

Databases 1 practice test

First part: „paper” exercises (6 points/exercise, sum: 54 points/9 exercises)

Consider the usual EMP and DEPT tables:

DEPT(DEPTNO, DNAME, LOC)

EMP(EMPNO, ENAME, JOB, MGR, HIRED, SAL, COMM, DEPTNO)

DEPT table

DEPTNO: department number. Unique ID of the department.

DNAME: department name. The name of the department like ACCOUNTING.

LOC: location of the department.

EMP table

EMPNO: employee number. Unique ID of the employee.

ENAME: employee name.

JOB: job of the employee, like CLERK

MGR: manager of the employee.

HIRED: hire date

SAL: salary

COMM: commission

DEPTNO: department number. Unique ID of a department.

Core Relational Algebra and Datalog:

You can use only CORE relational algebraic operators! Extended relational algebra cannot be used!

- a.) First write the relational algebra expression, you can use linear notation or expression tree notation for core relational algebraic expression (one of them),
- b.) then convert it into Datalog: write one or more Datalog rules that define the result.

1. Which jobs are done by at least two different employees?

1a.) in core relational algebra (6 points)

1b.) in Datalog (6 points)

2. Determine the cities (LOC) where at least one CLERK works, whose salary less, than 2000!

2a.) in core relational algebra (6 points)

2b.) in Datalog (6 points)

NAME:

NEPTUN CODE:

2/3

Extended Relational Algebra and SQL SELECT:

Use extended relational algebra, then convert the extended algebraic expression to SELECT statement.

3. Determine the average salary of SALESMAN's for every department (deptno) but only for those department where at least three SALESMAN work.

3a.) in extended relational algebra (6 points)

3b.) in SQL (6 points)

SQL queries:

Write the queries only in SQL: SELECT statement.

4. List the location of departments where the employee with highest salary works. (6p)

5. Find the employee who has the highest commission! (6p)

6. Find the departments where the average salary is at least the average salary of all employees! (6p)

NAME:

NEPTUN CODE:

3/3

ORACLE ID:

Second part: „ computer” SQL exercises

(6 points/exercise, sum: 36 points/9 exercises)

Total points: 90 points = 54 p (paper part) + 36 p (computer part)

Consider the following database schema:

```
EPerson(name, age, gender) // name is a key
EFrequents(name, pizzeria) // [name,pizzeria] is a key
EEats(name, pizza) // [name,pizza] is a key
EServes(pizzeria, pizza, price) // [pizzeria,pizza] is a key
```

create table script here: http://people.inf.elte.hu/sila/eduAB/create_epizza.txt

ORACLE SQL Developer

HOST: aramis.inf.elte.hu PORT: 1521 SEVICE: eszakigrid97

or (in case of emergency) HOST: tomx.inf.elte.hu PORT: 1522 SEVICE: ora11g

put the select statements into a text file: Neptun.txt (text-file)

and at the end move this file into this folder:

<\\nas2.inf.elte.hu\zh\sila>

or (in case of emergency) <\\inf.elte.hu\dfs\zh\sila>

1. Find all pizzas eaten by at least one female over the age of 20
2. Find all pizzerias that serve at least one pizza for less than \$10 that either Amy or Fay (or both) eat.
3. Find the age of the oldest person (or people) who eat mushroom pizza.
4. Find all pizzaterias where pepperoni pizza cost less than supreme pizza in the Pizza Hut pizzateria.
5. Determine the average price of pizza for every pizzas but only for those pizzas what is served at least three pizzaterias!
6. Find the pizzateria where the average price of the pizzas is the highest!