

Tema 1 laborator BAZE DE DATE

--ex 1

/*Ex1: Sa se afiseze numele si prenumele, codul jobului si data la care au inceput lucrul salariatii angajati intre 03 Ianuarie 1990 si 07 Decembrie 1994. Se vor exclude din rezultat capetele intervalului. Se va utiliza operatorul BETWEEN, iar rezultatul se va ordona descrescator dupa numarul zilei. Afisati numele si prenumele pe o singura coloana numita Nume si Prenume. */

/*Abordare: Se foloseste operatorul between pentru a gasi salariatii angajati intre 03 Ianuarie 1990 si 07 Decembrie 1994 si operatorii de comparatie > si < pentru a exclude din rezultat capetele intervalului*/

```
SELECT first_name || ' ' || last_name "Nume si Prenume", job_id, hire_date
FROM employees
WHERE hire_date between '03-Jan-1990' and '07-Dec-1994' and hire_date > '03-Jan-1990' and
hire_date < '07-Dec-1994'
ORDER BY to_char(hire_date, 'dd') desc;
```

--ex 2

/*Ex2: Sa se listeze numele si salariile angajatilor care castiga mai mult de 1500 si lucrează in departamentul 10 sau 30 si au exact doua caractere 'A' in nume (nu neaparat consecutive). Se vor eticheta coloanele drept Angajat si Salariu lunar.*/

/*Abordare: Cu operatorul and se verifica indeplinirea tuturor conditiilor, cu operatorul in se verifica daca departamentul are id-ul 10 sau 30, functia upper se foloseste pentru a putea numara atat A-urile, cat si a-urile din nume, iar operatorul LIKE se foloseste pentru a gasi numele cu exact doua caractere 'A'.

```
SELECT last_name "Angajat", salary "Salariu lunar"
FROM employees
WHERE (upper(last_name) LIKE 'A%A%' and salary>1500 and department_id in (10,30));
```

--ex3

/*EX3: Sa se afiseze numele, codul jobului si salariul pentru toti salariatii al
caror job contine sirul "CLERK" sau "REP" si salariul nu este egal cu 1000, 2000 sau
3000. Atentie la ordinea efectuarii operatiilor si la precedenta operatorilor. Explicati!*/

/*Abordare: S-a folosit aceasta ordine a efectuarii operatiilor si aceasta precedenta a operatorilor
pentru a se respecta ambele conditii din cerinta. Daca s-ar fi scris WHERE job_id LIKE
'%CLERK%' or job_id LIKE '%REP%' and salary not in (1000, 2000, 3000); nu s-ar fi respectat
si conditia referitoare la salariu, afisandu-se si angajatii cu salarii de 1000, 2000 sau 3000 de lei.*/

```
SELECT last_name, job_id, salary
FROM employees
WHERE salary not in (1000, 2000, 3000) and (job_id LIKE '%CLERK%' or job_id LIKE '%REP%');
```

--ex4

/*EX4: Scrieti o cerere care are urmatorul rezultat pentru fiecare angajat:

<prenume_angajat> <nume angajat> castiga <salariu> lunar dar doreste
<salariu de 3 ori mai mare>

Exemplu: Steven King castiga 24000 lunar dar doreste 72000

Etichetati coloana Salariu ideal. Pentru concatenare, utilizati atat functia

CONCAT cat si operatorul "||".*/

```
SELECT first_name||' '||last_name||' castiga '||salary||concat(' lunar, dar doreste ', 3*salary) "Salariu  
ideal"
FROM employees;
```

--ex5

/*EX5: Sa se afiseze, pentru angajatii cu prenumele „Steven”, codul, numele si prenumele acestora, precum si codul departamentului in care lucreaza. Cautarea trebuie sa nu fie case-sensitive, iar eventualele blank-uri (spatii) care preced sau urmeaza numelui trebuie ignorate. Adica se presupune ca exista spatii libere atat inainte, cat si dupa nume, si se sterg utilizandu-se o functie destinata prelucrarii sirurilor de caractere.*/

/*Abordare: Se foloseste functia RTRIM pentru a sterge spatiile de dupa nume, iar functia LTRIM pentru a le sterge pe cele dinaintea numelui. */

```
SELECT job_id, last_name, first_name, department_id
FROM employees
WHERE LTRIM(RTRIM(lower(first_name)))='steven';
```

--ex6

/*EX6:Sa se listeze numele, data angajarii si comisionul salariatilor care castiga comision. Să se eticheteze coloanele Nume angajat, Data angajarii si Comision. Utilizati functia RPAD pentru a determina ca data angajarii sa aiba lungimea de 20 de caractere, adaugand caracterul 'X'.*/

/*Abordare: Se foloseste functia RPAD(TO_CHAR(hire_date),20, 'X') pentru a fi sigur faptul ca data are 20 de caractere si este de forma specificata in cerinta. */

```
SELECT last_name "Nume angajat" , RPAD(TO_CHAR(hire_date),20, 'X') "Data angajarii" ,
commission_pct "Comision"
FROM employees
WHERE commission_pct IS NOT NULL;
```

--ex7

/*EX7: Sa se afiseze numarul de zile rămase pana la sfarsitul anului, utilizandu-se formatul 31-12-2021, in raport cu ziua curenta. Se va eticheta coloana Numar_Zile si se va rotunji cu doua zecimale dupa virgula.*/

/*Abordare: Cu sysdate from dual se poate afla data curenta. Se converteste ultima data din an cu to_date pentru a o putea scadea din ea pe cea curenta. Se adauga 1 pentru a afla cate zile sunt de la data curenta(inclusiv) pana la ultima zi din an. Functia round se foloseste pentru a rotunji rezultatul cu doua zecimale. */

```
SELECT round((to_date('31-12-2021', 'dd-mm-yyyy')-sysdate+1),2) "Numar_Zile"
FROM dual;
```

--ex8

/*EX8: Sa se listeze codul, numele departamentului si codul managerului de departament. In cazul in care un departament nu are manager se va afisa mesajul Departamentul <department_id> nu are manager (ex: DEPARTAMENTUL 120 nu are manager).*/

/*Abordare: Se foloseste functia NVL intrucat, daca departamentul are manager (adica manager_id diferit de null), atunci se afiseaza

manager_id, iar in caz contrar, afiseaza expresia a doua, adica mesajul "Departamentul <department_id> nu are manager". */

```
SELECT department_id, department_name,
NVL(TO_CHAR(manager_id),TO_CHAR('Departamentul ||department_id|| nu are manager'))
"Manager"
FROM departments;
```

--ex9

/*EX9: Scrieti o cerere care afiseaza numele angajatului, codul departamentului in care acesta lucreaza si numele colegilor sai de departament. Se vor eticheta coloanele corespunzator.*/

/*Abordare: Se folosesc etichetele a. si b. pentru a diferentia angajatul curent de colegul sau. Verificam daca cei doi au acelasi

department_id, adica daca sunt colegi de departament, precum si daca sunt persoane diferite (au employee_id-ul diferit). */

```
SELECT a.last_name "Angajat", a.department_id "Cod departament", b.last_name "Coleg"
FROM employees a, employees b
WHERE a.department_id = b.department_id
AND a.employee_id != b.employee_id
```

--ex10

/*EX10: Creati o cerere prin care sa se afiseze numele angajatilor, codul jobului, titlul job-ului, numele departamentului si salariul angajaților. Se vor include si angajatii al caror departament nu este cunoscut. */

/*Abordare: Se folosesc etichetele in mod sugestiv: e. pentru entitatile pe care le putem gasi in tabelul employees, j. pentru entitatea din tabelul jobs, d. pentru cea din tabelul departments. Se face left join la departments pentru a avea si angajatii al caror department nu este cunoscut. La al doilea join se foloseste on in loc de using pentru a afisa job_id.*/

```
SELECT e.last_name , e.job_id, j.job_title , d.department_name , e.salary
FROM employees e
LEFT JOIN departments d
USING(department_id)
JOIN jobs j
ON(j.job_id=e.job_id);
```

Alta varianta:

```
SELECT e.last_name, e.job_id, j.job_title, d.department_name, e.salary
```

```
FROM employees e, departments d, jobs j
```

```
WHERE e.job_id = j.job_id AND e.department_id = d.department_id(+);
```

/*Abordare: Se unesc tabelele employee si job prin job_id si tabelele employee si department prin department_id. Se foloseste (+) pentru a include si angajatii al caror departament nu este cunoscut.*/