## Exercitiul 1

select last\_name || ' ' || first\_name as "Nume\_PAM", hire\_date as "Data\_angajare\_PAM",

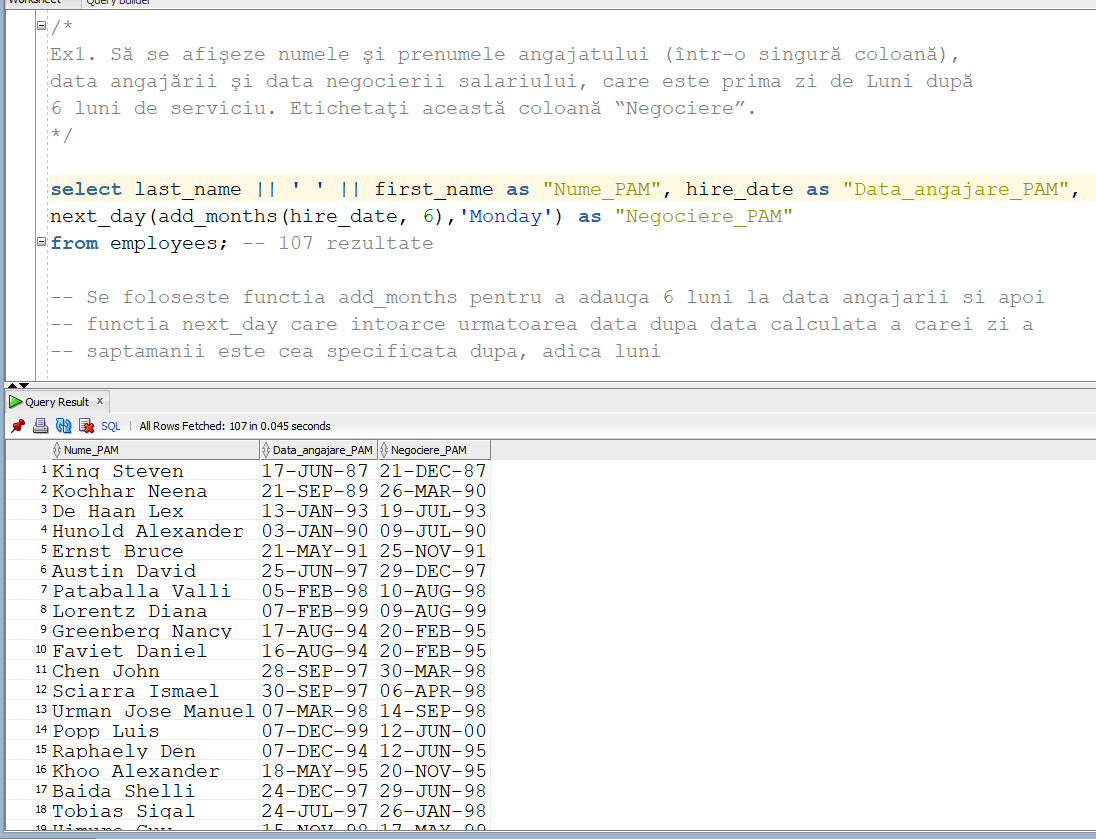
next\_day(add\_months(hire\_date, 6),'Monday') as "Negociere\_PAM"

from employees; -- 107 rezultate

-- Se foloseste functia add\_months pentru a adauga 6 luni la data angajarii si apoi

-- functia next\_day care intoarce urmatoarea data dupa data calculata a carei zi a

-- saptamanii este cea specificata dupa, adica luni



Exercitiul 2

select last\_name as "Nume\_PAM", round(months\_between(sysdate, hire\_date)) as "Luni\_lucrate\_PAM"

from employees

where to\_char(hire\_date, 'YYYY')<'1992'

order by 2; -- 5 rezultate

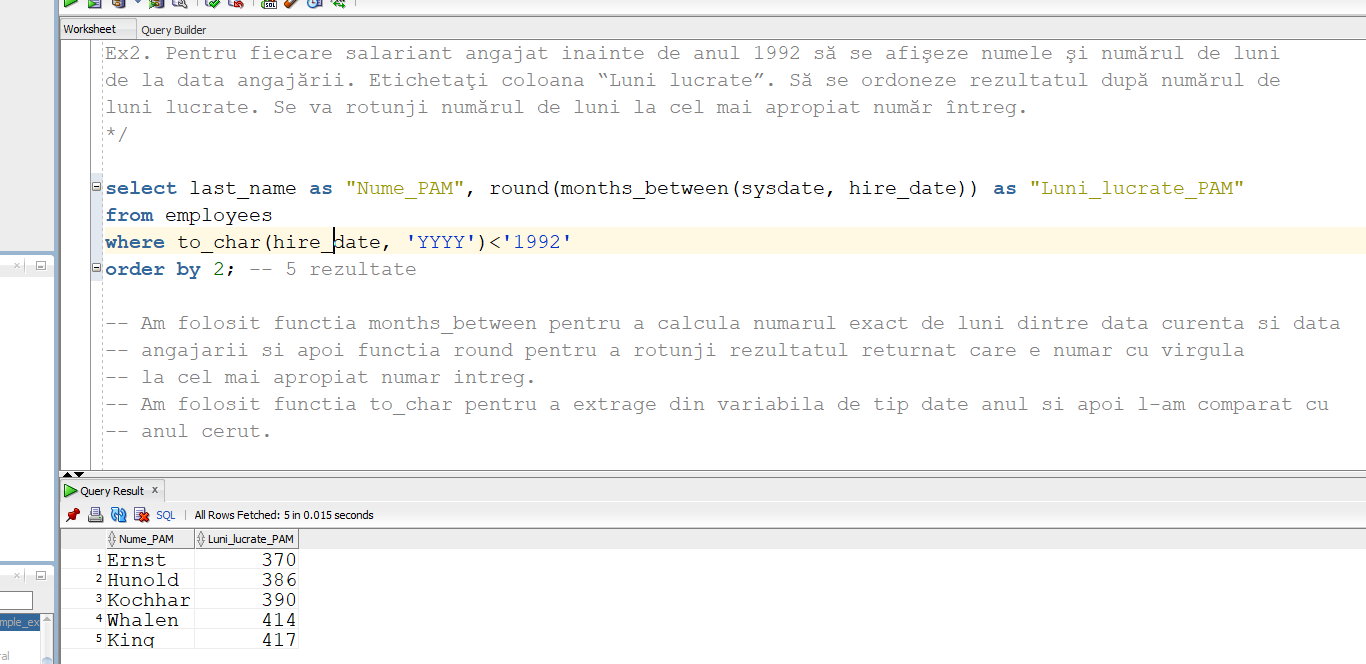
-- Am folosit functia months\_between pentru a calcula numarul exact de luni dintre data curenta si data

-- angajarii si apoi functia round pentru a rotunji rezultatul returnat care e numar cu virgula

-- la cel mai apropiat numar intreg.

-- Am folosit functia to\_char pentru a extrage din variabila de tip date anul si apoi l-am comparat cu

-- anul cerut.



## Exercitiul 3

select e.last\_name as "Nume\_PAM", d.department\_name as "Departament\_PAM", l.city as "Oras\_PAM"

from employees e, departments d, locations l

where e.department\_id = d.department\_id and d.location\_id=l.location\_id

and e.commission\_pct is not null and upper(e.first\_name) like ('\_E%'); -- 3 rezultate

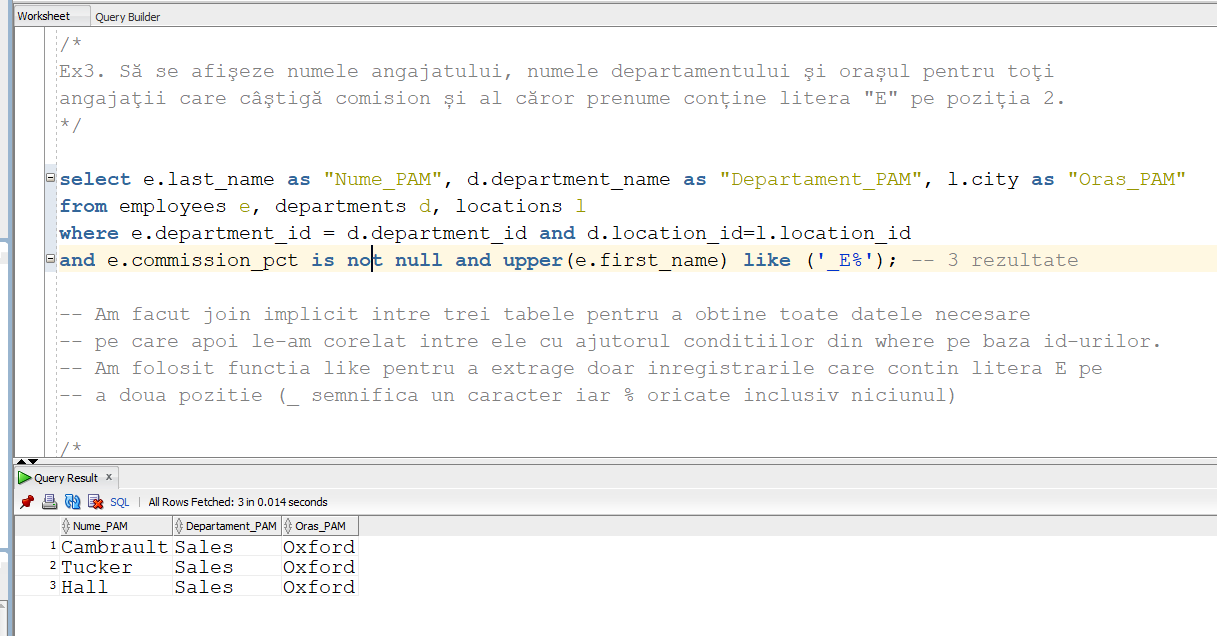
-- Am facut join implicit intre trei tabele pentru a obtine toate datele necesare

-- pe care apoi le-am corelat intre ele cu ajutorul conditiilor din where pe baza id-urilor.

-- Am folosit functia like pentru a extrage doar inregistrarile care contin litera E pe

-- a doua pozitie (\_ semnifica un caracter iar % oricate inclusiv niciunul)

-- Am verificat existenta unui comision prin compararea campului cu null.

****

## Exercitiul 4

select e.last\_name as "Nume\_PAM", d.department\_name as "Departament\_PAM", j.job\_title as "Nume\_job\_PAM"

from employees e, departments d, jobs j

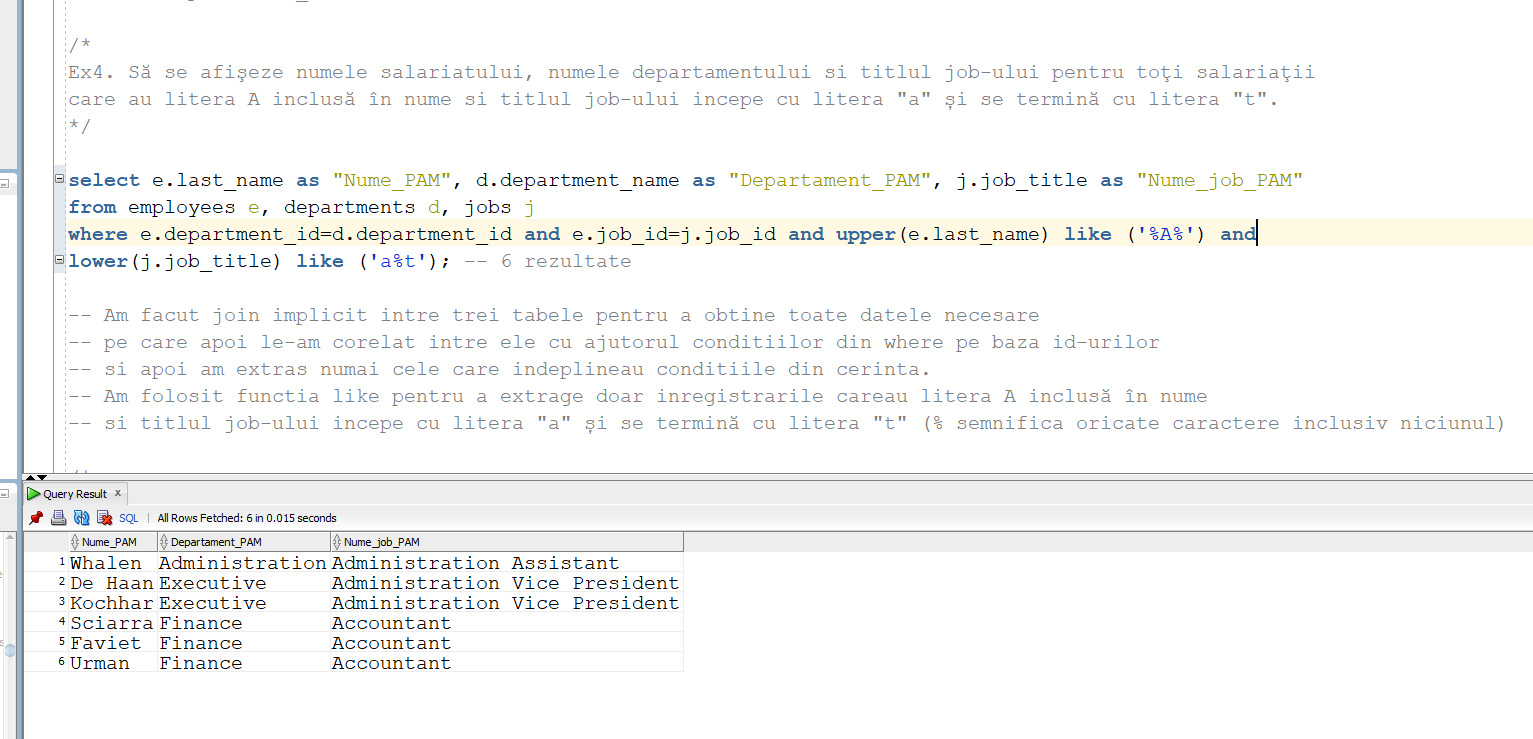
where e.department\_id=d.department\_id and e.job\_id=j.job\_id and upper(e.last\_name) like ('%A%') and lower(j.job\_title) like ('a%t'); -- 6 rezultate

-- Am facut join implicit intre trei tabele pentru a obtine toate datele necesare

-- pe care apoi le-am corelat intre ele cu ajutorul conditiilor din where pe baza id-urilor

-- si apoi am extras numai cele care indeplineau conditiile din cerinta.

-- Am folosit functia like pentru a extrage doar inregistrarile careau litera A inclusă în nume

-- si titlul job-ului incepe cu litera "a" și se termină cu litera "t" (% semnifica oricate caractere inclusiv

-- niciunul)

## Exercitiul 5

* Varianta 1 - Decode

select e.last\_name as "Nume\_PAM", e.hire\_date as "Data\_angajarii\_PAM", j.job\_title as "Nume\_job\_PAM", e.salary as "Salariu\_PAM", e.salary + decode(to\_char(e.hire\_date, 'YYYY'), '1989', e.salary\*0.2, '1990', e.salary\*0.15, '1991', e.salary \* 0.1, 0) as "Salariu\_marire\_PAM"

from employees e, jobs j

where e.job\_id = j.job\_id; -- 107 rezultate

-- Am folosit functia decode pentru a returna diverse valori in functie de valoarea pe care

-- o are anul angajarii. Primul argument este comparat cu al doilea si in caz de egalitate se

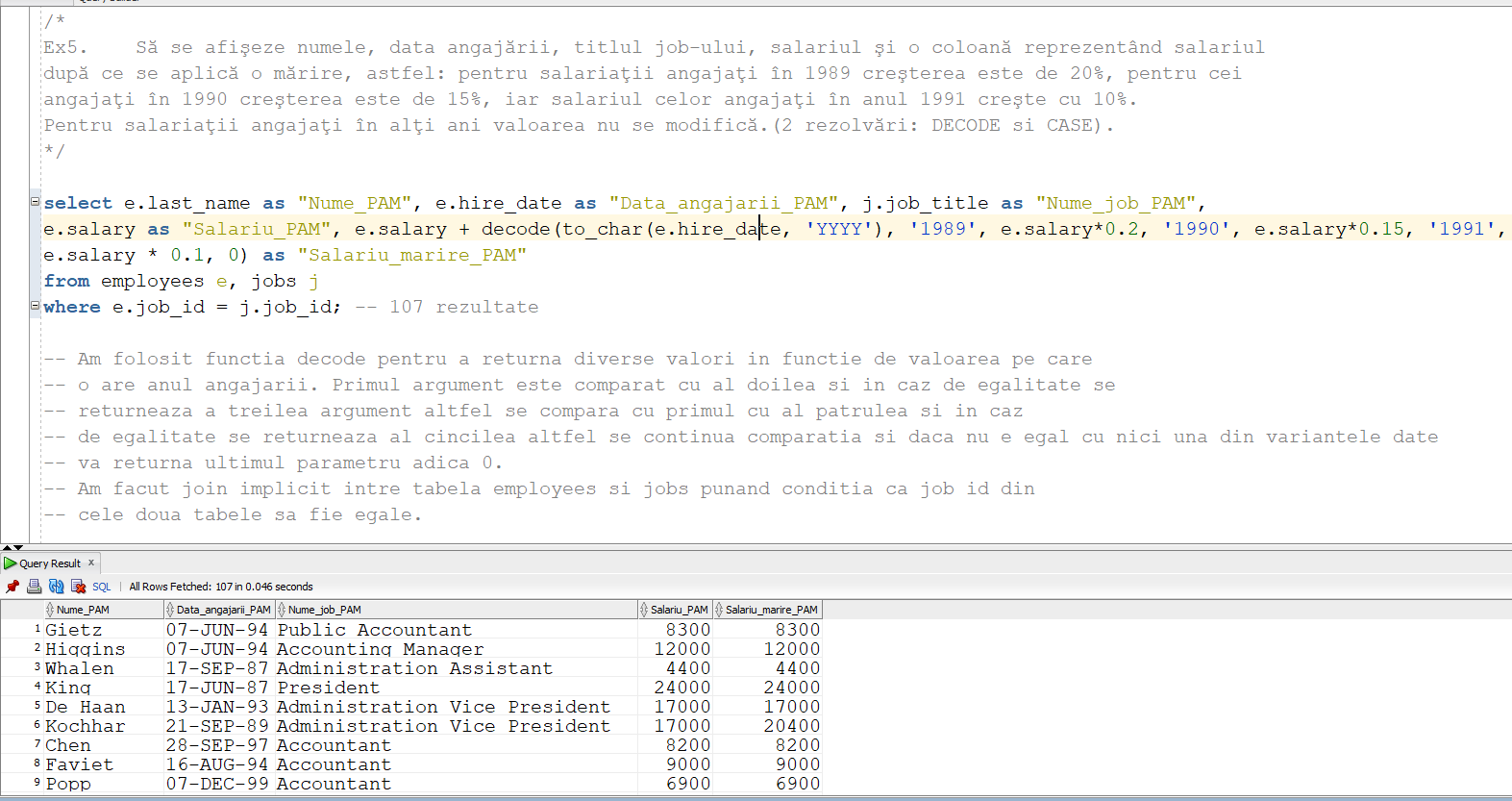
-- returneaza a treilea argument altfel se compara cu primul cu al patrulea si in caz

-- de egalitate se returneaza al cincilea altfel se continua comparatia si daca nu e egal cu nici una din

-- variantele date va returna ultimul parametru adica 0.

-- Am facut join implicit intre tabela employees si jobs punand conditia ca job id din

-- cele doua tabele sa fie egale.



* Varianta 2 - Case

select e.last\_name as "Nume\_PAM", e.hire\_date as "Data\_angajarii\_PAM", j.job\_title as "Nume\_job\_PAM",

e.salary as "Salariu\_PAM", e.salary +

case to\_char(e.hire\_date, 'YYYY')

when '1989' then e.salary\*0.2

when '1990' then e.salary\*0.15

when '1991' then e.salary\*0.1

else 0

end as "Salariu\_marire\_PAM"

from employees e, jobs j

where e.job\_id = j.job\_id; -- 107 rezultate

-- Am folosit clauza case pentru a returna diverse valori in functie de valoarea pe care

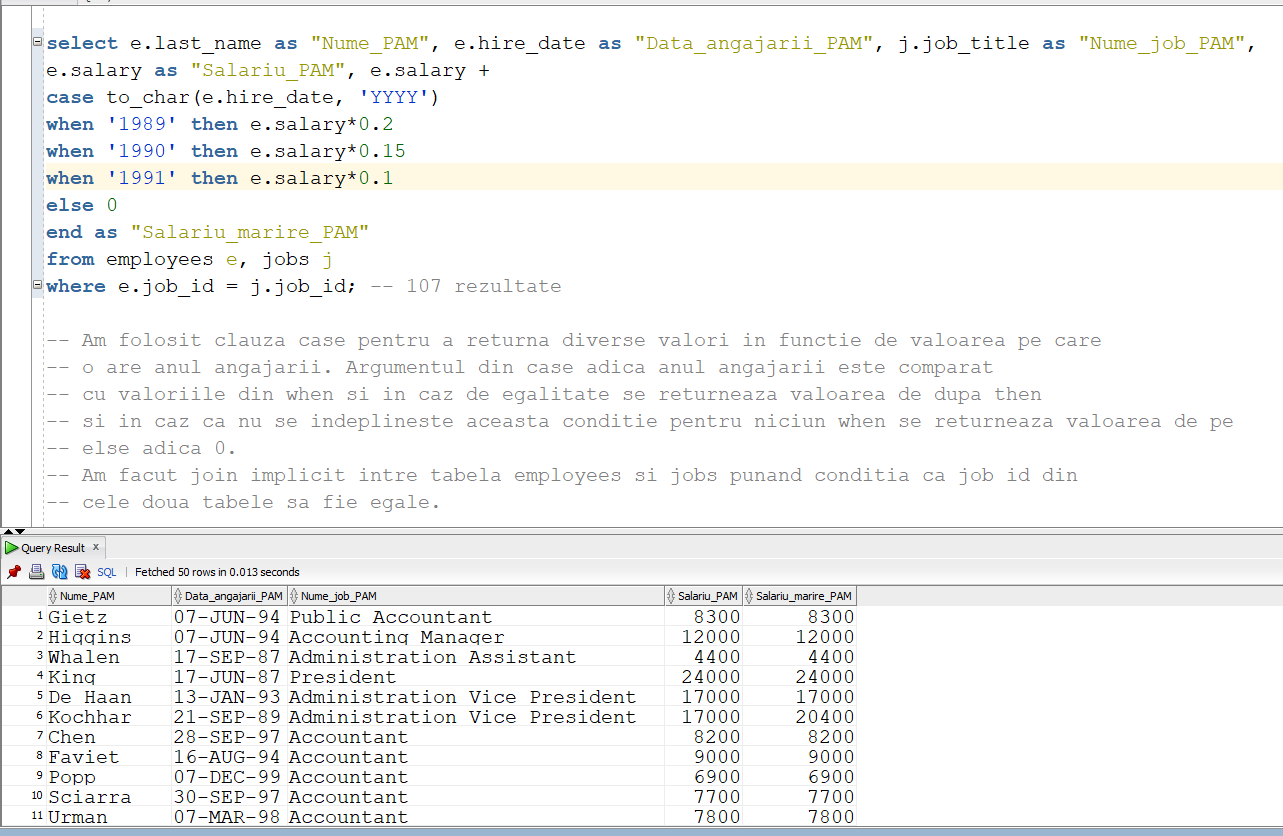
-- o are anul angajarii. Argumentul din case adica anul angajarii este comparat

-- cu valoriile din when si in caz de egalitate se returneaza valoarea de dupa then

-- si in caz ca nu se indeplineste aceasta conditie pentru niciun when se returneaza valoarea de

-- pe else adica 0.

-- Am facut join implicit intre tabela employees si jobs punand conditia ca job id din

-- cele doua tabele sa fie egale.