

DB 프로젝트 1-2 보고서

2015-11923 최한결

본 프로젝트에서는 1-1에서 만든 DBMS parser에 DDL 명령어를 수행할 수 있도록 기능을 추가하였다.

개발 환경

OS: MacOS catalina

IDE: Eclipse 4.15.0

핵심 모듈과 알고리즘에 대한 설명

Table에는 table의 이름과 Column들, Constraints Definition(primaryKeys, foreignKey), 테이블을 참조하는 다른 테이블명을 저장하는 referencedBy가 존재한다. 또 Column에는 column의 이름, datatype, null값이 가능한지를 나타내는 nullable, primary key임을 나타내는 pk, foreign key일 경우에 reference 하는 table이름,column이름을 나타내는 reference 가 있다.

Create Table시에 query가 완성되기 전까지 새로운 테이블을 만들고 입력 값에 따라 tableName, Column, Constraint Definition들을 새로 만든 테이블에 저장한다. 이 후 query가 완성되면 Constraint Definition (Table.primaryKeys, Table.foreignKeys)에 따라 각 Column.pk를 true로 바꾸고 Column.reference를 업데이트한다. 이 때 오류가 생길 경우 만들어 놓은 table은 db에 저장되지 못하고 사라진다.

구현 내용

Query가 완성되기 전에 데이터 조작이 일어나지 않게 하기 위해 모든 조건들을 저장해 두었다가 Query가 완성된 후 그 조건들을 사용해 테이블을 조작하는 방식이 필요했다. 따라서 기본적으로 세미콜론을 각 Query(Create table, drop table ...) Token으로 확인하고 명령의 완결을 확인하였고 입력 값들을 raw하게 저장 후 query가 완성된 후 데이터 무결성이 깨지지 않는 경우 저장한다.

Drop table 명령에서는 참조 무결성을 유지하기 위해 테이블마다 자신을 참조하고 있는 테이블들을 저장해 두고 create table 혹은 drop table이 일어날 때 각 table이 reference하고 있는 table들의 referencedBy를 업데이트해 참조 무결성이 깨지지 않도록 하였다.

가정한 것들

1. 모든 공백은 줄바꿈으로 대체 가능하다고 가정하였다.
Ex)
create
table
~ 와 같은 입력이 정상적으로 수행된다.
2. 각 query가 완성됨과 동시에 query가 수행되고 결과의 출력 또한 query가 완성됨과 동시에 일어난다고 가정한다.

Ex1)

```
> create table person(name char(10), age int, birthday date, primary key (name, age))
```

```
> abc;
```

와 같은 입력에서는 Syntax Error 가 일어나고 table이 완성되지 않는다.

Ex2)

```
DB_2015-11923> Desc
```

```
> a; show
```

```
-----
```

```
Table_name [a]
```

```
...
```

```
-----
```

```
> tables; desc
```

```
-----
```

```
a
```

```
b
```

```
-----
```

```
> b;
```

```
-----
```

```
table_name [b]
```

```
...
```

```
-----
```

```
DB_2015-11923>
```

컴파일과 실행 방법

PRJ1-2_2015-11923.jar가 들어있는 폴더에 db라는 하위 폴더를 생성한 후 java -jar PRJ1-2_2015-11923.jar 명령어를 통해 실행할 수 있다.

느낀 점

데이터 무결성을 유지하기 위해 엄청 많은 조건 확인이 필요했고 힘든 작업이었다. 하지만 위 조건 확인을 안했을 때는 더욱 알 수 없는 버그들이 나올 수 있다는 것을 확인했고 DBMS에 필수적인 부분이라고 생각했다.