--LABORATOR 5

Exerciţii:

0. a) Să se afişeze informaţii despre angajaţii al căror salariu depăşeşte valoarea medie a salariilor

colegilor săi de departament.

SELECT last\_name, salary, department\_id

FROM employees e

WHERE salary > (SELECT AVG(salary)

FROM employees

WHERE department\_id = e.department\_id

);

b) Analog cu cererea precedentă, afişându-se şi numele departamentului şi media salariilor acestuia

şi numărul de angajaţi.

--Soluţia 1 (subcerere necorelată în clauza FROM):

SELECT last\_name, salary, e.department\_id, department\_name, sal\_med, nr\_sal

FROM employees e, departments d, (SELECT department\_id, round(AVG(salary)) sal\_med, COUNT(\*) nr\_sal

FROM employees

GROUP BY department\_id

) ac

WHERE e.department\_id = d.department\_id

AND d.department\_id = ac.department\_id

AND salary > (SELECT AVG(salary)

FROM employees

WHERE department\_id = e.department\_id

);

-- sau:

SELECT last\_name, salary, e.department\_id, department\_name, sal\_med, nr\_sal

FROM employees e, departments d, (SELECT department\_id, round(AVG(salary)) sal\_med, COUNT(\*) nr\_sal

FROM employees

GROUP BY department\_id

) ac

WHERE e.department\_id = d.department\_id

AND d.department\_id = ac.department\_id

AND salary > sal\_med;

1. Să se afişeze numele şi salariul angajaţilor al căror salariu este mai mare decât salariile medii

din toate departamentele.

Se cer 2 variante de rezolvare: cu operatorul ALL sau cu funcţia MAX.

--ALL

select last\_name, salary

from employees

where salary > ALL (select round(avg(salary))

from employees

group by department\_id

);

--MAX

select last\_name, salary

from employees

where salary > (select round(max(avg(salary)))

from employees

group by department\_id

);

2. Sa se afiseze numele si salariul celor mai prost platiti angajati din fiecare departament.

--Soluţia 1 (cu sincronizare):

SELECT last\_name, salary, department\_id

FROM employees e

WHERE salary = (SELECT MIN(salary)

FROM employees

WHERE department\_id = e.department\_id);

--Soluţia 2 (fără sincronizare):

SELECT last\_name, salary, department\_id

FROM employees

WHERE (department\_id, salary) IN (SELECT department\_id, MIN(salary)

FROM employees

GROUP BY department\_id);

--Soluţia 3: Subcerere în clauza FROM

SELECT last\_name, salary, e.department\_id

FROM employees e join (select min(salary) sal, department\_id

from employees

group by department\_id) min\_sal

on(e.department\_id = min\_sal.department\_id)

WHERE e.salary = min\_sal.sal;

3. Sa se obtina numele salariatilor care lucreaza intr-un departament in care exista

cel putin 1 angajat cu salariul egal cu salariul maxim din departamentul 30.

--IN

select last\_name,salary, department\_id

from employees

where department\_id IN (select department\_id

from employees

where salary = (select max(salary)

from employees

where department\_id = 30

)

and department\_id != 30

);

--EXISTS

select last\_name,salary, department\_id

from employees e

where EXISTS (select 1

from employees

where e.department\_id = department\_id and

salary = (select max(salary)

from employees

where department\_id = 30

)

and department\_id != 30

);

4. Sa se obtina numele primilor 3 angajati avand salariul maxim.

Rezultatul se va afişa în ordine crescătoare a salariilor.

--Solutia 1: subcerere sincronizată

--numaram cate salarii sunt mai mari decat salariul de la linia la care a ajuns

select last\_name, salary, rownum

from (select last\_name, salary

from employees

order by salary desc

) e

where 3 > (select count(salary)

from employees

where salary > e.salary)

and rownum <= 3;

--Solutia 2: vezi analiza top-n (mai jos)

--dorim sa afisam angajatii care au primele 3 cele mai mari salarii din firma

select last\_name, salary, rownum

from employees

where rownum <= 3

order by salary desc;

-- aceasta varianta este GRESITA deoarece prima data se executa conditia din where

-- deci o sa ia primele 3 randuri gasite in tabel (in ordinea din tabel)

-- si la final o sa execute clauza order by

-- varianta corecta - subcerere in from

-- deoarece dorim sa ordonam descrescator inainte de a aplica conditia in where

select last\_name, salary, rownum

from (select last\_name, salary

from employees

order by salary desc

)

where rownum <= 3;

6. Să se determine locaţiile în care se află cel puţin un departament.

--IN

select location\_id

from locations

where location\_id IN (select location\_id

from departments);

--EXISTS

select location\_id

from locations loc

where EXISTS (select 1

from departments

where loc.location\_id = location\_id);

7. Să se determine departamentele în care nu există nici un angajat.

--NOT EXISTS

SELECT department\_id, department\_name

FROM departments d -- department\_id este cheie primara ceea ce inseamna

-- ca avem o lista unica a tuturor departamentelor

WHERE NOT EXISTS (SELECT 'x'

FROM employees -- aici department\_id este cheie externa deci inseamna

-- ca avem departamente in care lucreaza angajati

WHERE department\_id = d.department\_id);

--> in final obtinem exact lista departamentelor care nu au angajati

--NOT IN

SELECT department\_id, department\_name

FROM departments d

WHERE department\_id NOT IN (SELECT nvl(department\_id, 0)

FROM employees

);

--SAU:

SELECT department\_id, department\_name

FROM departments d

WHERE department\_id NOT IN (SELECT department\_id

FROM employees

WHERE department\_id is not null

);

-- atunci cand utilizam NOT IN trebuie sa eliminam sau sa inlocuim

--valorile null din subcerere

--MINUS

select department\_id

from departments -- din lista tuturor depart

MINUS -- eliminam

select department\_id

from employees; -- departamentele care au angajati

--> obtinem departamente care nu au angajati

8. Utilizând clauza WITH, să se scrie o cerere care afişează

numele departamentelor şi valoarea totală a salariilor din cadrul acestora.

Se vor considera departamentele a căror valoare totală a salariilor este mai mare decât

media valorilor totale ale salariilor tuturor angajatilor.

--numele departamentelor şi valoarea totală a salariilor din cadrul acestora

WITH val\_dep AS (SELECT department\_name, SUM(salary) AS total

FROM departments d join employees e ON (d.department\_id = e.department\_id)

GROUP BY department\_name

),

val\_medie AS (SELECT SUM(total)/COUNT(\*) AS medie

FROM val\_dep)

SELECT \*

FROM val\_dep

WHERE total > (SELECT medie

FROM val\_medie)

ORDER BY department\_name;

Tema: lab 5 -> ex: 5, 9, 10, 11, 12, 14

Deadline: duminica 12.04 inclusiv

5. Să se afişeze codul, numele şi prenumele angajaţilor care au cel puţin doi subalterni.

select employee\_id, last\_name, first\_name

from employees e join (select manager\_id, count(employee\_id)

from employees

group by manager\_id

having count(employee\_id)>2

) t on (e.manager\_id=t.manager\_id);

9. Să se afişeze codul, prenumele, numele şi data angajării, pentru angajatii condusi de Steven King care au cea mai mare

vechime dintre subordonatii lui Steven King. Rezultatul nu va conţine angajaţii din anul 1970.

with tabel as (select employee\_id, first\_name, last\_name, hire\_date

from employees

where manager\_id=( select employee\_id

from employees

where lower(last\_name) like'king' and lower(first\_name) like'steven'

)

and to\_char(hire\_date, 'yyyy')!='1970'

),

t as (select max(hire\_date) date\_max

from tabel

)

select employee\_id, first\_name, last\_name, hire\_date

from tabel , t

where hire\_date=date\_max;

10. Să se determine primii 10 cei mai bine plătiţi angajaţi.

select last\_name, salary, rownum

from (select last\_name, salary

from employees

order by salary desc

)

where rownum<=10;

11. Să se afişeze informaţii despre departamente, în formatul următor: „Departamentul <department\_name> este

condus de {<manager\_id> | nimeni} şi {are numărul de salariaţi <n> | nu are salariati}“.

select 'Departamentul '|| d.department\_name ||' este condus de '|| decode(e.manager\_id,NULL,' nimeni ',to\_char(e.manager\_id))

||' si are numarul de salariati '|| decode(NR , 0, 'nu are salariati' , to\_char(NR))

from departments d join employees e on (d.department\_id=e.department\_id)

join ( select count(employee\_id)NR, department\_id

from employees

group by department\_id

) t on (e.department\_id=t.department\_id) ;

12. Să se afişeze numele, prenumele angajaţilor

şi lungimea numelui pentru înregistrările în care aceasta este diferită de lungimea prenumelui.

select last\_name, first\_name, length(last\_name)

from employees

where length(last\_name)!=length(first\_name);

14. Să se afişeze:

a) suma salariilor, pentru job-urile care incep cu litera S;

select sum(salary)

from employees e join jobs j on (e.job\_id=j.job\_id)

where lower(j.job\_title) like 's%';

b) media generala a salariilor, pentru job-ul avand salariul maxim;

with t as (select job\_id, max(salary) sal\_max

from employees

group by job\_id

),

maxim as (select job\_id jobid

from t

where sal\_max =(select max(sal\_max)

from t

)

)

select round(avg(salary))

from employees , maxim

where job\_id=jobid;

c) salariul minim, pentru fiecare din celelalte job-uri.

with t as (select job\_id, max(salary) sal\_max, min(salary) sal\_min

from employees

group by job\_id

),

maxim as (select job\_id jobid

from t

where sal\_max =(select max(sal\_max)

from t

)

)

select distinct t.job\_id, sal\_min

from employees , t, maxim

where t.job\_id!=jobid;