NATURAL JOIN presupune existenţa unor coloane având acelaşi nume în ambele tabele.

Clauza determină selectarea liniilor din cele două tabele, care au valori egale în aceste coloane

---- urmatoarele variante sunt echivalente

Select e.last\_name, department\_id, manager\_id, department\_name

From employees e natural join departments d

Select e.last\_name, department\_id, manager\_id, department\_name

From employees e join departments d using (department\_id, manager\_id)

* **Acestea sunt coloanele egale**

Select e.last\_name, e.department\_id, e.manager\_id, d.department\_name

From employees e join departments d

On (e.department\_id = d.department\_id and e.manager\_id = d.manager\_id)

------------------------------

--ex2

Să se afişeze cel mai mare salariu, cel mai mic salariu, suma şi media salariilor tuturor angajaţilor. Etichetaţi coloanele Maxim, Minim, Suma, respectiv Media. Sa se rotunjeasca rezultatele.

Select max(salary) sal\_max, min(salary) sal\_min, sum(salary) sal\_suma, avg(salary) sal\_medie

From employees

--ex3

Să se afişeze minimul, maximul, suma şi media salariilor pentru fiecare job.

Select job\_id, salary

From employees

Order by job\_id

---- VS

Select job\_id, salary

From employees

group by job\_id

---- DE CE AM NEVOIE DE GROUP BY ???

select job\_id job, max(salary), min(salary) sal\_min, count(\*) nr\_ang, count(commission\_pct) ang\_cu\_comision, count(distinct manager\_id) coduri\_manager

from employees

group by job\_id

///\* cate elem sunt in acelasi grup

//// cate elem au commission pct

///cate elem cu manageri distincti sunt in acelasi gr

--ex5

Să se determine numărul de angajaţi care sunt şefi. Etichetati coloana “Nr. manageri”.

Select count(distinct manager\_id) nr\_manageri

From employees;

--ex6

Select max(salary) – min(salary) “diferenta salariu”

From employees

---GROUP BY -> remove duplicates of the values when joining;

Fiecare employee are departamentul sau, deci pt fiecare se afiseaza dep, insa mie imi trb doar sa stiu dep, nu si cat de folosite sunt

--ex7 Scrieţi o cerere pentru a se afişa numele departamentului, locaţia, numărul de angajaţi şi salariul mediu pentru angajaţii din acel departament. Coloanele vor fi etichetate corespunzător.

Select d.department\_id, d.department\_name

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

Group by d.department\_id, d.department\_name; -> pt a scoate datele ordonat, fara duplicate

**!!!! E recomandat sa adaug in group by toate elem mentionate in select, altfel uneori apar erori**

--ex8 Să se afişeze codul şi numele angajaţilor care câstiga mai mult decât salariul mediu din firmă. Se va sorta rezultatul în ordine descrescătoare a salariilor.

Select employee\_id, last\_name, salary

From employees

Where salary + salary \* nvl(commission\_pct, 0) >= (select avg(salary + salary \* nvl(commission\_pct, 0) )

From employees)

Select avg (commission\_pct), sum(commission\_pct)/count(\*) ->impartire la toate nr de pers

Sum(commission\_pct)/count(commission\_pct)

From employees

--ex9 Pentru fiecare şef, să se afişeze codul său şi salariul celui mai prost platit subordonat. Se vor exclude cei pentru care codul managerului nu este cunoscut. De asemenea, se vor exclude grupurile în care salariul minim este mai mic de 1000$. Sortaţi rezultatul în ordine descrescătoare a salariilor.

select ang.manager\_id, man.last\_name, man.first\_name, min(ang.salary)

from employees ang join employees man on (ang.manager\_id = man.employee\_id)

group by ang.manager\_id, man.last\_name, man.first\_name

having min(ang.salary) > 2500!!!!!!!!!!! CE E IN SELECT TRB SA APARA SI IN GROUP BY

mai jos, se vor grupa si angajatii care nu au manager\_id si se va afisa suma salariilor lor (exact ca un left join cu suma)

select manager\_id, min(salary)

from employees

where manager\_id is not null /// where inainte de gruparem altfel ..........

group by manager\_id

having min(salary) > 2500

order by min(salary) desc

---varianta echivalenta in care accept ang cu manager null

select ang.manager\_id, man.last\_name, man.first\_name, min(ang.salary)

from employees ang left join employees man on (ang.manager\_id = man.employee\_id)

group by ang.manager\_id, man.last\_name, man.first\_name;

having min(ang.salary) > 2500

-- HINT - ONLY join -> pt toti angajatii…

Group by -> pt fiecare departamnet, pt departamente…

--ex 10 Pentru departamentele in care salariul maxim depăşeşte 3000$, să se obţină codul, numele acestor departamente şi salariul maxim pe departament.

Select d.department\_name, d.department\_id, max(e.salary)

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

Group by d.department\_name, d.department\_id

Having max(e.salary) >3000

---ex 13 Să se afişeze maximul salariilor medii pe departamente.

Select max(medie)

From (

Select avg(salary) medie

From employees

Group by department\_id )

---14 Să se obtina codul, titlul şi salariul mediu al job-ului pentru care salariul mediu este minim.

Select e.job\_id, j.job\_title, avg(salary)

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

Group by e.job\_id, j.job\_title

Having avg(salary) = (Select min(medie) From (

Select avg(salary) medie

From employees

Group by job\_id ) )

)

//mininul din medie

////////////// 16 NU E INCOMPLET??

---ex16 . Să se afişeze nume dep si nume job si suma salariilor pe departamente şi, în cadrul acestora, pe job-uri.

select d.department\_name nume\_dep, j.job\_title title\_job, sum(e.salary) suma

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

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

group by e.department\_id, e.job\_id, d.department\_name, j.job\_title;

--- sa se afis angajatii care castiga cel mai mjult in dep in care lucreaza

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

From employees e

Where e.salary >= (select max(salary))

From employees

Where department\_id = e.departmnet\_id)

---sau

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

From employees e

Where (e.department\_id , e.salary) in (select department\_id, max(salary) from employees)

Group by department\_id

---ex 17 sa se afis denumirea , salariu max si nr de ang din departamentul unde se castiga sal min

->imi trb departamentul

->nu pot folosi la where subcererea pt ca ar ramane angajtii care sunt ok, nu dep lor

select d.department\_name nume\_dep,

count(\*) nr\_ang,

max(e.salary) sal\_max

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

group by d.department\_name

having min(salary) = (select min(salary) from employees);

// e.salary == salary pt having !!!!! DE CE ??????

---ex31 Să se afişeze numele, salariul, codul departamentului si salariul mediu din departamentul respectiv.

select e.last\_name, e.salary, e.department\_id, d.medie --> nu fol department, ci un nou tabel format din (... ) d , adica un table cu STRICT CE MA INTERESEAZA

From employees e join (Select department\_id, avg(salary) medie

From employees

Group by department\_id) d

On e.department\_id = d.department\_id;

!!!!!! am nevoie de grupare aici... pt ca avg sa stie care e criteriul de medie altfel

ERR : "not a single-group group function"

--18. Sa se afiseze codul, numele departamentului si numarul de angajati care lucreaza

--in acel departament pentru:

--a) departamentele in care lucreaza mai putin de 4 angajati;

--b) departamentul care are numarul maxim de angajati

--a)

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

from employees e join departments d on

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

group by d.department\_id, d.department\_name

having count (\*) < 4

--b)

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

from employees e join departments d on

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

group by d.department\_id, d.department\_name

having count (\*) = (

--subcerere care returneaza nr max de ang din dep

max ( ) nu poate fi accesat fara SELECT

select max(nr\_ang) from(

select count (\*) nr\_ang

from employees

group by department\_id

)

);

--19. Sa se afiseze salariatii care au fost angajati în aceeaşi zi a lunii în care cei mai multi

--dintre salariati au fost angajati.

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

from employees e join ( -- creez un tabel cu ziua "maxima"

select to\_char(hire\_date, 'dd') zi, count(\*)

from employees

group by to\_char(hire\_date, 'dd')

having count(\*) = (

select max(nr\_ang) from(

select count(\*) nr\_ang

from employees

group by to\_char(hire\_date, 'dd')

)

)

) zimax

on to\_char(e.hire\_date, 'dd') = zimax.zi;

--20. Să se obţină numărul departamentelor care au cel puţin 15 angajaţi.

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

from employees e join departments d on

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

group by d.department\_id, d.department\_name

having count (\*) >= 15; la sfr pt ca numara pe gruparea dupa departament

--21. Să se obţină codul departamentelor şi suma salariilor angajaţilor care lucrează în

--acestea, în ordine crescătoare. Se consideră departamentele care au mai mult de 10

--angajaţi şi al căror cod este diferit de 30.

select d.department\_id, d.department\_name, sum(e.salary)

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

where e.department\_id <> 30 //ceva local, care e o conditie de baza pt grupare

group by d.department\_id, d.department\_name

having count (employee\_id) > 10; la sfr pt ca numara pe gruparea dupa departament

--22. Sa se afiseze codul, numele departamentului, numarul de angajati si

--salariul mediu din departamentul respectiv.

--Se vor afişa şi departamentele fără angajaţi (outer join).

select d.department\_id, nvl(d.department\_name, 'dep\_necunoscut') dep,

count(e.employee\_id) nr\_ang, round (avg(salary), 2) medie

from employees e left join departments d on

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

group by d.department\_id, d.department\_name;

--27. Scrieţi o cerere pentru a afişa job-ul, salariul total pentru job-ul respectiv pe departamente

--si salariul total pentru job-ul respectiv pe departamentele 30, 50, 80.

--Se vor eticheta coloanele corespunzător. Rezultatul va apărea sub forma de mai jos:

--Job Dep30 Dep50 Dep80 Total

-- .........................................

SELECT DISTINCT

TO\_CHAR(e.hire\_date, 'dd') zi\_luna,

(

SELECT

COUNT(\*)

FROM

employees

WHERE

TO\_CHAR(hire\_date, 'dd') = TO\_CHAR(e.hire\_date, 'dd')

AND department\_id = 80

) nr\_ang\_dep\_80,

(

SELECT

COUNT(\*)

FROM

employees

WHERE

TO\_CHAR(hire\_date, 'dd') = TO\_CHAR(e.hire\_date, 'dd')

AND department\_id = 50

) nr\_ang\_dep\_50,

(

SELECT

COUNT(\*)

FROM

employees

WHERE

TO\_CHAR(hire\_date, 'dd') = TO\_CHAR(e.hire\_date, 'dd')

AND department\_id = 60

) nr\_ang\_dep\_60,

(

SELECT

COUNT(\*)

FROM

employees

WHERE

TO\_CHAR(hire\_date, 'dd') = TO\_CHAR(e.hire\_date, 'dd')

) total

FROM

employees e

ORDER BY

zi\_luna;

====

select to\_char(e.hire\_date, 'dd'), e.department\_id,

decode(e.department\_id, 50, 1, 0 ),

decode(e.department\_id, 80, 1, 0 ),

decode(e.department\_id, 60, 1, 0 ),

e.employee\_id

from employees e

order by to\_char(e.hire\_date, 'dd'), department\_id;

select to\_char(e.hire\_date, 'dd'),

sum(decode(e.department\_id, 50, 1, 0 )) dep\_50,

sum(decode(e.department\_id, 80, 1, 0 )) dep\_80,

sum(decode(e.department\_id, 60, 1, null )) dep\_60, sau 0 in loc de null

count(\*) total

from employees e

group by to\_char(e.hire\_date, 'dd');

///DC NU ASA?

select e.job\_id

sum(decode(e.department\_id, 50, 1, 0 )) dep\_50,

sum(decode(e.department\_id, 80, 1, 0 )) dep\_80,

sum(decode(e.department\_id, 60, 1, null )) dep\_60, sau 0 in loc de null

count(\*) total

from employees e

group by job\_id;

!!!

where nu fct direct cu sum avg, min , max, insa poate fct daca am subcerere

having fct cu ele

--28. Să se creeze o cerere prin care să se afişeze numărul total de angajaţi şi, din acest total,

--numărul celor care au fost angajaţi în 1997, 1998, 1999 si 2000. Denumiti capetele de

--tabel in mod corespunzator.

select count (decode(to\_char(hire\_date, 'yyyy'), '1997' , 1, null )) "1997",

count (decode(to\_char(hire\_date, 'yyyy'), '1998' , 1, null )) "1998",

count (decode(to\_char(hire\_date, 'yyyy'), '1999' , 1, null )) "1999",

count (decode(to\_char(hire\_date, 'yyyy'), '2000' , 1, null )) "2000",

count(\*) total

from employees;