

# Project 1-2: Implementing DDL

**Due: 2020/05/13 (Wed), 11:59 PM**

이번 프로젝트의 목표는 프로젝트 1-1에서 구현한 SQL 파서 프로그램에 기능을 추가하여 스키마를 저장하고 스키마에 접근할 수 있도록 하는 것이다.

구현한 프로그램은 4개의 DDL 구문(create table, drop table, desc, show tables)에 대해 처리할 수 있어야 한다.

테이블 스키마는 파일에 저장되어 프로그램이 종료되어도 사라지지 않아야 한다. 저장에는 Berkeley DB를 이용한다.

프로젝트 1-2의 코드를 이용하여 프로젝트 1-3에서 실제로 레코드를 삽입하고 삭제하는 기능을 구현할 예정이므로, 이를 고려하면서 구현하도록 한다.

## 1. 요구 사항

- 프로젝트 1-1에서 구현한 SQL 파서 프로그램을 이용하여야 한다.
- 2장에 나열된 모든 DDL 구문들을 처리할 수 있어야 한다.
- 테이블 스키마를 파일(단일 파일 혹은 여러 개의 파일)에 저장하여야 한다.
  - DBMS 콘솔을 종료한 후 다시 실행하더라도 테이블 스키마가 남아 있어야 한다.
  - 테이블 스키마 저장에는 Berkeley DB를 이용한다.

2장에는 실행 예시 및 각 상황 별로 출력해야 하는 메시지의 종류가 나열되어 있다. 메시지의 내용은 메시지 정의 파일(2020\_1-2\_Messages.docx)을 참조한다.

## 2. SQL

### 2.1 CREATE TABLE

- 실행 예시

```
DB_2020-12345> create table account
(
    account_number int not null,
    branch_name char(15),
    primary key(account_number)
);
'account' table is created
```

- 입력한 쿼리가 올바르다면, 테이블 정보를 저장하고 CreateTableSuccess(#tableName)에

해당하는 메시지를 출력한다.

- 입력한 쿼리에 오류가 있다면, 오류에 맞는 적절한 에러 메시지를 출력한다.
  - 컬럼의 이름이 중복될 경우, `DuplicateColumnDefError`에 해당하는 메시지 출력
  - Primary key 정의가 여러 번 나온 경우, `DuplicatePrimaryKeyDefError`에 해당하는 메시지 출력
  - Foreign key의 타입과 foreign key가 참조하는 컬럼의 개수나 타입이 서로 다른 경우, `ReferenceTypeError`에 해당하는 메시지 출력
  - Foreign key가 primary key가 아닌 컬럼을 참조한다면, `ReferenceNonPrimaryKeyError`에 해당하는 메시지 출력  
(Foreign key가 composite primary key의 일부만을 reference하는 경우에도 `ReferenceNonPrimaryKeyError`에 해당하는 메시지 출력)
  - Foreign key가 존재하지 않는 컬럼을 참조한다면, `ReferenceColumnExistenceError`에 해당하는 메시지 출력
  - Foreign key가 존재하지 않는 테이블을 참조한다면, `ReferenceTableExistenceError`에 해당하는 메시지 출력
  - 존재하지 않는 컬럼을 primary key로 정의한 경우, `NonExistingColumnDefError(#colName)`에 해당하는 메시지 출력
  - 존재하지 않는 컬럼을 foreign key로 정의한 경우, `NonExistingColumnDefError(#colName)`에 해당하는 메시지 출력
  - 이미 같은 이름의 테이블이 존재할 경우, `TableExistenceError`에 해당하는 메시지 출력
  - char 타입의 길이를 1보다 작게 지정한 경우, `CharLengthError`에 해당하는 메시지 출력
  - 본 문서에서 정의된 오류 상황 이외의 오류 상황이 있다고 판단될 경우 그 오류의 이름과 메시지를 직접 정의하고 보고서에 명시할 것 (단, 채점은 본 문서에 나오는 오류 상황을 제대로 처리했는지만 볼 것임. 즉 본 문서에 정의되지 않은 상황에 대해서는 채점하지 않음)
- 테이블을 생성할 때에는 다음과 같은 가정을 따른다.
  - Primary key로 지정된 컬럼은 따로 not null로 지정하지 않더라도 자동적으로 not null인 것으로 본다.
  - null 값을 가질 수 있는 컬럼은 primary key가 될 수 없다.
  - Foreign key는 다른 테이블의 primary key를 참조하여야 한다.
  - Foreign key와 foreign key가 참조하는 컬럼의 타입은 서로 같아야 한다.

- ◆ char 타입의 경우, 길이가 다르다면 서로 다른 타입으로 본다.
- null 값을 가질 수 있는 컬럼도 foreign key가 될 수 있다.
- 하나의 테이블은 여러 개의 foreign key를 가질 수 있다.
- Foreign key는 자신과 같은 테이블에 있는 컬럼을 참조할 수 없다.
- char 타입의 길이는 0보다 커야 한다.
- 테이블 이름과 컬럼 이름은 대소문자를 구분하지 않는다. (case insensitive)
- Foreign key constraint에 대한 자세한 설명 및 스펙은 (About Referential Integrity Constraints.pdf) 문서를 참고

## 2.2 DROP TABLE

- 실행 예시

```
DB_2020-12345> drop table account;
'account' table is dropped
```

- 입력한 쿼리가 올바르다면, 테이블 정보를 삭제하고 DropSuccess(#tableName)에 해당하는 메시지를 출력한다.
- 입력한 쿼리에 오류가 있다면, 오류에 맞는 적절한 에러 메시지를 출력한다.
  - 테이블이 존재하지 않을 경우, NoSuchTable에 해당하는 메시지 출력
  - 다른 테이블이 참조하고 있는 테이블을 삭제하려고 할 경우, DropReferencedTableError(#tableName)에 해당하는 메시지 출력

## 2.3 DESC

- 실행 예시

```
DB_2020-12345> desc account;
-----
table_name [account]
column_name      type           null          key
account_number   char(10)       N             PRI
branch_name      char(15)       N             PRI/FOR
balance          int            Y
```

- 테이블의 스키마 정보를 출력한다.
  - 컬럼 이름, 타입, null 값 허용 여부, key 정보(primary key, foreign key)를 포함하여야 함
  - 출력 형식은 위와 같아야 함 (단, 각 필드 사이의 공백은 원하는 대로 정의할 것)
- 테이블이 존재하지 않는다면 NoSuchTable에 해당하는 메시지를 출력한다.

## 2.4 SHOW TABLES

- 실행 예시

```
DB_2020-12345> show tables;
```

```
-----  
branch  
customer  
loan  
borrower  
account  
depositor  
-----
```

- DB에 존재하는 모든 테이블의 이름을 출력한다.
- 테이블이 존재하지 않는다면 ShowTablesNoTable에 해당하는 메시지를 출력한다.

### 3. 개발 환경

- Java
- Eclipse
- Oracle Berkeley DB API
  - ◆ Berkeley DB는 데이터를 key-value의 pair로 저장함
  - ◆ SQL API를 통하여 데이터에 접근하면 안됨! (직접 구현하여야 함)

### 4. 제출

1. Executable jar 파일
  - 파일명: PRJ1-2\_학번.jar (예: PRJ1-2\_2020-12345.jar)
2. Eclipse 프로젝트
  - 소스 파일은 반드시 적절한 주석을 포함하여야 함
  - 프로젝트 이름: PRJ1-2\_학번 (예: PRJ1-2\_2020-12345)
3. 리포트
  - 파일명: PRJ1-2\_학번.docx (예: PRJ1-2\_2020-12345.docx)
  - 반드시 포함되어야 하는 내용
    - ◆ 개발 환경(os, ide version등)
    - ◆ 핵심 모듈과 알고리즘에 대한 설명
    - ◆ 구현한 내용에 대한 간략한 설명

- ◆ (제시된 요구사항 중 구현하지 못한 부분이 있다면) 구현하지 못한 내용
- ◆ 가정한 것들
- ◆ 컴파일과 실행 방법
- ◆ 프로젝트를 하면서 느낀 점
- 위의 3가지 파일을 압축하여 etl Project1-2로 제출
  - 파일명: PRJ1-2\_학번.zip (예: PRJ1-2\_2020-12345.zip)

## 5. 성적 관련 사항

- 제출 기한 이후 24시간 이내 제출시 10% 감점
- 제출 기한 이후 24시간 이후 48시간 이내 제출시 20% 감점
- 제출 기한 48시간 이후에는 점수 없음
- 부정 행위는 0점 처리
  - ◆ 다른 사람의 코드를 참조하는 행위
  - ◆ 이전에 수강한 사람의 코드를 참조하는 행위
  - ◆ 제출한 소스코드에 대해 표절 방지 프로그램을 돌릴 예정
- 본 문서에 명시되어 있는 출력 양식을 지키지 않을 시 감점
- Jar 파일 미작동시 감점

## 6. References

- Oracle Berkeley DB
  - <http://www.oracle.com/technetwork/database/berkeleydb/overview/index.html>

\*프로젝트에 관한 전반적인 것 관련 문의는 마감 전 월요일(5/11)까지 받습니다.

\*또한 프로젝트 전반적인 것 외 본인의 코드 구현에 따른 문의는 형평성을 위해 답변이 원칙적으로 불가능합니다.