Tema

1) Utilizand interogarea de la punctul 3f) scrieti o functie care sa afiseze urmatorea situatie pentru un

produs. Functia se va apela dintr-o procedura nestocata.

CREATE OR REPLACE FUNCTION f\_stoc\_produs(p\_cod ctl\_XY.cod\_p%TYPE, p\_stoc OUT NUMBER) RETURN NUMBER IS

v\_cod\_p ctl\_XY.cod\_p%TYPE;

v\_den\_p ctl\_XY.den\_prod%TYPE;

A picture containing text, font, white, typography

Description automatically generatedv\_total\_intrari NUMBER;

v\_total\_iesiri NUMBER;

v\_stoc\_produs NUMBER;

BEGIN

SELECT ta.cod\_p, ta.den\_prod, Total\_apr,NVL(Total\_vanz,0), Total\_apr-NVL(Total\_vanz,0) INTO v\_cod\_p, v\_den\_p, v\_total\_intrari, v\_total\_iesiri, v\_stoc\_produs

FROM

(SELECT ctl\_XY.cod\_p, den\_prod, tip\_op, SUM(cant) AS Total\_apr

FROM ctl\_XY INNER JOIN detalii\_misc\_XY ON ctl\_XY.cod\_p=detalii\_misc\_XY.cod\_p

GROUP BY ctl\_XY.cod\_p, den\_prod, tip\_op

HAVING UPPER (tip\_op)='A'

ORDER BY 1) ta

LEFT JOIN

(SELECT ctl\_XY.cod\_p, den\_prod, tip\_op, SUM(cant) AS Total\_vanz

FROM ctl\_XY INNER JOIN detalii\_misc\_XY ON ctl\_XY.cod\_p=detalii\_misc\_XY.cod\_p

GROUP BY ctl\_XY.cod\_p, den\_prod, tip\_op

HAVING UPPER (tip\_op)='V'

ORDER BY 1) tv

ON ta.cod\_p=tv.cod\_p

WHERE ta.cod\_p=p\_cod;

DBMS\_OUTPUT.PUT\_LINE('Cod produs Denumire Total incasari Total iesiri');

DBMS\_OUTPUT.PUT\_LINE(' '||v\_cod\_p||' '||v\_den\_p||' '||v\_total\_intrari||' '||v\_total\_iesiri);

p\_stoc:=v\_stoc\_produs;

RETURN p\_stoc;

END;

/

A picture containing text, receipt, font, white

Description automatically generatedDECLARE

v\_slct\_cod ctl\_XY.cod\_p%TYPE;

v\_rezultat NUMBER;

BEGIN

v\_slct\_cod:=&dati\_codul;

v\_rezultat:=f\_stoc\_produs(v\_slct\_cod, v\_rezultat);

DBMS\_OUTPUT.PUT\_LINE('Stocul produsului cu codul ' || v\_slct\_cod || ' este ' || v\_rezultat);

END;

/

Suplimentar incercati sa obtineti si raspunsul (tratarea erorii de negasire a unui cod).

CREATE OR REPLACE FUNCTION f\_stoc\_produs\_exceptie(p\_cod ctl\_XY.cod\_p%TYPE, p\_stoc OUT NUMBER) RETURN NUMBER IS

v\_cod\_p ctl\_XY.cod\_p%TYPE;

v\_den\_p ctl\_XY.den\_prod%TYPE;

v\_total\_intrari NUMBER;

v\_total\_iesiri NUMBER;

v\_stoc\_produs NUMBER;

BEGIN

SELECT ta.cod\_p, ta.den\_prod, Total\_apr,NVL(Total\_vanz,0), Total\_apr-NVL(Total\_vanz,0) INTO v\_cod\_p, v\_den\_p, v\_total\_intrari, v\_total\_iesiri, v\_stoc\_produs

FROM

(SELECT ctl\_XY.cod\_p, den\_prod, tip\_op, SUM(cant) AS Total\_apr

FROM ctl\_XY INNER JOIN detalii\_misc\_XY ON ctl\_XY.cod\_p=detalii\_misc\_XY.cod\_p

GROUP BY ctl\_XY.cod\_p, den\_prod, tip\_op

HAVING UPPER (tip\_op)='A'

ORDER BY 1) ta

LEFT JOIN

(SELECT ctl\_XY.cod\_p, den\_prod, tip\_op, SUM(cant) AS Total\_vanz

FROM ctl\_XY INNER JOIN detalii\_misc\_XY ON ctl\_XY.cod\_p=detalii\_misc\_XY.cod\_p

GROUP BY ctl\_XY.cod\_p, den\_prod, tip\_op

HAVING UPPER (tip\_op)='V'

ORDER BY 1) tv

ON ta.cod\_p=tv.cod\_p

WHERE ta.cod\_p=p\_cod;

DBMS\_OUTPUT.PUT\_LINE('Cod produs Denumire Total incasari Total iesiri');

DBMS\_OUTPUT.PUT\_LINE(' '||v\_cod\_p||' '||v\_den\_p||' '||v\_total\_intrari||' '||v\_total\_iesiri);

p\_stoc:=v\_stoc\_produs;

RETURN p\_stoc;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('Produsul cu codul '||p\_cod||' nu exista');

RETURN 0;

END;

/

A picture containing text, font, white, receipt

Description automatically generatedDECLARE

v\_slct\_cod ctl\_XY.cod\_p%TYPE;

v\_rezultat NUMBER;

BEGIN

v\_slct\_cod:=&dati\_codul;

v\_rezultat:=f\_stoc\_produs\_exceptie(v\_slct\_cod, v\_rezultat);

END;

/

2) Pornind de la prima si a treia interogare a punctului 4c), scrieti o procedura fara parametru, cu cursor, care sa afiseze urmatoarea situatie.

CREATE OR REPLACE PROCEDURE p\_pret AS

CURSOR c IS

SELECT ctl\_XY.cod\_p, den\_prod, tip\_op, SUM(pret\*cant) AS Total\_val

FROM ctl\_XY, detalii\_misc\_XY

WHERE ctl\_XY.cod\_p=detalii\_misc\_XY.cod\_p

AND UPPER (tip\_op)='A'

AND pret<=(SELECT AVG(pret)

FROM detalii\_misc\_XY

WHERE UPPER (tip\_op)='A')

GROUP BY ctl\_XY.cod\_p, den\_prod, tip\_op

ORDER BY 1;

v\_pret\_mediu NUMBER:=0;

v\_total\_val NUMBER:=0;

BEGIN

SELECT ROUND(AVG(pret),2) INTO v\_pret\_mediu

FROM detalii\_misc\_XY

WHERE UPPER (tip\_op)='A';

DBMS\_OUTPUT.PUT\_LINE('Pretul mediu al aprovizionarilor este '||v\_pret\_mediu);

DBMS\_OUTPUT.PUT\_LINE('--------------------------------------------------------------------------');

DBMS\_OUTPUT.PUT\_LINE('Cod produs Denumire produs Tip Operatiune Valoare');

FOR i IN c LOOP

DBMS\_OUTPUT.PUT\_LINE(i.cod\_p||' '||i.den\_prod||' '||i.tip\_op||' '||i.Total\_val);

v\_total\_val:=v\_total\_val + i.Total\_val;

END LOOP;

DBMS\_OUTPUT.PUT\_LINE('--------------------------------------------------------------------------');

DBMS\_OUTPUT.PUT\_LINE('Totalul valori al aprovizionarilor sub pretul mediu este '||v\_total\_val);

END;

/

EXECUTE p\_pret;

A picture containing text, receipt, screenshot, font

Description automatically generated