-- LABORATOR 6

DIVISION

SELECT employee\_id

FROM works\_on

MINUS

SELECT employee\_id

From works\_on

WHERE project\_id IN

(SELECT project\_id

( SELECT employee\_id, project\_id

FROM (SELECT DISTINCT employee\_id FROM works\_on) t1,

(SELECT project\_id FROM projects WHERE to\_char(start\_date,'dd/mm/yyy')<'01/07/2006' and to\_char(start\_date,'dd/mm/yyy')>='01/01/2006') t2

MINUS

SELECT employee\_id, project\_id FROM works\_on

) t3;

SELECT employee\_id

FROM works\_on

WHERE project\_id IN

(SELECT project\_id

FROM projects

WHERE to\_char(start\_date,'dd/mm/yyy')<'01/07/2006' and to\_char(start\_date,'dd/mm/yyy')>='01/01/2006'

)

GROUP BY employee\_id

HAVING COUNT(project\_id)= (SELECT COUNT(\*)

FROM projects

WHERE to\_char(start\_date,'dd/mm/yyy')<'01/07/2006' and to\_char(start\_date,'dd/mm/yyy')>='01/01/2006'

);

desc projects;

desac works\_on;

select \* from projects;

select \* from works\_on;

Exemplu: Să se obţină codurile salariaţilor ataşaţi tuturor proiectelor

pentru care s-a alocat un buget egal cu 10000.

--Metoda 1 (utilizând de 2 ori NOT EXISTS):

SELECT DISTINCT employee\_id

FROM works\_on a

WHERE NOT EXISTS

(SELECT 1

FROM projects p

WHERE budget = 10000

AND NOT EXISTS

(SELECT 'x'

FROM works\_on b

WHERE p.project\_id = b.project\_id

AND b.employee\_id = a.employee\_id

)

);

-- metoda 1 este identinca cu metoda 4

Exemplu: Să se obţină codurile salariaţilor ataşaţi tuturor proiectelor

pentru care s-a alocat un buget egal cu 10000.

select \* from projects; -- lista tuturor proiectelor

-- observam ca p2 si p3 au buget egal cu 10k

select \* from works\_on; -- lista angajatilor care lucreaza la proiecte

-- observam ca angajatii 145, 148, 101, 200 lucreaza la TOATE proiectele cu buget de 10k

--Metoda 2 (simularea diviziunii cu ajutorul funcţiei COUNT):

SELECT employee\_id

FROM works\_on -- lista tuturor angajatilor care lucreaza la proiecte

WHERE project\_id IN

(SELECT project\_id

FROM projects

WHERE budget = 10000 -- proiectele care au buget EGAL de 10k (p2 si p3)

) -- daca ne oprim cu solutia in acest punct obtinem angajatii care lucreaza atat la

-- toate proiectele cu buget egal cu 10k, cat si la o parte din proiectele

-- cu buget egal cu 10k

GROUP BY employee\_id -- grupand putem numara pentru fiecare angajat proiectele cu buget de 10k la care lucreaza

HAVING COUNT(project\_id)= (SELECT COUNT(\*)

FROM projects

WHERE budget = 10000

); -- si daca nr proiectelor la care lucreaza este egal cu nr proiectelor cu

-- buget egal cu 10k => ca ang lucreaza la TOATE proiectele care au buget

-- egal cu 10k si nu doar la o parte din ele

Exemplu: Să se obţină codurile salariaţilor ataşaţi tuturor proiectelor

pentru care s-a alocat un buget egal cu 10000.

select \* from projects; -- lista tuturor proiectelor

-- observam ca p2 si p3 au buget egal cu 10k

select \* from works\_on; -- lista angajatilor care lucreaza la proiecte

-- observam ca angajatii 145, 148, 101, 200 lucreaza la TOATE proiectele cu buget de 10k

--Metoda 3 (operatorul MINUS):

SELECT employee\_id

FROM works\_on -- lista tuturor angajatilor care lucreaza la proiecte

MINUS -- dorim sa eliminam angajatii care lucreaza la proiecte cu buget DIFERIT de 10k

-- sau pe cei care lucreaza doar la o parte din proiectele cu buget EGAL cu 10k

-- in final dorim sa obtinem doar angajatii care lucreaza la TOATE proiectele cu buget de 10k

SELECT employee\_id from

( SELECT employee\_id, project\_id

FROM (SELECT DISTINCT employee\_id FROM works\_on) t1,

(SELECT project\_id FROM projects WHERE budget = 10000) t2

MINUS

SELECT employee\_id, project\_id FROM works\_on -- (\*)

) t3;

in baza de date avem (luam 2 exemple):

ang 148 care lucreaza la p2 si p3

ang 176 care lucreaza la p3 (\*) - lista angajatilor care lucreaza la proiecte

t1 - o lista de angajati care lucreaza la proiecte

t2 - lista proiectelor cu buget de 10k

produsul cartezian t1 cu t2 - (luand elementele 148 si 176 ca exemplu)

rezulta -> 148 - p2 / 148 - p3

176 - p2 / 176 - p3

etc (combinand toti angajatii cu proiectele cu buget de 10k)

-> din produsul cartezian se elimina (\*) care reprezinta lista angajatilor care lucreaza la proiecte

rezulta => un ang care lucreaza la toate proiectele cu buget de 10k o sa fie eliminat (ex: ang 148)

176 ramane in lista deoarece el lucreaza doar la o parte din proiectele cu buget de 10k

Prin urmare in t3 ramane o lista cu angajatii care lucreaza la proiecte cu buget diferit de 10k

sau doar la o parte din proiectele cu buget egal cu 10k !!!!!!!!!

In final obtinem doar angajatii care lucreaza la TOATE proiectele cu buget de 10k

--Metoda 4 (A include B => B\A = Ø):

SELECT DISTINCT employee\_id

FROM works\_on a -- lista tuturor ang care lucreaza la proiecte

WHERE NOT EXISTS

(

(SELECT project\_id

FROM projects p

WHERE budget = 10000 -- proiectele cu buget egal cu 10k (p2 si p3)

)

MINUS -- din lista proiectelor cu buget de 10k elimin proiectele la care lucreaza ang curent

-- atunci daca ang curent lucreaza la toate proiectele cu buget de 10k, in subcerere se obtine

-- multimea vida

-- daca ang lucreaza la o parte din proiectele cu buget de 10k sau la alte proiecte cu buget

-- diferit de 10k, atunci angajatul respectiv o sa ramana in lista obtinuta in subcerere

(SELECT p.project\_id

FROM projects p, works\_on b

WHERE p.project\_id = b.project\_id

AND b.employee\_id = a.employee\_id -- lista proiectelor la care lucreaza angajatul CURENT (sincronizare)

) -- in subcerere raman ang care lucreaza la proiecte cu buget diferit de 10k sau

-- doar la o parte din proiectele cu buget de 10k

);

-- in final se obtin ang care lucreaza la TOATE proiectele cu buget de 10k.

Exerciţii (DIVISION + alte cereri):

1. Să se listeze informaţii despre angajaţii care au lucrat în TOATE proiectele

demarate în primele 6 luni ale anului 2006.

select \* from projects;

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

from employees

where employee\_id in (select employee\_id

from works\_on -- angajati care lucreaza la proiecte

where project\_id in (select project\_id

from projects

where start\_date >= to\_date('01-jan-06')

and start\_date <= to\_date('30-jun-06')

-- lista proiectelor demarate in primele 6 luni(p1 si p2)

)

-- daca ne oprim cu solutia in acest punct obtinem angajatii care lucreaza

-- la toate proiectele demarate in primele 6 luni ale lui 2006

-- dar si pe cei care lucreaza doar la o parte din proiectele demarate in primele 6 luni din 2006

group by employee\_id -- grupand putem numara pt fiecare angajat proiectele la care lucreaza

having count(project\_id) = (select count(project\_id)

from projects

where start\_date >= to\_date('01-jan-06')

and start\_date <= to\_date('30-jun-06')

) -- daca nr de proiecte este egal cu nr proiectelor demarate in primele

-- 6 luni din 2006 => ca ang lucreaza la TOATE proiectele acestea

);

2. Să se listeze informaţii despre proiectele la care au participat toţi angajaţii

care au deţinut alte 2 posturi în firmă.

select \*

from projects -- lista tuturor proiectelor

where project\_id in (select project\_id

from works\_on -- proiecte la care lucreaza angajati

where employee\_id in

(select employee\_id

from job\_history

group by employee\_id

having count(job\_id) = 2

) --angajaţii care au deţinut alte 2 posturi în firmă (101, 176, 200)

group by project\_id -- grupand putem numara angajatii pentru fiecare proiect

having count(employee\_id) = (select count(count(employee\_id))

from job\_history

group by employee\_id

having count(job\_id) = 2

)

-- daca nr de ang care lucreaza la un proiecte este egal

-- cu nr de angajati care au detinut alte doua posturi in firma

-- atunci inseamna ca la acel proiect au participat toti ang care

-- au detinut alte doua posturi in firma

);

3. Să se obţină numărul de angajaţi care au avut cel puţin trei job-uri, luându-se în considerare şi job-ul curent.

-- inseamna cel putin doua joburi in trecut - in job\_history

select count(count(employee\_id)) NrAng

from job\_history

group by employee\_id

having count(job\_id) >= 2;

4. Pentru fiecare ţară, să se afişeze numărul de angajaţi din cadrul acesteia.

select count(employee\_id)

from employees join departments using (department\_id)

join locations using (location\_id)

right join countries using (country\_id)

group by country\_id;

Tema: laborator 6: ex: 5, 6, 7

Deadline: 25.04 (sambata inclusiv)

5. Să se listeze codurile angajaţilor şi codurile proiectelorpe care au lucrat.

Listarea va cuprinde şi angajaţii care nu au lucrat pe nici un proiect.

select e.employee\_id, project\_id

from employees e left join works\_on w on ( e.employee\_id=w.employee\_id);

6. Să se afişeze angajaţii care lucrează în acelaşi

departament cu cel puţin un manager de proiect.

select employee\_id

from employees

where department\_id IN (select department\_id

from employees

where employee\_id IN (

select project\_manager

from projects

)

);

7. Să se afişeze angajaţii care nu lucrează în acelaşi departament cu nici un manager de proiect

select employee\_id

from employees

where department\_id not in(select department\_id

from employees

where employee\_id in (select project\_manager

from projects

)

);