Task 4

Được làm bởi TEAM B gổm:  
Nguyễn Cảnh Toàn-Nhóm Trưởng

Lâm Kim Long-Thành viên

Kim Ngân-Thành viên

Nguyễn Phúc-Thành viên

# Mục lục:

Table of Contents

[1. Mục lục: 1](#_Toc157156750)

[2. Bật, tắt thông báo cuộc trò chuyện: 1](#_Toc157156751)

[3. Bật, tắt trạng thái gửi tin nhắn: 1](#_Toc157156752)

[4. Ẩn cuộc trò chuyện bằng mã pin: 1](#_Toc157156753)

[5. Tham gia nhóm bằng link mời: 1](#_Toc157156754)

[6. Quản lí nhóm: 1](#_Toc157156755)

[7. Ghim cuộc trò chuyện: 1](#_Toc157156756)

[8. Tạo ghi chú: 1](#_Toc157156757)

[9. Tạo bình chọn: 1](#_Toc157156758)

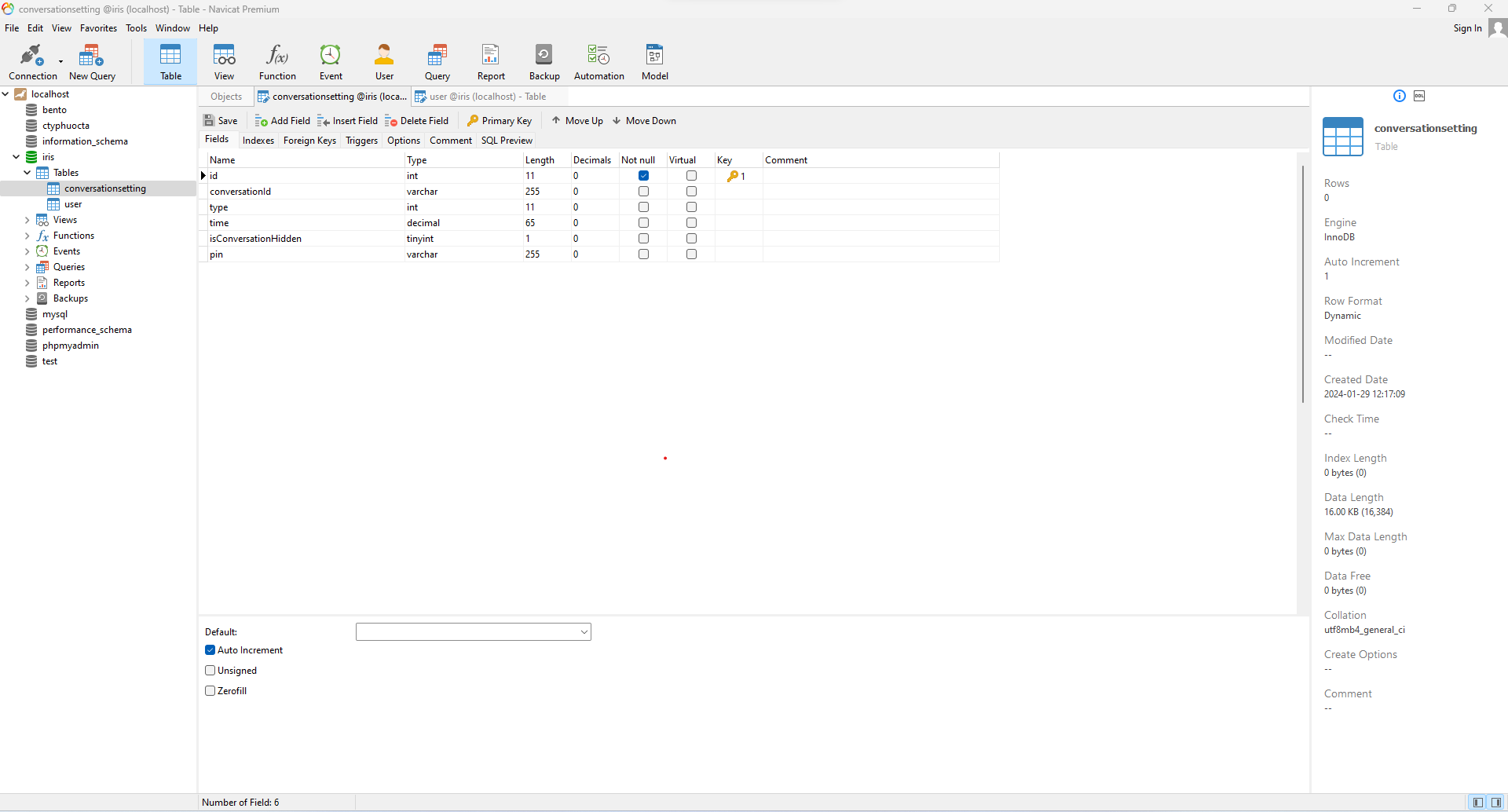
[10. Đăng nhập: 1](#_Toc157156759)

[11. Đăng kí: 1](#_Toc157156760)

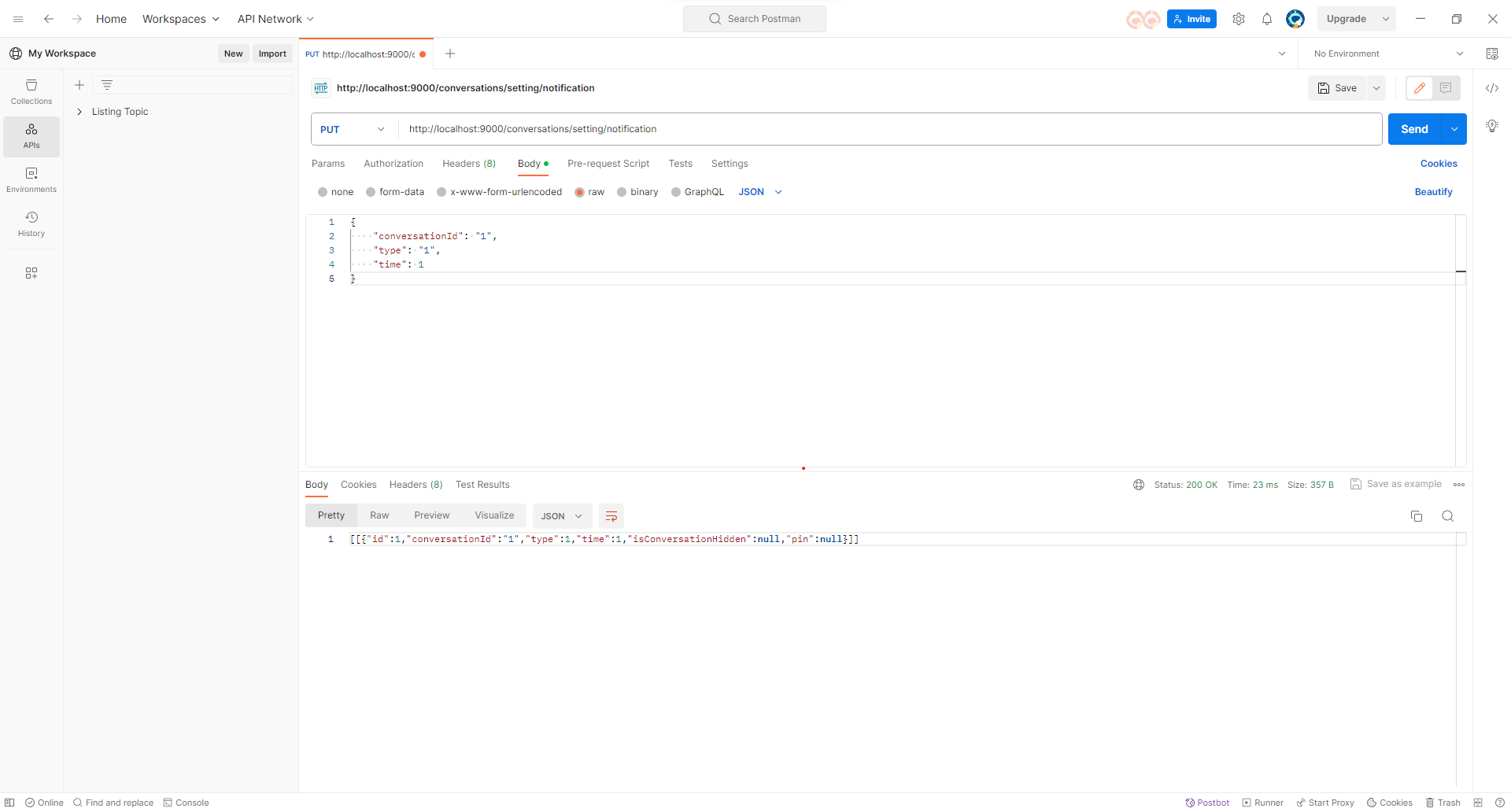
# Bật, tắt thông báo cuộc trò chuyện:

Trong bài task 4 này, em sử dụng nodejs làm backend và em sử dụng Visual Studio Code làm IDE cho bài của em, còn phần database thì em sử dụng Xamqq để làm control panel và Navicat để lưu dữ liệu.

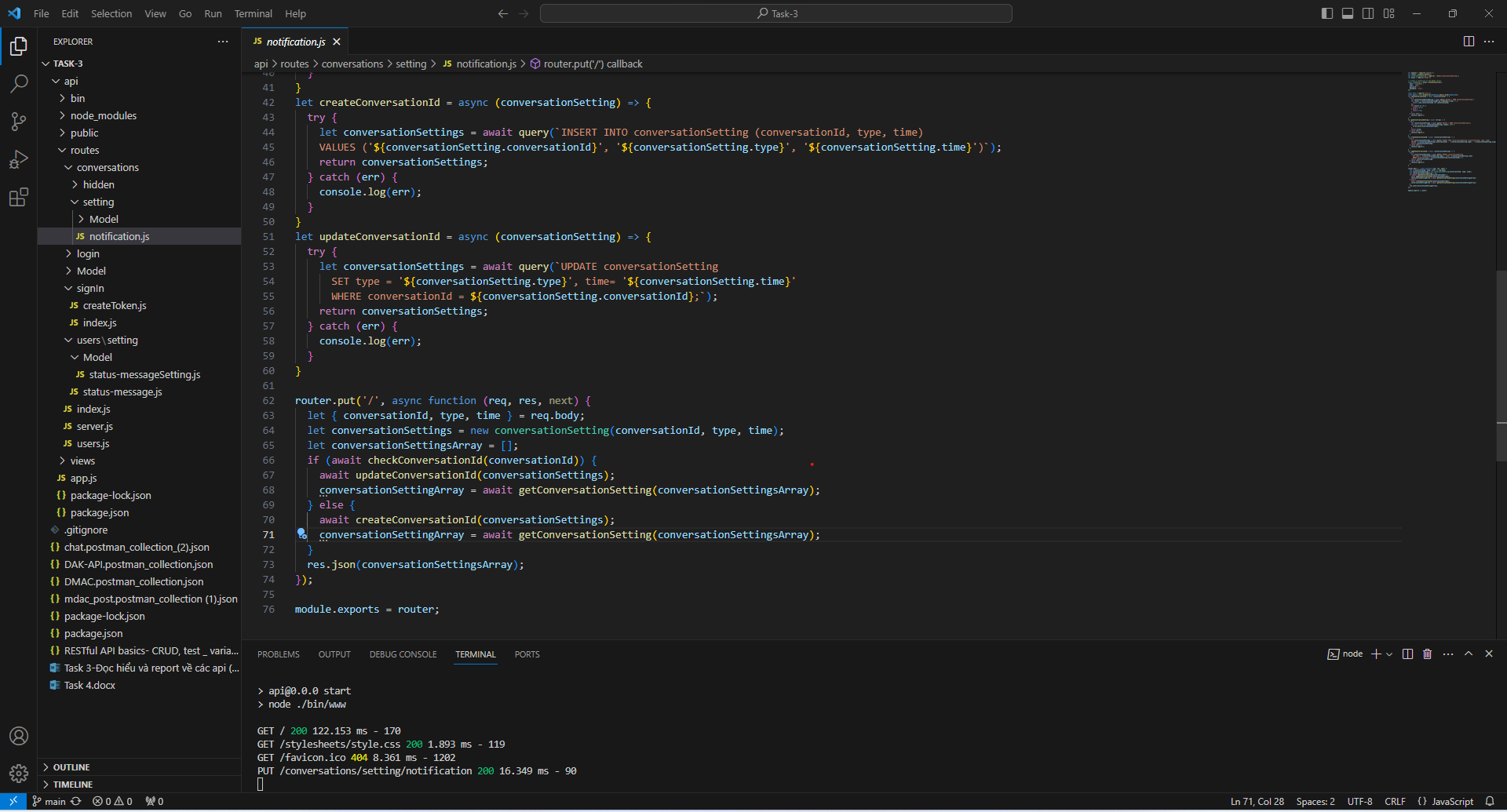
Về phần bật tắt thông báo thì đầu tiên cần phải tạo database có tên là conversationSetting và trong conversationSetting tạo trường id, conversationId, type, time, isConversationHidden, pin. Sau khi hoàn thành thì database sẽ hiển thị như sau:



Sau đó trong postman phần body bạn gửi raw body json object có 3 địa chỉ là conversationId, type, time. Sau khi thành công thì postman sẽ hiện ra như sau:



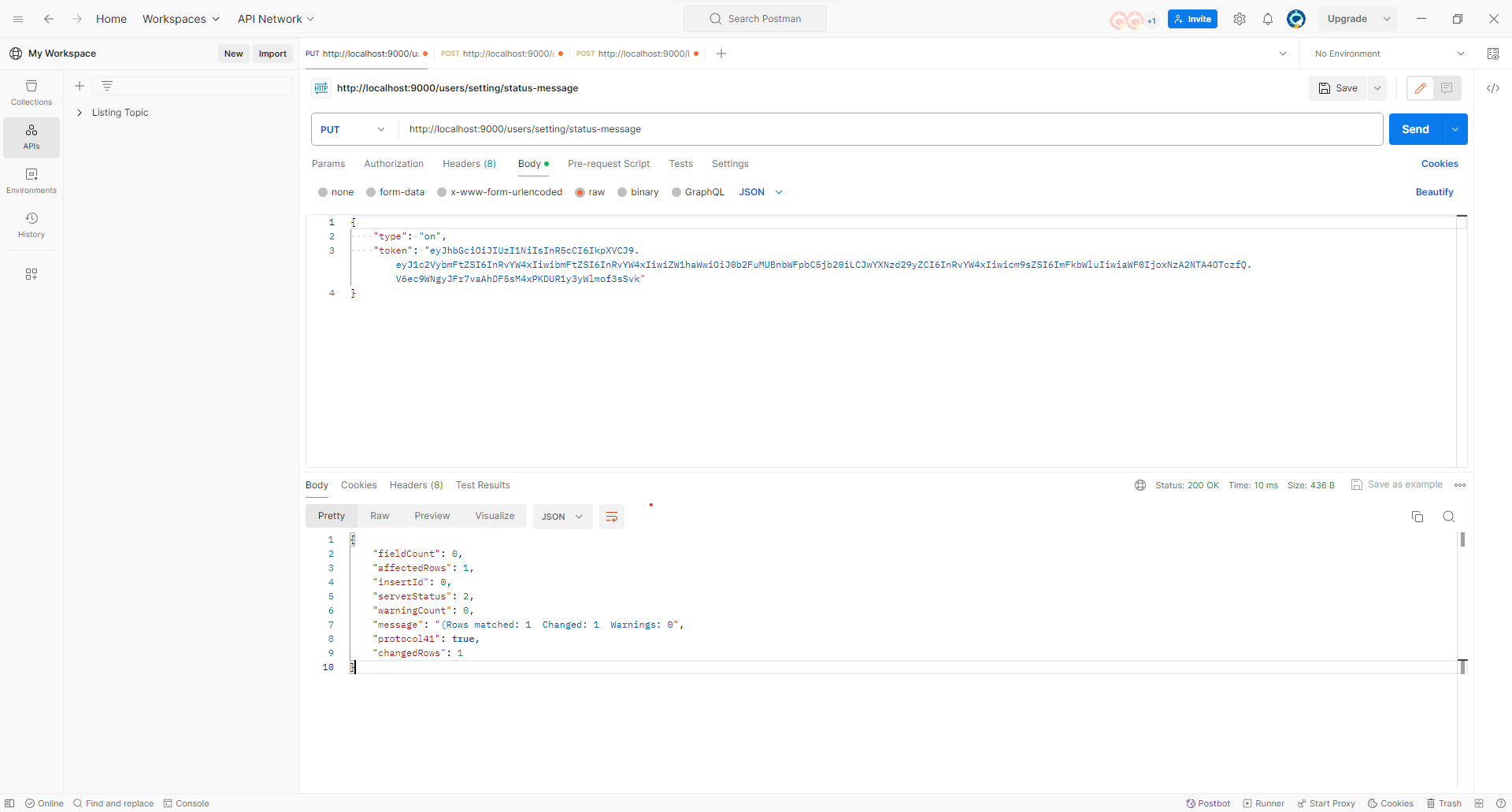
Trong phần code BE thì nodejs làm 3 việc là check thử coi có conversationId đó ko, nếu có thì update lại cái conversations có conversationId cũ còn nếu không thì tạo ra 1 biến mới trong conversationSetting. Đây là code của em:



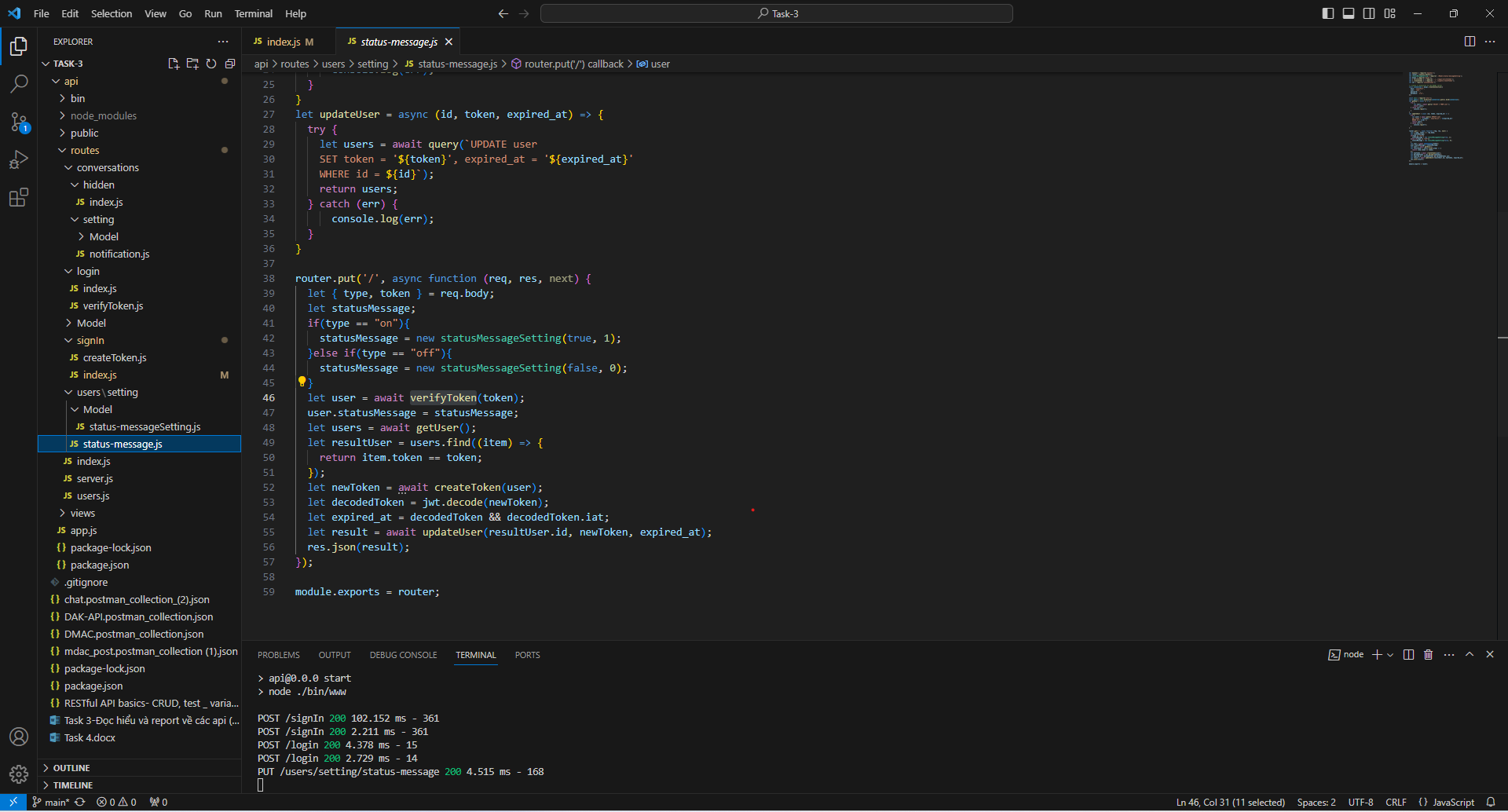
# Bật, tắt trạng thái gửi tin nhắn:

Đầu tiên để bật tắt trạng thái gửi tin nhắn thì người dùng cần phải nhập vào token của người dùng và type bật tắt.

Về phần token của người dùng thì bạn cần phải copy token khi vừa mới đăng kí còn type on để bật, off để tắt là được. Khi thành công thì postman sẽ hiện ra như sau:



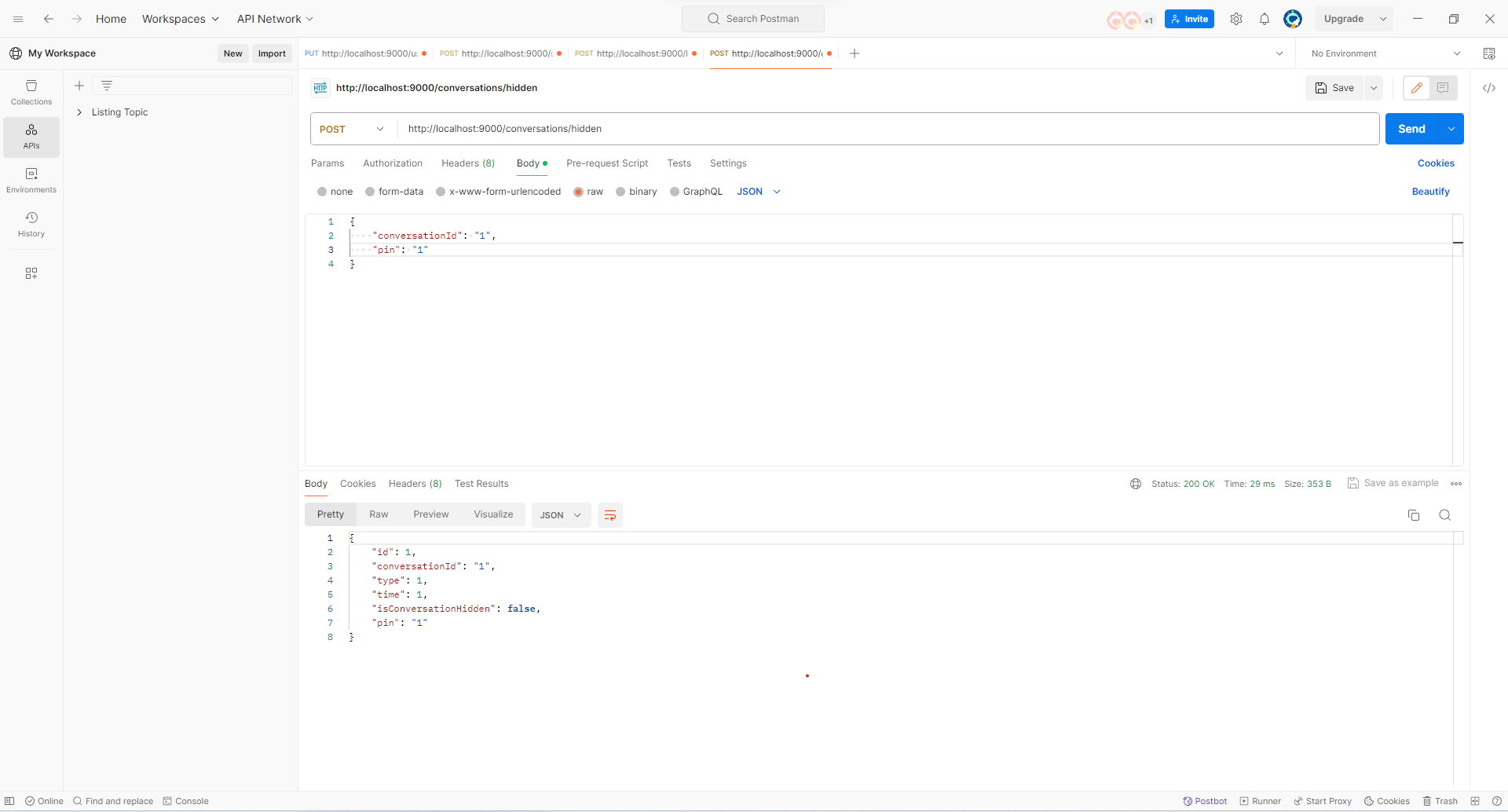
Về phần BE thì em sử dụng jwttoken của phần đăng nhập để lấy được thông tin người đăng nhập token. Sau đó em cho type và value vào thông tin người đăng nhập và lấy id của của cái token đó để cập nhật vào id của cái token đó 1 cái token mới tạo và ngày hết hạn mới. Cụ thể thì code post của em như sau:



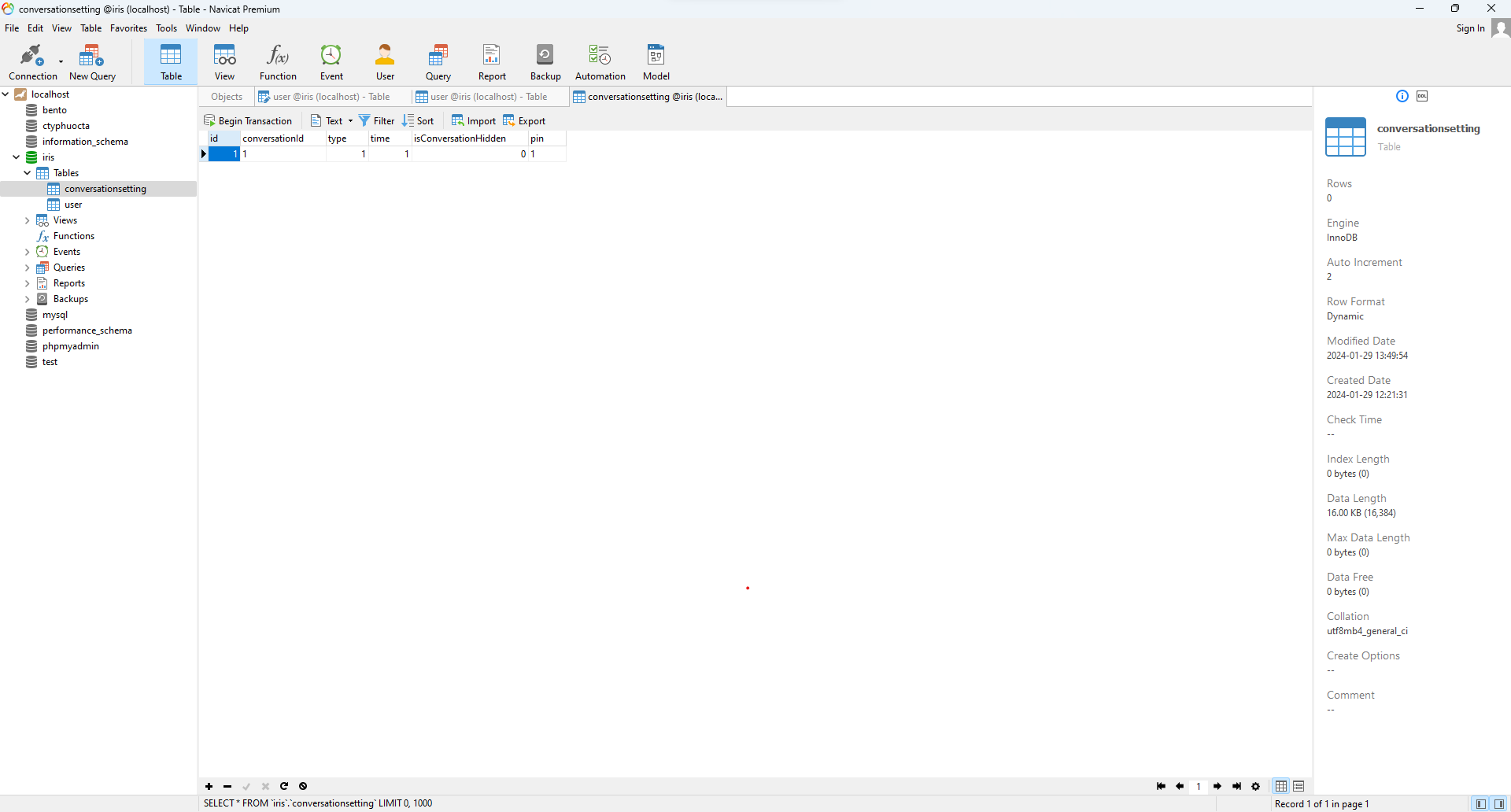
# Ẩn cuộc trò chuyện bằng mã pin:

Đầu tiên để ẩn cuộc trò chuyện bằng mã pin thì người dùng cần phải nhập vào postman conversationId, pin.

Về phần postman thì cần phải nhập conversationId, pin vào body => raw và nhập vào json object như ở dưới. Sau khi thành công thì postman sẽ hiện ra như sau:

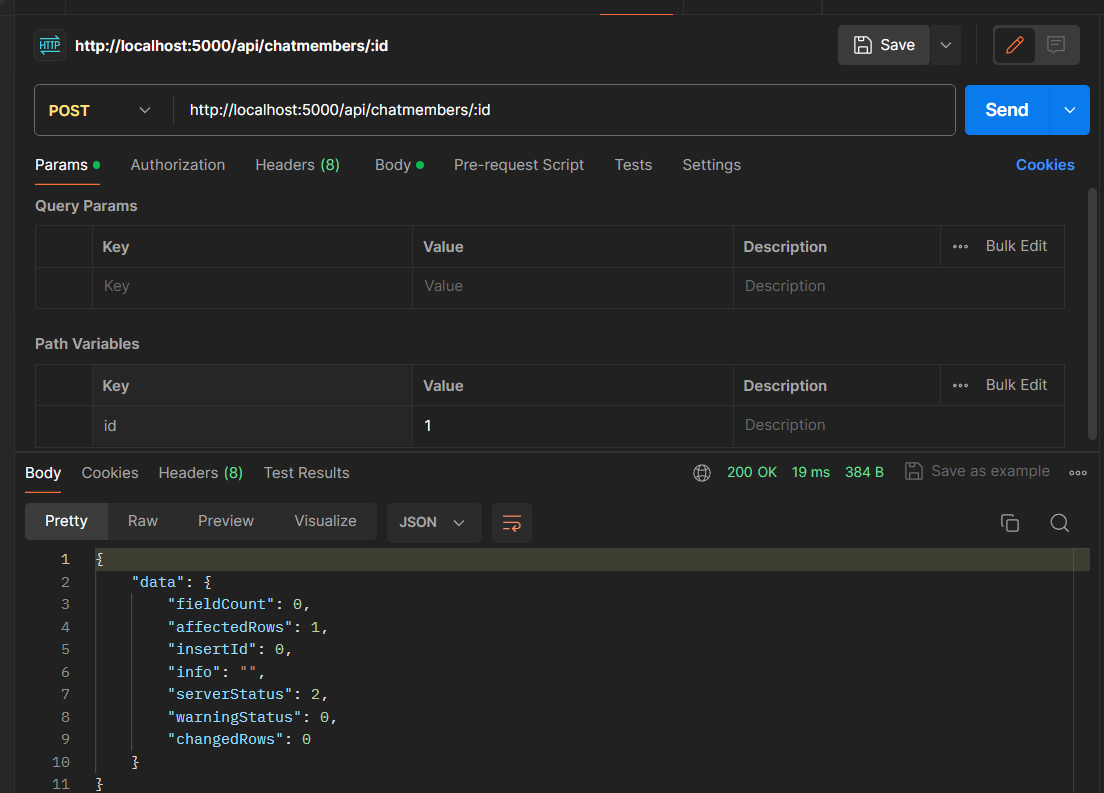


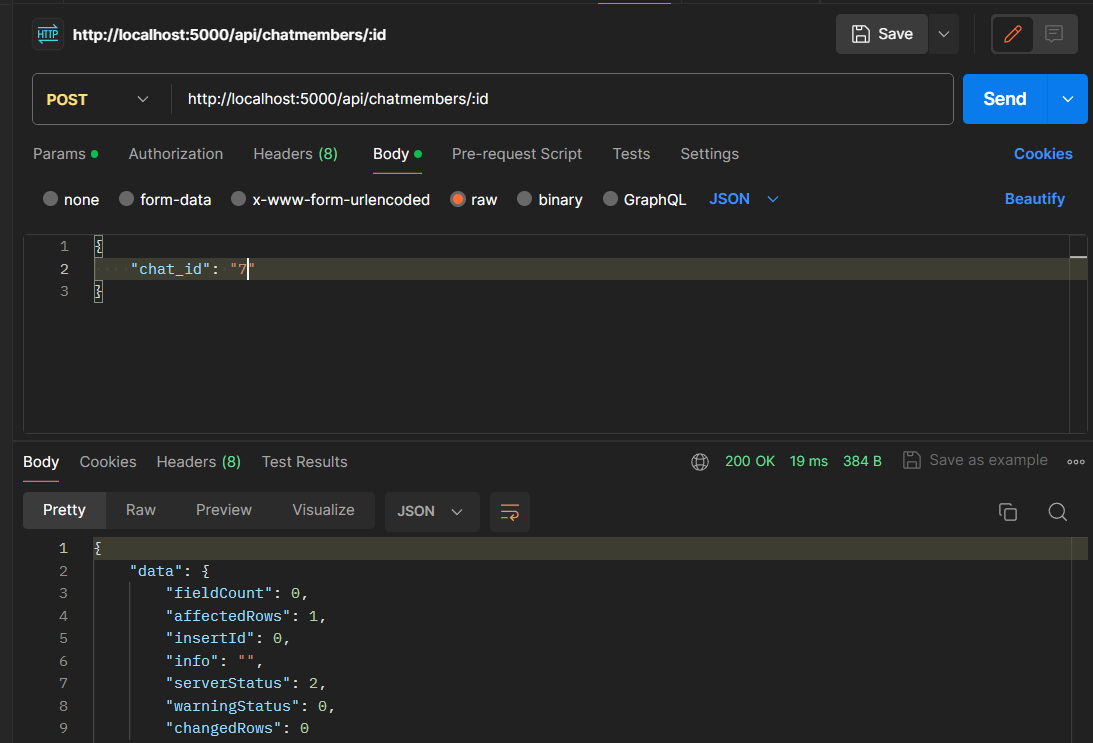
Về phần database thì khi nhận được giá trị thì trường pin sẽ chuyển thành pin mới và isConversationHidden sẽ chuyển từ null sang true hoặc false. Khi thành công thì navicat sẽ hiện ra như sau:



# Tham gia nhóm bằng link mời:

Tham gia bằng link mời (id là user\_id)





# Quản lí nhóm:

…

# Ghim cuộc trò chuyện:

…

# Tạo ghi chú:

…

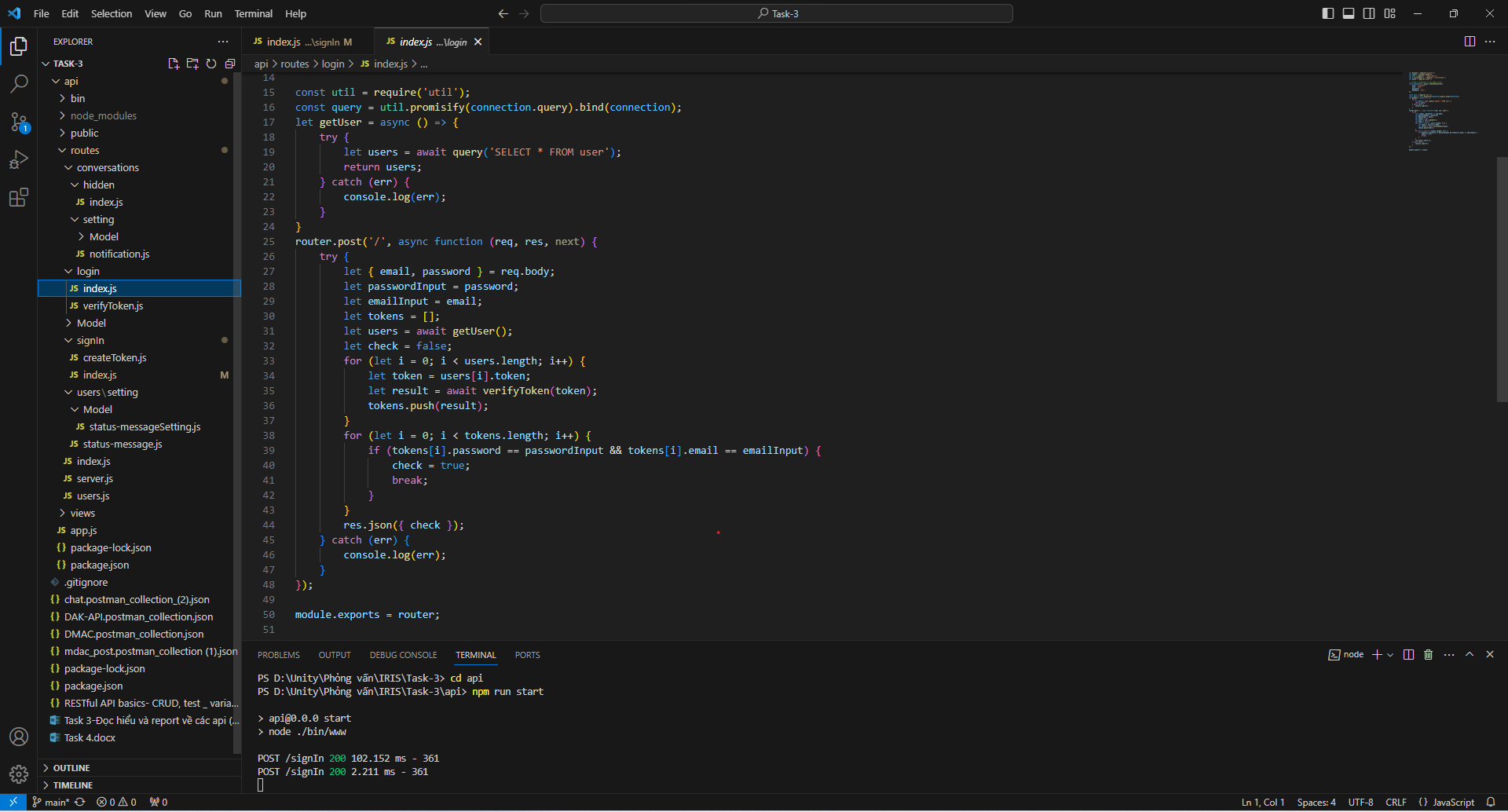
# Tạo bình chọn:

…

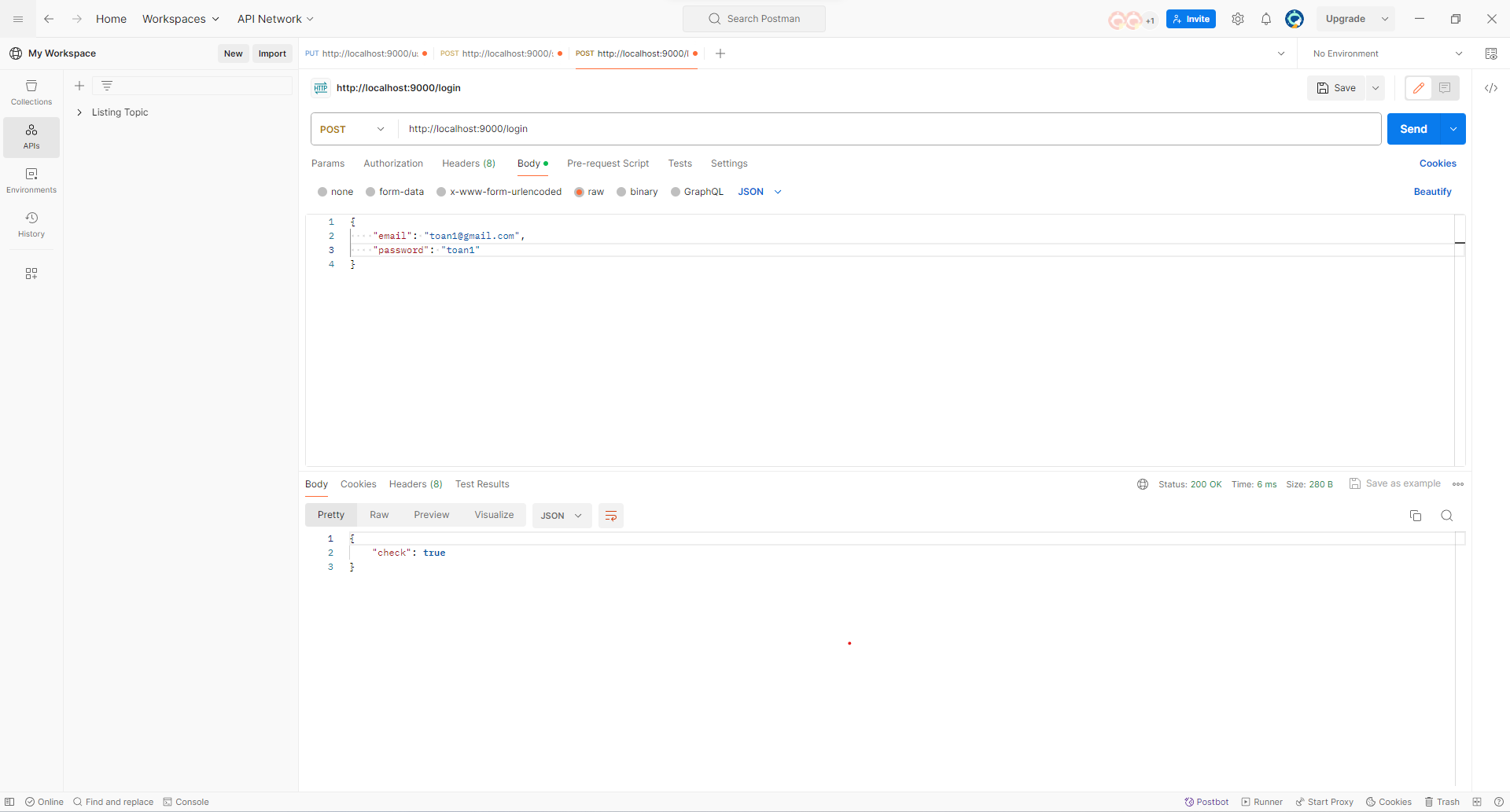
# Đăng nhập:

Đầu tiên để đăng nhập được thì người dùng cần phải đăng kí trên postman trước và sau đó lấy email và password khi đăng nhập để nhập vào postman.

Về phần đăng nhập thì trong code backend, em chỉ cần lấy tất cả user và giải mã tất cả user và cho từng user vào 1 cái mảng rổng để kiểm tra xem có cái nào trùng email với password với cái vừa đăng nhập ko nếu có thì postman sẽ xuất ra là yes. Code em sẽ như sau:

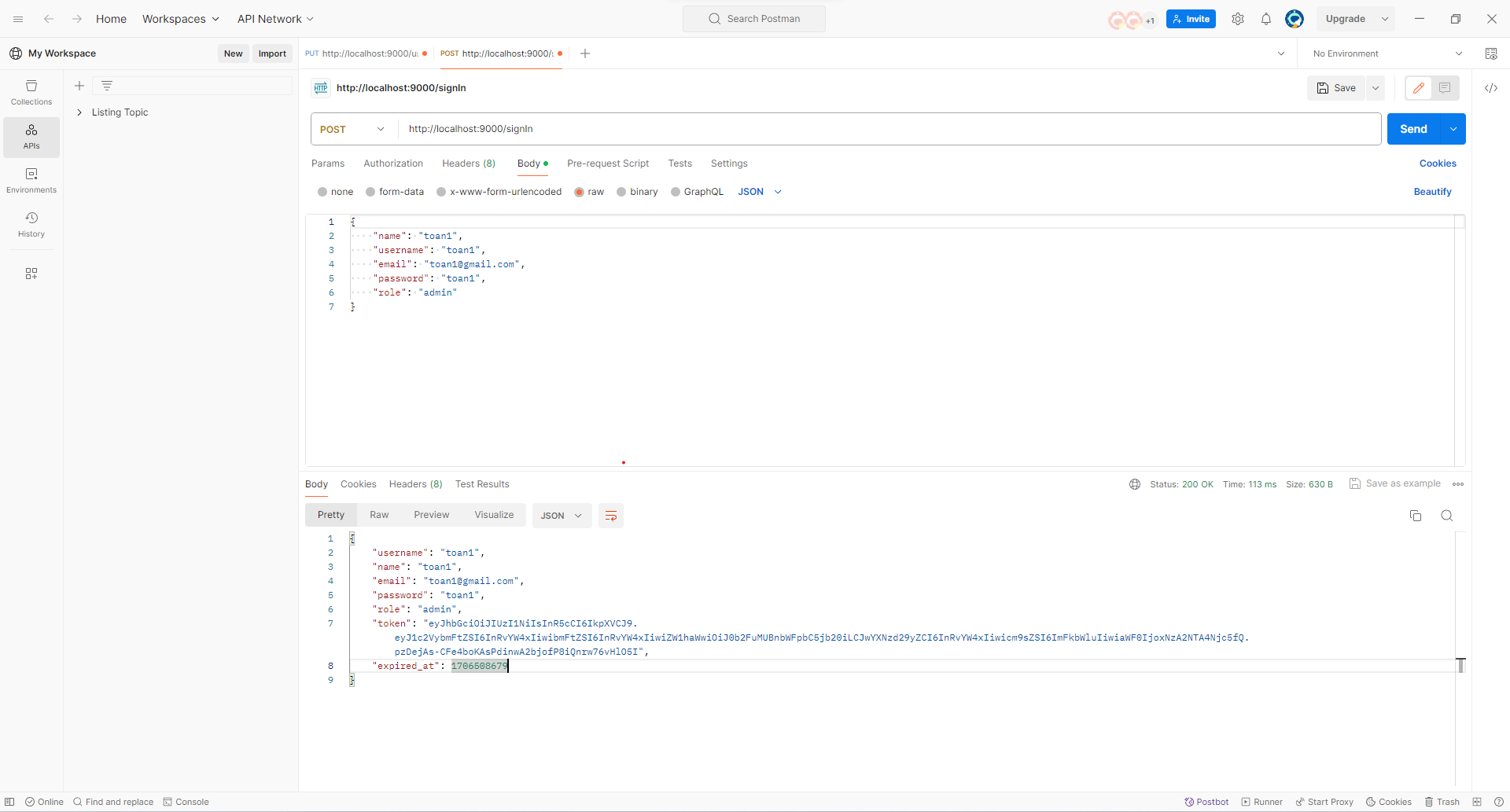


Trong postman thì em cần phải nhập email và password vào phần body và khi thành công postman sẽ hiện ra như sau:

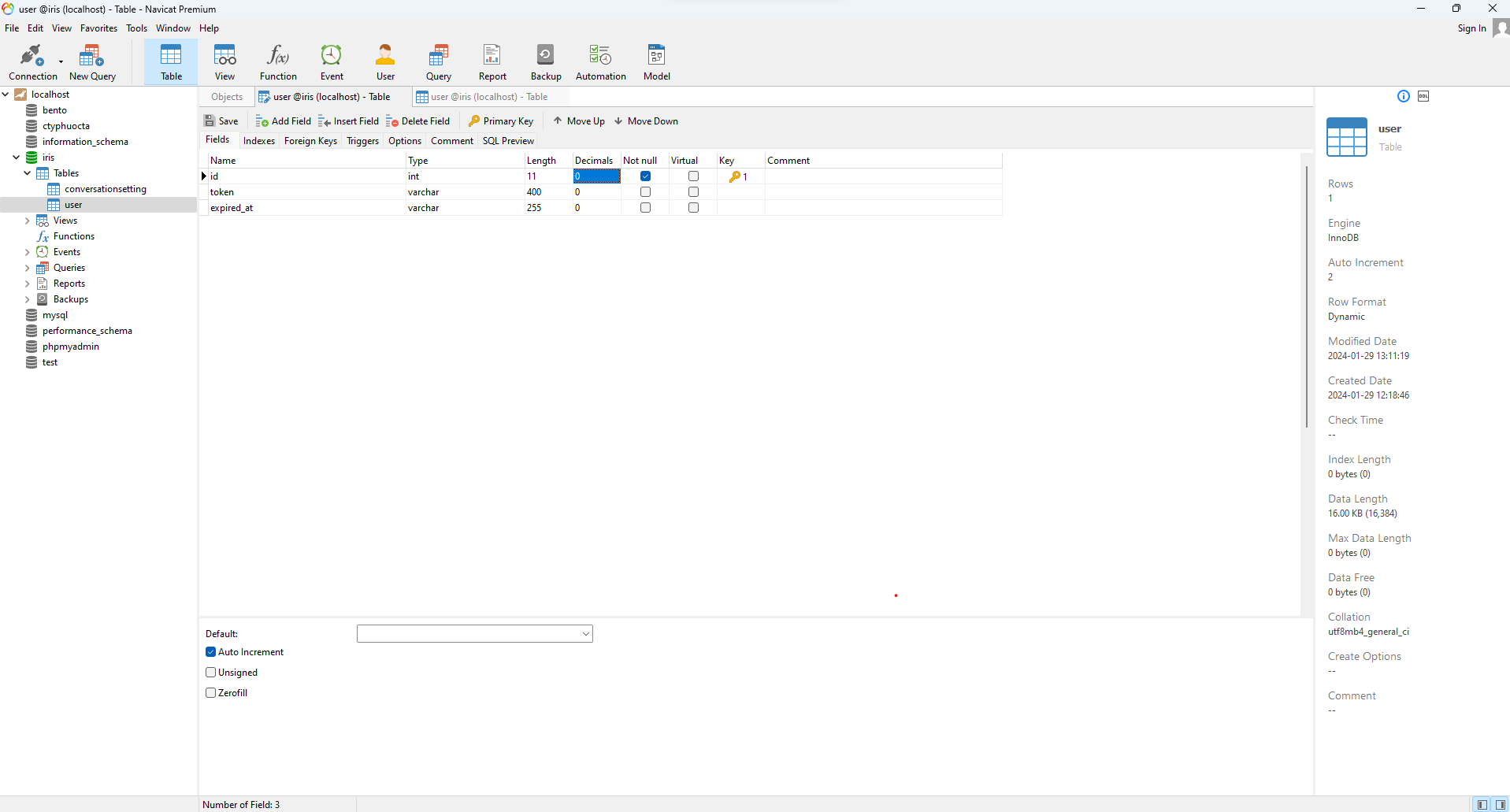


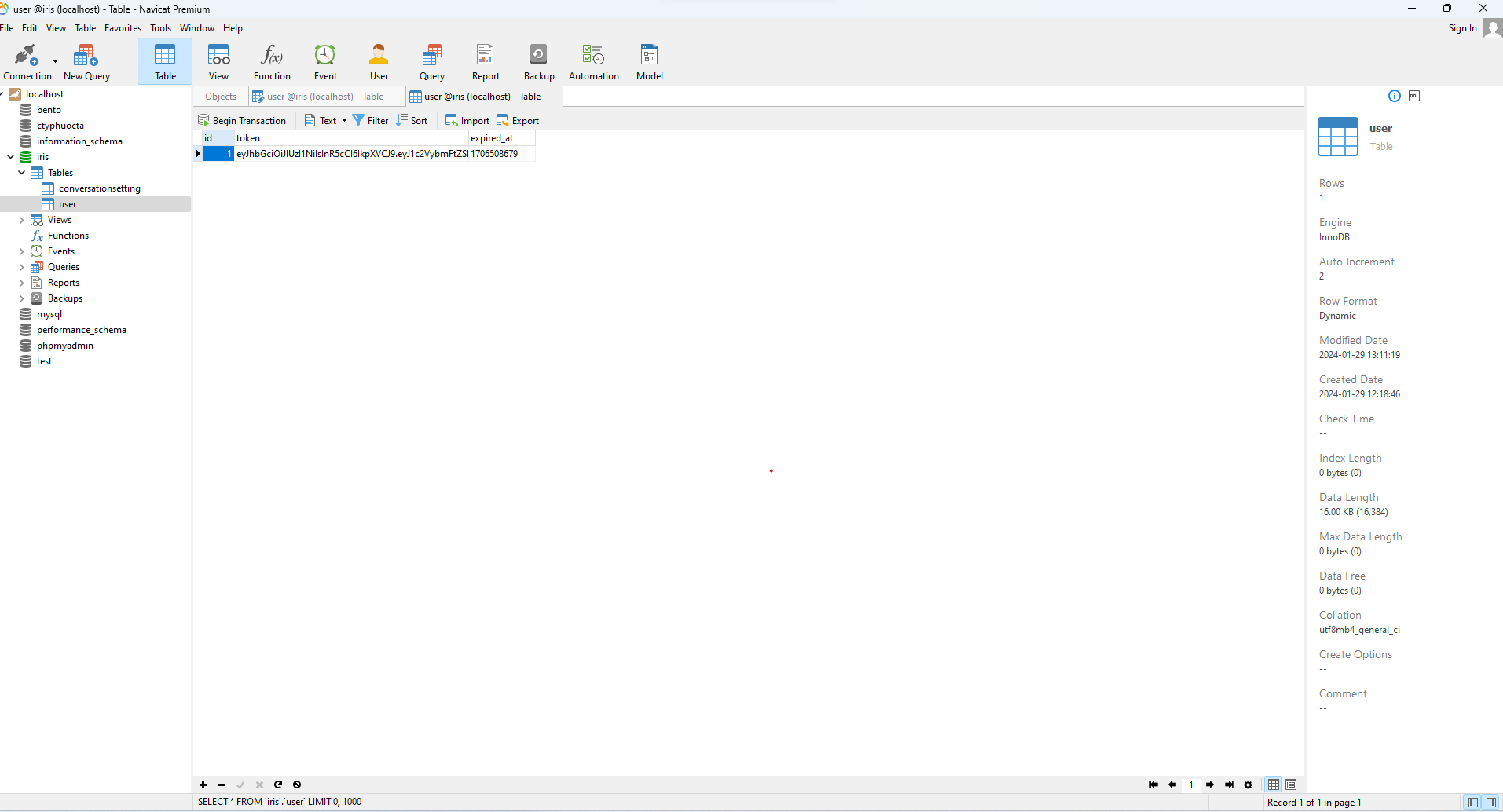
# Đăng kí:

Đầu tiên để đăng kí được thì trong postman em bắt user nhập name, username, email, password, role vào địa chỉ json object và sử dụng method post và sau khi thành công thì postman sẽ hiện ra như hình sau:



Trong phần database navicat thì em tạo ra database mới là user có 3 trường là id, token, expired\_at. Token sẽ lưu hết tất cả thông tin về user khi postman gửi về và sau khi postman gửi về thành công thì database sẽ hiện ra như sau:





Trong phần backend, em sử dụng jwtoken với payload là dữ liệu postman nhập vô, secret là Canhtoan111, em sử dụng mã HS256 và hết hạn sau 1h để tạo token. Sau đó em lấy cái token đó giải mã ra để lấy expired\_at để lưu vô database. Code em như hình ở dưới đây:

