Exercises (SQL and relational algebra)

```
NIKOVITS.EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)
NIKOVITS.DEPT(deptno, dname, loc)
NIKOVITS.SAL_CAT(category, lowest_sal, highest_sal)
NIKOVITS.LIKES(name, fruits)
```

1.
List the department number, department name and location for the departments having an employee with salary category 1. (**deptno, dname, loc**)

Tmp = π deptno→dno σ category=1 AND sal >= lowest_sal AND sal <= highest_sal (Emp × Sal_cat)
π Dept.deptno,dname,loc σ dno=Dept.deptno (Tmp × Dept)

```
SELECT deptno, dname, loc from dept WHERE deptno IN
  (SELECT deptno FROM emp, sal_cat
   WHERE category = 1 AND sal BETWEEN lowest_sal AND highest_sal);
```

Other solution:

```
SELECT dept.deptno, dname, loc from dept,
  (SELECT deptno FROM emp, sal_cat
   WHERE category = 1 AND sal BETWEEN lowest_sal AND highest_sal) tmp
WHERE dept.deptno=tmp.deptno;
```

Attention: See the difference in case of multiset meaning. -> use **DISTINCT or δ**

2.
List the department number, department name and location for the departments **having no employee** with salary category 1. (**deptno, dname, loc**)

Tmp = π deptno→dno σ category=1 AND sal >= lowest_sal AND sal <= highest_sal (Emp × Sal_cat)
T2 = π Dept.deptno,dname,loc σ dno=Dept.deptno (Tmp × Dept)
Dept - T2

```
SELECT deptno, dname, loc from dept
 MINUS
SELECT deptno, dname, loc from dept WHERE deptno IN
  (SELECT deptno FROM emp, sal_cat
   WHERE category = 1 AND sal BETWEEN lowest_sal AND highest_sal);
```

Other solution:

```
SELECT deptno, dname, loc from dept WHERE deptno NOT IN
  (SELECT deptno FROM emp, sal_cat
   WHERE category = 1 AND sal BETWEEN lowest_sal AND highest_sal);
```

3.
List the department number, department name and location for the departments **having at least two employees** with salary category 1. (**deptno, dname, loc**)

Tmp = γ deptno;count(ename)→cnt σ category=1 AND sal >= lowest_sal AND sal <= highest_sal (Emp × Sal_cat)
π deptno,dname,loc σ cnt>=2 (Tmp ⋈ Dept)

```
SELECT deptno, dname, loc from dept WHERE deptno IN
  (SELECT deptno FROM emp, sal_cat
   WHERE category = 1 AND sal BETWEEN lowest_sal AND highest_sal
   GROUP BY deptno HAVING COUNT(ename) >= 2);
```

4.
List the employees who have maximal salary within their own department. Give the department number, employee name and salary for them. (**deptno, ename, sal**)

Tmp = ρ dno←deptno (γ deptno;max(sal)→ms (Emp))
π deptno,ename,sal σ dno=Emp.deptno AND Emp.sal = ms (Tmp × Emp)

```
SELECT e.deptno, ename, sal
FROM emp e, (SELECT deptno, MAX(sal) ms FROM emp GROUP BY deptno) tmp
WHERE e.deptno = tmp.deptno and e.sal = ms;
```

5.
List the jobs where this job occurs only on one department and give the name of this department too. **(job, dname)**
Projection is important below, because of duplicate elimination!

Tmp = σ cnt=1 γ job;count(deptno)→cnt (π job,deptno Emp)
π job, dname ((Tmp ⋈ Emp) ⋈ Dept)

```
SELECT DISTINCT job, dname FROM emp NATURAL JOIN dept
WHERE job IN
  (SELECT job FROM emp
   GROUP BY job HAVING COUNT(DISTINCT deptno) = 1);
```

6.
Give the names who like every fruit. **(name)** (see Likes(name, fruits) relation)

Tmp = γ count(fruits)->fr (π fruits (Likes))
π name (Tmp ⋈ (γ name;count(fruits)->fr Likes))

```
SELECT name FROM likes
GROUP BY name
HAVING COUNT(fruits) = (SELECT COUNT(DISTINCT fruits) FROM likes);
```

Other solution without grouping:

NotLikes = (π l1.name, l2.fruits (ρ l1 Likes x ρ l2 Likes)) - Likes
π name Likes - π name NotLikes

```
SELECT name FROM likes
 MINUS
SELECT DISTINCT name FROM
(SELECT DISTINCT l1.name, l2.fruits FROM likes l1, likes l2
  MINUS
 SELECT name, fruits FROM likes) NotLikes;
```

7.
Give the salary and salary category of the employees who have the lowest salary among the employees having a subordinate. (**sal, category**)

Manager = ρ empno←mgr π mgr Emp
Tmp = γ min(sal)→ms (Manager ⋈ Emp)
π ms,category σ ms >= lowest_sal AND ms <= highest_sal (Sal_cat x Tmp)

```
SELECT minsal, category FROM sal_cat,
(SELECT MIN(sal) AS minsal
 FROM emp NATURAL JOIN (SELECT mgr AS empno FROM emp)) tmp
WHERE tmp.minsal BETWEEN lowest_sal AND highest_sal;
```

Other solution:

```
SELECT minsal, category FROM sal_cat,
(SELECT MIN(sal) minsal FROM emp WHERE empno IN (SELECT mgr FROM emp)) t
WHERE t.minsal BETWEEN lowest_sal AND highest_sal;
```

8. (Only SQL because relational algebra does not have built in functions)
Give the month names (January, February etc.) in which at least two employees started to work (hiredate shows the start of work) and give the number of such employees by month.
(Month_name, Num_emps)

```
SELECT to_char(hiredate, 'Month') Month_name,
       COUNT(distinct empno) Num_emps
FROM emp
GROUP BY to_char(hiredate, 'Month')
HAVING COUNT(empno) >=2;
```