Database Project 1-3 Report

이선재 2020-16634

Main Additions Since Project 1-2

- Evaluation of WHERE clause, shared for DELETE and SELECT queries
- Check for integrity constraints in DELETE and INSERT queries

Algorithm for WHERE Clause Evaluation

1. Retrieve All Records:
    - Retrieve all records from the specified table or joined records from a list of tables.
    - Initialize an empty list to store records that match the WHERE clause conditions.
2. Extract and Validate WHERE Clause:
    - Check if a WHERE clause is provided in the SQL query.
    - If no WHERE clause is provided, all records from the table are marked for deletion or selection.
    - If a WHERE clause is provided, extract the conditions from the WHERE clause using the `extract_conditions` function.
3. Validate Conditions:
    - Iterate through the extracted conditions and validate each condition using the `validate_condition` function.
    - Ensure that table and column references are correct and that comparisons are valid.
4. Evaluate Conditions Against Records:
    - Iterate through the initial records retrieved from step 1.
    - For each record, evaluate the conditions using the `evaluate_conditions` function.
    - If a record meets all conditions, add it to the collecting list.

Discussion

The biggest challenge in project 1-3 was designing the algorithm for the WHERE clause as there are so many cases and combinations to consider, as well as the fact that this is shared by DELETE and SELECT. I first designed the WHERE clause algorithm while working on the DELETE query, but I had to go back and make changes to the algorithm when I later worked on the SELECT query.

Furthermore, I think this project provided a good opportunity to review comparison predicates and three value logic. As I went through test cases, I got a more solid understanding of confusing cases involving null, such as the result of `SELECT * from students where name != ‘John’` not including records with null values for name as the corresponding comparisons would evaluate to UNKNOWN.

Custom Errors (carried over from Project 1-2 Report)

Message Type	**Message**	**Example Case**
ReferenceColumnCountMismatchError	Create table has failed: number of referencing columns does not match number of referenced columns	Create table student(id int, foreign key(id) references person(id, age));
ReferenceTableSelfError	Create table has failed: foreign key cannot reference its own table	Create table student(id int, foreign key(id) references student(id));
InsertTableDuplicateColumnError	Insert has failed: column name is duplicated	Insert into person (id, id) values (101, 201);