1

Select distinct a.nume, a.prenume, c.denumire

//distinct pt ca un ang poate lucre la mai multe proiecte pt un anumit client

From angajat a join lucreaza l on (a.cod = l.cod\_angajat)

Join client c on (c.cod = l.cod\_client)

Join proiect p on (p.cod\_proiect = l.cod\_proiect)

Where data\_ef is null;

2

Select a.nume, a.prenume

From angajat a ///pt un a se intampla…

Where exists ( select ‘p’

from proiect p join sarcina s on (p.cod = s.cod\_proiect)

Where cod\_manager = a.cod

Group by p.cod

Having count( s.cod\_sarcina) > 20

)

And not exists ( select ‘p’

from proiect p join sarcina s on (p.cod = s.cod\_proiect)

Where cod\_manager = a.cod

Group by p.cod

Having count( s.cod\_sarcina) < =20

)

3. Grouping sets = group -uri legate prin union

Select a.nume, p.denumire, sum(l.data\_end – l.data\_start)nr\_zile

From angajat a join lucreaza l on (a.cod= l.cod\_angajat)

Join proiect p on (l.cod\_proiect = p.cod)

Group by grouping sets( (a.nume, p.denumire),( a.nume), (p.denumire) );

4.

Create table lucreaza (cod\_angajat number (7),

Cod\_proiect number (7) references proiect(cod),

Cod\_sarcina number (7),

Data\_start date,

Data\_end date,

Constraint pk\_l primary\_key (cod\_angajat, cod\_sarcina, data\_start),

Constraint fk\_l1 foreign\_key (cod\_angajat), references angajat(cod),

Constraint fk\_l2 foreign\_key (cod\_sarcina), references sarcina(cod),

Constraint ck\_l check (data\_start <data\_end);

5

Update proiect p

Set cod\_manager = ( select cod

From angajat

Where salariu = (select max (salariu)

From angajat)

)

Where p.data\_start = ( select max(data\_start)

From proiect

Where to\_char(data\_start, ‘yyyy’) = ‘2014’

)