

# 5DV119 Introduction to Database Management Spring 2015

## Obligatory Exercise 1

Submission deadline to avoid lateness penalty: February 02, 2015 at 0800 (8am)

**Note:** The submission deadline for this assignment has been extended by 24 hours, to February 03, 2015 at 08:00, due to delays in making the submission system available.

## 1 Problem Statement

Given is the following relational database schema:

Airport(Code, City, Country, Latitude, Longitude)

Flight(**Airline**, FlightNumber, **Origin**, **Destination**)

Schedule(**Airline**, **FlightNumber**, Date, DepartureTime, ArrivalTime)

Airline(Name, Abbreviation, HomeCountry, WebSite)

Ticket(Number, **Airline**, **FlightNumber**, **Date**, Cost)

In the above schema, note the following conventions:

- The primary key of each relation is underlined.
- Foreign keys are shown in **boldface**. Specifically:
  - The attribute **Airline** in the Flight relation is a foreign key which references the Airline relation.
  - The attributes **Origin** and **Destination** in the Flight relation are each foreign keys which reference the Airport relation.
  - The attribute pair (**Airline**, **FlightNumber**) in the Schedule relation is a foreign key which references the Flight relation.
  - The attribute triple (**Airline**, **FlightNumber**, **Date**) in the Ticket relation is a foreign key which references the Schedule relation.

If possible, find a solution, in SQL, to each of the queries in Section 2 which works with both PostgreSQL and MySQL. If necessary (but only if so), give two solutions, one for PostgreSQL and one for MySQL. If two separate solutions are provided, a comment (in the SQL source code) as to why that was necessary must be provided.

Exception: for queries 3 and 7, it is not necessary to give a solution which works with MySQL (because the techniques necessary to solve these using MySQL will not yet been covered in the lectures one week before this assignment is due).

Revised to include submission information: January 29, 2015

## 5DV119, Obligatory Exercise 1, page 2

Unless stated specifically to the contrary, each solution must eliminate all duplicates from the result and must provide meaningful column names. Furthermore, to keep the solutions simple and consistent, the following two restrictions apply:

- (i) The solution must consist of a single SQL directive. Creation and subsequent use of temporary tables is not permitted.
- (ii) Subqueries involving `SELECT` are allowed only in the `WHERE` and `HAVING` fields of queries and subqueries, and in the `SET` fields of `UPDATE` commands.

## 2 Queries to Be Solved

1. Find the code, latitude, and longitude of each airport which located in a city named Paris or Berlin. (This includes all cities named Paris or Berlin, regardless of the country.)
2. Find the names of those airlines which have either a flight with origin the airport with code TXL or a flight with origin the airport with code SXF.
3. Find the names of those airlines which have both a flight with origin the airport with code TXL and a flight with origin the airport with code SXF. (Hint: This may be done without using embedded subqueries by using the `INTERSECT` directive. Since MySQL does not support this directive, it is sufficient to give a solution which works only with PostgreSQL.)
4. Find the names of all airlines which have a scheduled flight on the date 2015-11-12.
5. Find the airport code, city, and country for all airports which have a departure for which some ticket with ticket number less than 200 costs more than 4000.
6. Find the names of those airlines which have a flight whose destination is an airport which is located in Germany or France.
7. Find the codes of those airports located in Berlin, Germany which do not have any scheduled departures. (Hint: This may be done without using embedded subqueries by using the `EXCEPT` directive. Since MySQL does not support this directive, it is sufficient to give a solution which works only with PostgreSQL.)
8. Find the names of those airlines with either the string “Air” or else the string “Luft” (both case insensitive) in their names.
9. Reduce the price of all tickets issued by Scandinavian airlines for the interval 2015-11-01 to 2015-11-20 inclusive by 20%. (Note: You may use the code “SK” in your query. It is not necessary to pattern match on “Scandinavian”)
10. Find the flights for the airline with code SK which are scheduled on 2015-11-12 with a departure time before noon (12:00). In addition to the flight number, give the airport codes for both the origin and the destination.

### 3 Submission Rules

1. SQL code which generates a test database for this exercise is available on the course Web page.
2. Unless stated specifically to the contrary, all solutions must be valid under both PostgreSQL and MySQL, using the database servers of the Department of Computing Science. This means the server `postgres` for PostgreSQL and the server `mysql` for MySQL. If you develop your solutions on other systems, make sure to test them on the database servers of the Department of Computing Science, as different versions of PostgreSQL and MySQL may support different features.
3. The SQL code for the solutions of each of the ten queries must be placed in two text files, one which runs under PostgreSQL and the other which runs under MySQL. These files must be runnable as input source to both PostgreSQL (using the `\i` directive) and MySQL (using the `\.` directive), respectively. This means in particular that all comments must be in SQL-compatible format. The solutions must be in order, from Query 1 to Query 10. For each query, the following three items must be given, in order.
  - (a) A comment which states the query, in English.
  - (b) A line of the form `SELECT 'Query n';`, with `n` the number of the query.
  - (c) The SQL solution to the query.

As an example, for a fictitious Query 11:

```
-- Find the names of all airlines which are located
-- in Bohemia.
SELECT 'Query_11';
SELECT Name FROM Airline WHERE (HomeCountry='Bohemia');
```

The purpose of the second line is to indicate the query number in the output when the file is run as input source.

4. Solutions must be uploaded using the submission system found at the following URL:  
<https://www8.cs.umu.se/kurser/5DV119/VT15/handin/>  
Note the following regarding this system:
  - (a) The system will prompt for three items: a file `postgresql.sql`, a file `mysql.sql`, and a list of group members.
  - (b) The corresponding files on your system are selected via a Web-based file browser. The names of the files on your system need not match those names; the uploaded files will be renamed automatically.
  - (c) The ID of the user who is uploading the submission is included automatically in the list of group members, and so need not be repeated (although it causes no harm to do so). However, the IDs of the other group members must be given with the first submission.

## 5DV119, Obligatory Exercise 1, page 4

- (d) The ID of each group member **must** be the user ID of that person at `cs.umu.se`; that is, the `xxx` part of the e-mail address `xxx@cs.umu.se`.
  - (e) All group members **must** be registered for the course; otherwise, the system will not accept the submission.
  - (f) Once a submission is made, new group members may not be added for this assignment.
  - (g) The system does some basic syntax checking of the submission. Thus, it is strongly recommended that the initial submission be made ahead of the deadline, so any problems in format may be fixed without the risk of a lateness penalty.
  - (h) Resubmissions may be made at any time, but those submitted after the deadline will be flagged as late.
  - (i) Whether or not a submission has been made for a particular student may be checked using the *Labres* system. A link will be provided on the course home page.
5. No report or file containing results is necessary, and paper submission is neither required nor possible. It suffices to submit the two solution files electronically.
  6. Remember that a correct solution must work for all instances of the database, and not just for the test database provided.
  7. This exercise may be done either individually or in a group consisting of no more than four (4) individuals.

## 4 Further Notes

- Remember that there are point penalties for late submission. See the course syllabus.
- It is not allowed to copy the work of others. The submission must be the original work of the individual(s) in the working group.
- The grader reserves the right to interview members of the working group about the solution.
- So that solutions may be discussed in a class meeting, students and/or groups that are very late in preparing a solution may be required to solve an alternate problem to receive credit for this exercise.