

Baze de Date

Solutii – Laborator 2 – Saptamana 3

-- LABORATOR 2 - SAPTAMANA 3

-- LABORATOR 1

15. Să se afișeze numele și job-ul pentru toți angajații care nu au manager.

```
SELECT last_name, job_id, manager_id  
FROM employees  
WHERE manager_id is null;
```

```
SELECT * FROM EMPLOYEES;
```

16. Să se afișeze numele, salariul și comisionul pentru toți salariații care câștigă comision.

Să se sorteze datele în ordine descrescătoare a salariilor și comisioanelor.

```
SELECT last_name, salary, commission_pct  
FROM employees  
WHERE commission_pct is not null  
ORDER BY salary desc, commission_pct desc;
```

```
SELECT last_name, salary, commission_pct
FROM employees
--WHERE commission_pct is not null
ORDER BY commission_pct desc;
```

-- LABORATOR 2 - SAPTAMANA 3

```
SELECT TO_DATE('18-03-2022', 'DD-MM-YYYY')
FROM DUAL;
```

```
SELECT TO_CHAR(SYSDATE, 'DD/MM/YYYY')
FROM DUAL;
```

```
SELECT TO_NUMBER ('-25.789', 99.999)
FROM DUAL;
```

```
SELECT RTRIM ('info ')
FROM DUAL;
```

```
SELECT INSTR (LOWER('AbC aaacDe'), 'aa', 5, 3)
FROM DUAL;
```

```
SELECT TRIM (BOTH 'X' FROM 'XinfoXxX')  
FROM DUAL;
```

```
-- expr_date -/+ expr_number
```

```
SELECT '07-03-2023' + 3  
FROM DUAL; -- EROARE
```

```
SELECT TO_CHAR(TO_DATE('07/03/2023', 'DD/MM/YYYY') + 3,  
'DD/MM/YYYY') DATA  
FROM DUAL;
```

```
-- expr_date1 – expr_date2
```

```
SELECT ROUND(SYSDATE - TO_DATE('10-07-2020', 'DD-MM-YYYY'),  
2) "NR ZILE"  
FROM DUAL;
```

NVL(expr_1, expr_2) -- tipurile celor doua expresii trebuie sa fie compatibile

-- sau expr 2 sa se converteasca automat la expr 1

-- sau vom aplica o conversie explicita

```
SELECT NVL (1, 'a')
```

FROM DUAL; -- eroare deoarece tipurile de date ale celor doua expresii nu sunt compatibile

-- si expr 2 nu se converteste automat la expr 1

-- trebuie aplicata o conversie explicita

```
SELECT NVL (TO_CHAR(1), 'a')
```

```
FROM DUAL;
```

2. Scrieți o cerere prin care să se afișeze prenumele salariatului cu prima litera majusculă și toate celelalte litere minuscule, numele acestuia cu majuscule și lungimea numelui, pentru angajații al căror nume începe cu J sau M sau care au a treia literă din nume A. Rezultatul va fi ordonat descrescător după lungimea numelui. Se vor eticheta coloanele corespunzător. Se cer 2 soluții

(cu operatorul LIKE și funcția SUBSTR);

-- LIKE

```
SELECT INITCAP(first_name) "Prenume", upper(last_name) "Nume",  
length(last_name) "Lungime Nume"
```

```
FROM employees
```

```
WHERE upper(last_name) like 'J%'
```

OR upper(last_name) like 'M%' -- pentru angajații al căror nume începe cu J sau M

OR upper(last_name) like '___A%' -- care au a treia literă din nume A

```
ORDER BY "Lungime Nume" desc;
```

```
-- SUBSTR
```

```
-- SUBSTR(string, start [,n])
```

```
SELECT INITCAP(first_name) "Prenume", upper(last_name) "Nume",  
       length(last_name) "Lungime Nume"
```

```
FROM employees
```

```
WHERE substr(upper(last_name), 1, 1) = 'J'
```

```
      OR substr(upper(last_name), 1, 1) = 'M' -- pentru angajații al căror nume  
începe cu J sau M
```

```
      OR substr(upper(last_name), 3, 1) = 'A' -- care au a treia literă din nume A
```

```
ORDER BY "Lungime Nume" desc;
```

4. Să se afișeze pentru toți angajații al căror nume se termină cu litera 'e',
codul, numele, lungimea numelui și poziția din nume în care apare prima dată
litera 'A'. Utilizați alias-uri corespunzătoare pentru coloane.

```
-- poziția din nume în care apare prima dată litera 'A'
```

```
-- INSTR(string, 'chars' [,start [,n]])
```

```
INSTR(upper(last_name), 'A', 1, 1)
```

-- angajații al căror nume se termină cu litera 'e'

lower(last_name) like '%e'

substr (lower(last_name), -1) = 'e'

--cererea

select employee_id "Cod ang", last_name "Nume Ang", length(last_name)
"Lung Nume",

INSTR(upper(last_name), 'A', 1, 1) "Pozitie litera A"

from employees

where --substr (lower(last_name), -1) = 'e';

lower(last_name) like '%e';