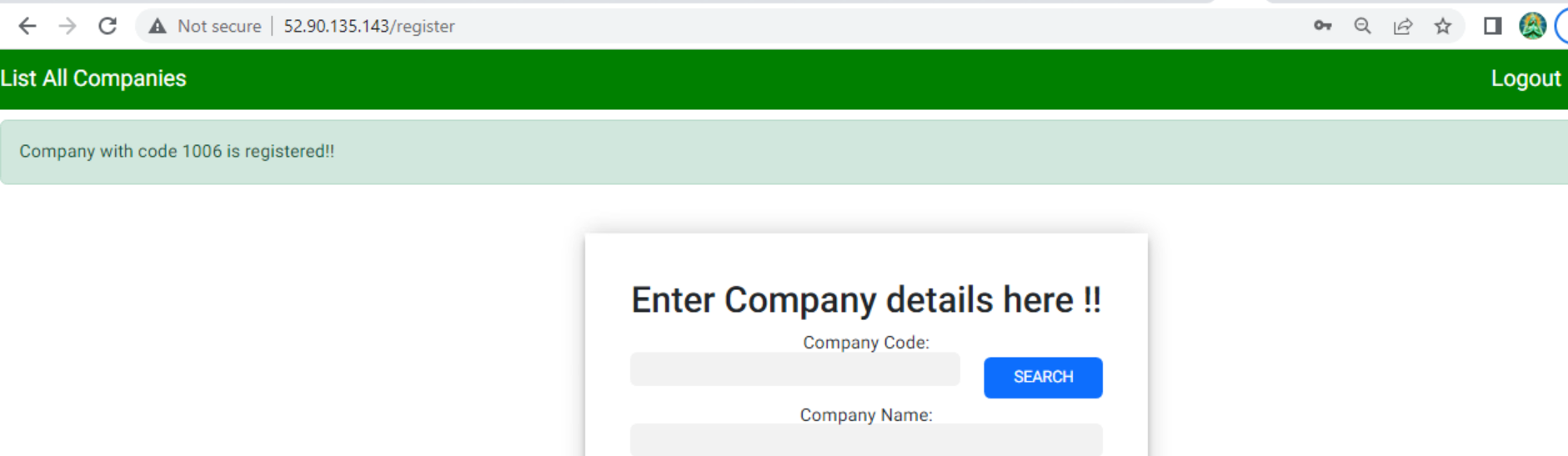


2. User signup



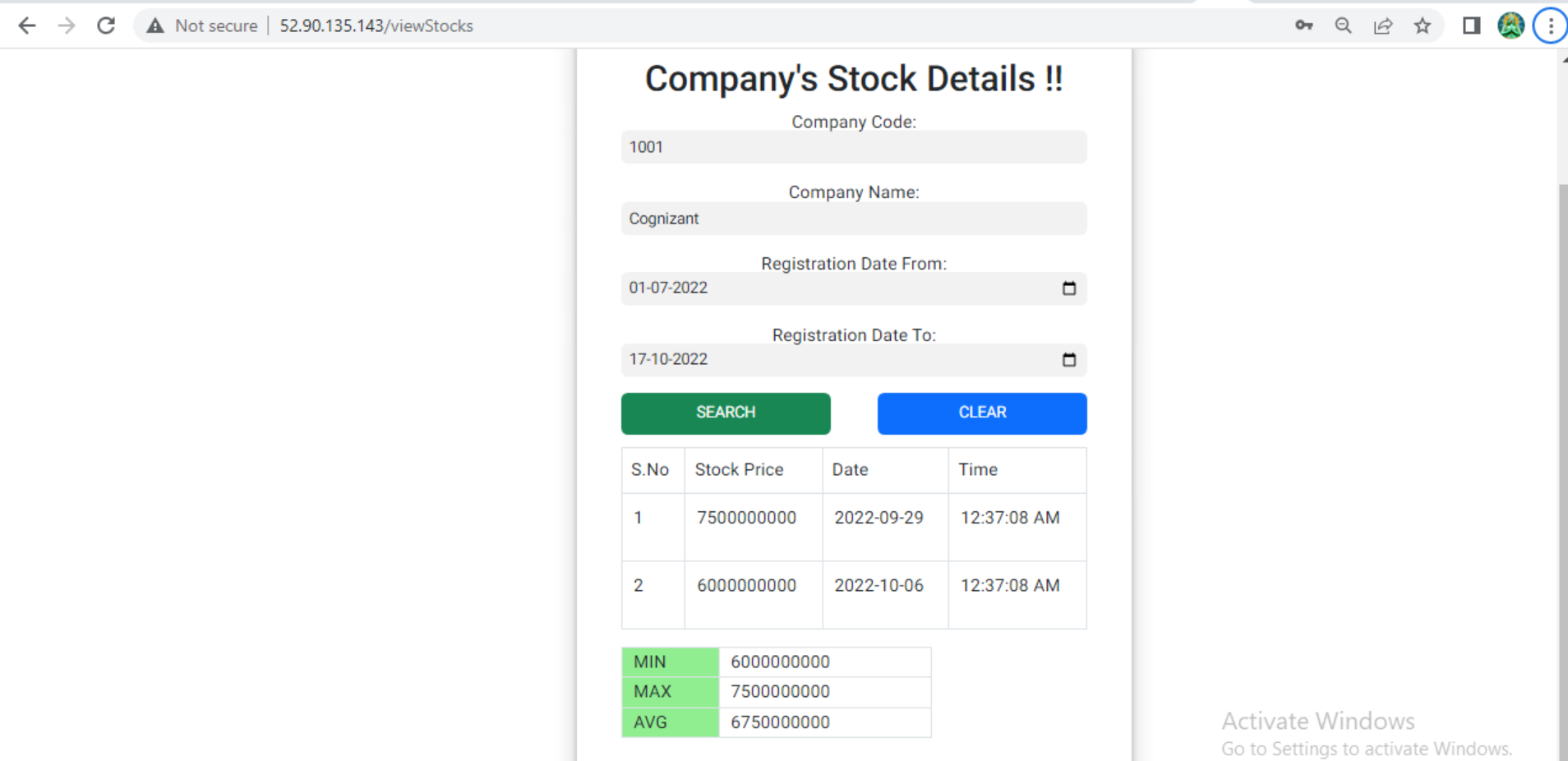
3. Company Registration





4. List all company stock details with aggregate value





**AWS Cloud Deployment:**

