--outer join

-- nume ang titluri fct ocupate anterior si dep in care au lucrat

-- daca un ang nu a ocupat alte fct in afara de cea prezenta se va afisa titlul jobului si dep actual

Select e.first\_name, e.last\_name, j.job\_title, d.department\_name, jh.job\_id, jh.department\_id, da.department\_id department\_istoric, nvl(ja.job\_title, j.job\_title), job\_istoric

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

Join jobs j on (e.job\_id = j.job\_id)

left Join job\_history jh on (e.employee\_id = jh.employee\_id) /\* aici se asigura ca pt ang fara istoric li se asigneaza coloane nule \*/

left join jobs ja on (ja.job\_id = jh.job\_id)

left join departments da on (da.department\_id = jh.department\_id)

/\* se vor folosi iarasi departamentulsi jobul pentru ca ele trb sa ap si la istoric si la jobul actual \*/

--ex 2

Să se afişeze codul şi numele angajaţilor care lucrează în acelasi departament cu cel puţin un angajat al cărui nume conţine litera “t”. Se vor afişa, de asemenea, codul şi numele departamentului respectiv. Rezultatul va fi ordonat alfabetic după nume. Se vor da 2 soluţii pentru join (condiţie în clauza WHERE şi sintaxa introdusă de standardul SQL3).

Select distinct e.last\_name, e.department\_id, d.department\_name /\* ca sa nu apara degeaba de mai multe ori ang \*/

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

Join employees c on (c.department\_id = e.department\_id and c.last\_name like ‚%t%’)

Order by e.last\_name

--ex 3

Sa se afiseze numele, salariul, titlul job-ului, oraşul şi ţara în care lucrează angajatii condusi direct de King.

Select e.last\_name, e.salary, j.job\_title, l.city

From employees e join employees k on (e.manager\_id = k.employee\_id and k.last\_name = ‚King’),

Join departments d on (e.department\_id = d.department\_id)

Join locations l on (d.location\_id = l.location\_id)

Join jobs j on (e.job\_id = j.job\_id)

--ex 4, 5

Select e.last\_name, tochar(e.salary, ‚$99,999.00’), j.job\_title, l.city

From employees e join employees k on (e.manager\_id = k.employee\_id and k.last\_name = ‚King’),

Join departments d on (e.department\_id = d.department\_id)

Join locations l on (d.location\_id = l.location\_id)

Join jobs j on (e.job\_id = j.job\_id)

Where l.city = ‚Oxford’

Order by e.last\_name;

--Ex 6

Afisati codul, numele si salariul tuturor angajatilor care castiga mai mult decat salariul mediu si lucreaza intr-un departament cu cel putin unul din angajatii al caror nume contine litera “t”.

-- avg(salary), min(salary), max(salary), sum(salary), count(\*) nr\_ang/count (commission\_pct) nr\_ang\_cu\_comision

Select e.employee\_id, e.last\_name

From employees e Join employees c on (e.departments\_id = c.department\_id and c.last\_name like ‘%t%’)

Where e.salary >= (select avg(salary) from employees) /\* trebuie neaparat sa fol fct cu o grupare -> select e 100% necesar\*/

Order by e.employee\_id, e.last\_name

------varianta fara join

Select distinct e.employee\_id, e.last\_name

From employees e

Where e.salary >= (select avg(salary) from employees) /\* trebuie neaparat sa fol fct cu o grupare -> select e 100% necesar\*/ !!!!! NU am e.salary pt ca ar fi trb sa am automat o grupare , dar cerinta nu specifica ca ar fi necesara

And department\_id in (select department\_id from employees where last\_name like ‘%t%’)

Order by e.employee\_id, e.last\_name

--numele dep in care nu lucreaza angajati

Select department\_name

From departments

Where department\_id noi in (select department\_id from employees where department\_id is not null) ------- IS NOT NULL necesar pt a nu avea eroarea de a nu aparea niciun rez

--ex17 Folosind subcereri, să se afişeze numele şi salariul angajaţilor conduşi direct de preşedintele companiei (acesta este considerat angajatul care nu are manager).

Select e.last\_name, to\_char(e.salary, ‚$99,999.00’), j.job\_title, l.city oras, l.country\_id tara

From employees e

Join departments d on (e.department\_id = d.department\_id)

Join locations l on (d.location\_id = l.location\_id)

Join jobs j on (e.job\_id = j.job\_id)

Where e.manager\_id in (select employee\_id

From employees

Where initcap(last\_name = ‚king’) and manager\_id is null)

--subcereri == joinuri

--15 Folosind subcereri, să se afişeze numele şi data angajării pentru salariaţii care au fost angajaţi după Gates. ( Pot fol op >= pt date)

select e.last\_name, e.hire\_date

from employees e

where to\_char(e.hire\_date, 'dd-mm-yyyy')

> (select to\_char(hire\_date, 'dd-mm-yyyy')

from employees

where last\_name like 'Gates');

--18 Scrieti o cerere pentru a afişa numele, codul departamentului si salariul angajatilor al caror număr de departament si salariu coincid cu numarul departamentului si salariul unui angajat care castiga comision.

Select last\_name, department\_id

From employees

Where (department\_id, salary) in ( select department\_id , salary

From employees

Where commission\_pct is not null)

Select last\_name, department\_id

From employees

Where salary >= ALL(select salary from employees)

->mai mare decat MAX select max(salary) from employees

Where salary >= any(select salary from employees)

* Echiv Where salary = in (select salary from employees)

---10 Se cer codurile departamentelor al căror nume conţine şirul “re” sau în care lucrează angajaţi având codul job-ului “SA\_REP”. Cum este ordonat rezultatul?

Select department\_id

From departments

Where departments\_name like ‚%re%’

Union select department\_id

From employees where job\_id like ‚%SA\_REP%’

Order by 1

* Utilizand union all apar duplicate

->UNION COMBINA REZULTATELE OBT DIN 2 SAU MAI MULTE INTEROGARI

-> UNION = SAU (op pe multimi)

--12. Sa se obtina codurile departamentelor in care nu lucreaza nimeni (nu este introdus niciun salariat in tabelul employees). Se cer două soluţii.

Select department\_id

From departments

Minus

Select department\_id

From employees

Order by 1

---SAU TODO

select d.department\_name, count (\*) nr

from departments d join employees e

on (e.department\_id = d.department\_id)

group by d.department\_name

--13 Se cer codurile departamentelor al căror nume conţine şirul “re” şi în care lucrează angajaţi având codul job-ului “HR\_REP”.

Select department\_id

From departments

Where lower (department\_name) like ‚%re%’

Intersect

Select department\_id

From employees

where job\_id like ‚%HR\_REP%’

Order by 1

--media salariilor celor care au fost ang in 2000 si media tuturor ang

Select round(avg(salary), 2) valoare, media\_salariilor\_2000 info

From employees

Where to\_char(hire\_date, ‚yyyy’) = ‚2000’

Union

Select round(avg(salary), 2) valoare, media\_salariilor\_2000 info

From employees

---ex 20

Scrieti o cerere pentru a afisa angajatii care castiga mai mult decat oricare functionar (job-ul conţine şirul “CLERK”). Sortati rezultatele dupa salariu, in ordine descrescatoare. (ALL). Ce rezultat este returnat dacă se înlocuieşte “ALL” cu “ANY”?

Select e.first\_name, e.last\_name,

From employees e join jobs j on (e.job\_id = j.job\_id and lower(j.job\_name) like ‚%clerk%’)

Where salary = MAX( select salary from employees)

Order by 1 descending

--ex 21

Scrieţi o cerere pentru a afişa numele, numele departamentului şi salariul angajaţilor care nu câştigă comision, dar al căror şef direct coincide cu şeful unui angajat care câştigă comision.

Select e.last\_name, e.first\_name, d.department\_name, e.salary

From employees e join employees m on (e.manager\_id = m.employee\_id)

Join employees a on (m.employee\_id = a.manager\_id and a.commission\_pct is not null)