Databases 1 Exam

Name:
Neptun code:

Copy your solutions into this file and send it to the following email address: <u>nikovits@inf.elte.hu</u>. You should send the **results (output) of the queries** too. If you do not send the output, I will give you much fewer points!!! You can use ARAMIS or ULLMAN database.
Each exercise counts 10 points, so altogether you can get 60 points.

Grades are the following: 2 -> 20 points, 3-> 30 points, 4 -> 40 points, 5 -> 50 points

---

Exercise 1. SQL WITH statement (Tables needed: nikovits.emp, nikovits.sal_cat)

Give the following result for which you can use the WITH statement. Give the employees for whom it is true that his salary is less than the average salary of the employees falling in the same salary category as his salary category. In the result give the employee's name, his salary, and the average salary of his salary category. **(ename, sal, category_avg)**

Exercise 2. Recursion in Datalog (Tables needed: nikovits.flight)

We have a relation FLIGHT(Airline, Orig, Dest, Cost) which contains the flights and their costs from a city (Orig) to another city (Dest). Write a recursive SQL query (a similar Datalog query can help you) which gives the destination cities that can be reached from 'Denver' with at most 1200 cost. The output should contain the destination city and the cost of the whole journey. **(Dest_city, Cost)**.

Exercise 3. DML (Tables needed: nikovits.emp)

Create a table EMP2 which has the same tuples as nikovits.EMP, then write an UPDATE statement (not a pl/sql program !!!) on this table which increases the salaries of the employees for whom it is true that the total summarized salaries of his direct subordinates are larger than 6000 (>6000). The increment is the average salary of the employee's direct subordinates. After the update, give the result of the following query:
```
SELECT avg(sal) FROM emp2;
``` **(Avg_Sal)**

Exercise 4. Functional Dependencies

Consider a relation with schema S(A, B, C, D) and FD's A->B, B->C, B->D.
a) What are all the keys of S?
b) What are all the superkeys that are not keys?

Exercise 5. Normal Forms

Consider a relation with schema R(A, B, C, D, E) and FD's AB->C, DE->C, B->D.
a) Indicate the BCNF violations.
b) Decompose the relations, as necessary, into collections of relations that are in BCNF.

Convert the following E-R diagram to a relational database schema following the **object-oriented approach**. Give the CREATE TABLE statements with **primary key** and **foreign key** definitions. The statements should be syntactically correct, please run them in Oracle to test it. (You can use any Oracle datatype for the columns. You can change the attribute names if needed, for example from "number" to "Cnumber".)