Databases 1 Exam 2020.05.28

Name: Zurab Tsinadze
Neptun code: PRRXNH

Copy your solutions into this file and send it to the following email address: nikovits@inf.elte.hu. You should send the **results of the queries** too. You can use ARAMIS or ULLMAN database.
Deadline: 15:40.
Each exercise counts 10 points, so altogether you can get 60 points.

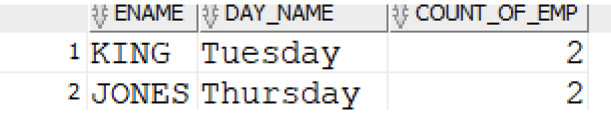
Grades are the following: 2 -> 20 points, 3-> 30 points, 4 -> 40 points, 5 -> 50 points

---

Exercise 1. SQL (Tables needed: nikovits.emp)

Give the employees who have at least two subordinates who started to work on dates that have the same name (e.g. Monday, so the name of the days should be the same for the 2 subordinates.) For these employees give the name of the employee (that is the name of the manager), the name of the day, and the number of subordinates starting to work on this day.
(Ename, Day_name, Count_of_emp)

```
SELECT b.ename ename, to_char(a.hiredate,'Day') Day_name , count(*) Count_of_emp
from nikovits.emp a , nikovits.emp b
where a.mgr = b.empno
GROUP BY b.mgr , b.ename , to_char(a.hiredate,'Day')
HAVING count(*) >= 2;
```



	ENAME	DAY_NAME	COUNT_OF_EMP
1	KING	Tuesday	2
2	JONES	Thursday	2

Exercise 2. Datalog

We have two relations: Q(A,B) and R(C,D). Write an expression of relational algebra that defines the same relation as the head of the following Datalog rule:

P(x,y) ← Q(x,z) AND R(z,y) AND x < y

π A,D (σ (B=C and A<D) (R×S))

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

Create a table EMP2 which has the same tuples as nikovits.EMP, then write an UPDATE statement on this table which increases the salaries of the employees falling into salary category 2. (see nikovits.SAL_CAT table) The increment is the average salary of the employee's own department. After the update, select the new average salary of the employees and give it too. (**Avg_Sal**)

```
drop table emp2;
create table emp2 as select * from nikovits.emp;

update emp2 a
   set sal = sal + (select avg(sal)
                from emp2
                where a.deptno = deptno)
   where (sal, 2) in (select sal, category
                        from emp join sal_cat on (sal between lowest_sal and highest_sal));
select avg(sal) from emp2;
```

Answer: 2549.51

Exercise 4. PL/SQL (Tables needed: nikovits.emp)

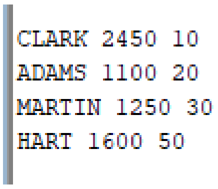
Write a PL/SQL program (procedure or anonymous block) which prints out the following data. The name, the salary, and the department number of the employees whose salary is the second largest within their own department. **(Ename, Sal, Deptno)**

If there are ties (equality) between salaries, then any name can be printed from them.

```
set serveroutput on;
CREATE OR REPLACE PROCEDURE printEmp IS
    CURSOR cur is
        select * from nikovits.emp order by deptno, sal;
    emp2_rec emp2%rowtype;
    cur_dep Integer := 0;
    i Integer := 0;
begin
    open cur;

    loop
        if i = 0 then
            fetch cur into emp2_rec;
            i := 1;
            exit when cur%notfound;
        end if;
        fetch cur into emp2_rec;
        exit when cur%notfound;
        dbms_output.put_line(emp2_rec.ename || ' ' || emp2_rec.sal || ' ' || emp2_rec.deptno);
        cur_dep := emp2_rec.deptno;
        while cur_dep = emp2_rec.deptno loop
            fetch cur into emp2_rec;
            exit when cur%notfound;
        end loop;
    end loop;
    close cur;

end;
/
execute printEmp;
```



```
CLARK 2450 10
ADAMS 1100 20
MARTIN 1250 30
HART 1600 50
```

I know it's a complicated solution, but works.

Exercise 5. Normal Forms

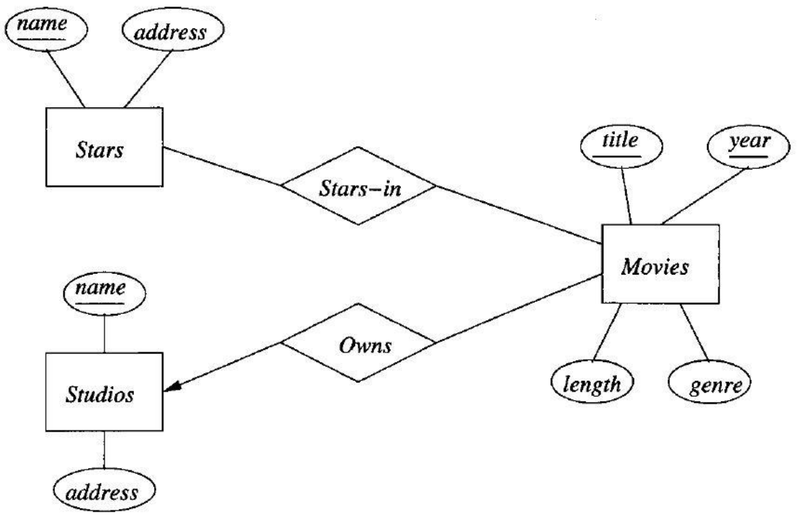
Let R(A,B,C,D) be decomposed into relations R1={A,B}, and R2={B,C,D}.
For the following FD's {B -> C, C -> D} use the chase test to tell whether the decomposition of R is lossless. If not lossless, give an example of an instance of R that returns more than R when projected onto the decomposed relations and rejoined.

	A	B	C	D
R1	a	b	c1=c	d1=d
R2	a2	b	c2=c	d2=d

Answer: lossless join
Why:
Because of B->C we can say that C-s are same in both R1 and R2.
Now because of C->D rule we can say that d-s are also same.

Exercise 6. E-R models and DDL

Convert the following E-R diagram to a relational database schema. 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.)



```
  CREATE TABLE Stars(
  name varchar(25),
  address varchar(30),
  PRIMARY KEY (name)
);

CREATE TABLE Studios(
  name varchar(25),
  address varchar(30),
  PRIMARY KEY (name)
);

CREATE TABLE Movies(
  title varchar(20),
  year varchar(4),
  length number(3),
  genre varchar(10),
  PRIMARY KEY (title, year),
  FOREIGN KEY (studio) references Studios(name)
);


CREATE Table MovieAndStar (
    title varchar(20),
    star varchar(25),
    PRIMARY KEY (title, star),
    FOREIGN KEY (title) references Movies(title),
    FOREIGN KEY (star) references Stars(name)
);
```