

Simple SQL Query Tool

Báo cáo đề tài thử thách

Dương Gia Bảo

TickLab

Ngày 5 tháng 10 năm 2024



- ➊ Mở đầu
- ➋ Timelines
- ➌ Kết luận

① Mở đầu

② Timelines

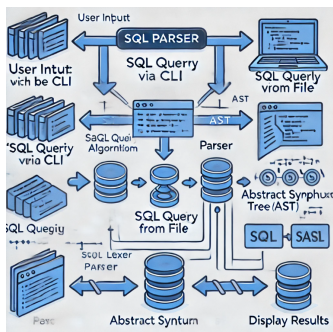
③ Kết luận

Mục tiêu:

- Xây dựng công cụ truy vấn CSV sử dụng cú pháp SQL
- Nâng cao khả năng lập trình với C++

Nội dung:

- Thiết kế và xây dựng chương trình exql
- Bộ phân tích cú pháp truy vấn
- Đọc và xử lý tệp CSV



1 Mở đầu

2 Timelines

SQL Lexer

SQL Parser

Handle file

3 Kết luận

1 Mở đầu

2 Timelines

SQL Lexer

SQL Parser

Handle file

3 Kết luận

Mục tiêu

Triển khai giao diện dòng lệnh (CLI) và SQL Lexer cho dự án.

Các bước thực hiện:

- Định nghĩa các token:
 - ✓ Tạo một liệt kê cho các loại token (từ khóa, định danh, literal, toán tử, dấu phân cách, khoảng trắng).
- Triển khai các hàm phân tích token
 - ✓ Từ khóa và Định danh: Phân tích các từ khóa SQL như SELECT, FROM và các định danh (tên cột và bảng).
 - ✓ Literal: Xử lý các literal kiểu chuỗi và số.
 - ✓ Xử lý các toán tử.

- Xử lý lỗi
 - ✓ Triển khai xử lý lỗi cho các token không được hỗ trợ hoặc không mong muốn.
 - ✓ Định nghĩa các loại lỗi và một lớp để theo dõi vị trí và loại lỗi trong truy vấn SQL.
- Kiểm thử
 - ✓ Kiểm thử lexer với các truy vấn SQL khác nhau để đảm bảo nó có thể phân tích đúng các token.
 - ✓ Đảm bảo lexer xử lý đúng các token hợp lệ và đưa ra lỗi thích hợp cho các đầu vào không hợp lệ.

1 Mở đầu

2 Timelines

SQL Lexer

SQL Parser

Handle file

3 Kết luận

Mục tiêu

Xây dựng trình phân tích cú pháp SQL để chuyển đổi luồng mã thông báo thành Cây cú pháp trừu tượng (AST), thể hiện cấu trúc của truy vấn SQL theo cách có ý nghĩa.

- Trong giai đoạn Lexer trước đó, chúng ta đã biến chuỗi truy vấn thô thành một chuỗi các token. Tuy nhiên, việc nhận dạng token không đảm bảo rằng chuỗi này là một truy vấn hợp lệ và có ý nghĩa.
- SQL Parser đảm bảo rằng chuỗi token tuân theo các quy tắc ngữ pháp của ngôn ngữ truy vấn CSV, xây dựng một biểu diễn cú pháp dưới dạng Abstract Syntax Tree (AST).

Các bước thực hiện:

- Định nghĩa cấu trúc dữ liệu cho AST:
 - ✓ SQL Statements.
 - ✓ Biểu thức (Expressions).
- Định nghĩa Parser:
 - ✓ Xây dựng các phương thức như `parse_statement()` và `parse_expression()`.
- Xử lý lỗi.
- Kiểm thử.

1 Mở đầu

2 Timelines

SQL Lexer

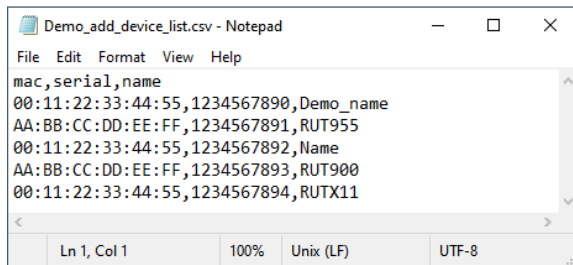
SQL Parser

Handle file

3 Kết luận

Mục tiêu

Xây dựng chức năng xử lý tệp cho SQL Parser, cho phép người dùng nhập truy vấn từ tệp và thực thi chúng một cách hiệu quả.



```
Demo_add_device_list.csv - Notepad
File Edit Format View Help
mac,serial,name
00:11:22:33:44:55,1234567890,Demo_name
AA:BB:CC:DD:EE:FF,1234567891,RUT955
00:11:22:33:44:55,1234567892,Name
AA:BB:CC:DD:EE:FF,1234567893,RUT900
00:11:22:33:44:55,1234567894,RUTX11
Ln 1, Col 1 100% Unix (LF) UTF-8
```

Các bước thực hiện:

- Định Nghĩa Chức Năng Đọc Tập.
- Xử Lý Tập Đầu Vào.
- Phân Tích Truy Vấn:
 - ✓ Sau khi đọc được nội dung từ tập hoặc chuỗi đầu vào, sử dụng Parser để phân tích truy vấn và xây dựng AST. Đảm bảo xử lý các trường hợp lỗi nếu có vấn đề trong quá trình phân tích.
- Xử Lý Kết Quả.

- 1 Mở đầu
- 2 Timelines
- 3 Kết luận

Hoàn thành:

- Xử lý Lexer
- Phân tích Parser Statement
- Handle file

Chưa hoàn thành:

- Phân Tích Parser Expression
- Handle Error trong Parser
- Xử Lý Parser Statement trên nhiều dòng

Thank You