TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

**KHOA VẬT LÝ**

-----------------🕮----------------

**

**PROJECT KẾT THÚC MÔN HỌC**

**THỰC TẬP TIN HỌC ỨNG DỤNG**

Sinh viên thực hiện: Văn Đức Hải Đăng

Lớp: K66 Kĩ thuật điện tử và tin học

Mã sinh viên: 21000423

**HÀ NỘI - 2024**

MỞ ĐẦU

Qua dự án này, em đã thiết lập và chạy SparkSQL trong vùng chứa Docker và cài đặt các phần phụ thuộc cần thiết trong vùng chứa để thực hiện các truy vấn SQL trên cơ sở dữ liệu được lưu trữ trên container.

Vì kiến thức còn hạn hẹp, nếu có sai sót trong quá trình hoàn thành project. Rất mong có sự thông cảm và đóng góp nhận xét của thầy.

Em xin chân thành cảm ơn.

MỤC LỤC

[DANH MỤC HÌNH VẼ 3](#_Toc169124969)

[1. SETUP DOCKER 4](#_Toc169124970)

[2. SPARK INSTALLATION 5](#_Toc169124971)

[3. DATABASE SETUP 8](#_Toc169124972)

[4. DEPENDENCIES 10](#_Toc169124973)

[5. CONFIGURATION 11](#_Toc169124974)

[6. RUNNING QUERIES 13](#_Toc169124975)

[KẾT LUẬN 16](#_Toc169124976)

# DANH MỤC HÌNH VẼ

[Hình 1: Docker desktop 4](#_Toc169124996)

[Hình 2: Khởi tạo Dockerfile 4](#_Toc169124997)

[Hình 3: Sau khi build image 4](#_Toc169124998)

[Hình 4: image đã chạy 4](#_Toc169124999)

[Hình 5: Container 4](#_Toc169125000)

[Hình 6: Dockerfile 6](#_Toc169125001)

[Hình 7: Build image 6](#_Toc169125002)

[Hình 8: Container 6](#_Toc169125003)

[Hình 9: Cài đặt Spark 7](#_Toc169125004)

[Hình 10: Dockerfile 8](#_Toc169125005)

[Hình 11: Nhập data 9](#_Toc169125006)

[Hình 12: Dockerfile 10](#_Toc169125007)

[Hình 13: Spark 11](#_Toc169125008)

[Hình 14: Lệnh kết nối spark tới cơ sở dữ liệu 11](#_Toc169125009)

[Hình 15: test.py 11](#_Toc169125010)

[Hình 16: Kết quả chạy 12](#_Toc169125011)

[Hình 17: Khởi tạo Spark 13](#_Toc169125012)

[Hình 18: Read 13](#_Toc169125013)

[Hình 19: UPDATE.py 14](#_Toc169125014)

[Hình 20: Kết quả update 14](#_Toc169125015)

[Hình 21: delete.py 14](#_Toc169125016)

[Hình 22: Kết quả delete 15](#_Toc169125017)

# 1. SETUP DOCKER

(1) Tải Docker Desktop web



#### Hình 1: Docker desktop

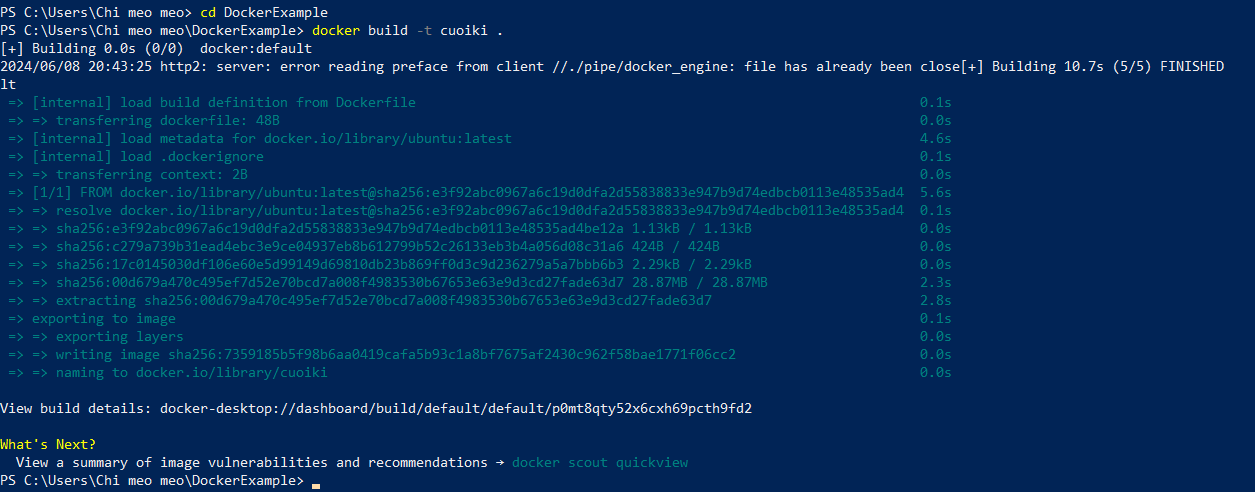
(2) Tạo thư mục tên DockerExample rồi vào DockerExample tạo file tên Dockerfile



#### Hình 2: Khởi tạo Dockerfile

(3) Lưu file và build image với dòng lệnh

docker build -t cuoiki .



#### Hình 3: Sau khi build image



#### Hình 4: image đã chạy

(4) Chạy container với dòng lệnh

docker run -it cuoiki bash



#### Hình 5: Container

# 2. SPARK INSTALLATION

(1) Một số công cụ cần thiết cần cài để chạy Spark trên container. Nhập lệnh vào Dockerfile ở trên:

*ENV DEBIAN\_FRONTEND=noninteractive*

*RUN apt-get update \*

*&& apt-get install -y curl tar wget vim \*

*python3 python3-pip \*

*&& ln -s /usr/bin/python3 /usr/bin/python \*

*&& apt-get clean \*

*&& rm -rf /var/lib/apt/lists/\**

*RUN apt-get update \*

*&& apt-get install -y default-jdk \*

*&& apt-get clean \*

*&& rm -rf /var/lib/apt/lists/\**

*RUN wget -qO- "https://downloads.apache.org/spark/spark-${SPARK\_VERSION}/spark-${SPARK\_VERSION}-bin-hadoop${HADOOP\_VERSION}.tgz" | tar -xz -C /opt/*

Dòng đầu tiên được dùng để không bị hỏi khi cài đặt thư viện

Spark phiên bản 3.5.1

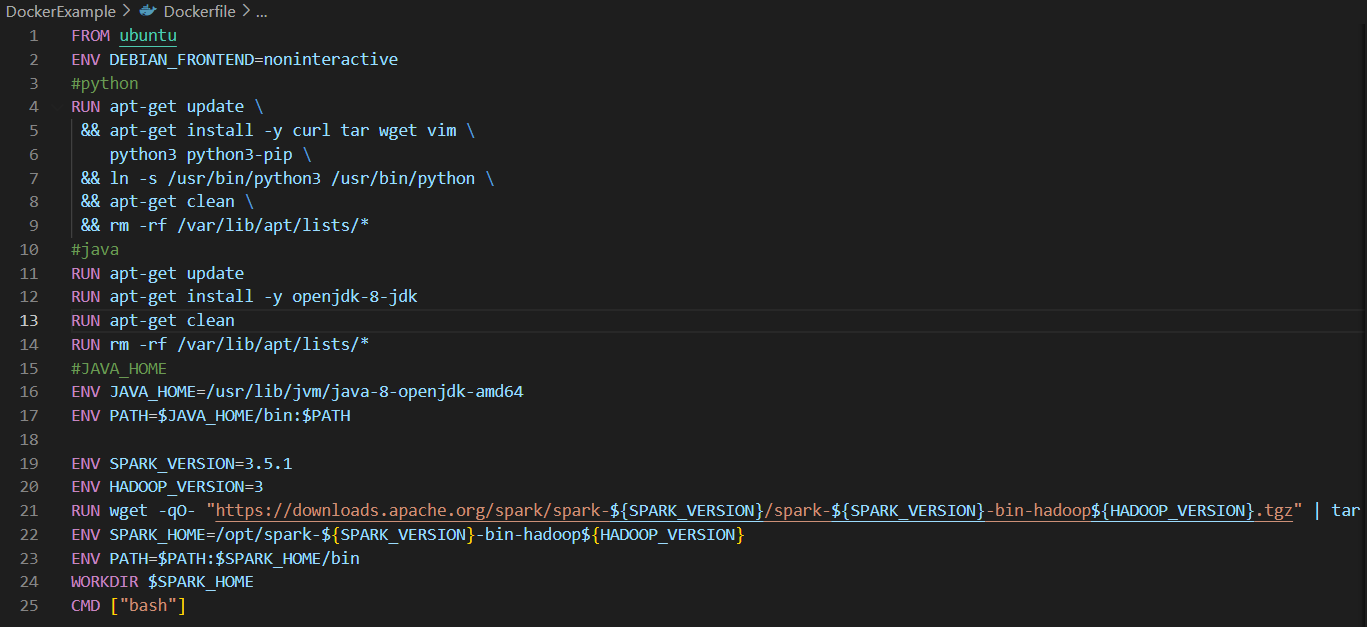
(2) Cấu hình đường dẫn cho Spark trong Dockerfile sau khi chọn các gói thư viện và công cụ

*ENV JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64*

*ENV PATH=$JAVA\_HOME/bin:$PATH*

*ENV SPARK\_HOME=/opt/spark-${SPARK\_VERSION}-bin-hadoop${HADOOP\_VERSION}*

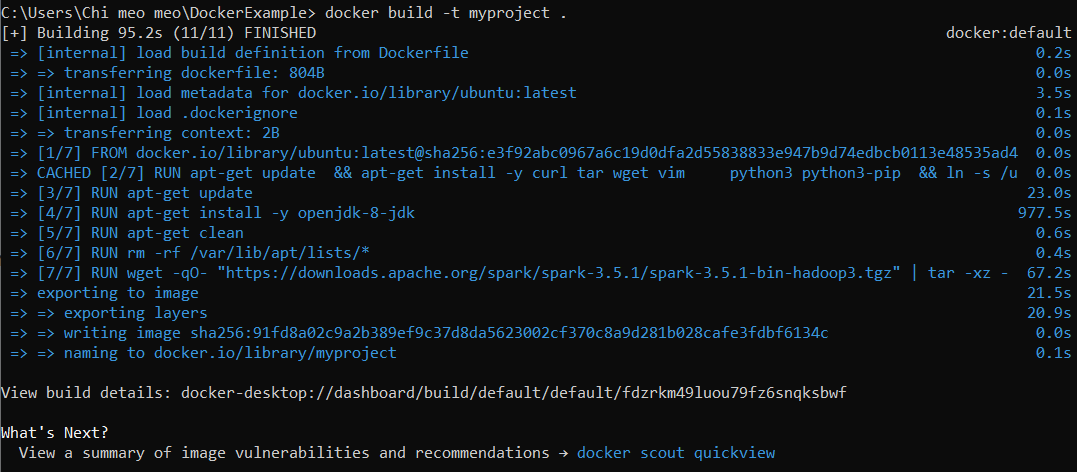
*ENV PATH=$PATH:$SPARK\_HOME/bin*

**

#### Hình 6: Dockerfile

(3) Build image với dòng lệnh

docker build -t myproject .

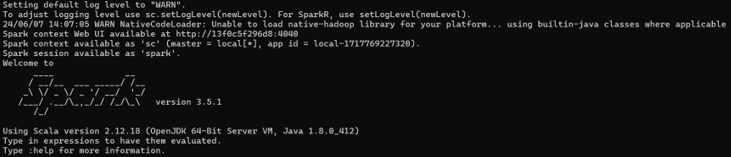


#### Hình 7: Build image

(4) Chạy container



#### Hình 8: Container



#### Hình 9: Cài đặt Spark

# 3. DATABASE SETUP

(1) Bổ sung lệnh vào Dockerfile

*RUN apt-get update \*

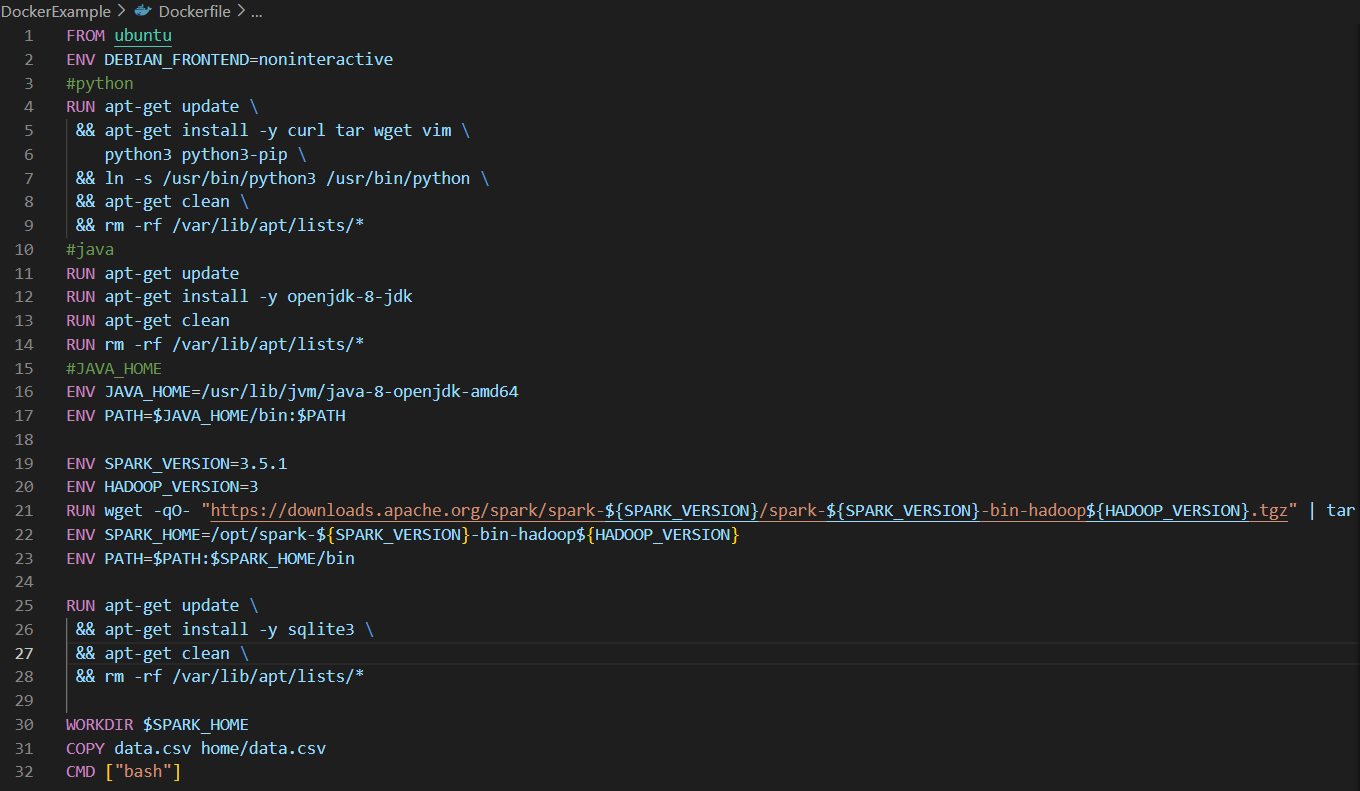
*&& apt-get install -y sqlite3 \*

*&& apt-get clean \*

*&& rm -rf /var/lib/apt/lists/\**

(2) Copy file tên data.csv vào thư mục home với tên là data.csv

(3) Lưu file và build lại image myproject



#### Hình 10: Dockerfile

(4) Chạy container

*Docker run –t myproject .*

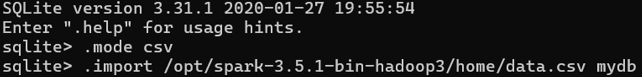
(5) Tạo cơ sở dữ liệu

*sqlite3 /home/test.db*

(6) Đưa file data.csv vào cơ sở dữ liệu

*.mode csv*

*.import /opt/spark-3.5.1-bin-hadoop3/home/data.csv mydb*



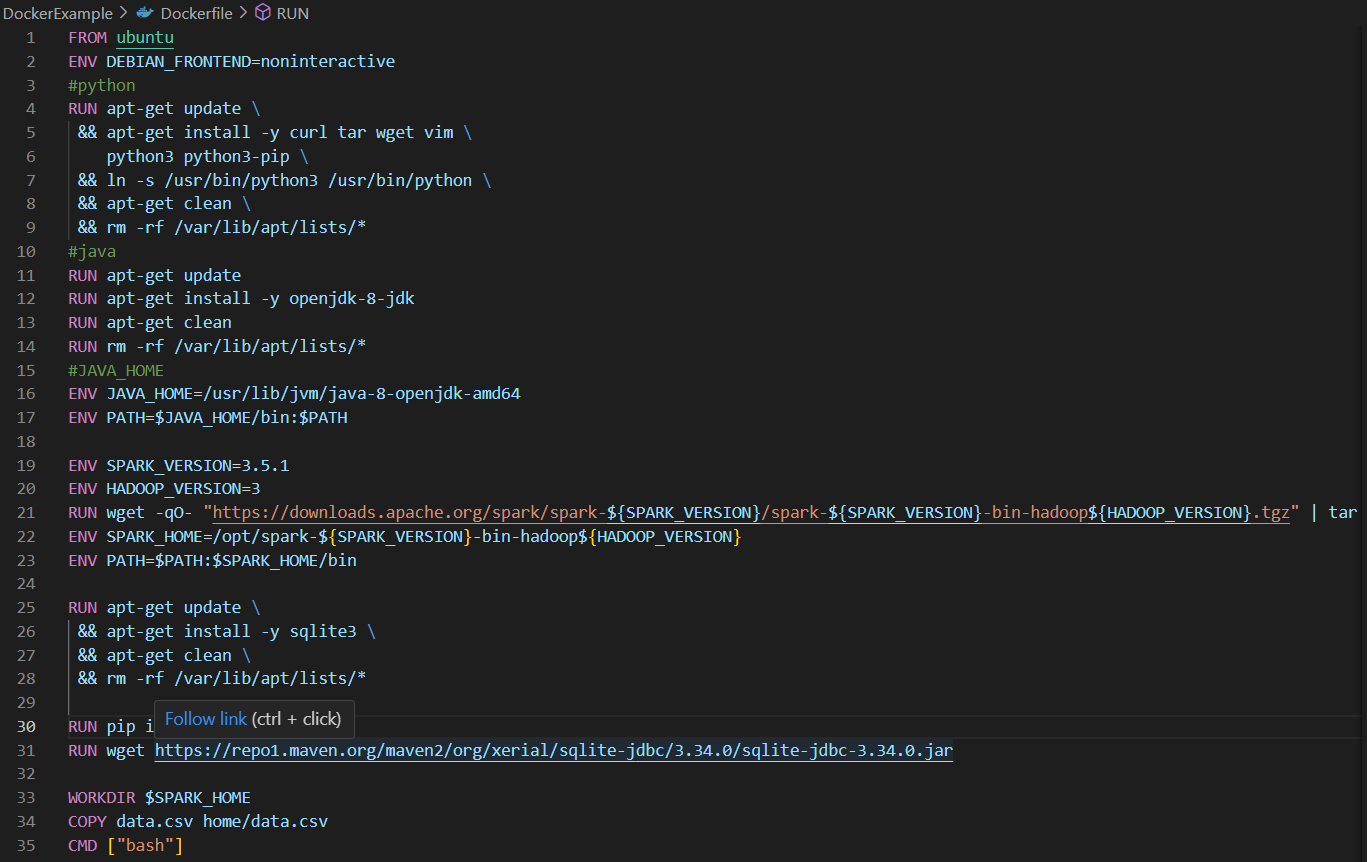
#### Hình 11: Nhập data

# 4. DEPENDENCIES

(1) Cài JDBC với các dòng lệnh sau thêm vào Dockerfile

*RUN pip install pyspark*

*RUN wget* [*https://repo1.maven.org/maven2/org/xerial/sqlite-jdbc/3.34.0/sqlite-jdbc-3.34.0.jar*](https://repo1.maven.org/maven2/org/xerial/sqlite-jdbc/3.34.0/sqlite-jdbc-3.34.0.jar)



#### Hình 12: Dockerfile

(2) Build image myproject như phần trên

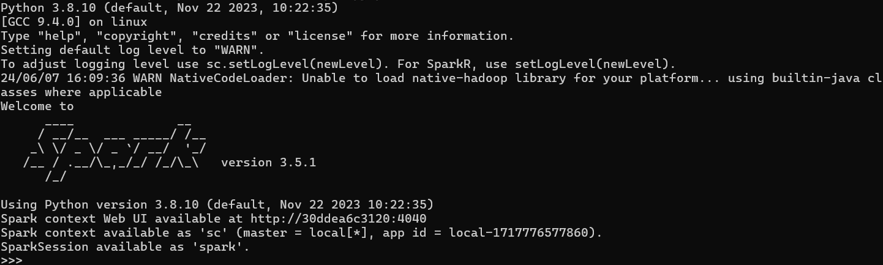
(3) Chạy container

# 5. CONFIGURATION

(1) Chuyển JDBC vào jars để Spark kết nối với cơ sở dữ liệu với lệnh:

*mv sqlite-jdbc-3.34.0.jar /opt/spark/jars*

(2) Nhập lệnh spark sau khi chạy container và build image



#### Hình 13: Spark

(3) Kết nối spark tới cơ sở dữ liệu và thực hiện đọc dữ liệu bảng trong cơ sở dữ liệu với lệnh:

*df=spark.read.format("jdbc").option("url",f"jdbc:sqlite:/home/test.db").*

*opion("dbtable","mydb").load()*



#### Hình 14: Lệnh kết nối spark tới cơ sở dữ liệu

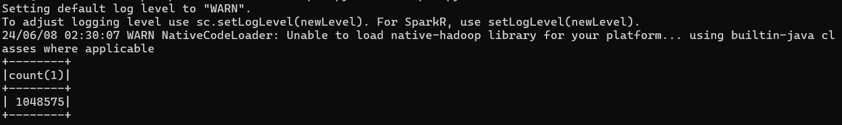
(4) Tạo file test.py để kết nối SparkSQL và thực hiện truy vấn



#### Hình 15: test.py

(5) Chạy test.py với lệnh:

*python3 test.py*

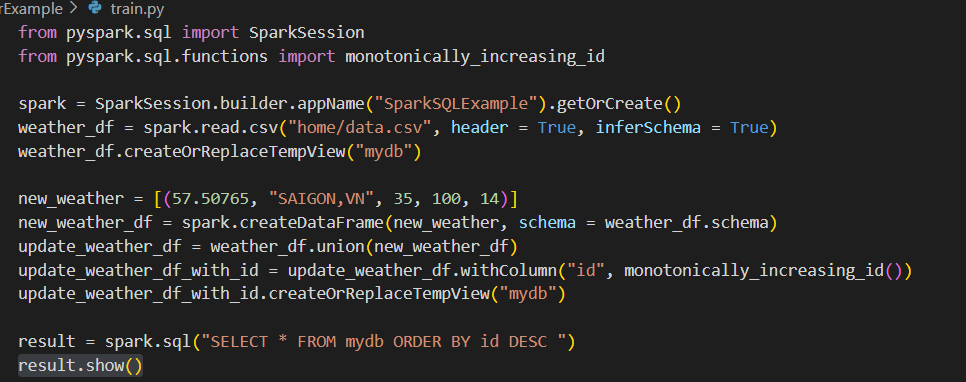
Hình 16: Kết quả chạy

# 6. RUNNING QUERIES

CREATE:

Thêm hàng và cột

Tạo file train.py và thực hiện ghi dòng lệnh sau để khởi tạo Spark và đọc dữ liệu từ bảng



#### Hình 17: Khởi tạo Spark

READ:

Đọc từ bảng mydb với tham chiếu là cột id và cho xếp từ cao xuống thấp

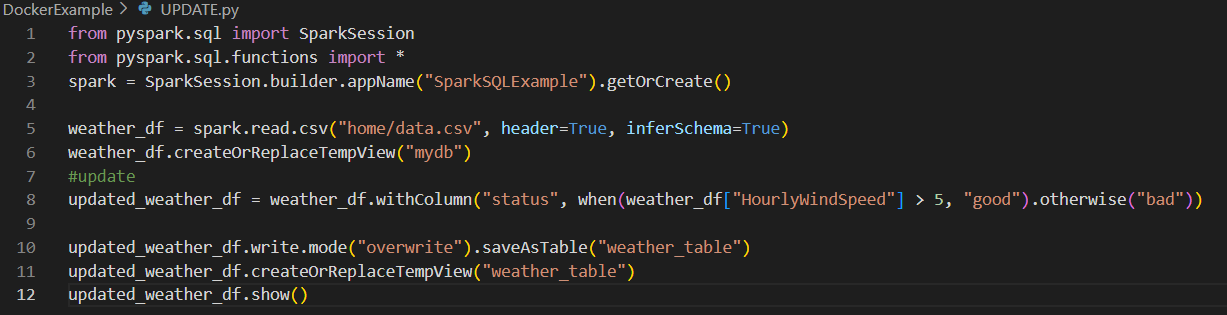


#### Hình 18: Read

Viết 2 câu lệnh rồi chạy ra kết quả

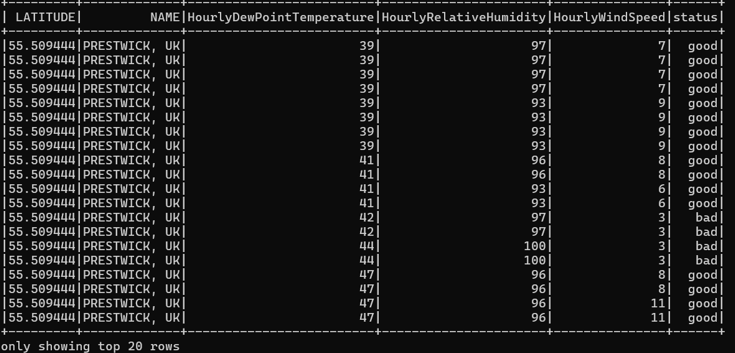
Update:

Tạo file UPDATE.py



#### Hình 19: UPDATE.py

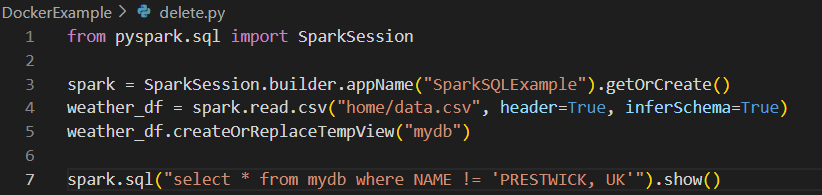
Kết quả:



#### Hình 20: Kết quả update

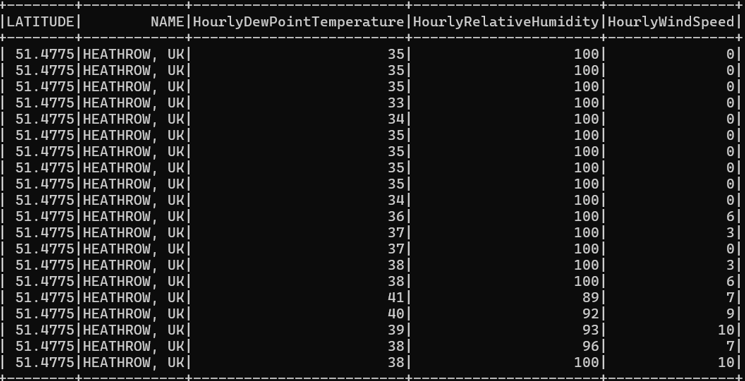
Delete:

Tạo file delete.py



#### Hình 21: delete.py

Kết quả:



#### Hình 22: Kết quả delete

# KẾT LUẬN

Qua việc làm project này, em đã biết cách chạy một docker container, dùng một tập cơ sở dữ liệu cho trước trong docker, biết cách build image, biết cách dùng Spark trong docker.