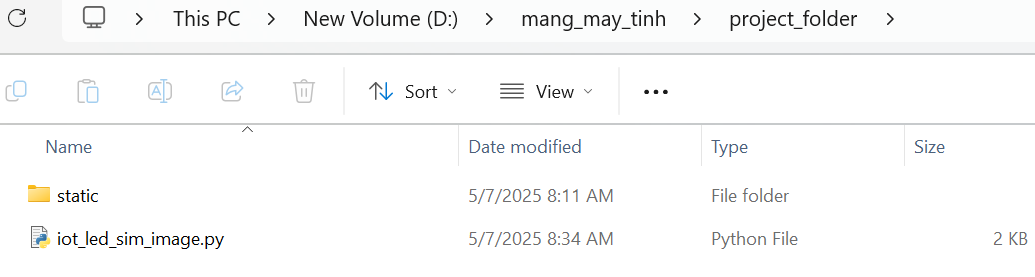
**Lê Thị Lan – 22174600093 – DHKL16A1HN**

**BÁO CÁO**

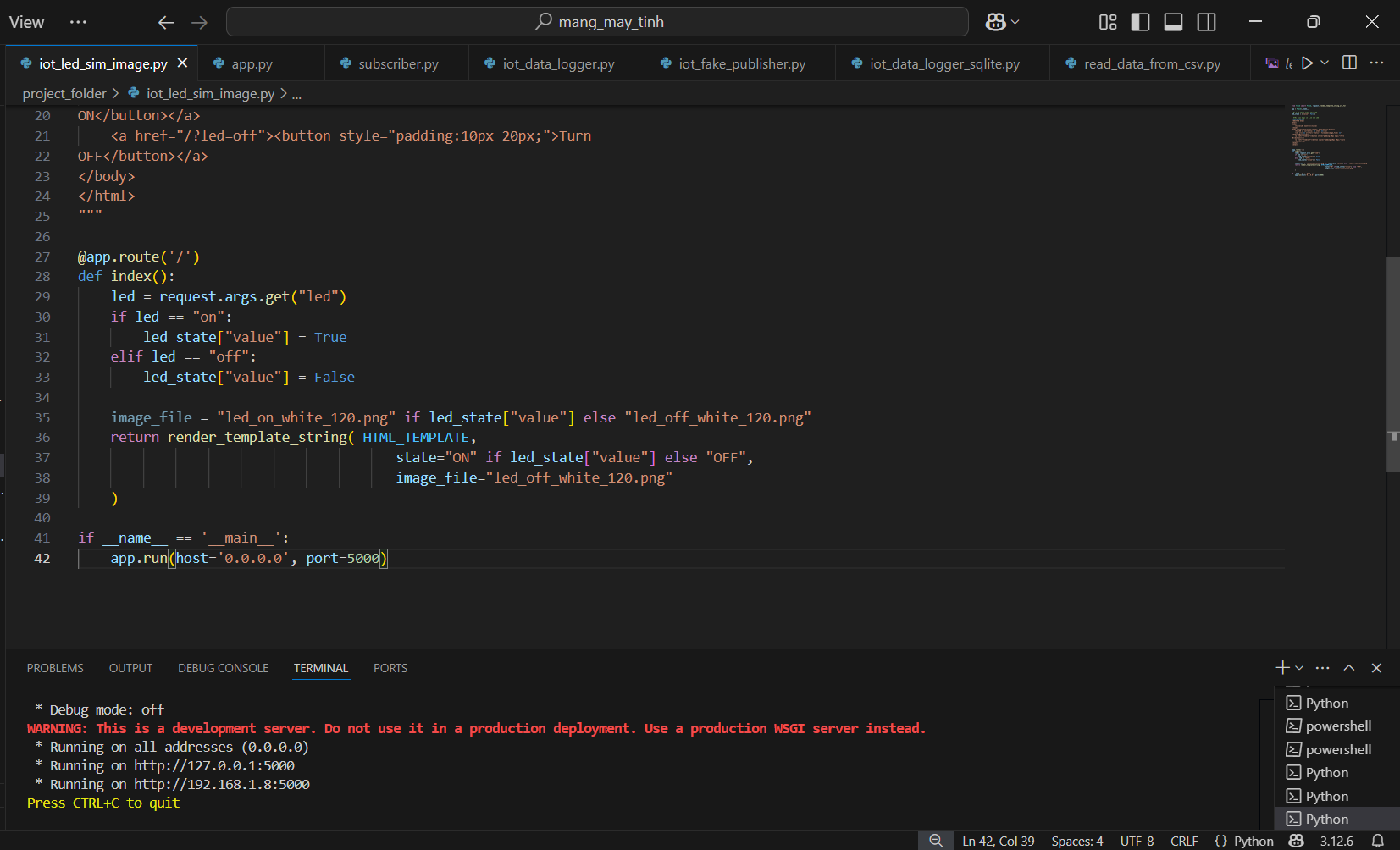
**MẠNG MÁY TÍNH VÀ TRUYỀN SỐ LIỆU**

**BÀI LAB 1.2: MÔ PHỎNG ĐIỀU KHIỂN LED QUA HTTP BẰNG FRAMEWORK FLASK**

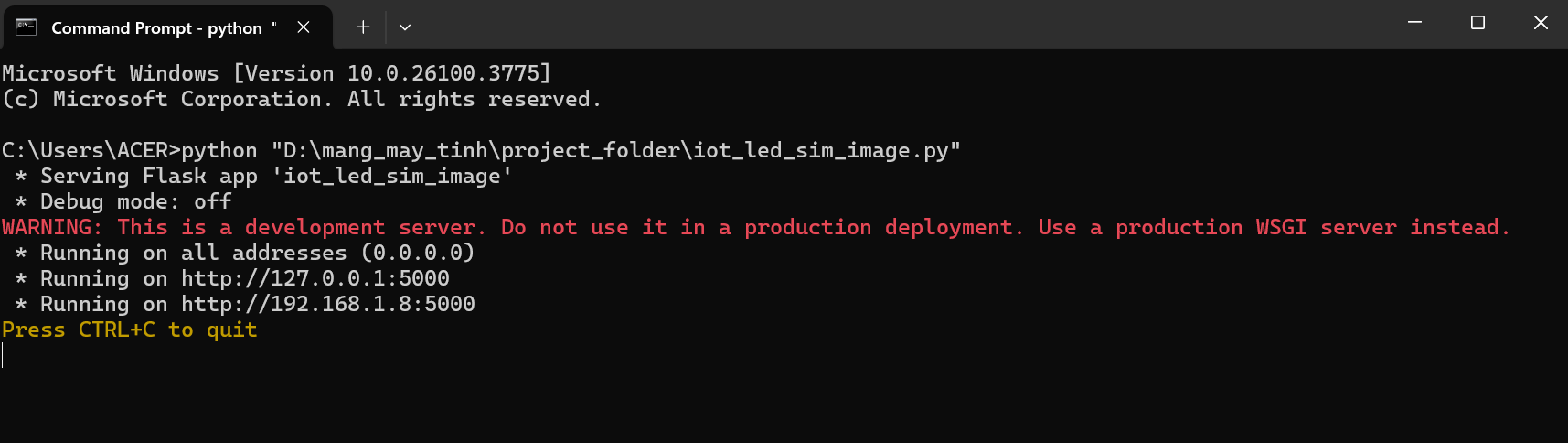
Sau khi tạo thư mục có cấu trúc sau:

****

Sau khi chạy file code iot\_led\_sim\_image ta có kết quả sau:

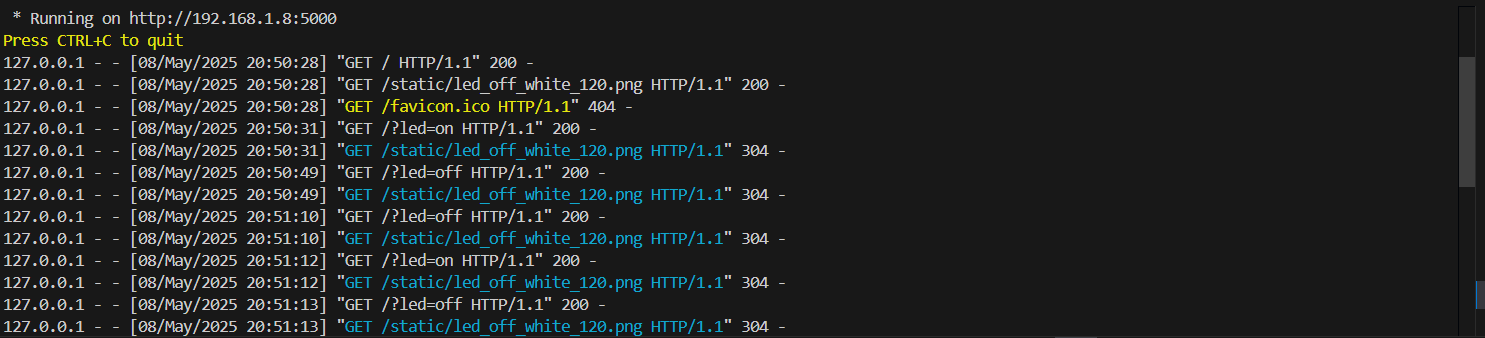
****

Kết quả chạy trên cửa số CMD:

****

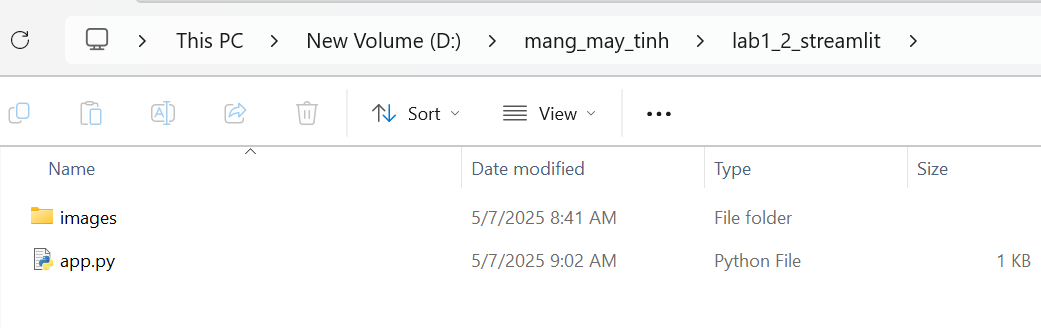
Quan sát giao diện web hiển thị trạng thái LED và các nút điều khiển.

|  |  |
| --- | --- |
|  |  |

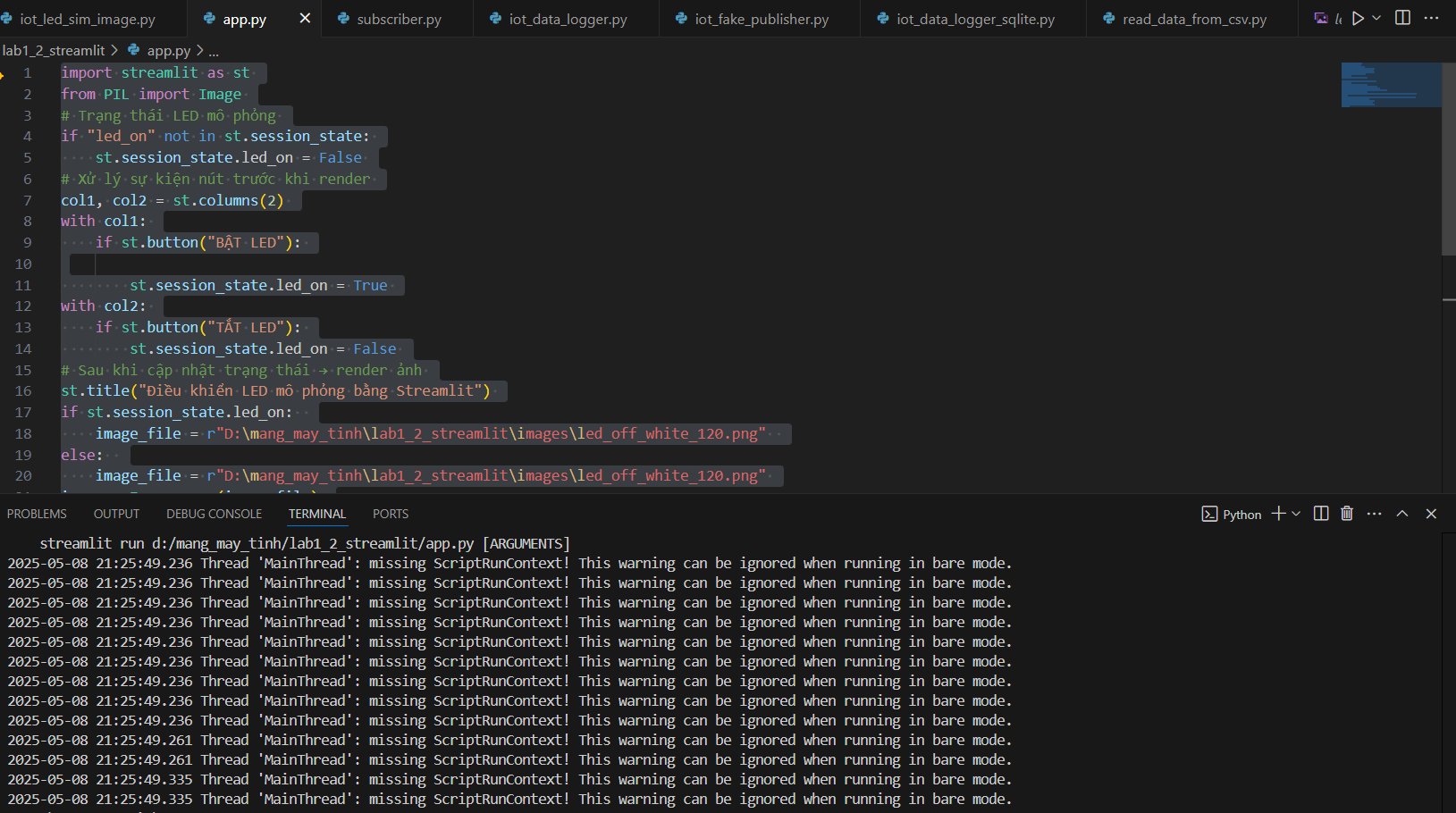
****

**BÀI LAB 1.2 PHIÊN BẢN MÔ PHỎNG ĐIỀU KHIỂN LED QUA HTTP BẰNG STREAMLIT**

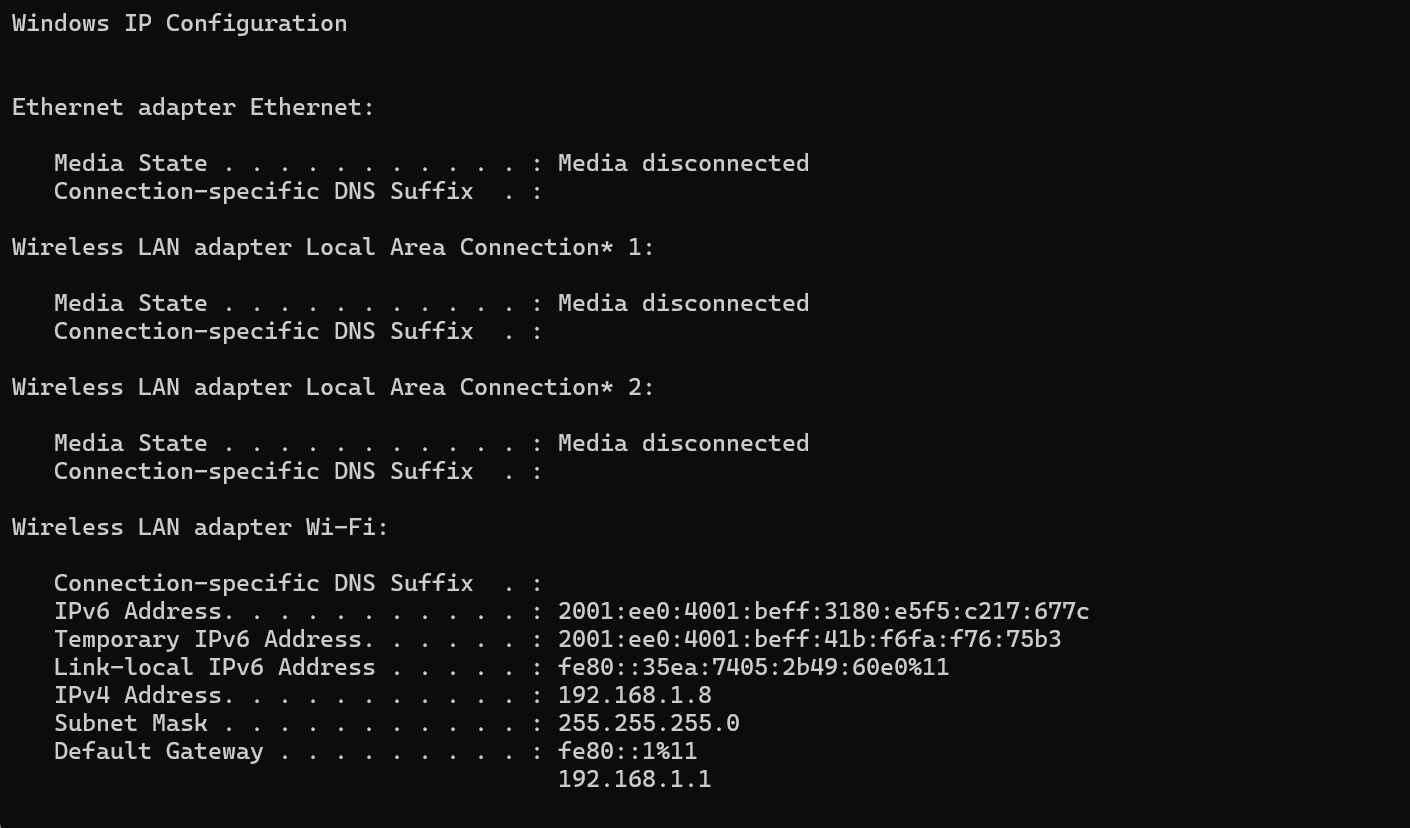
Dưới đây là cấu trúc thư mục được tạo để tiến hành thực hiện bài tập thực hành:

****

Sau khi chạy đoạn mã app.py ta có kết quả sau:

****

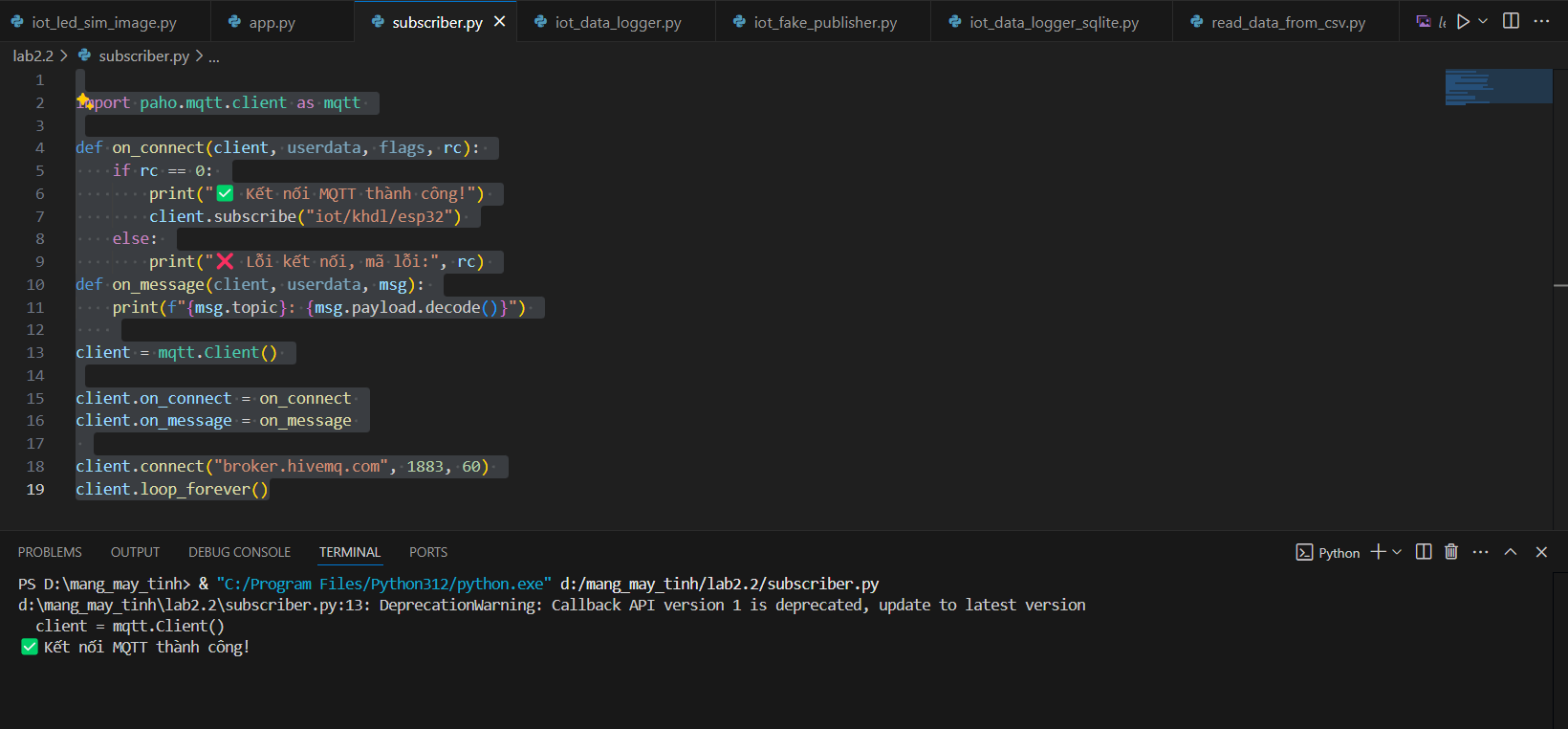
Sau khi chạy trên cửa sổ CMD bằng lệnh ipconfig ta có kết quả sau:

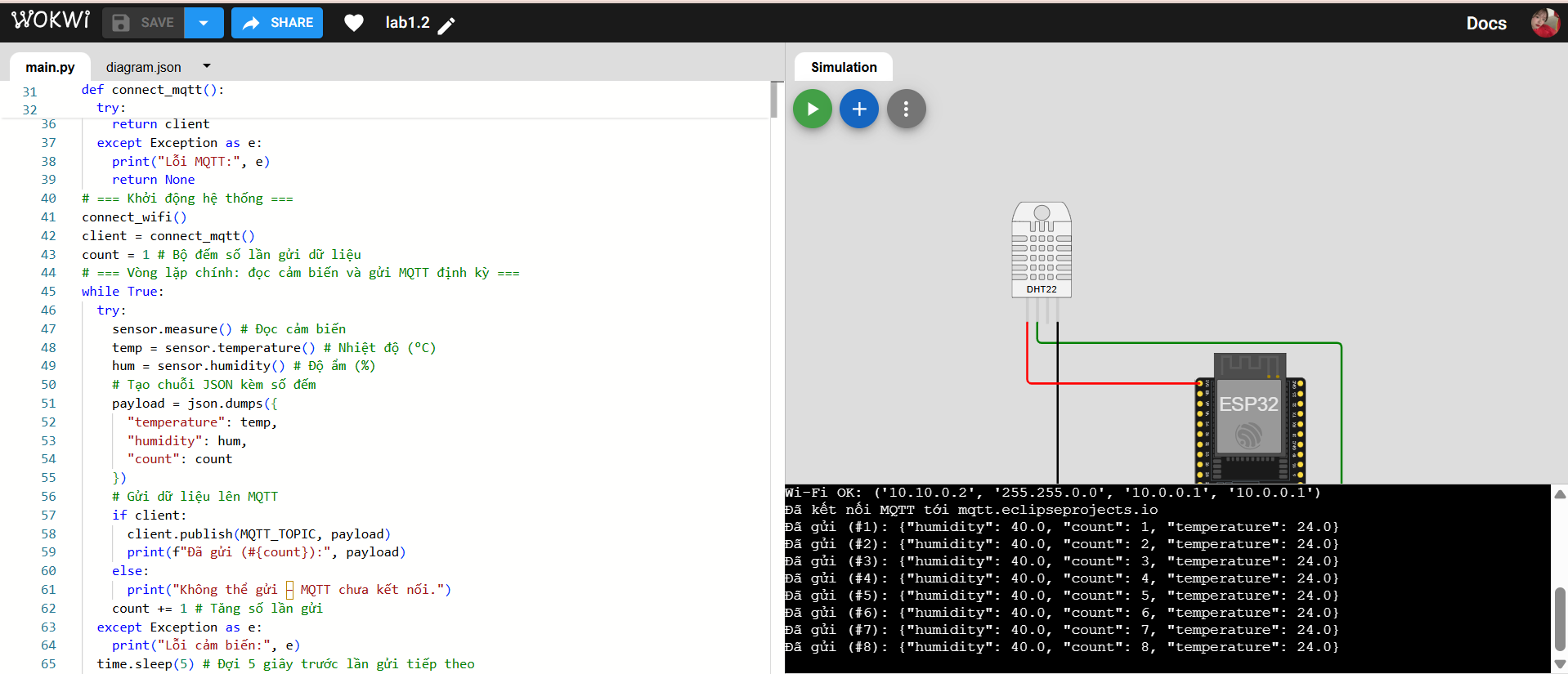
****

Ta có địa chỉ Ipv4 là: 192.168.1.8

**LAB 2.2a – GỬI DỮ LIỆU CẢM BIẾN LÊN MQTT BROKER**

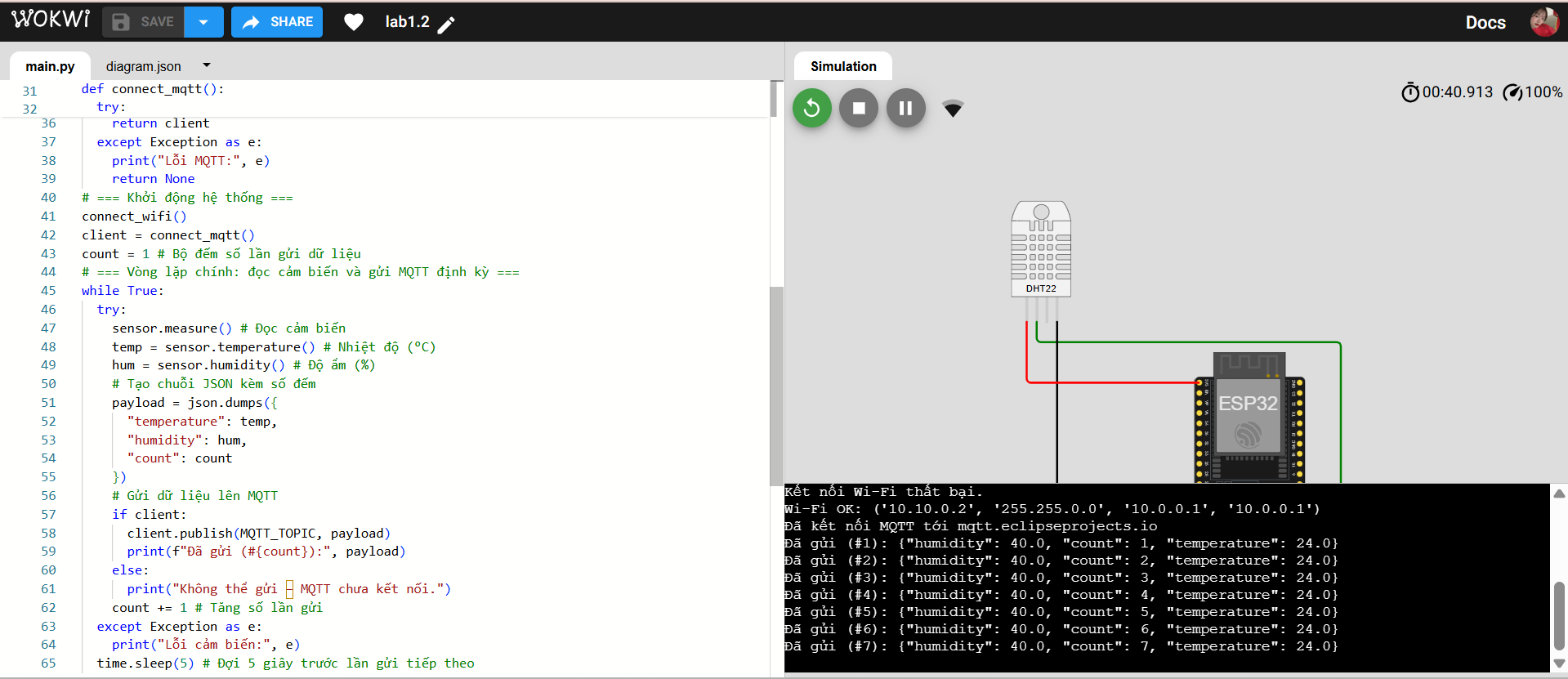
Dưới đây làESP32 mô phỏng (trên Wokwi) đọc dữ liệu từ cảm biến DHT22, kết nối Wifi và gửi kết quả dữ liệu như hình sau:



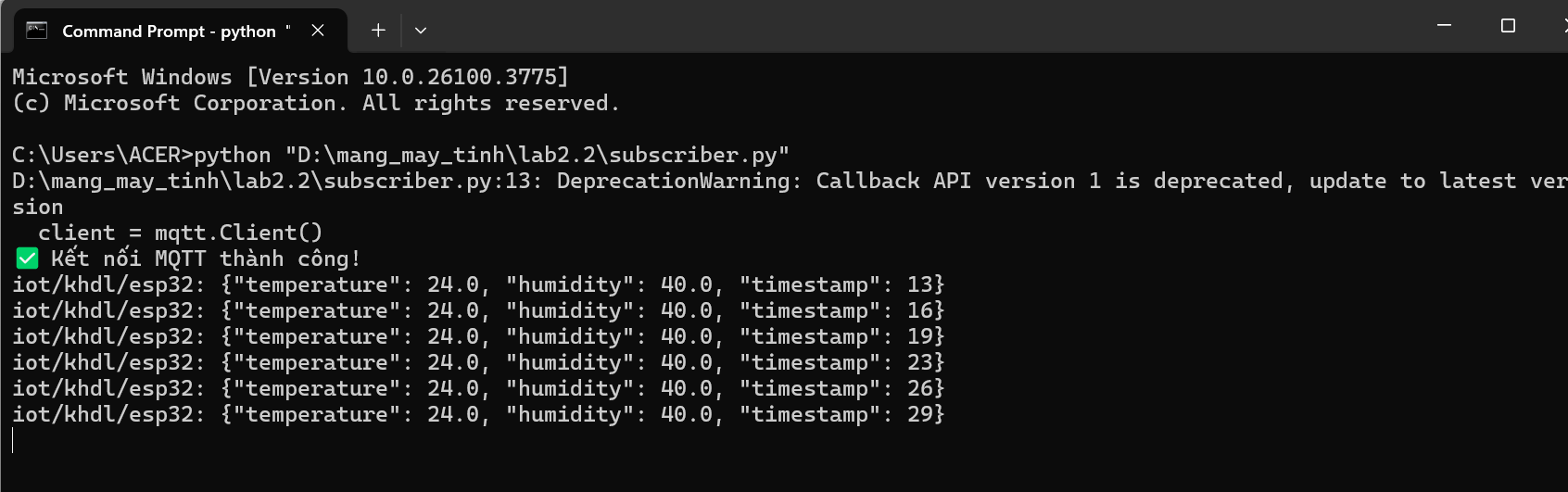
****

**LAB 2.2b – SUBSCRIBE DỮ LIỆU CẢM BIẾN TỪ MQTT BẰNG PYTHON TRÊN PC**

Dưới đây là mô phỏng ESP32+DHT22 trong wokwi:

****

Tiếp theo, ta sẽ thấy dòng JSON xuất hiện liên tục mỗi 2 giây khi mở terminal hoặc CMD và chạy lệnh: python subscriber.py

****

**LAB 2.4: LƯU TRỮ VÀ TRỰC QUAN HÓA DỮ LIỆU IoT**

**Lab 2.4a: Mô phỏng với wokwi (DHT22+ESP 32)**

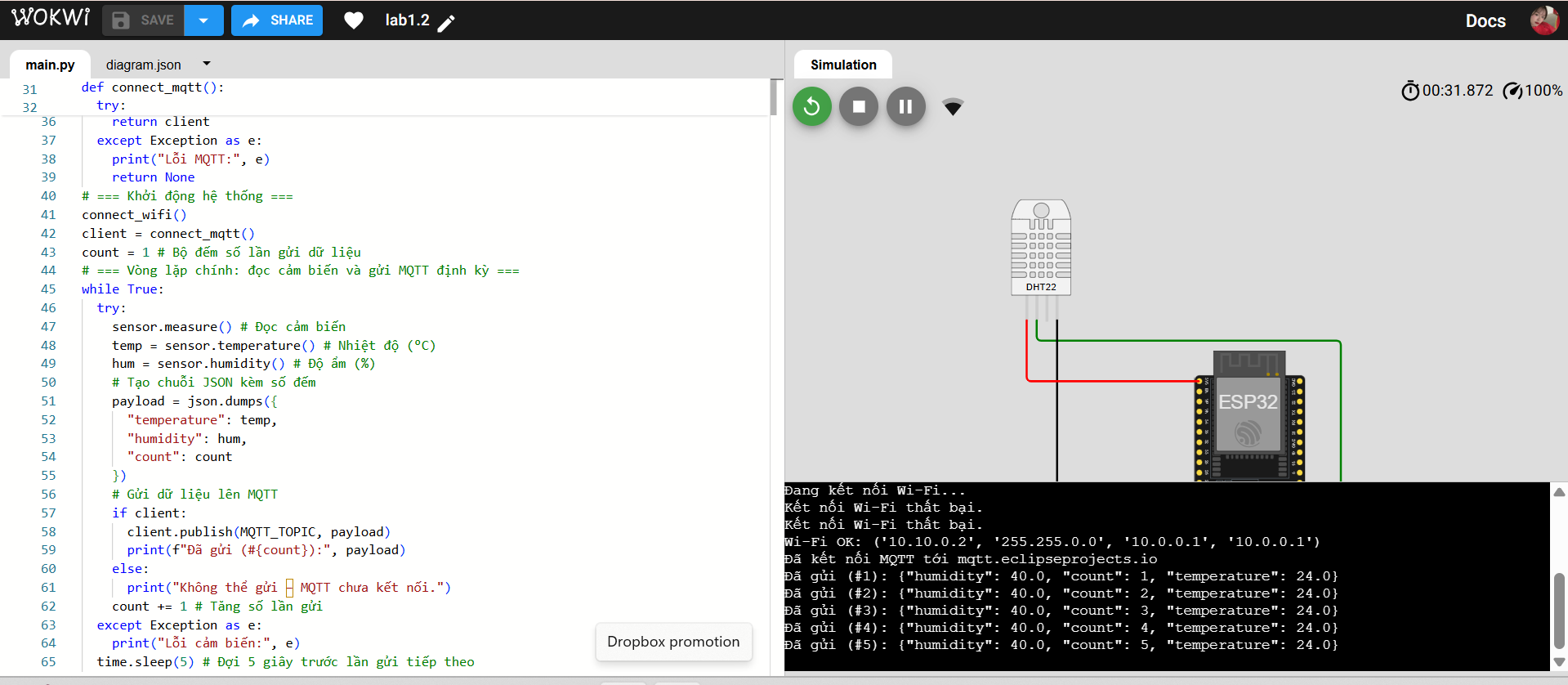
Kiểm thử Lab 2.4a

*Bước 1: Khởi động Publisher*

Chạy mô phỏng trong Wokwi với ESP32 + DHT22 (main.py đang publish JSON lên iot/khdl/esp32)

Đảm bảo mỗi 2 giây có 1 bản tin JSON gửi lên.

Dưới đây là kết quả:



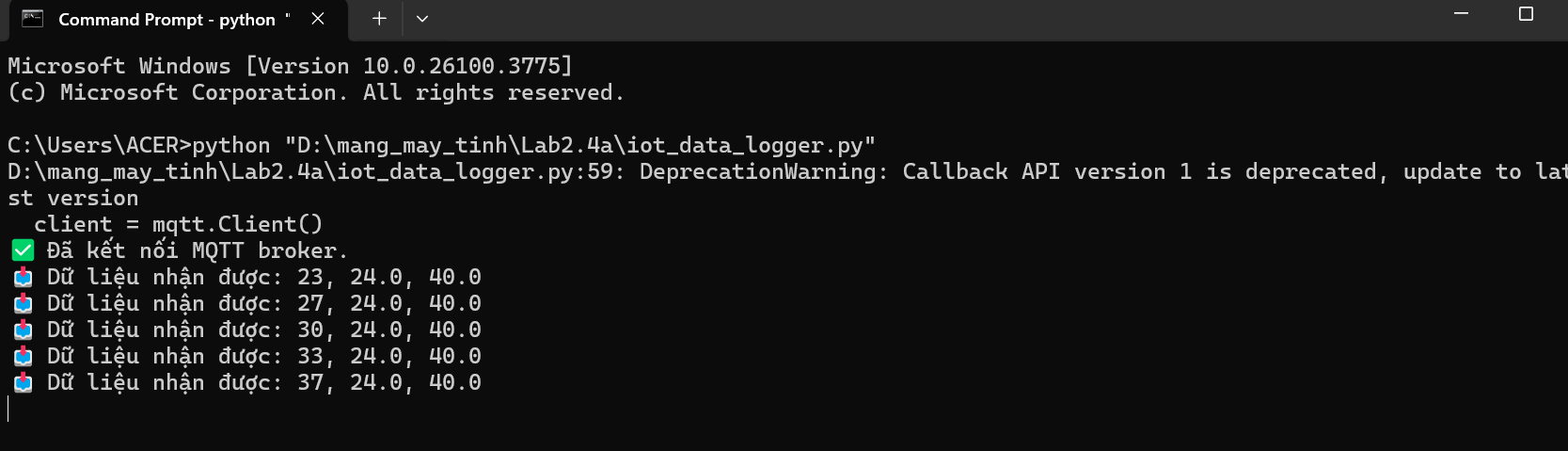
*Bước 2: Kiểm tra dữ liệu hiển thị trong terminal*

Chạy iot\_data\_logger.py Vào thư mục ~\lab2.4\có chứa file iot\_data\_logger.py Gõ lệnh: python iot\_data\_logger.py

Quan sát log hiển thị: 📥 Dữ liệu nhận được: timestamp, nhiệt độ, độ ẩm

Nếu có lỗi, kiểm tra định dạng JSON hoặc kết nối MQTT

Dưới đây là kết quả:

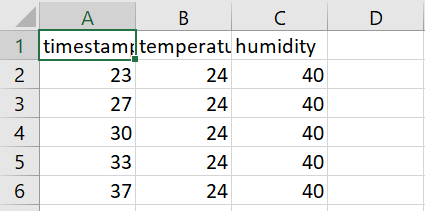


*Bước 3: Kiểm tra file CSV*

Mở file sensor\_data.csv bằng Excel hoặc Notepad

Kiểm tra dữ liệu có được ghi dòng mới đúng định dạng không

Dưới đây là kết quả:



*Bước 4: Kiểm tra vẽ biểu đồ*

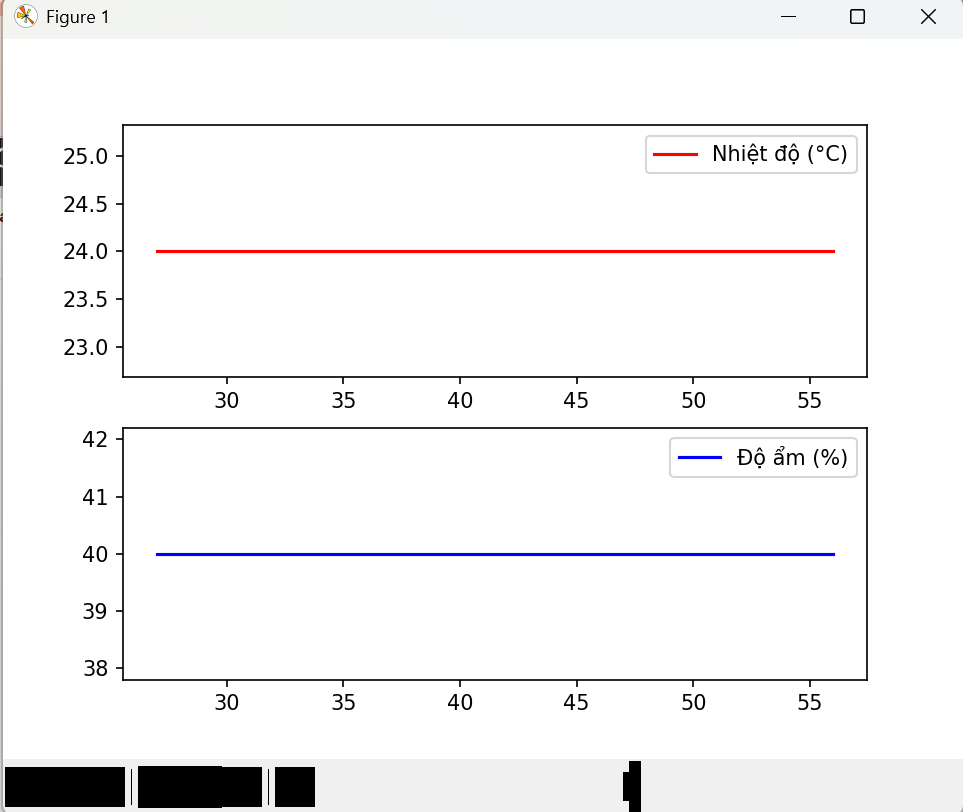
Sau 10 bản tin: xuất hiện cửa sổ vẽ matplotlib

Biểu đồ gồm 2 phần:

Trên: nhiệt độ theo thời gian

Dưới: độ ẩm theo thời gian

Dưới đây là kết quả:



*Bước 5: Dừng thử nghiệm*

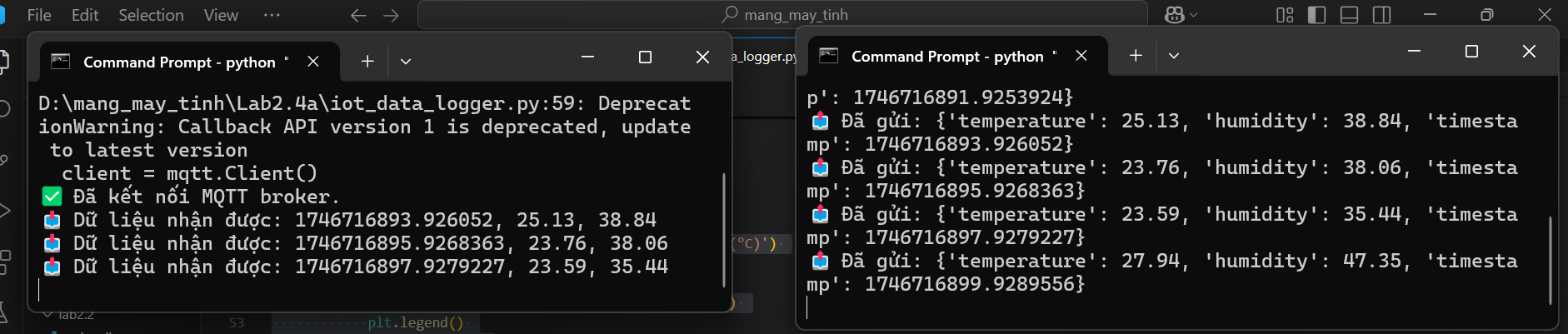
Nhấn Ctrl+C trong terminal để dừng script

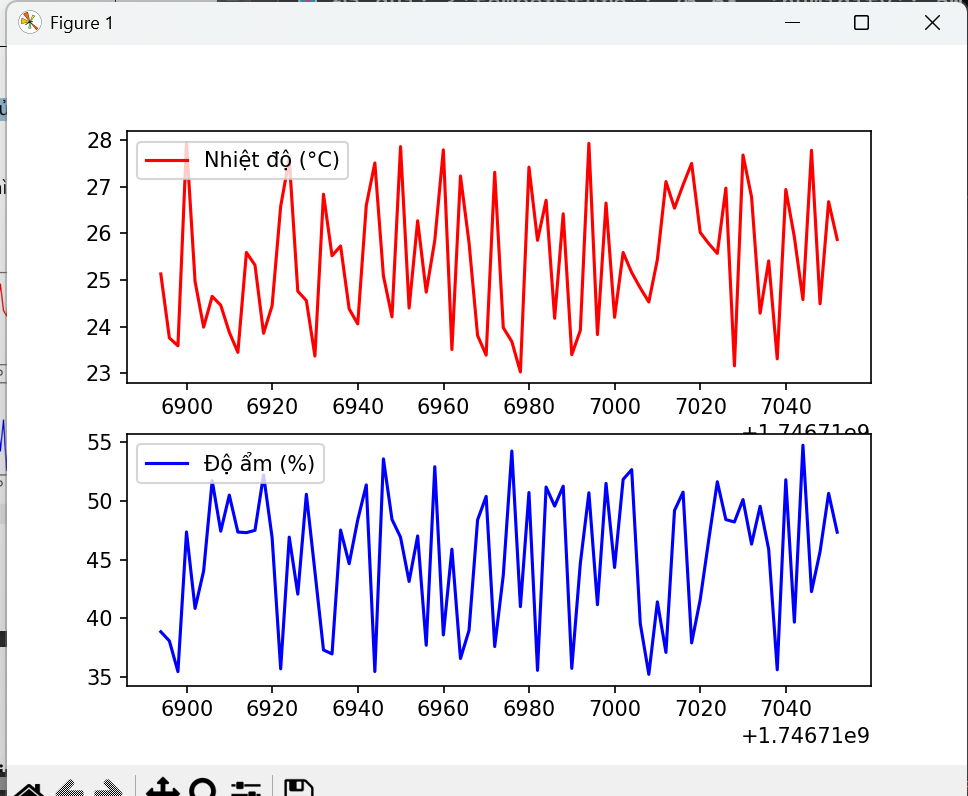
**Lab 2.4b: Mô phỏng dữ liệu ngẫu nhiên ngay trong publisher**

Bước 1 Khởi động Publisher: Thay vì chạy wokwi, chúng ta chạy một cửa sổ terminal: terminal 1: chạy: python iot\_fake\_publisher.py

Mở Terminal 2: chạy: python iot\_data\_logger.py

Dưới đây là kết quả:

****

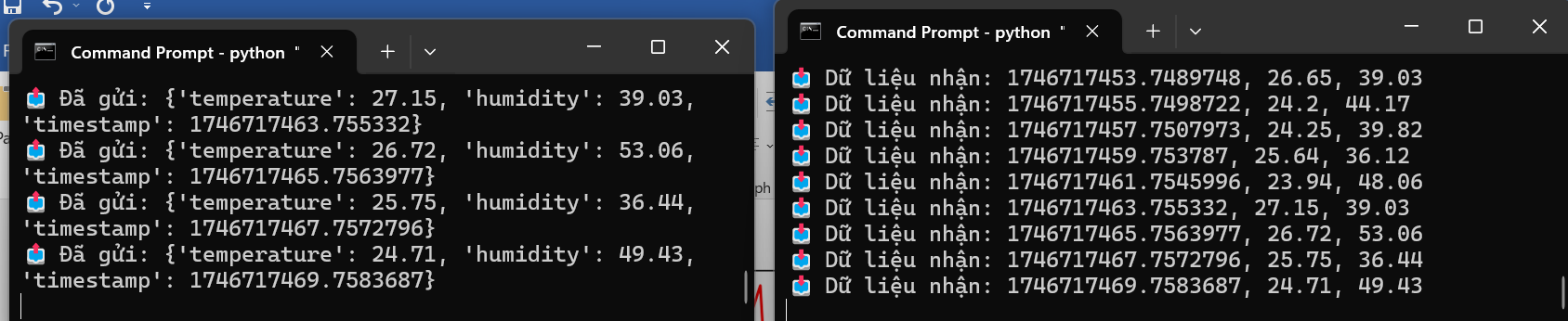
****

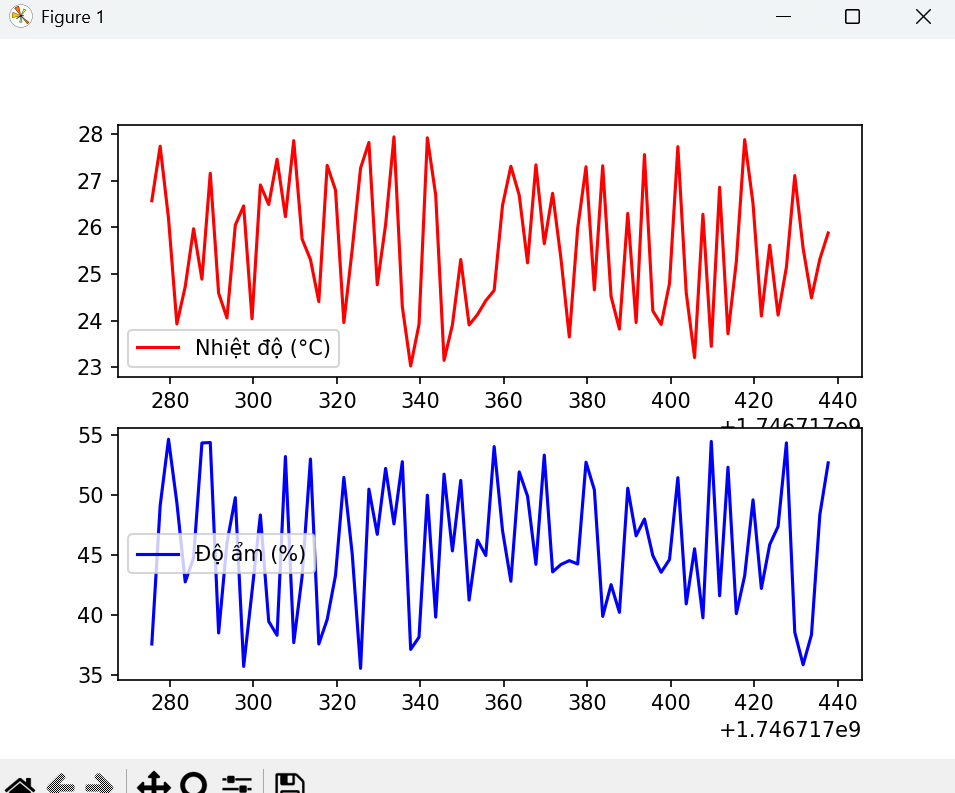
**Lab 2.4c. Lưu dữ liệu cảm biến vào SQLite và vẽ biểu đồ realtime**

Bước 1: Terminal 1: python iot\_fake\_publisher.py

Terminal 2: python iot\_data\_logger\_sqlite.py

Các bước còn lại thực hiện tương tự Lab 2.4a

****

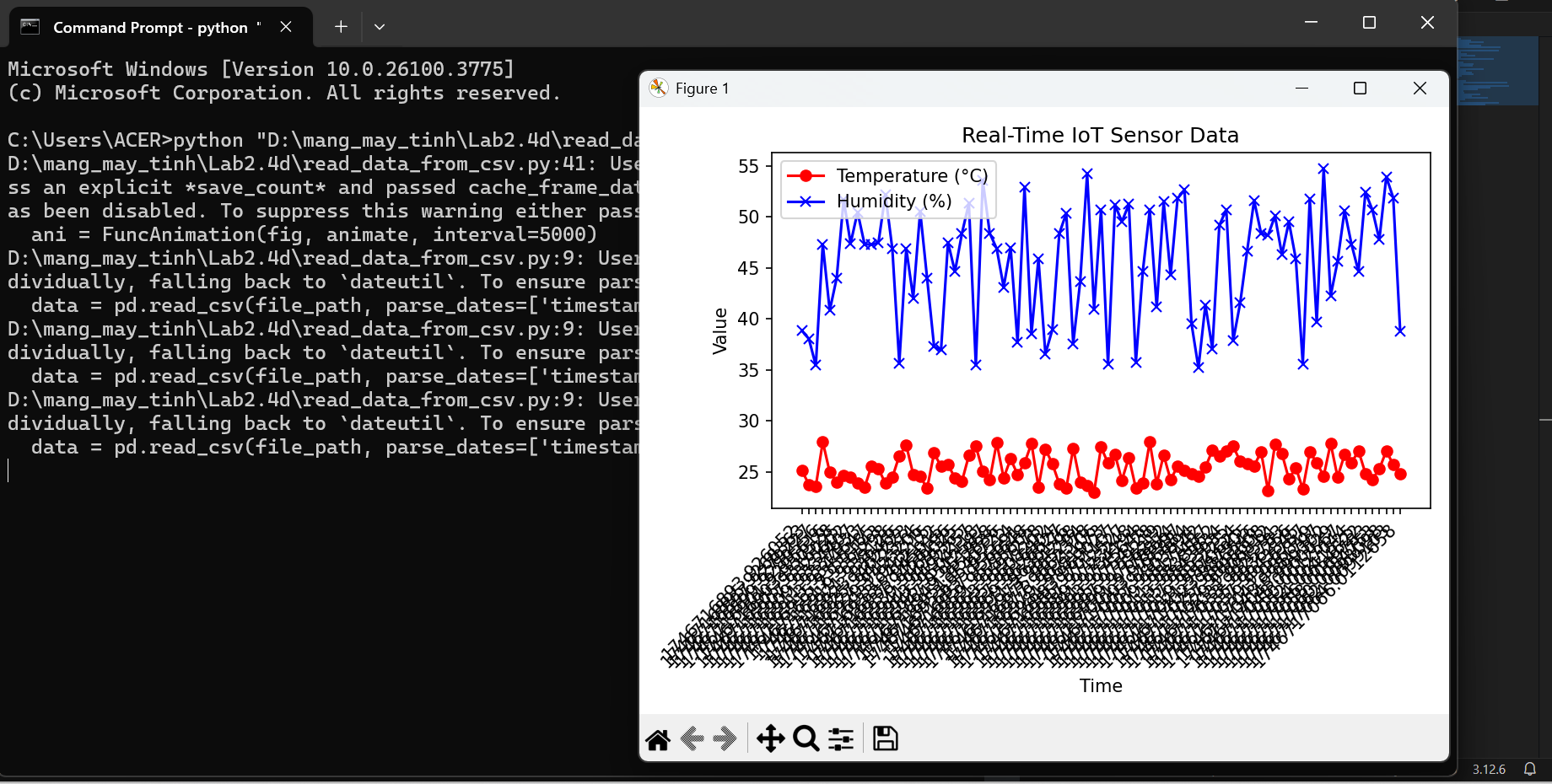
****

**Lab 2.4. d. Đọc dữ liệu từ file .csv**

Giả sử file sensor\_data.csv có nội dung như sau:

|  |  |
| --- | --- |
| 1 | timestamp,temperature,humidity |
| 2 | 2023-07-15 10:30:00,25.5,60.0 |
| 3 | 2023-07-15 10:31:00,25.7,59.8 |

Sau khi chạy đoạn code read\_data\_from\_csv.py ta có kết quả sau:

****