SET SERVEROUTPUT ON;

SET VERIFY OFF;

-- 4. Definiți un bloc anonim în care să se afle numele restaurantului cu cei mai mulți angajați.

-- Comentați cazul în care există cel puțin două restaurante cu număr maxim de angajați.

-- 5. Rezolvați problema anterioară utilizând variabile de legătură. Afișați rezultatul atât din bloc, cât

-- și din exteriorul acestuia.

-- nu functioneaza... nu stiu de ce

VAR restaurant NUMBER(10, 0);

BEGIN

    WITH counts AS (

        SELECT id\_restaurant, COUNT(\*) nr\_angajati

        FROM ANGAJAT

        GROUP BY (id\_restaurant)

    )

    SELECT id\_restaurant

    INTO :restaurant

    FROM counts

    WHERE nr\_angajati = (SELECT MAX(nr\_angajati) FROM counts);

    DBMS\_OUTPUT.PUT\_LINE('Restaurantul '||:restaurant);

EXCEPTION

    WHEN TOO\_MANY\_ROWS THEN

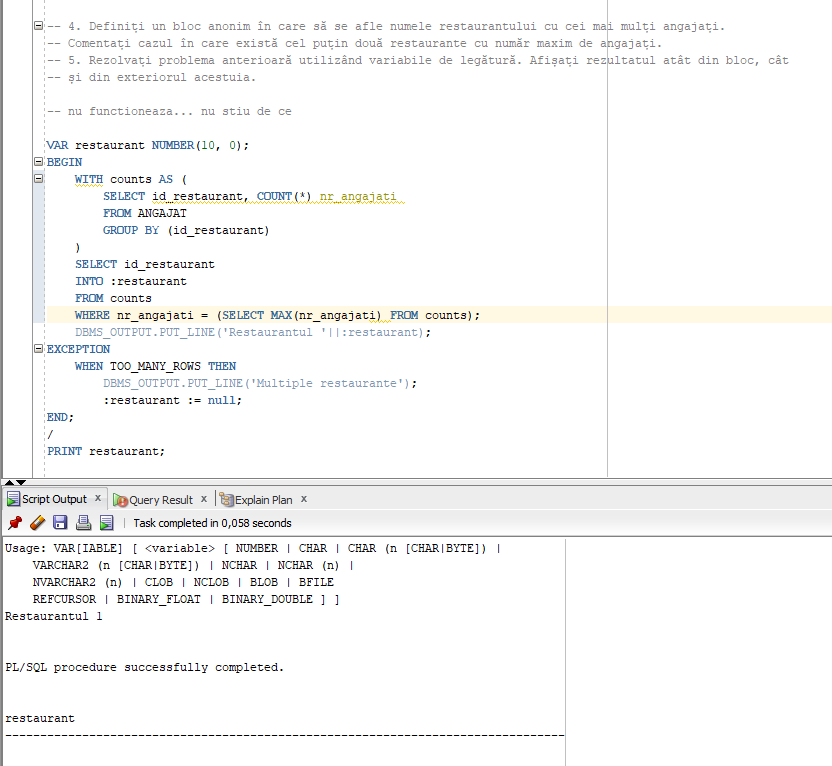
        DBMS\_OUTPUT.PUT\_LINE('Multiple restaurante');

        :restaurant := null;

END;

/

PRINT restaurant;



SET SERVEROUTPUT ON;

SET VERIFY OFF;

-- 7. Determinați bonusul pe care îl primește un casier al cărui cod este dat de la

-- tastatură. Bonusul este determinat astfel: dacă numarul de comenzi procesate este cel puțin 8, atunci bonusul

-- este 200; dacă este cel puțin 5 si cel mult 7 atunci bonusul este 100,

-- iar dacă este cel putin 1 si cel mult 4, atunci bonusul este 50, altfel 0. Afișați bonusul obținut.

-- Comentați cazul în care nu există niciun angajat cu codul introdus

DECLARE

    v\_cod casier.id\_angajat%TYPE := &p\_cod;

    v\_bonus BINARY\_INTEGER;

    v\_nr\_comenzi BINARY\_INTEGER;

    v\_check\_exists BINARY\_INTEGER;

BEGIN

    SELECT COUNT(id\_angajat)

    INTO v\_check\_exists

    FROM CASIER

    WHERE id\_angajat = v\_cod;

    IF v\_check\_exists = 0 THEN

        RAISE NO\_DATA\_FOUND;

    END IF;

    SELECT COUNT(id\_comanda)

    INTO v\_nr\_comenzi

    FROM COMANDA

    WHERE id\_casier = v\_cod;

    IF v\_nr\_comenzi >= 8 THEN

        v\_bonus := 200;

    ELSIF v\_nr\_comenzi >= 5 THEN

        v\_bonus := 100;

    ELSIF v\_nr\_comenzi >= 1 THEN

        v\_bonus := 50;

    ELSE

        v\_bonus := 0;

    END IF;

    DBMS\_OUTPUT.PUT\_LINE('Bonusul este '||v\_bonus);

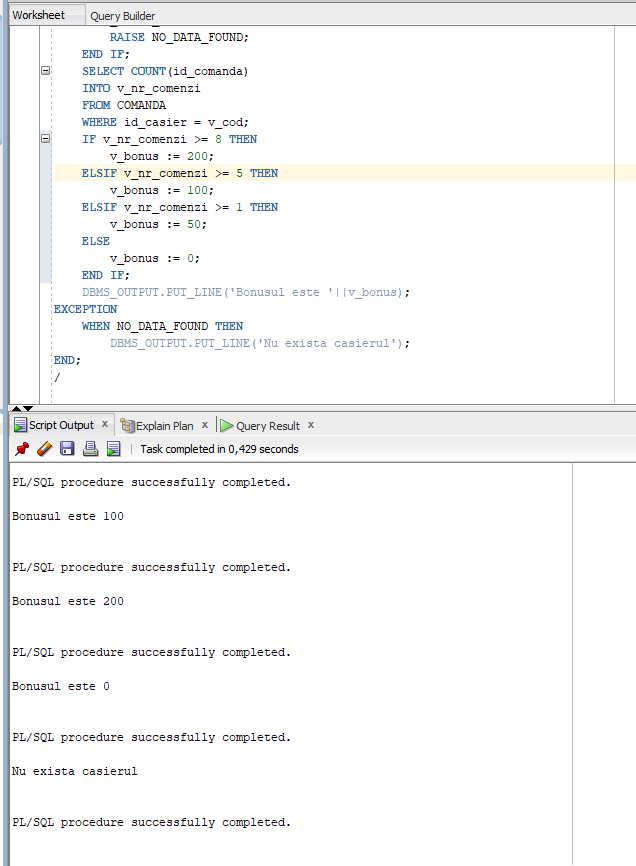
EXCEPTION

    WHEN NO\_DATA\_FOUND THEN

        DBMS\_OUTPUT.PUT\_LINE('Nu exista casierul');

END;

/



SET SERVEROUTPUT ON;

SET VERIFY OFF;

-- 9. Scrieți un bloc PL/SQL în care stocați prin variabile de substituție un cod de angajat si un cod de

-- restaurant. Să se mute angajatul în noul restaurant. Daca modificarea s-a putut realiza

-- (există în tabel ANGAJAT un salariat având codul respectiv) să se afișeze mesajul “Actualizare

-- realizata”, iar în caz contrar mesajul “Nu exista un angajat cu acest cod”. Anulați modificările

-- realizate.

DECLARE

    v\_cod\_ang angajat.id\_angajat%TYPE := &p\_cod\_ang;

    v\_cod\_rst angajat.id\_restaurant%TYPE := &p\_cod\_rst;

BEGIN

    UPDATE ANGAJAT

    SET id\_restaurant = v\_cod\_rst

    WHERE id\_angajat = v\_cod\_ang;

    IF SQL%ROWCOUNT = 0 THEN

        DBMS\_OUTPUT.PUT\_LINE('Nu exista un angajat cu acest cod');

    ELSE

        DBMS\_OUTPUT.PUT\_LINE('Actualizare realizata');

    END IF;

    ROLLBACK;

END;

