COSC3380
HW1: SQL Queries
Due Sep/24 7pm

1 Introduction

You will develop SQL queries to answer information requests written in natural language (plain English). That is, you need to translate the request into a query that returns the desired result. Requests can be understood as questions on the database content.

2 Input and output

The input is a relational database consisting of several tables. Most tables are normalized. The output for each query will be a table, as returned by SQL.

Bookings database: The specific database will contain ticket and booking information for an airline in Russia. The connection and schema information will be explained in the newsgroup.

3 Output

You need to produce one table storing the query result. The table name and column names will depend on the query and will be fixed to enable testing. That is, you can create any intermediate results or tables with any column names, but the final table should have exactly the same column names.

4 Requests

The precise list of requests is posted in the newsgroup to avoid misunderstandings. Some of the wording has been changed to improve clarity.

5 Requirements

- Programming language: SQL. For this homework no other language is necessary (e.g. Python, Java).
- Relational operations: union, intersection, selection, projection and join. Projection includes GROUP BY aggregation.
- You will store query results in one output table. The table name and column names will be specified in the newsgroup by the TAs. Name discrepancies will result in failing tests. Also, avoid including extra columns not requested in the query. Sorting rows in the output tables is unnecesary.

You can create temporary tables or you can solve the query with a single query and nested SELECTs. You can create temp tables to solve each set operation step by step.

- Store each query in one file.sql in a folder. Names and folder name will be specified by the TAs.
- SQL code must have comments, including your name.
- SQL must be indented, following the style from the textbook.
- If you find some requirement difficult and you do not implement it you can include a comment in your README file explaining why.
- Correctness is the most important requirement: TEST your program with many expressions. Your program should not crash or produce exceptions.
- Query execution: Your queries must be correct (syntax, valid tables, valid columns).
- Folder and file name to be specified by TA. Example: folder: /hw1 and queries q01.sql, q02.sql, ...