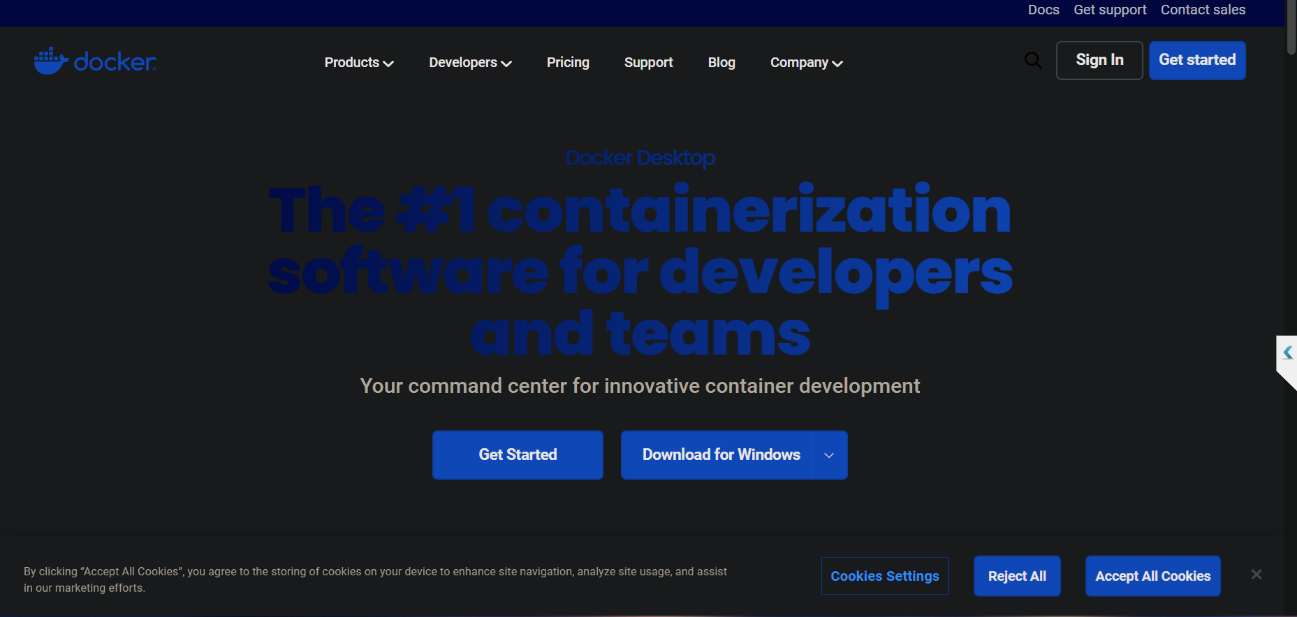
SparkSQL in Docker Project

Mục tiêu: Mục tiêu của project này là cài đặt và chạy SparkSQL trong một Docker container. Container này sẽ lưu trữ một cơ sở dữ liệu (cụ thể là SQLite database) và cho phép thực hiện các truy vấn SQL sử dụng SparkSQL.

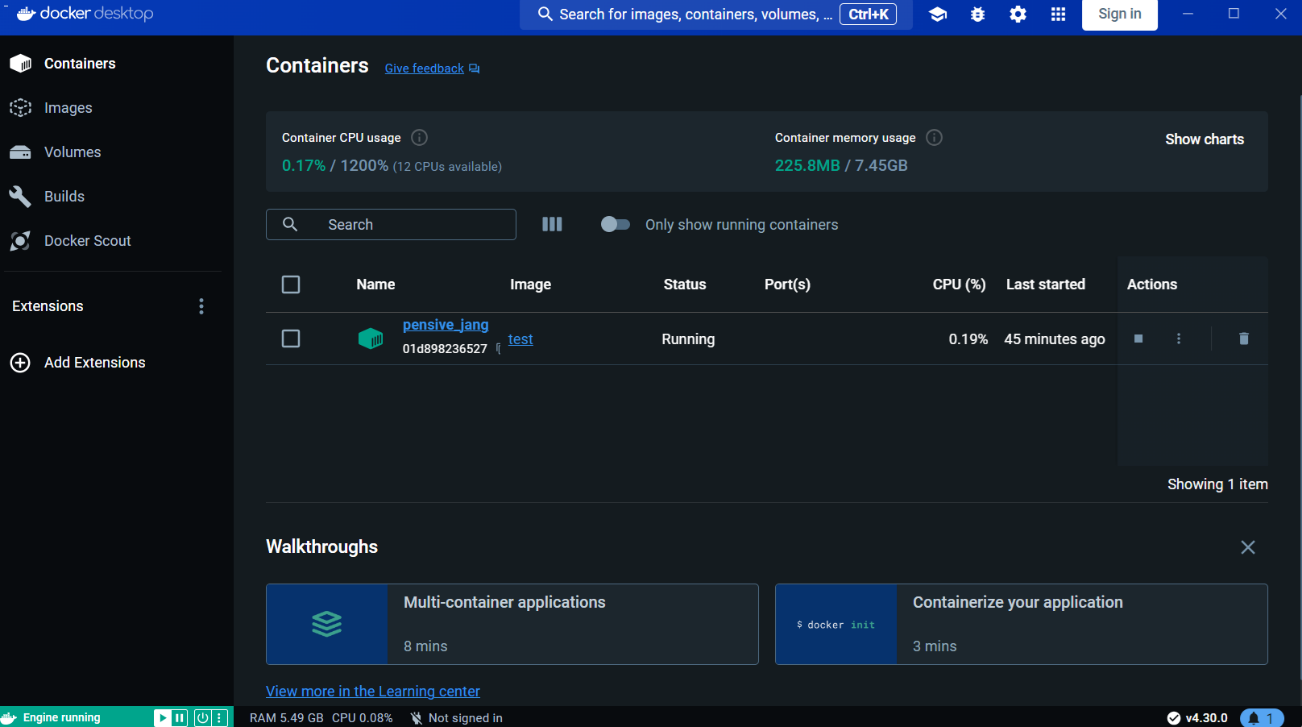
Setup:

Tải dockerdestop tùy theo hệ điều hành của máy.

<https://www.docker.com/products/docker-desktop/>



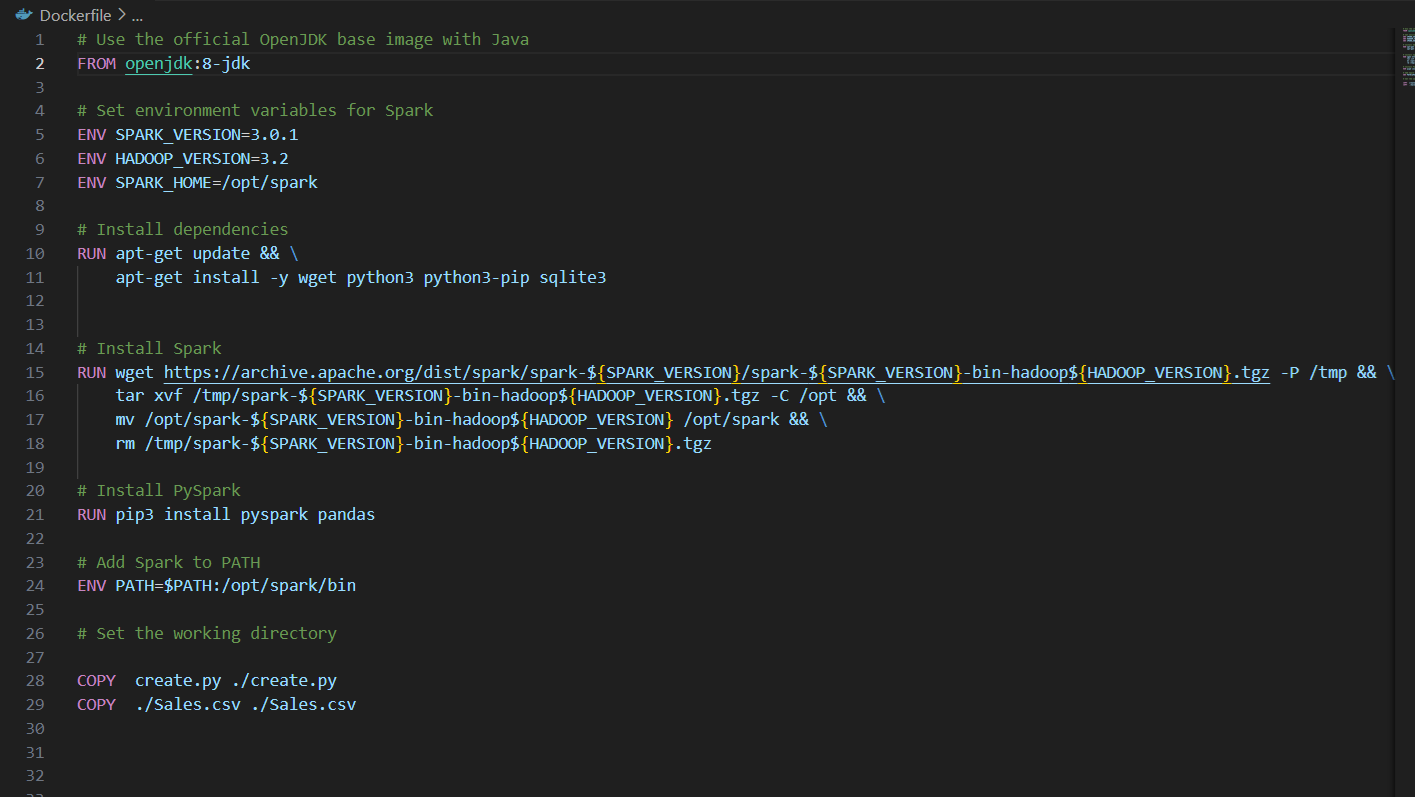
Đây là giao diện.



Ta tiến hành dow và cài đặt chương trình.

Sau đó em sử dụng VS Code để tạo và cài đặt cho chương trình.

Đầu tiên ta tạo file tên Dockerfile với nội dung như sau:



Ở đây, đầu tiên sử dụng hình ảnh chính thức của OpenJDK với Java 8 làm cơ sở để xây dựng Image.

Sau đó thiết lập môi trường cần thiết cho spark, SPARK\_VERSION xác định phiên bản Spark mà bạn muốn cài đặt, HADOOP\_VERSION xác định phiên bản Hadoop và SPARK\_HOME là đường dẫn nơi Spark sẽ được cài đặt.

Chạy lệnh apt-get update để cập nhật danh sách các gói và sau đó cài đặt các gói cần thiết như wget, python3, python3-pip, và sqlite3. Cờ -y được sử dụng để tự động xác nhận các lời nhắc trong quá trình cài đặt.

Cài đặt spark:

Tải xuống file tar.gz của Spark từ trang lưu trữ của Apache về thư mục /tmp.

Giải nén file tar.gz vào thư mục /opt.

Đổi tên thư mục vừa giải nén thành /opt/spark để dễ dàng tham chiếu.

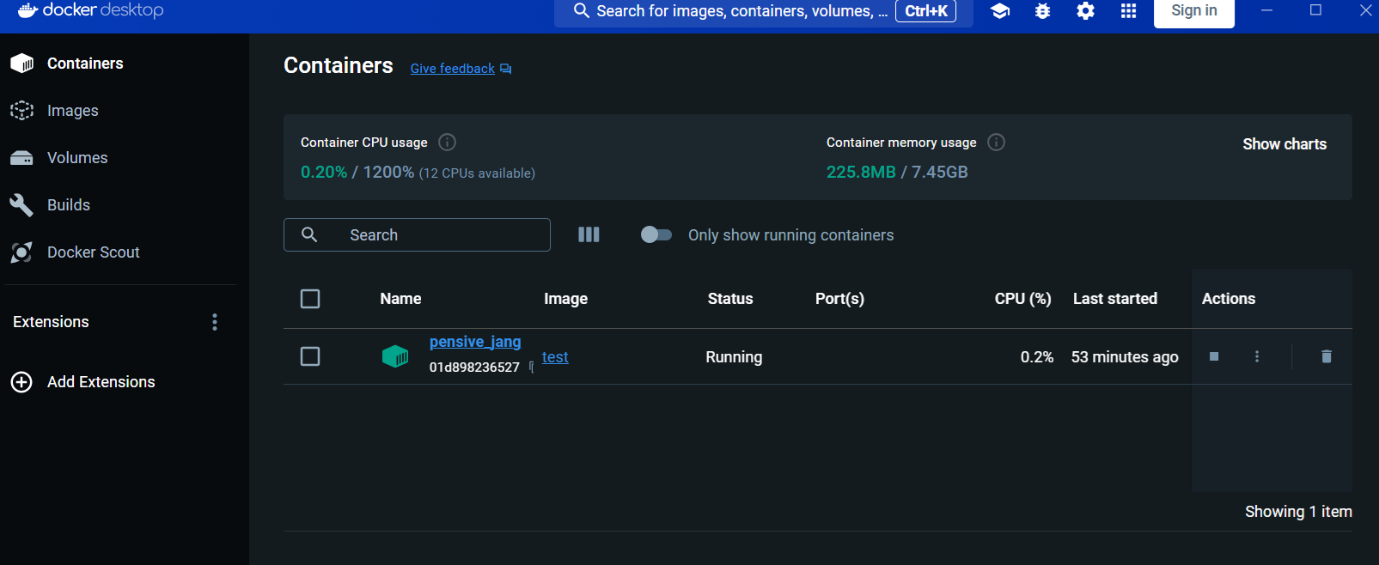
Xóa file tar.gz đã tải về để tiết kiệm không gian đĩa.

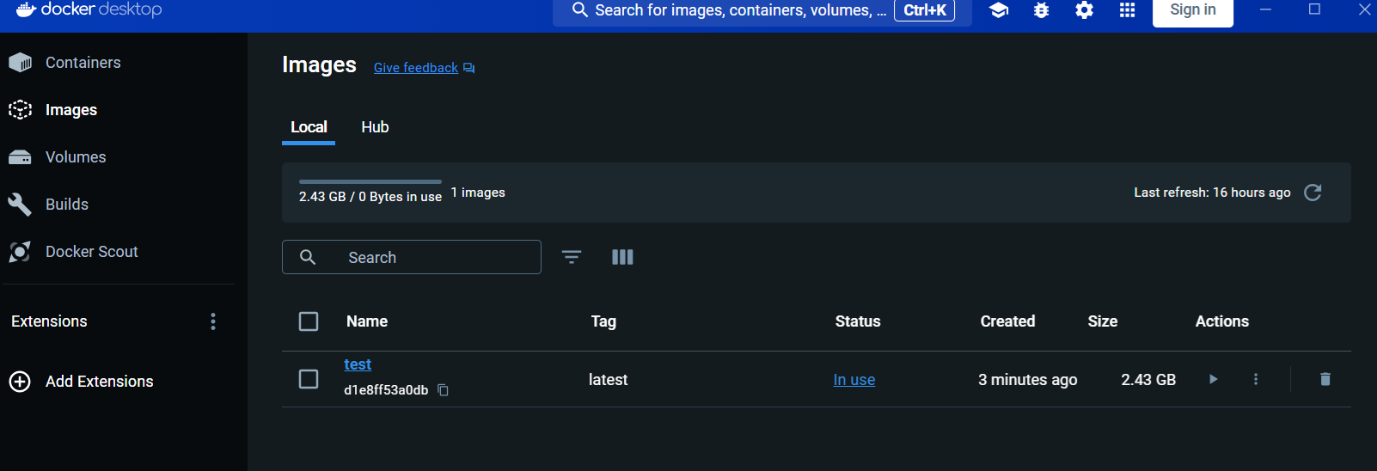
Sử dụng pip3 để cài đặt các gói Python cần thiết cho PySpark và pandas.

Cập nhật biến môi trường PATH để bao gồm đường dẫn đến thư mục bin của Spark. Điều này cho phép bạn gọi các lệnh của Spark từ bất kỳ đâu trong container.

Sao chép file create.py và Sales.csv từ máy của bạn vào thư mục làm việc mặc định trong container. Những file này sẽ có sẵn để sử dụng trong container.

Sau khi setup xong dùng lệnh ‘docker build -t test .’ và ‘docker run -it test’ để chạy và cài containers và image trong dockerdestop.

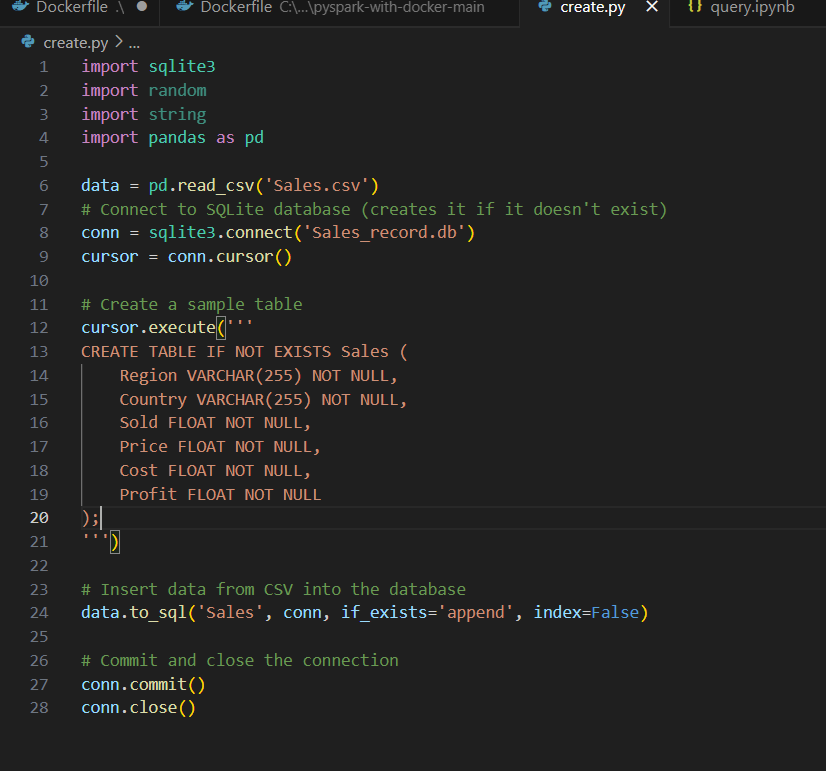




**Thực thi:**

Tạo cơ sở dữ liệu:

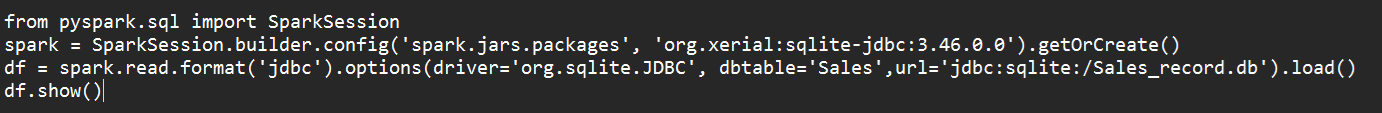
Chuẩn bị 1 file có tên create.py giúp tatạo ra một cơ sở dữ liệu SQLite bằng sqlite3.



Đầu tiên tạo database tên là ‘Sales\_record’ sau đó tạo table tên ‘Sales’ bao gồm 6 cột Region, Country, Sold, Price, Cost, Profit.

Ta đọc file ‘Sales.csv’ bằng pandas bao gồm 1,000,000 dòng.

Sử dụng những câu lệnh này để kết nối với database.



Để kết nối với database SQLite, sử dụng một chuẩn API để tương tác với cơ sở dữ liệu có tên là JDBC driver. Cụ thể dự án sử dụng sqlite jdbc driver 3.46.0.0.

Do code trưc tiếp lên terminal lên hơi khó quan sát.

A screenshot of a computer program

Description automatically generated

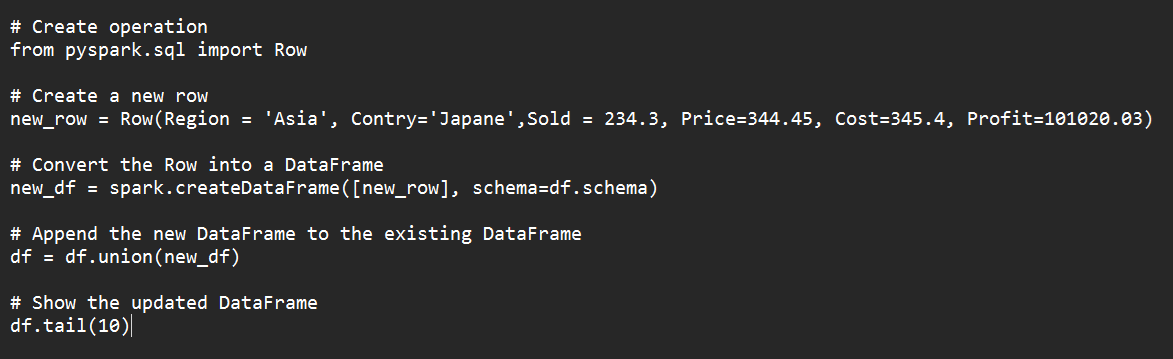
A screenshot of a computer

Description automatically generated

Thực hiện các thao tác CRUD.

Để thực hiện các thao tác CRUD hiệu quả, dự án sử dụng DataFrame của pyspark vì DataFrame đã được xây dựng để hoạt động hiệu quả với dataset lớn hay Big Data.

Create



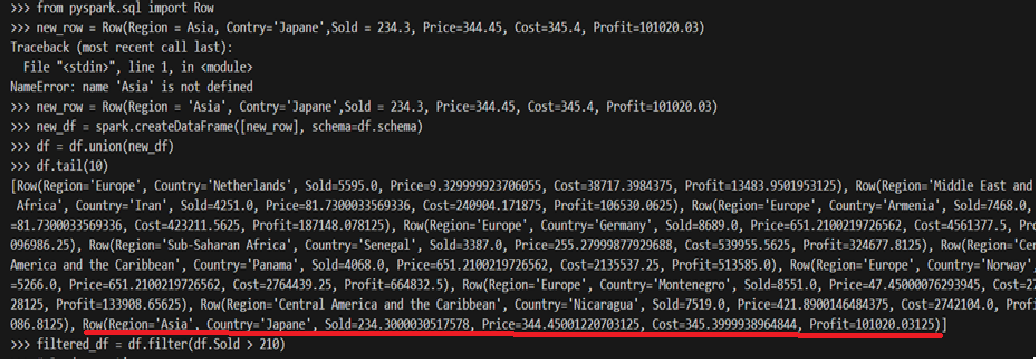
Tạo SparkSession

Tạo một dòng mới, Ở đây, Row được sử dụng để tạo một dòng mới với các giá trị được chỉ định.

Chuyển đổi Row thành DataFrame, dòng này chuyển đổi Row thành DataFrame mới với cùng schema như df.

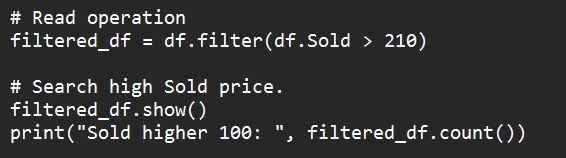
Thêm DataFrame mới vào DataFrame hiện có, dòng này thêm new\_df vào df hiện có.

Hiển thị DataFrame cập nhật.



Dòng gạch đỏ là dòng đã được thêm.

Read.



A screenshot of a computer screen

Description automatically generated

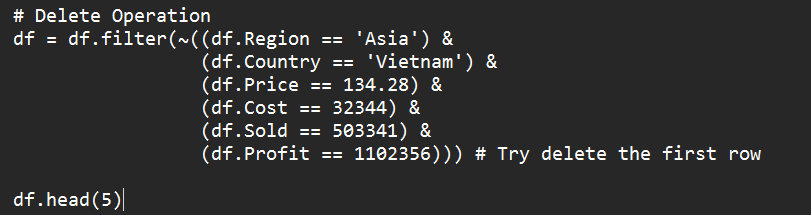
Update:



A black screen with white text

Description automatically generated

Delete



A black screen with white text

Description automatically generated