

데이터베이스 프로젝트 보고서 1-3

2016-11464 정재훈

1. 핵심 모듈과 알고리즘에 대한 설명

이제 우리는 table에 데이터를 가지고 있다. 그리고 char string이 특수문자들을 포함하므로, delimiter를 잘 정의할 필요성이 있다. 따라서 char string에 쓰이지 않는 “(double quote)를 delimiter로 삼아서 query string을 만들었고, 이를 기준으로 parse하였다. insert의 경우 별도의 알고리즘은 필요 없고 reference 하고 있는 table들을 순회하는 것만 주의하면 쉽게 처리할 수 있었고, delete나 select의 경우 where clause를 포함하는데, 이가 or-and-not-괄호 순의 tree 구조를 이루고 있어서 이를 기준으로 parse하고 괄호 안의 query의 경우 재귀함수를 통해 해결하였다. 이 과정에서 수시로 and/or 의 shortcut을 사용 가능한 record들을 확인하여 그들은 query를 evaluate 하는 과정을 생략하도록 구현하였다. delete의 referential constraint의 경우 insert와 비슷하게 현재 table을 참조하고 있는 모든 table들을 순회하며 데이터를 필요 시 null로 대체하였다.

2. 구현한 내용에 대한 간략한 설명

INSERT에서는 values 앞에 column을 지정하는 경우와 그렇지 않은 경우 모두 받되, 지정하지 않은 경우에 대해서는 개수가 맞지 않으면 error을, 지정하는 경우에 개수가 전체 column의 개수와 다르면 없는 행에 null을 자동 삽입하도록 구현하였으며, primary key 와 foreign key 관련 위배되는 사항이 존재하나 확인하여 존재하면 error을 출력하였다. DELETE에서는 where 절이 없으면 전체를, where 절이 있다면 해당되는 부분만을 지우려고 시도하는데, 이 과정에서 foreign key 관련 위배되는 사항이 존재하면 그 행을 지우지 않으며, 지워진다면 해당 데이터를 쓰는 테이블의 그 데이터를 null로 바꿔주고 지운다. SELECT에서는 * 가 들어오면 전체 column을 아니면 선택된 것들만 출력하며 where 관련은 DELETE와 같게 구현하였다. AS 로 다른 이름으로 바꾼 경우 column은 해당 이름으로 출력하며 table은 그 이름으로 where문에 입력해주었다.

3. 구현하지 못한 내용

(5/13일자 etl 게시판의 질문 답변 사항 || 과제에서 명시된) 모든 사항을 구현함

4. 가정한 것들

명시되지 않았지만 사람에 따라 오류라고 판단 가능한 사항(as 뒤의 table이 duplicated 등)은 오류가 아니고 사용자의 의도라고 가정하였다. select문에서 table을 a as b로 선언하면 where 문에서는 b.xxx와 xxx 만 사용 가능하고 a.xxx 는 input으로 들어오지 않는다고 가정하였다. (SQL 실습 사이트의 동작과 동일) insert, delete, select의 가정은 1-3 과제 스펙 문서의 가정을 따라 정의하였다(동일한 column은 insert의 입력으로 들어오지 않는다, unknown 연산 원리 등) select로 출력할 때 출력 양식은 MySQL을 따른다고 가정하였다. +9와 9는 insert로 입력 받을 때 같은 데이터라고 가정한다, ‘a’ 와 ‘a ’도 같다 가정한다.

5. 컴파일과 실행 방법

executable jar 파일을 실행한다. 단, jar 파일 옆에 db directory가 있어야한다.

(command line에서 `java -jar 파일이름`)

6. 프로젝트를 하면서 느낀 점

SQL의 data를 manipulate하는 과정에서 생각보다 많은 error들을 control 해야 한다는 것을 알 수 있었고, 주어진 문제 조건이 다양한 error 처리 경우를 갖는 경우 코딩이 얼마나 시간이 많이 걸리고 힘든 것인지 알 수 있었다. 주어진 문제를 해결하는 과정에서 testcase를 직접 만들어보면서 생각지도 못한 error들이 많다는 것을 알 수 있었고, 질문 게시판을 보면서 혼자서는 수많은 error들을 다 잡아내기 힘들어서 협력이 필요하다는 것을 깨달았다. INSERT, DELETE, SELECT 세 구문을 처리하는 데에 나흘 동안 40시간 이상을 할애했는데 아직도 error로부터 완전하다고 자신할 수 없으며, 이렇듯 간단한 sql 구현 parser를 만드는 것만으로도 오랜 시간이 걸렸는데 MySQL 등의 완전한 sql을 구현한 분들이 대단하다.