C++ SQL Parser

This is a SQL Parser for C++. It parses queries given as strings into C++ structs. From here you may view the details of your query as well as edit them.

Usage

Statement Type:

Any string can be converted into a ‘statement’ and then converted into it’s applicable statement type.

Ex. statement stmt = statement("DROP TABLE tab;");

dropStmt stmt = dropStmt(stmt);

DROP Statement

dropStmt stmt = dropStmt(“DROP TABLE tab;”)

Modifiable parameters:

1. tableName
   * stmt.getTableName();
   * stmt.setTableName(“newTableName”);

Print Function (Returns a string of the full query):

* stmt.printDropStmt();

TRUNCATE Statement

truncateStmt stmt = truncateStmt(“TRUNCATE TABLE tab;”)

Modifiable parameters:

1. tableName
   * stmt.getTableName();
   * stmt.setTableName(“newTableName”);

Print Function (Returns a string of the full query):

* stmt.printTruncateStmt();

ALTER Statement

alterStmt stmt = alterStmt("ALTER TABLE IF EXISTS ONLY tab1 \* RENAME COLUMN col1name TO col2name;")

Modifiable parameters:

1. tableName
   * stmt.getTableName();
   * stmt.setTableName(“newTableName”);
2. Might need more

Print Function (Returns a string of the full query):

* stmt.printAlterStmt();

CREATE Statement

createStmt stmt = createStmt("CREATE TABLE tab AS TABLE old\_tab WITH NO DATA;")

Or

createStmt stmt2 = createStm("CREATE TABLE tab (nums INT, names VARCHAR, circ circle);"

Modifiable parameters:

1. tableName
   * stmt.getTableName();
   * stmt.setTableName(“newTableName”);
2. colVec

* stmt2.getColVec();
* stmt.setColVec(vector<columnDets> ex);

Usage Ex:

vector<columnDets> cv = stmt2.getColVec(); //Get the vector of columnDets

columnDets colDets1 = cv[0]; //Set an empty columnDets to the index you want to change (this will allow the existing parameters to persist)

colDets1.setColName("newNums"); //Set the new colName

colDets1.setColData("INT8"); //Set the new colData

columnDets colDets2 = cv[1]; //Set another columnDets to next index

colDets2.setColData("TEXT"); //Set new colData (not necessary to change both name and data)

stmt2.setColVec({colDets1, colDets2, cv[2]}); //Set the new colVec by inputting a vector of columnDets

Print Function (Returns a string of the full query):

* stmt.printCreateStmt();

INSERT Statement

insertStmt stmt = insertStmt("INSERT INTO tab(nums, title) VALUES ('105', 'Banana');")

Modifiable parameters:

1. tableName
   * stmt.getTableName();
   * stmt.setTableName(“newTableName”);
2. colNames

* stmt.getColNames();
* stmt.setColNames(vector<string> ex);

1. colValues

* stmt.getValues();
* stmt.setValues(vector<string> ex);

Usage Ex.

vector<string> newVec1={"newNums", "newTitle", "thirdCol"}; //Create string vector

inEditTest1.setColNames(newVec1); //Set colNames

vector<string> newVec2={"'999'", "'The banana'", "'idk'"}; //Create string vector

inEditTest1.setValues(newVec2); //Set values

Print Function (Returns a string of the full query):

* stmt.printInsertStmt();

DELETE Statement

deleteStmt stmt = deleteStmt("DELETE FROM ONLY tab WHERE nums <> 101;")

Modifiable parameters:

1. tableName
   * stmt.getTableName();
   * stmt.setTableName(“newTableName”);
2. WhereEquation
   * stmt.getWhere(); //Returns the where equation as type Equation
   * stmt.setWhereLeft(“newNums”);
   * stmt.setWhereSymbol(“=”);
   * stmt.setWhereRight(“10”);

Print Function (Returns a string of the full query):

* stmt.printDeleteStmt();

SELECT Statement

selectStmt stmt = selectStmt("SELECT \* FROM tab;")

Modifiable parameters:

1. tableName
   * stmt.getTableName();
   * stmt.setTableName(“newTableName”);
2. Select from data (\* in above example)
   * stmt.getFrom();
   * stmt.setFrom(“name”);

Print Function (Returns a string of the full query):

* stmt.printSelectStmt();

UPDATEStatement

updateStmt stmt = updateStmt("UPDATE ONLY tab1 \* AS tabby SET col1 = 'name1' WHERE col1 = 'name5';")

Modifiable parameters:

1. tableName
   * stmt.getTableName();
   * stmt.setTableName(“newTableName”);
2. Set Equation (col1=’name1’)
   * stmt.getSet();
   * stmt.setSetLeft(“newCol”);
   * stmt.setSetSymbol(“=”);
   * stmt.setSetRight(“‘notName5’”);
3. Where Equation (col1=’name5’;)
   * stmt.getWhere();
   * stmt.setWhereLeft(“newCol”);
   * stmt.setWhereSymbol(“=”); //Should not change this
   * stmt.setWhereRight(“‘name2’”);

Print Function (Returns a string of the full query):

* stmt.printUpdateStmt();

Helper Structs:

Equation - Used for equations such as xxx = yyy

ColumnDets - Used for storing detailed column information

Limitations:

Everything not listed above