# Databases – Exercise 5 : DDL and SQL

Due date: Thursday 8.12.16

Exercise submission:

- In this exercise you should create a single zip file – ex5.zip – out of the files of Question 1 and Question 2, and submit the zip file via ex5 submission link on the course homepage, before 23:55.
- You should name the submitted files, the tables, and each attribute in each table, **exactly** as instructed in each question. You should not submit the files in folders.

## Question 1 (30 points):

Consider the following relational schema. Trips have tnum which identifies them, a location where the trip is (string), duration in days (integer), and difficulty (integer). Tourism companies have cid that identifies them, a name (string), and the year of establishment (integer). The relation Offer contains pairs of cid,tnum such that the company with that cid offers the trip with number tnum (a company can offer several trips, and several companies can offer the same trip).

- Trip (tnum: integer, location: string, duration: integer, difficulty: integer)
- Company (cid: integer, name: string, estYear: integer)
- Offer (cid, tnum)

Difficulty is a number between 1 and 7. Duration has default value of 5. Duration is at least 3. Trips of duration more than 6 days have difficulty of at least 3. All the fields are always known except for difficulty.

a) Write DDL statements in a file **create.sql** that creates these tables. In your DDL commands, make sure to include any constraints (key, foreign key, check, etc.) that should be defined over the tables.

b) Write DDL statements in a file **drop.sql** that drop these tables.

Running create.sql and then drop.sql should cause all tables created to be dropped. Note that we are not requiring you to insert any tuples into the tables in this exercise.

## Question 2 (70 points):

Consider the following relations:

- Hobby (hnum, hname)
- Person (pid, pname, age)
- Hobbies (pid, hnum, frequency)
  [frequency is a positive integer representing the number of days that the person with pid does the hobby with hnum during a week]

Write the following queries in SQL. Pay careful attention to these guidelines, as this exercise will be tested automatically:

1. Write each query in a separate file named **querya.sql**, **queryb.sql**, etc.
2. In all queries, **remove** duplicates and **order** the result, ascending, by the first output column.
3. Do not use views to define your queries.
4. To test your queries, you should create the tables and insert tuples. However, only submit the files of your queries.
5. You may assume that the three tables are not empty, and that none of them contains NULL values.

The queries:

a) Find all the names of the persons who have both 'Cooking' and 'Volleyball' as their hobbies.

b) Find all the hobbies (hname) of teenagers (ages between 12 and 19 including), in other words hobbies of at least one teenager.

c) Find the names of the persons who have all the hobbies Idan has.
   - In case Idan is not in the database or doesn't have time for hobbies, the query answer should be all the persons in Person.
   - You may assume there is at most one person named Idan in the database.

d) Find all the hobbies (hname) that are done at least 3 times a week by all the persons with that hobby (hobbies in the database that don't appear in Hobbies are always included in the query answer).

e) Find the names of persons who have more than two hobbies.

f) Find the hobby (hname) with the minimal average of ages of the persons with that hobby (in case there are more than one hobby with minimal average of ages – show them all).

g) For each hobby (hnum) in Hobbies, find the name of the oldest person who does the hobby, and the name of the youngest person who does the hobby. If there are several oldest/youngest people, return the person whose name is first, lexicographically.

Good luck!