Tài liệu Hướng dẫn Cài đặt và Chạy SparkSQL với SQLite trong Docker

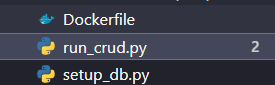
## Tổng quan

Tài liệu này cung cấp hướng dẫn từng bước về cách thiết lập môi trường Docker, kết nối SparkSQL với cơ sở dữ liệu SQLite, và thực hiện các thao tác CRUD cùng với các bài kiểm tra hiệu suất. Các phần sau đây sẽ bao gồm quá trình cài đặt, hướng dẫn xây dựng và chạy container Docker, và các ví dụ về truy vấn SQL.

## Yêu cầu

Đã cài đặt Docker trên máy của bạn.

Các file cần chuẩn bị là Dockerfile, setup\_db.py, run\_crud.py. Trong đó, Dockerfile xây dựng image Docker; file setup\_db.py để tạo dữ liệu, trong đó, dữ liệu bao gồm: id, Tên, tuổi, quốc tịch. Trong đó, các thông tin được điền tuần tự, các giá trị về tuổi, quốc tịnh được chọn ngẫu nhiên. File run\_crud.py là file thực hiện các thao tác lên trên database như là đọc, thêm, chỉnh sửa, xóa và đánh giá khác biệt của khi dùng WHERE và không dùng WHERE.



Yêu cầu tạo thư mục có directory tree như hình.

## Quá trình cài đặt

### 1. Thiết lập Môi trường Docker

Đoạn mã Dockerfile này làm các công việc sau:

*FROM ubuntu:20.04:* Đây là base image của Dockerfile, được xây dựng trên hệ điều hành Ubuntu 20.04.

*ENV DEBIAN\_FRONTEND=noninteractive:* Thiết lập biến môi trường DEBIAN\_FRONTEND với giá trị noninteractive, để tránh việc apt-get yêu cầu xác nhận người dùng trong quá trình cài đặt.

*Cài đặt các gói và công cụ cần thiết:* Cài đặt các gói và công cụ như curl, tar, wget, vim, python3 và python3-pip để chuẩn bị môi trường. Đồng thời thiết lập symlink từ python3 đến python.

*Cài đặt OpenJDK 8:* Cài đặt OpenJDK 8, một môi trường Java được sử dụng trong ứng dụng. Đồng thời thiết lập biến môi trường JAVA\_HOME để chỉ định đường dẫn đến JDK.

*Cài đặt Apache Spark:* Tải xuống và cài đặt Apache Spark phiên bản 3.5.1 và liên kết biến môi trường SPARK\_HOME đến thư mục cài đặt của Spark.

*Cài đặt SQLite và JDBC driver:* Cài đặt SQLite3 để sử dụng trong cơ sở dữ liệu và tải xuống JDBC driver cho SQLite.

*Thiết lập thư mục làm việc và sao chép các script:* Thiết lập thư mục làm việc thành thư mục cài đặt của Spark, sau đó sao chép các tệp setup\_db.py và run\_crud.py vào thư mục này.

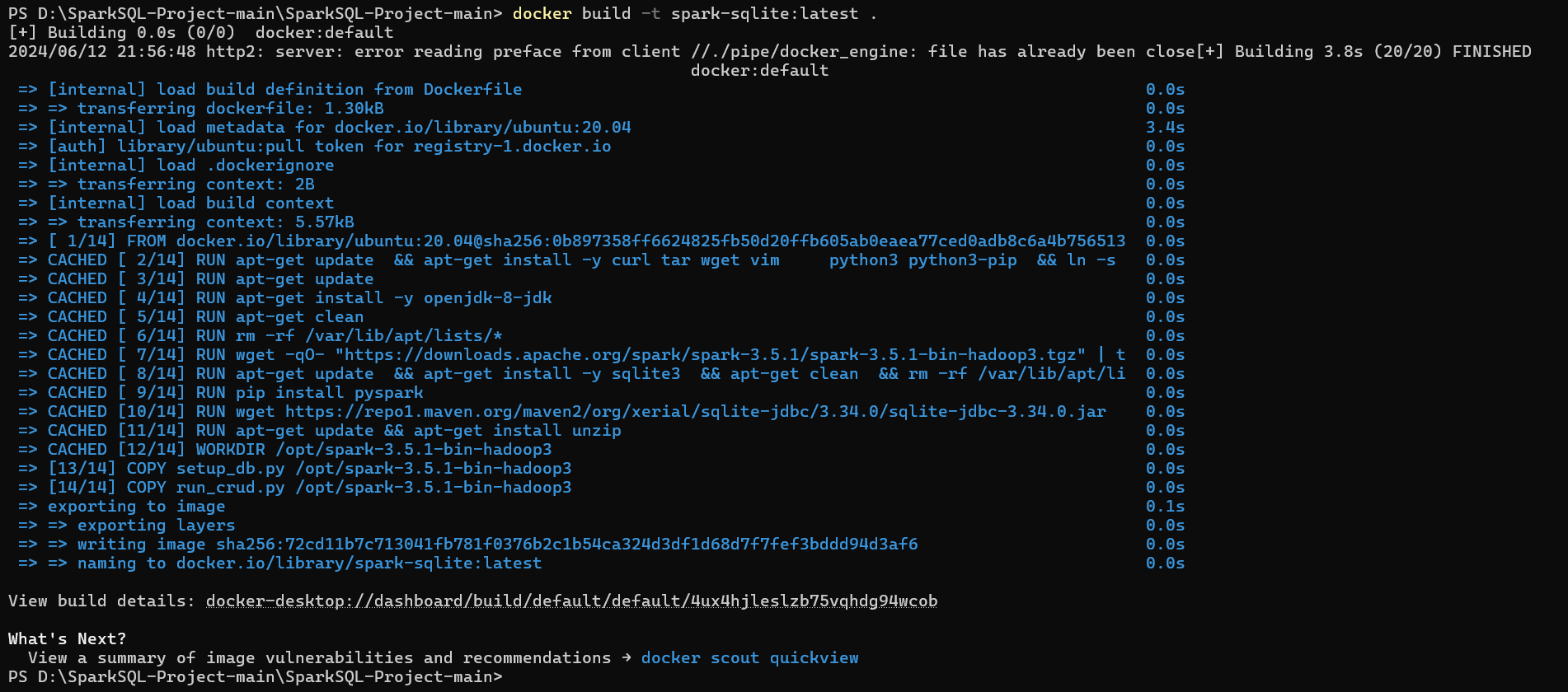
*ENTRYPOINT ["tail", "-f", "/dev/null"]:* Đây là lệnh mặc định sẽ được thực thi khi container được khởi chạy. Trong trường hợp này, nó chỉ là để giữ cho container chạy, vì nó sẽ không kết thúc ngay lập tức sau khi được khởi chạy.

Dockerfile này đang chuẩn bị một môi trường để phát triển và triển khai các ứng dụng sử dụng Apache Spark và SQLite.

### 2. Xây dựng Hình ảnh Docker

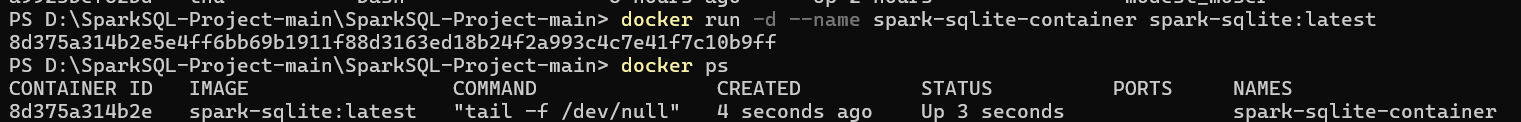
Chạy lệnh sau trong thư mục chứa Dockerfile để xây dựng Docker image:

***docker build -t spark-sqlite:latest .***



Đây là hình ảnh sau khi Docker Image được xây dựng hoàn thiện. Tiếp tục chạy container Docker với lệnh sau:

***docker run -d --name spark-sqlite-container spark-sqlite:latest***



***docker exec -it spark-sqlite-container /bin/bash***

Không có mô tả.

## Thiết lập Cơ sở dữ liệu

Đảm bảo cơ sở dữ liệu được thiết lập với lược đồ và dữ liệu ban đầu cần thiết. Sử dụng tập tin `setup\_db.py` để khởi tạo cơ sở dữ liệu SQLite.

### 1. Import các thư viện cần thiết:

sqlite3: Thư viện để làm việc với cơ sở dữ liệu SQLite.

random: Thư viện để sinh ra các giá trị ngẫu nhiên.

### 2. Hàm initialize\_db():

Đây là hàm chính để khởi tạo cơ sở dữ liệu. Hàm này thực hiện các bước tạo bảng và chèn dữ liệu mẫu vào.

### 3. Kết nối đến cơ sở dữ liệu SQLite:

Kết nối đến cơ sở dữ liệu SQLite và tạo một con trỏ (cursor) để thực hiện các thao tác SQL.

### 4. Tạo bảng people:

Tạo bảng people với các cột: id, name, age, và country.

### 5. Danh sách các quốc gia có thể:

Tạo một danh sách các quốc gia để chèn ngẫu nhiên vào bảng people.

### 6. Chèn dữ liệu mẫu:

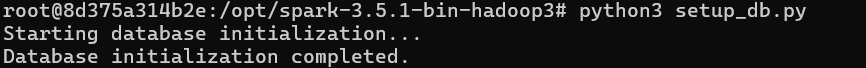
Vòng lặp này tạo ra 1,000,000 bản ghi với tên ngẫu nhiên, tuổi ngẫu nhiên từ 1 đến 100, và quốc gia ngẫu nhiên từ danh sách countries.

### 7. Lưu và đóng kết nối:

Lưu các thay đổi vào cơ sở dữ liệu và đóng kết nối.

Khi vào bên trong docker, ta sử dụng lệnh sau để tạo và điền vào database 1 triệu dòng:

***python3 /opt/setup\_db.py***



Khi thấy thông báo này, database đã được hoàn thành, ta sử dụng lệnh sau để cấp các quyền cho phép truy cập chỉnh sửa database:

***chmod 777 /opt/spark/examples/src/main/resources/people.db***

## Chạy các thao tác CRUD của SparkSQL

Tập tin run\_crud.py thực hiện các thao tác CRUD và kiểm tra hiệu suất trên cơ sở dữ liệu SQLite bằng cách sử dụng SparkSQL.

File run\_crud.py chứa các hàm để thực hiện các thao tác CRUD (Create, Read, Update, Delete) với SparkSQL và cơ sở dữ liệu SQLite.

### 1. Cấu hình logging:

Cấu hình logging để ghi lại thông tin trong quá trình thực hiện các thao tác.

### 2. Hàm create\_spark\_session():

Tạo một SparkSession để làm việc với Spark. Đồng thời, cấu hình Spark để sử dụng JDBC driver cho SQLite.

### 3. Hàm connect\_to\_db():

Kết nối đến cơ sở dữ liệu SQLite và trả về kết nối.

### 4. Hàm create\_temp\_view():

Tạo một temporary view từ bảng people trong cơ sở dữ liệu SQLite để có thể thực hiện các truy vấn SQL với Spark.

### 5. Hàm read\_data():

Đọc và hiển thị 5 dòng dữ liệu cuối từ bảng people.

### 6. Hàm insert\_data():

Chèn thêm hai bản ghi mới vào bảng people.

### 7. Hàm update\_data():

Cập nhật tuổi của bản ghi có id = 1000001 thành 40.

### 8. Hàm delete\_data():

Xóa các bản ghi có id >= 1000001 bằng cách lọc và ghi đè lại bảng people.

### 9. Hàm performance\_test():

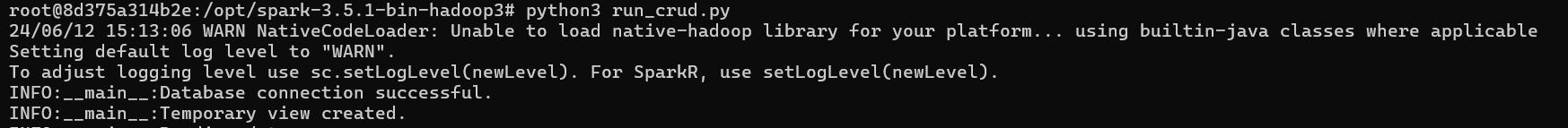
Đo lường thời gian thực hiện các truy vấn đếm tổng số bản ghi và đếm số bản ghi có age > 50.

### 10. Hàm show\_bottom\_rows():

Hiển thị 5 bản ghi cuối cùng của bảng people theo thứ tự giảm dần của id.

Sử dụng lệnh sau để chạy file run\_crud. Sau đây là kết quả khi chạy file:

***python3 /opt/run\_crud.py***



Thông báo về việc đã kết nối database và tạo temporary view.

A screenshot of a computer screen

Description automatically generated

Thông báo về việc đọc và in ra thông tin ở 5 dòng cuối.

A screen shot of a computer

Description automatically generated

Thông báo thêm thông tin và in 5 dòng cuối sau khi thêm thông tin.

A computer screen with white text

Description automatically generated

Thông báo về việc update thông tin về tuổi của id 1000001 và hiện thị thông tin sau thay đổi.

A computer screen shot of a black screen

Description automatically generated

Xóa 2 dòng cuối đã thêm vào và in ra 5 dòng cuối để kiểm tra.

A screen shot of a computer code

Description automatically generated

Kiểm tra thời gian count toàn bộ record và count có điều kiện WHERE và đóng kết nối server với database.