

CAPITOLUL 4

FUNCTII

In acest capitol, sunt introduse functiile. Functiile fac ca cererile de baza sa devina mai puternice si sint folosite pentru manipularea valorilor. Acest capitol acopera functiile numerice si pe caracter. Functiile de conversie si care opereaza pe tipuri de date sunt tratate in cap.5. In final functiile de grup sunt tratate in cap.6.

INTRODUCERE

Functiile sint folosite pentru manipularea datelor. Ele accepta unul sau mai multe argumente si intorc o valoare. Un argument este o constanta , variabila sau o referire de coloana. Formatul pentru functie este urmatorul: `function_name(arg1,arg2,...)`

Functiile pot fi folosite pentru:

- a calcula
- a modifica datele individuale
- a manipula iesirea pentru grupuri de siruri
- a schimba formatul datelor pentru afisare
- a converti tipurile de date

Exista diferite tipuri de functii:

- CHARACTER
- NUMBER
- DATE
- CONVERSION
- FUNCTION THAT ACCEPT ANY DATA TYPE AS INPUT
- GROUP

Unele functii opereaza pe un singur sir , altele pe grupuri de siruri.

Cele mai folosite functii sunt prezentate in acest manual. Puteti recurge la SQL*Plus REFERENCE GUIDE pentru lista completa a functiilor.

Functiile care se aplica unui singur sir vor fi discutate in capitolul prezent si in cap.5. Functiile de grup vor fi tratate in cap.6.

Functii care se aplica unui singur sir.

- actioneaza asupra fiecarui sir returnat din cerere
- intoarce un rezultat pentru fiecare sir
- asteapta unul sau mai multe argumente
- pot fi imbricate
- pot fi folosite acolo unde se folosesc variabile , coloane sau expresii , de exemplu , in constructiile :SELECT, WHERE, ORDER BY.

Explicatia notatiei:

Notatie + Semnificatie:

col
un nume de coloana dintr-o baza de date
value(valoare)
orice valoare literale(caracter/data/numar)
n
reprezinta un sir
'string'
reprezinta caracterul string
chars
reprezinta un numar de caractere specificate
date
reprezinta o coloana date sau o valoare date

Functii numerice si pe caracter

Acestea accepta la intrare un caracter si poate returna si caracter si valoare numerica.

Urmatoarele functii prezentate sunt:

LOWER(col/value)

forteaza caracterele alfa care sunt scrise cu litere mari sau mixte in caractere scrise cu litere mici.

Pentru a afisa cu litere mici numele departamentului ca si sirul 'SQL COURSE' introduceti:

```
SELECT LOWER (DNAME) ,LOWER('SQL course')  
FROM DEPT;
```

```
LOWER(DNAME)          LOWER(SQL COURSE)  
-----  
research              sql course  
sales                 sql course  
operations            sql course  
accounting            sql course
```

UPPER(col/value)

forteaza caracterele alfa care sunt scrise cu litere mici , sau o combinatie de litere mici si mari in caractere scrise cu litere mari.

In urmatorul exemplu , functia UPPER e folosita pentru a forta intrarea utilizatorului la litere mari.

```
SELECT ENAME
FROM EMP
WHERE ENAME =UPPER('&ENAME');
```

Enter value for ename :smith

```
ENAME
-----
SMITH
```

INITCAP(col/value)

forteaza prima litera a fiecarui cuvint in litera mare

Pentru afisarea numelor departamentelor si locatiile , introduceti:

```
SELECT      INITCAP(DNAME),  INITCAP(LOC)
FROM        DEPT;
```

```
INITCAP(DNAME)  INITCAP(LOC)
-----
Accounting      New York
Research        Dallas
Sales           Chicago
Operations       Boston
```

CONCAT(char 1,char2)

intoarce char1 concatenat cu char2 (Alternativa a operatorului ||)

```
SELECT CONCAT (ename,job) "JOB"
FROM EMP
WHERE EMPNO =7900;
```

```
JOB
-----
JAMESCLERK
```

Functiile LPAD si RPAD aduc sirurile de caractere la o lungime specificata.

LPAD(col/value,n,'string')

adauga la coloana sau la valoarea literala spre stinga pina la lungimea totala n. Primele spatii sunt umplute cu 'string'. Daca 'string' e omis atunci sunt umplute cu blancuri.

```
SELECT LPAD(DNAME,20,'*'),LPAD(DNAME,20),LPAD(DEPTNO,20,'. ')
FROM DEPT;
```

```
LPAD(DNAME,20,'*')  LPAD(DNAME,20)          LPAD(DEPTNO,20,'. ')
-----
*****RESEARCH
RESEARCH.....20
*****SALES
SALES.....30
*****OPERATIONS
OPERATIONS.....40
```

```

*****ACCOUNTING
ACCOUNTING.....10

```

Observati ca a doua coloana e completata in stinga cu spatii, implicit , si ca a treia coloana este de tipul numar.

RPAD(col/value,n,'string')

adauga la coloana sau la valoarea literala spre dreapta pina la lungimea totala n. Ultimele pozitii sunt umplute cu 'string' sau daca acesta e omis cu blankuri.

```

SELECT RPAD (DNAME,20, '*') ,RPAD (DNAME,20) ,RPAD (DEPTNO,20, '. ')
FROM DEPT;

RPAD (DNAME,20, '*')  RPAD (DNAME,20)      RPAD (DEPTNO,20, '. ')
-----
RESEARCH***** RESEARCH
20.....
SALES***** SALES
30.....
OPERATIONS***** OPERATIONS
40.....
ACCOUNTING***** ACCOUNTING
10.....

```

In acest caz a doua coloana are in dreapta blankuri puse implicit.

Urmatoarele functii presupun ca caracterele din string sunt numerotate de la stinga la dreapta , incepind cu 1.

SUBSTR(col/value,pos,n)

intoarce un string de n caractere lungime dintr-o coloana sau valoare literala , incepind de la pozitia pos. Daca n e omis, e extras sirul din pozitia pos la sfirsit.

Urmatorul exemplu afiseaza urmatoarele "substringuri":

- 4 caractere din literalul 'ORACLE' incepind de la al doilea caracter
- continutul lui DNAME incepind cu al doilea caracter
- 5 caractere din DNAME incepind cu al treilea caracter

```

SELECT SUBSTR ('ORACLE',2,4) ,SUBSTR (DNAME,2) ,
SUBSTR (DNAME,3,5)
FROM DEPT;

SUBSTR ('ORACLE',2,4)  SUBSTR (DNAME,2)  SUBSTR (DNAME,3,5)
-----
RACL                  ESEARCH          SEARC
RACL                  ALES            LES
RACL                  PERATIONS       ERATI

```

RACL

CCOUNTING

COUNT

Observati ca valorile sint aliniate la stinga. Aceasta deoarece SQL*Plus intotdeauna afiseaza de la stinga, implicit.

INSTR(col/value,'string')

gaseste pozitia caracterului in care apare prima data 'string'.

INSTR(col/value,'string',pos,n)

gaseste pozitia caracterului pentru a n-a aparitie a lui string in coloana sau valoarea literala incepind din pozitia pos.

```
SELECT DNAME, INSTR(DNAME, 'A'),
INSTR(DNAME, 'ES'), INSTR(DNAME, 'C', 1, 2)
FROM DEPT;
```

DNAME	INSTR(DNAME, 'A')	INSTR(DNAME, 'ES')
ACCOUNTING	1	0
RESEARCH	5	2
SALES	2	4
OPERATIONS	5	0

O folosire obisnuita a lui INSTR este de a determina daca inceputul utilizatorului contine un caracter particular sau mai multe. In exemplul anterior, de exemplu, expresia INSTR(DNAME,'ES') e 0 pentru ACCOUNTING deoarece acesta nu contine sirul dat.

LTRIM si RTRIM sterg caracterele specificate din sir.

LTRIM (col/value,'char/s')

sterge de la stinga prima aparitie a caracterului specificat(sau o combinatie a caracterelor specificate). Daca nu e specificat nici un caracter sterge toate blancurile din stinga.

```
SELECT DNAME, LTRIM(DNAME, 'A'), LTRIM(DNAME, 'AS'),
LTRIM(DNAME, 'ASOP')
FROM DEPT;
```

DNAME	LTRIM(DNAME, 'A')	LTRIM(DNAME, 'AS')	LTRIM(DNAME, 'ASOP')
RESEARCH	RESEARCH	RESEARCH	RESEARCH
SALES	SALES	LES	LES
OPERATIONS	OPERATIONS	OPERATIONS	ERATIONS
ACCOUNTING	CCOUNTING	CCOUNTING	CCOUTING

RTRIM (col/value,'char/s')

sterge de la dreapta, aparitiile lui char (sau combinatie de caractere specifice) .Daca nu e specificat 'char/s' atunci sterge blancurile.

```
SELECT DNAME, RTRIM(DNAME, 'G'), RTRIM(DNAME, 'GHS')
RTRIM(DNAME, 'N')
FROM DEPT;
```

DNAME	RTRIM(DNAME, 'G')	RTRIM(DNAME, 'GHS')	RTRIM(DNAME, 'N')
RESEARCH	RESEARCH	RESEARC	RESEARCH
SALES	SALES	SALE	SALES
OPERATIONS	OPERATIONS	OPERATION	OPERATIONS
ACCOUNTING	ACCOUNTIN	ACCOUNTIN	ACCOUNTING

RTRIM poate fi in mod particular folositor in a sterge blancurile de la sfirsit la coloane.De exemplu , presupunind ca in transferul datelor la tabela 'emp' blancurile au fost adaugate neintentionat dupa fiecare ENAME.Urmatoarea comanda va sterge toate blancurile finale:

```
UPDATE EMP
SET ENAME =RTRIM(ENAME);
```

Observatie: comanda UPDATE e tratata complet mai tirziu.

SOUNDEX(col/value)

intoarce un sir de caractere reprezentind pronuntia(sunetul) cuvintului pentru fiecare coloana sau valoare literala.Aceasta functie intoarce o reprezentare fonetica a fiecarui cuvint si ne permite sa comparam cuvinte care sunt scrise diferit, dar se pronunta(suna) la fel.

```
SELECT ENAME, SOUNDEX(ENAME)
FROM EMP
WHERE SOUNDEX(ENAME) = SOUNDEX('FRED');
```

ENAME	SOUNDEX(ENAME)
FORD	F630

LENGTH(col/value)

intoarce numarul de caractere(sau digiti) din coloana sau din valoarea literala.

```
SELECT LENGTH('SQL COURSE'), LENGTH(DEPTNO), LENGTH(DNAME)
FROM DEPT;
```

LENGTH('SQLCOURSE')	LENGTH(DEPTNO)	LENGTH(DNAME)
10	2	8
10	2	5
10	2	10
10	2	10

Observati ca LENGTH la fel ca si functia INSTR intoarce o valoare numerica.

Functiile TRANSLATE si REPLACE sunt folosite pentru substitutia caracterelor.

TRANSLATE(col/value,from,to) translateaza la iesire caracterele 'from' la cele 'to'.Mai mult de un caracter , se poate potrivi. Toate operatiile lui 'from' sunt inlocuite de corespondentul din 'to'. Daca caracterul corespunzator din 'to'nu e rezervat , caracterul din 'from'e sters(vezi functiile imbricate).

```
SELECT ,ENAME,TRANSLATE(ENAME,'C','P'),JOB
TRANSLATE(JOB,'AR','IT')
FROM EMP
WHERE DEPTNO=10;

ENAME          TRANSLATE(ENAME,'C','P')  JOB
TRANSLATE(JOB,'AR','IT')
-----
--
CLARK          PLARK                        MANAGER      MINIGET
KING           KING                         PRESIDENT    PTESIDENT
MILLER         MILLER                       CLERK        CLETK
```

REPLACE(col/value,string,replacement_string)
intoarce col/value cu orice aparitie a 'string-ului', inlocuita cu replacement_string.Daca 'replacement_string' e omis, toate operatiile lui 'string'sunt sterse.Daca sunt emise si 'string' si 'replacement_string'atunci se semnaleaza eroare.

```
SELECT JOB , REPLACE(JOB,'SALESMAN','SALESPERSON'),
ENAME ,REPLACE(ENAME,'CO','PX')
FROM EMP;

JOB REPLACE(JOB,'SALESMAN','SALESPERSON') ENAME
REPLACE(ENAME,'CO','PX')
-----
ANALYST

ANALYST          SCOTT      SPXTT
SALESMAN  SALESPERSON      TURNER  TURNER
SALESMAN  SALESPERSON      ALLEN   ALLEN
MANAGER   MANAGER           CLARK   CLARK
```

Functia REPLACE suplimenteaza functionalitatea furnizata de functia TRANSLATE.TRANSLATE furnizeaza caractere singulare , unul cite unul , pentru substitutie.REPLACE va permite sa substituiti un sir cu altul ca si sa stergeti un sir (daca nu specificati 'replacement_string'). Observati ca 'string' si 'replacement_string'

pot fi de orice lungime. REPLACE poate sa faca si conversia caracter la caracter REPLACE('C', IP).

FUNCTII IMBRICATE

Funcțiile singulare aplicate liniilor(sirurilor) pot fi imbricate la orice adincime.Daca functiile sunt imbricate, ele sunt evaluate din interior spre exterior.

Sa presupunem ca vreti sa aflati nr. de ori in care un caracter apare intr-un sir.Cum veti proceda? Puteti imbrica functiile LENGHT si TRANSLATE pentru a obtine rezultatul dorit. Urmatorul exemplu va permite sa numarati de cite ori apare S intr-un sir.

```

SELECT DNAME, LENGTH (DNAME) , LENGTH (DNAME)
LENGTH (TRANSLATE (DNAME, 'AS', 'A '))
FROM DEPT;

DNAME          LENGTH (DNAME) LENGTH (DNAME) -
LENGTH (TRANSLATE (DNAME, 'AS', 'A '))
-----
RESEARCH                8
1 SALES                  5
2 OPERATIONS            10
1 ACCOUNTING            10
0

```

Pasii pentru a obtine rezultatul sunt:

1. Folositi LENGTH pentru a identifica nr. de caractere din sir.
2. Apoi folositi TRANSLATE pentru a lua fiecare aparitie a lui S din sir.
- 3.
- 4.
5.

```
SELECT TRANSLATE (DNAME, 'AS', 'A ')
```
6.

```
FROM DEPT;
```
- 7.
8.

```
TRANSLATE (DNAME, 'AS', 'A ')
```
9. -----
10. RESEARCH
11. ALE
12. OPERATION
13. ACCOUNTING
- 14.
15. Observati ca A e inlocuit cu A si S nu are un caracter corespunzator cu care sa fie inlocuit . S e inlocuit cu nimic - e sters din sir. A serveste de plasare ?
16. Acum, scadeti din lungimea initiala a sirului pe cea din care am scos toate aparitiile lui S. LENGTH(DNAME)-LENGTH(TRANSLATE(DNAME,'AS','A'))
17. Rezultatul e o valoare reprezentind nr. de aparitii ale lui S in sir.

O tehnica alternativa este de a folosi functia REPLACE.

1. Folositi REPLACE pentru a sterge fiecare aparitie a lui S din sir. Valoarea intoarsa va reprezenta nr. de caractere care au ramas dupa ce S a fost sters. Cu alte cuvinte , o valoare minus nr. de operatii ale lui S in sir.

```
2.
3.      SELECT DNAME, LENGTH(DNAME), LENGTH(REPLACE(DNAME, 'S'))
4.      FROM DEPT;
```

```
5.
6.      DNAME          LENGTH(DNAME)    ENGTH(REPLACE(DNAME, 'S'))
7.      -----
8.      ACCOUTING      10                                10
9.      RESEARCH       8                                 7
10.     SALES          5                                 3
11.     OPERATIONS    10                                9
12.
```

13. Apoi scadeti din lungimea totala , lungimea sirului fara S.

```
14.
15.     SELECT DNAME, LENGTH(DNAME), LENGTH(DNAME)
16.     LENGTH(REPLACE(DNAME, 'S'))
17.     FROM DEPT;
18.
19.     DNAME          LENGTH(DNAME)  LENGTH(DNAME) -
20.     LENGTH(REPLACE(DNAME, 'S'))
-----
```

```
21.     ACCOUTING      10
22.     RESEARCH       8
23.     SALES          5
24.     OPERATIONS    10
25.
```

26. Rezultatul e o valoare care reprezinta nr. de operatii ale lui S in sir .

FUNCTII NUMERICE

Acestea accepta intrare numerica si intorc ca rezultat un numar. Aceasta sectiune descrie citeva din functiile numerice. ROUND(col/value,n) rotujeste coloana , expresie sau valoare la n zecimale. Daca n e omis nu are zecimale , daca e negativ , numarul din stinga punctului zecimal e rotunjit.

```
SELECT      ROUND(45.923,1),
            ROUND(45.923),
            ROUND(45.323,1),
            ROUND(42.323,-1),
            ROUND(SAL/32,2)
FROM EMP
WHERE DEPTNO =10;
```

```
ROUND(45.923,1) ROUND(45.923) ROUND(45.323,1) Round(42.323,-
1) ROUND(SAL/32,2)
```

76.56	45.9	46	45.3	40
156.23	45.9	46	45.3	40
40.63	45.9	46	45.3	40

TRUNC(col/value,n) trunchiaza coloana sau valoarea la n zecimala, sau daca nu e omis, fara zecimala. Daca n e negativ, numarul din stanga punctului zecimal e trunchiat la zero.

```
SELECT      TRUNC(45.923,1),
            TRUNC(45.923),
            TRUNC(45.323,1),
            TRUNC(42.323,-1),
            TRUNC(SAL/32,2)
FROM EMP
WHERE DEPTNO= 10;
```

```
TRUNC(45.923,1) TRUNC(45.923) TRUNC(45.323,1) TRUNC(42.323,-1) TRUNC(SAL/32,2)
```

76.56	45.9	45	45.3	40
156.25	45.9	45	45.3	40
40.62	45.9	45	45.3	40

CEIL(col/value) gaseste cel mai mic intreg mai mare sau egal cu coloana, expresie sau valoare.

```
SELECT CEIL(SAL), CEIL(99.9), CEIL(101.76), CEIL(-11.1)
FROM EMP
WHERE SAL BETWEEN 3000 AND 5000;
```

CEIL(SAL)	CEIL(99.9)	CEIL(101.26)	CEIL(-11.1)
3000	100	102	-11
5000	100	102	-11
3000	100	102	-11

FLOOR(col/value) gaseste cel mai mare intreg mai mic sau egal cu coloana expresie sau valoare.

```
SELECT FLOOR(SAL), FLOOR(99.9), FLOOR(101.76), FLOOR(-11.1)
FROM EMP
WHERE FLOOR(SAL) BETWEEN 3000 AND 5000;
```

FLOOR(SAL)	FLOOR(99.9)	FLOOR(101.76)	FLOOR(-11.1)
3000	99	101	-12
5000	99	101	-12
3000	99	101	-12

POWER(col/value,n) ridica coloana, expresia sau valoarea la puterea n. Poate fi negativa.

```

SELECT SAL,POWER(SAL,2), POWER(SAL,3), POWER(50,5)
FROM EMP
WHERE DEPTNO=10;

```

SAL	POWER(SAL,2)	POWER(SAL,3)	POWER(50,5)
2450.00	6002500	14706125000	312500000
5000.00	25000000	125000000000	312500000
1300.00	1690000	2197000000	312500000

EXP(n) intoarce e ridicat la puterea n
e=2.71828183.

```

SELECT EXP(4)
FROM DUAL;

```

```

EXP(4)
-----
54.59815

```

SQRT(col/value) gaseste radacina patrata a coloanei sau
valorii.

Daca col/value e NULL sau negativ atunci e
intors rezultatul NULL.

```

SELECT SAL, SQRT(SAL),SQRT(40), SQRT(COMM)
FROM EMP
WHERE COMM>0;

```

SAL	SQRT(SAL)	SQRT(40)	SQRT(COMM)
1600.00	40	6.32455532	17.3205081
1250.00	35.3553391	6.32455532	22.3606798
1250.00	35.3553391	6.32455532	37.4165739

SIGN(col/value) intoarce -1 daca e coloana, expresie sau
valoare e
un nr. negativ , intoarce 0 daca e zero,+1 daca
e
nr.pozitiv.

```

SELECT SAL -COMM,SIGN(SAL-COMM),COMM-SAL, SIGN(COMM-SAL)
FROM EMP
WHERE DEPTNO=30;

```

SAL-COMM	SIGN(SAL-COMM)	COMM-SAL	SIGN(COMM-SAL)
1300	1	-1300	-1
750	1	-750	-1
-150	-1	150	1
1500	1	-1500	-1

Frecvent , functia SIGN este folosita pentru a testa daca o
valoare
este mai mica ca , mai mare ca sau egala cu a doua valoare.

Urmatorul exemplu tipareste toti angajatii al caror salariu
este
mai mare decit comisionul lor.

```
SELECT ENAME ,SAL,COMM
FROM EMP
WHERE SIGN (SAL-COMM)=1;
```

ENAME	SAL	COMM
ALLEN	1600	300
WARD	1250	500
TURNER	1500	0

ABS(col/value) gaseste valoarea absoluta a coloanei sau valorii.

```
SELECT SAL,COMM,COMM-SAL,ABS (COMM-SAL) ,ABS (-35)
FROM EMP
WHERE DEPTNO =30;
```

SAL	COMM	COMM-SAL	ABS (COMM-SAL)	ABS (-35)
1600.00	300.00	-1300	1300	35
1250.00	500.00	-750	750	35
1250.00	1400.00	150	150	35
2850.00				35
1500.00	00	-1500	1500	35
950.00				35

MOD(value1,value2) gaseste restul impartirii lui 'value1'la 'value2'.

```
SELECT SAL , COMM , MOD (SAL,COMM) , MOD (100,40)
FROM EMP
WHERE DEPTNO =30
ORDER BY COMM;
```

SAL	COMM	MOD (SAL,COMM)	MOD (100,40)
2,850.00			20
950.00			20
1,600.00	300.00	100	20
1,250.00	500.00	250	20
1,250.00	1,400.00	1250	20
1,500.00	00	1500	20

Alte citeva functii matematice:

LOG(m,n)

intoarce logaritmul cu baza m si argument n.

SIN(n)

intoarce sinusul lui n

SINH(n)

intoarce sinusul hiperbolic al lui n

TAN(n)

intoarce tangenta lui n

TANH(n)

intoarce tangenta hiperbolica a lui n

COS(n)

cosinus de n
COSH(n)
cosinus hiperbolic al lui n

Exercitii-cap.4- Folosirea functiilor

Aceste exercitii acopera folosirea functiilor nu doar in constructiile SELECT dar si in WHERE si ORDER BY. Daca sunt folosite aliasuri ale coloanelor in rezultat , folo-siti-le in constructia SELECT a propozitiei SQL.

TEME

1. Listati pentru toti angajatii numele si salariul marit cu 15 % si exprimat ca un nr. de dolari.

DEPTNO	ENAME	PCTSAL
20	SMITH	920
30	ALLEN	1840
30	WARD	1438
20	JONES	3421
30	MARTIN	1438
30	BLAKE	3278
10	CLARK	2818
20	SCOTT	3450
10	KING	5750
30	TURNER	1725
20	ADAMS	1265
30	JAMES	1093
20	FORD	3450
10	MILLER	1495

14 selectate.

2. Obtineti urmatoarea iesire:

```
EMPLOYEE_AND_JOB
-----
SMITH      CLERK
ALLEN     SALESMAN
WARD      SALESMAN
JONES     MANAGER
MARTIN   SALESMAN
BLAKE    MANAGER
CLARK    MANAGER
SCOTT    ANALYST
KING     PRESIDENT
TURNER   SALESMAN
ADAMS    CLERK
JAMES    CLERK
FORD     ANALYST
MILLER   CLERK
```

3. Obtineti urmatoarea iesire

```
EMPLOYEE
-----
SMITH(Clerk)
```

ALLEN (Salesman)
 WARD (Salesman)
 JONES (Manager)
 MARTIN (Salesman)
 BLAKE (Manager)
 CLARK (Manager)
 SCOTT (Analyst)
 KING (President)
 TURNER (Salesman)
 ADAMS (Clerk)
 JAMES (Clerk)
 FORD (Analyst)
 MILLER (Clerk)

4. **Faceti o cautare pentru a obtine o lista a angajatilor cu functia 'job' specificata de utilizator.**

```

Enter value for job :clerk
EMPNO  ENAME    JOB    MGR HIREDATE          SAL      COMM
DEPTNO
-----
---
      7369  SMITH    CLERK   7902 17-DEC-80         1000
20
      7876  ADAMS    CLERK   7788 12-JAN-83         1100
20
      7900  JAMES    CLERK   7698 03-DEC-81       1092.5
30
      7934  MILLER   CLERK   7782 23-JAN-82         1300
10
  
```

5. **S- a descoperit ca nu toti angajatii din departamentul 30 sunt barbati .Obtineti urmatoarea iesire:**

```

ENAME          DEPTNO      JOB
-----
ALLEN           30      Salesperson
WARD            30      Salesperson
MARTIN          30      Salesperson
BLAKE           30      Manager
TURNER          30      Salesperson
JAMES           30      Clerk
  
```

SOLUTII Cap.4

```

1. SELECT  DEPTNO,ENAME, ROUND(SAL*1.15) PCTSAL
   FROM    EMP;
2. SELECT  RPAD(ENAME,10)||LPAD(JOB,10) EMPLOYEE_AND_JOB
   FROM    EMP;
3. SELECT  ENAME || '('||initcap(job)||')' EMPLOYEE
   FROM    EMP;
4. SELECT  *
   FROM    EMP
   WHERE   UPPER(JOB)=UPPER('&JOB');
5. SELECT  ENAME,DEPTNO,INITCAP
   (REPLACE(JOB,'SALESMAN','SALESPERSON')) JOB
   FROM    EMP
   WHERE   DEPTNO =30;
  
```