

## **Baze de date-Anul 1**

### **Solutii – Lab2**

---

--LABORATOR 2 - continuare

5. Să se afișeze detalii despre salariații care au lucrat un număr întreg de săptămâni până la data curentă.

```
SELECT *  
FROM employees  
WHERE MOD(ROUND(sysdate - hire_date), 7) = 0;
```

6. Să se afișeze codul salariatului, numele, salariul, salariul mărit cu 15%,  
exprimat cu două zecimale și numărul de sute al salariului nou rotunjit la 2 zecimale.  
Etichetați ultimele două coloane "Salariu nou", respectiv "Numar sute".  
Se vor lua în considerare salariații al căror salariu nu este divizibil cu 1000.

```
SELECT employee_id, last_name, salary,  
       round(salary + 0.15 * salary, 2) "Salariu Nou",  
       round((salary + 0.15 * salary) / 100, 2) "Numar sute"  
FROM employees  
WHERE MOD(salary, 1000) != 0;
```

--MOD(salary, 1000) <> 0

--MOD(salary, 1000) ^= 0

/\*

salary + 15 % \* salary

salary + 0.15 \* salary

sau:

salary \* 1.15

\*/

8. Să se afișeze data (numele lunii, ziua, anul, ora, minutul si secunda) de peste 30 zile.

```
SELECT TO_CHAR(SYSDATE + 30, 'MONTH DD YYYY HH24:MI:SS') "Data"
FROM DUAL;
```

10. a) Să se afișeze data de peste 12 ore.

Data de peste 12 ore -> adica 12 ore dintr-o zi => 12 ore din 24 de ore -> 12/24 -> 1/2

```
SELECT TO_CHAR(SYSDATE + 12/24, 'DD/MM HH24:MI:SS') "Data"
FROM DUAL;
```

b) Să se afișeze data de peste 5 minute

Obs: Cât reprezintă 5 minute dintr-o zi?

Cat reprezinta 5 min dintr-o ora?

1h ----- 60 min

x ----- 5 min

$$x = 5/60 = 1/12$$

Cat reprezinta 5 minute dintr-o zi (24 de ore)?

$$(1/12) / 24 = 1/12 * 1/24 = 1/288;$$

```
SELECT TO_CHAR(SYSDATE + 1/288, 'DD/MM HH24:MI:SS') "Data"
FROM DUAL;
```

--11

```
SELECT concat(last_name, first_name), hire_date,  
       NEXT_DAY(ADD_MONTHS(hire_date, 6), 'monday') "Negociere"  
FROM employees;
```

--12

-- prima varianta de ordonare

```
SELECT last_name, round(months_between(sysdate, hire_date)) "Luni lucrate"  
FROM employees  
ORDER BY MONTHS_BETWEEN(SYSDATE, hire_date);
```

-- a doua varianta de ordonare

```
SELECT last_name, round(months_between(sysdate, hire_date)) "Luni lucrate"  
FROM employees  
ORDER BY 2;
```

-- a treia varianta de ordonare

```
SELECT last_name, round(months_between(sysdate, hire_date)) "Luni lucrate"  
FROM employees  
ORDER BY "Luni lucrate";
```

-- ATENTIE!!!

13. Să se afișeze numele angajaților și comisionul.

Dacă un angajat nu câștigă comision, să se scrie "Fara comision".

Etichetați coloana "Comision".

NVL (expr1, expr2) - daca expr1 este NULL, intoarce expr2,

- altfel intoarce expr1

!!! Tipurile celor două expresii trebuie să fie compatibile

sau expr\_2 să poată fi convertit implicit la expr\_1.

```
SELECT last_name, NVL(TO_CHAR(commission_pct), 'Fara comision') "Comision"
FROM employees;
```

14. Să se listeze numele, salariul și comisionul tuturor angajaților al căror venit lunar (salariu + valoare comision) depășește 10 000;

```
SELECT last_name, salary, commission_pct, salary + salary * NVL(commission_pct, 0) "Venit lunar"
FROM employees
WHERE salary + salary * NVL(commission_pct, 0) > 10000;
```

```
SELECT 10 + NULL
FROM DUAL;
```

```
SELECT 10 + 0
FROM DUAL;
```

```
DESC EMPLOYEES;
```

```
-- JOIN
```

```
SELECT * FROM employees;
```

```
SELECT * FROM departments;
```

```
--EXEMPLU:
```

```
SELECT employee_id, d.department_id, last_name, department_name
FROM employees e, departments d
WHERE department_id = department_id and d.department_id = 20;
```

--INNER JOIN returnează rândurile (rows) pentru care condiția de join este îndeplinită.

-- I. Condiția de Join este scrisă în clauza WHERE a instrucțiunii SELECT

```
SELECT employee_id, d.department_id, last_name, department_name  
FROM employees e, departments d  
WHERE e.department_id = d.department_id;
```

--JOIN SCRIS IN FROM (standardul SQL3) - folosind ON

```
SELECT employee_id, d.department_id, last_name, department_name  
FROM employees e JOIN departments d ON (e.department_id = d.department_id);
```

--JOIN SCRIS IN FROM (standardul SQL3) - folosind USING

-- USING SE UTILIZEAZA ATUNCI CAND coloana de legatura are aceeasi denumire in ambele tabele

```
SELECT employee_id, department_id, last_name, department_name  
FROM employees JOIN departments USING (department_id);
```

Cele doua variante (join in where si join in from) sunt echivalente.

Sunt afisati 106 angajati din totalul de 107 -> De ce?

Ce observați având în vedere numărul de rânduri returnate?

R: pot fi angajati fara departament

--CUM II AFLAM SI PE ACESTIA?

-- pentru a afisa si angajatii care nu au departament se utilizeaza

-- simbolul (+) in partea deficitara de informatie

-- deficit de informatie -> angajati FARA departament

```
SELECT employee_id, d.department_id, last_name, department_name  
FROM employees e, departments d  
WHERE e.department_id = d.department_id (+);
```

-- deficit de informatie -> departamente FARA angajati

```
SELECT employee_id, d.department_id, last_name, department_name  
FROM employees e, departments d  
WHERE e.department_id (+) = d.department_id;
```

17. Să se listeze codurile și denumirile job-urilor care există în departamentul 30.

```
select e.job_id, job_title  
from jobs j, employees e  
where department_id = 30 and j.job_id = e.job_id;
```