Acest proiect se referă la proiectarea unui model de date ce furnizează informații despre un Batalion militar american din timpul celui de-al Doilea Razboi Mondial.

Vom prezenta modelul de date, restricţiile pe care trebuie să le respecte şi vom încerca, într-o manieră didactică, să construim diagrama E/R corespunzătoare.

Inițial vom lua în considerare anumite situații care nu sunt optime, în sensul că pot genera anomalii la reactualizări sau nu permit anumitor interogări asupra modelului. Vom încerca să arătăm care sunt deficienţele modelului, situaţiile care le-au generat şi cum pot fi corectate (parţial sau total) anomaliile respective.

Modelul de date va gestiona informații despre organizarea batalionului, soldații din respectivul batalion cât și informații despre echipamentele, rangurile și medaliile deținute de aceștia. Batalionul este format din plutoane, care este cea mai mică formă de organizare militară. Plutonul este format din mai mulți soldați (cel puțin unul) și este staționat într-un singur oraș, posibil alături de alte plutoane. Echipamentele “de grupare” (artilerie, mitraliere grele, vehicule) atribuite unui pluton cât și personalul administrativ (excepție fiind medicul) nu fac parte din proiectarea modelului.

Orașele se află obligatoriu într-o țară introdusă în tabelul Tara, iar țările sunt localizate obligatoriu într-o regiune introduse în tabelul Regiune.

De asemenea modelul de date analizează informații legate de fiecare soldat din cadrul batalionului, cum ar fi date personale, rangul, plutonul din care fac parte, echipamentul acestora cât și medaliile pe care aceștia le-au câștigat de-a lungul timpului.

Un soldat poate avea un singur rang, însă mai mulți soldați pot avea același rang. Salariul unui soldat este stabilit având în considerare doar rangul acestuia, orice alte lucruri cum ar fi ani de experiență sau vârstă nu sunt luate în considerare la calcularea salariului.

Medaliile sunt înmânate soldaților care au săvârșit anumite acte, fie de curaj, sacrificiu sau eroism, ele pot fi acordate mai multor soldați și nu toți soldații trebuie să aibă medalii, deținerea medaliilor nu acordă niciun privilegiu soldatului posesor.

Echipamentul fiecărui soldat este standardizat, există câteva combinații de echipamente prestabilite pe care un soldat le poate primi. Echipamentele sunt alcătuie din armă și grenade (restul conținuturilor echipamentului nu sunt acoperite în modelul de date, considerându-se că fiecare soldat va primi aceeași uniformă, aceleași unelte, rații, etc.), fiecare echipament incluzând o singură armă.

Fiecare armă este fabricată într-o țară care trebuie să fie inclusă în tabelul Tara (este considerat că o armă este produsă într-o singură țară ci nu în mai multe) și de asemenea folosește un anumit tip de muniție cu același calibru, care trebuie inclusă în tabelul Munitie (muniție referindu-se strict la cartușe, nu sunt luate în considerare rachete, grenade pentru lansatoare, etc.), însă muniția fiecărei arme nu trebuie să fie unică, însă nu există armă căreia să nu îi fie asociată o muniție.

Muniția este standardizată, fiecare muniție fiind asociată cel puțin unei arme și este specificată și cantitatea de cartușe care sunt incluse pentru fiecare armă (am ales să fie fiecare muniție asociată cel puțin unei arme deoarece producerea unui tip de cartuș care nu este folosit de nicio armă nu ar avea sens).

Modelul de date respectă anumite restricţii de funcţionare.

* + Fiecare pluton este alcătuit din cel puțin un soldat și este staționat într-un singur oraș.
  + Fiecare oraș se află într-o singură țară și poate să găzduiască mai multe plutoane, sau niciunul.
  + Fiecare țară este localizată într-o singură regiune și poate produce arme, însă producerea de arme nu este o necesitate.
  + În fiecare regiune este localizată cel puțin o țară (deoarece nu ar avea sens introducerea ei în baza de date).
  + Fiecare rang este deținut de cel puțin un soldat și același rang poate fi acordat mai multor soldați.
  + Fiecare soldat are fix un rang asociat și un echipament asociat și poate să aibă mai multe medalii, una, sau niciuna. De asemenea fiecare soldat va avea un superior, mai puțin soldatul cu rangul de locotenent-colonel, care este liderul batalionului.
  + 2 sau mai mulți soldați nu pot avea același nume și prenume (nu putem introduce 2 soldați cu numele de Popescu Ion, dar putem introduce Popescu Ion, Popescu Vasile, sau Popescu Ion, Ionescu Ion).
  + Un echipament are fix o armă asociată și poate fi acordat mai multor soldați același echipament.
  + O armă poate fi inclusă în unul sau mai multe echipament, sau în niciunul (cazul armelor care nu mai sunt în folosință de forțele armate americane). De asemenea, fiecare armă are o țară unică în care este fabricată și poate folosi mai multe tipuri de muniție.
  + Fiecare tip de muniție este folosit de cel puțin o armă.

**Entități:**

Pentru modelul de date regeritor la batalionul american din Al Doilea Război Mondial, avem ca entități structurile: Pluton, Oras, Rang, Soldat, Tara, Medalie, Regiune, Echipament, Arma, Munitie.

Pluton = cea mai mică formă de organizare militară, ea este condusă în mod normal de un căpitan sau de un locotenent ea poate fi folosită pentru a grupa un număr restrâns de soldați și de a le desemna o armă de grup (de obicei un mortier, sau o mitralieră grea). Cheia primară a entității este cod\_pluton.

Oras = așezare importantă, cu populație numeroasă, cu întreprinderi și instituții, constituind, de obicei, un centru industrial, comercial, cultural, politic și administrativ, entitatea este utilizată pentru a determina locul specific unde este staționat unul sau mai multe plutoane militare. Cheia primară a entității este cod\_oras.

Rang = treapta de ierarhie militară pe care o ocupă o persoană din forțele armate. De obicei, uniforma indică rangul purtătorului prin însemne particulare aplicate pe uniforma respectivă. Cheia primară a entității este cod\_rang.

Soldat = persoană înrolată în armată ca voluntar sau recrutată pe timp de război indiferent dacă au ales sau nu. Cheia primară a entității este cod\_soldat.

Tara = teritoriu locuit de un popor organizat din punct de vedere administrativ și politic într-un stat, este folosit pentru a determina unde sunt localizate orașele în care sunt staționați soldații din batalion. Cheia primară a entității este cod\_tara.

Medalie = decorațiune primită de către un soldat în urma săvârșirii unei acțiuni notabile, cum ar fi curajul, eroismul sau în urma unei răni sau unui deces. Cheia primară a entității este cod\_medalie.

Regiune = teritoriu vast care poate include una sau mai multe țări. Folosind entitatea Regiune putem determina care țară este principalul inamic din punct de vedere militar al Statelor Unite ale Americii din respectiva regiune. Cheia primară a entității este cod\_regiune.

Echipament = totalitatea obiectelor de îmbrăcăminte, de încălțăminte și de accesorii cu care este dotat un militar, principalele piese de echipament de luptă fiind arma și numărul de grenade care îi sunt atribuite unei persoane din forțele armate active. Cheia primară a entității este cod\_echipament.

Arma = obiect care servește în lupta împotriva unui inamic, în special armă de foc, care folosește pulbere explozivă pentru a lansa un proiectil, care are un calibru determinat de diametrul țevii de tragere. Cheia primară a entității este cod\_arma.

Munitie = cartuș, principala resursă consumabilă a soldaților, ele fiind diferențiate prin calibrul acestora cât și prin arma pentru care sunt utilizate, de asemenea ne indică și numărul de cartușe care sunt incluse într-un echipament (nu poate sa fie NULL deoarece trebuie sa fie alocate un numar de cartuse diferit de 0). Cheia primară a entității este cod\_munitie.

**Relații:**

Vom prezenta relațiile modelului de date, dând o descriere completă a fiecăreia. Denumirile acestor legături sunt sugestive, reflectând conținutul acestora și entitățile pe care le leagă.

Pentru fiecare relație vom preciza cardinalitatea minimă si maximă.

Tara\_este\_localizata\_in\_Regiune = relație de tipul many-to-one care leagă entitățile Tara și Regiune, reflectând legătura dintre acestea (în ce regiune este localizată o anumită țară). Ea are cardinalitatea minima 1:1 (minim o țară este localizată într-o regiune și o regiune conține cel puțin o țară) și cardinalitatea maximă n:1 (o regiune poate conține mai multe țări, dar o țară poate fi localizată într-o singură regiune).

Oras\_se\_afla\_in\_Tara = relație de tipul many-to-one care leagă entitățile Oras și Tara, reflectând legătura dintre acestea (în ce țară se află un anumit oraș). Ea are cardinalitatea minima 1:1 (minim un oraș se află într-o singură țară și o țară conține cel puțin un oraș) și cardinalitatea maximă n:1 (o țară poate conține mai multe orașe, dar un oraș se poate afla într-o singură țară).

Pluton\_este\_stationat\_in\_Oras = relație de tipul many-to-one care leagă entitățile Pluton și Oras, reflectând legătura dintre acestea (în ce oraș este staționat un anumit pluton). Ea are cardinalitatea minima 0:1 (un pluton este staționat într-un oraș și un oraș nu trebuie să aibă staționat niciun pluton) și cardinalitatea maximă n:1 (un oraș poate staționa mai multe plutoane, dar un pluton poate fi staționat într-un singur oraș).

Soldat\_face\_parte\_din\_Pluton = relație de tipul many-to-one care leagă entitățile Soldat și Pluton, reflectând legătura dintre acestea (din ce pluton face parte un anumit soldat). Ea are cardinalitatea minima 1:1 (un soldat trebuie sa facă parte dintr-un pluton și un pluton trebuie să fie constituit din minim un soldat) și cardinalitatea maximă n:1 (mai mulți soldați pot face parte din același pluton, dar un soldat poate sa facă parte dintr-un singur pluton).

Soldat\_detine\_Rang = relație de tipul many-to-one care leagă entitățile Soldat și Rang, reflectând legătura dintre acestea (ce rang deține un anumit soldat). Ea are cardinalitatea minima 1:1 (un soldat trebuie să aiba un rang și un rang trebuie deținut de cel puțin un soldat) și cardinalitatea maximă n:1 (mai mulți soldați pot avea același rang, dar un soldat poate avea un singur rang).

Soldat\_a\_fost\_decorat\_cu\_Medalie = relație de tipul many-to-many care leagă entitățile Soldat și Medalie, reflectând legătura dintre acestea (ce medalii a primit un anumit soldat). Ea are cardinalitatea minima 0:0 (un soldat poate să nu fi primit nicio medalie și o medalie poate să nu fie acordată nimănui) și cardinalitatea maximă n:m (mai mulți soldați pot primi aceeași medalie și o medalie poate fi acordată mai multor soldați).

Soldat\_ii\_este\_dat\_Echipament = relație de tipul many-to-one care leagă entitățile Soldat și Echipament, reflectând legătura dintre acestea (ce echipament îi este atribuit fiecărui soldat). Ea are cardinalitatea minima 0:1 (unui soldat i se dă un echipament, dar un echipament poate să nu fie acordat nimănui) și cardinalitatea maximă n:1 (unui soldat i se dă un singur echipament, iar un echipament poate fi atribuit mai multor soldați).

Echipament\_are\_inclusa\_Arma = relație de tipul many-to-one care leagă entitățile Echipament și Arma, reflectând legătura dintre acestea (ce armă este inclusă în fiecare echipament). Ea are cardinalitatea minima 0:1 (un echipament are inclusă o armă, dar o armă poate sa nu fie inclusă în niciun echipament) și cardinalitatea maximă n:1 (un echipament are inclusă o singură armă, iar o armă poate să fie inclusă în mai multe echipamente).

Arma\_este\_fabricata\_in\_Tara = relație de tipul many-to-one care leagă entitățile Arma și Tara, reflectând legătura dintre acestea (în ce țară este fabricată o anumită armă). Ea are cardinalitatea minima 0:1 (o armă este fabricată într-o țară, dar o țară poate sa nu fabrice nicio armă) și cardinalitatea maximă n:1 (o armă este fabricată într-o singură țară, iar o țară poate să fabrice mai multe arme).

Arma\_utilizeaza\_Munitie = relație de tipul many-to-many care leagă entitățile Arma și Munitie, reflectând legătura dintre acestea (ce muniție utilizează o anumită armă). Ea are cardinalitatea minima 1:1 (o armă folosește cel puțin un tip de muniție și o muniție este utilizată de minim o armă) și cardinalitatea maximă n:m (mai multe arme pot folosi aceeași muniție și mai multe muniții pot fi folosite de aceeași armă).

**Atribute:**

Entitatea Pluton are ca atribute:

cod\_pluton = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unui pluton.

numar\_pluton = variabilă de tip întreg, de lungime maximă 2, care reprezintă numărul unui pluton. Are ca și constrângeri NOT NULL.

cod\_oras = variabilă de tip întreg, de lungime maximă 5, care reprezintă codul orașului în care se află plutonul. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Oras. Are ca și constrângeri NOT NULL.

Entitatea Oras are ca atribute:

cod\_oras = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unui oraș.

nume\_oras = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele unui oraș. Are ca și constrângeri NOT NULL.

zona\_de\_conflict = variabilă de tip caracter, de lungime maximă 2 care spune dacă un oraș se află sau nu într-o zonă de conflict. Are ca și valori posibile: “Da”, “Nu” și nu poate fi NULL.

cod\_tara = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul țării în care se află orașul. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Tara. Are ca și constrângeri NOT NULL.

Entitatea Rang are ca atribute:

cod\_rang = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unui rang.

nume\_rang = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele unui oraș. Are ca și valori posibile: “soldat”, “soldat primă clasă”, “caporal”, “sergent”, “prim sergent”, “locotenent”, “căpitan”, “maior”, “locotenent-colonel” (Alte ranguri poziționate mai sus în ierarhia militară nu sunt relevante, ele fiind atașate unităților de dimensiuni mai mari). Are ca și constrângeri NOT NULL.

salariu = variabilă de tip întreg, de lungime maximă 5, care reprezintă salariul unui soldat acordat unui soldat cu un anumit rang. Are ca și constrângeri NOT NULL.

Entitatea Soldat are ca atribute:

cod\_soldat = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unui soldat.

nume = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele unui soldat. Are ca și constrângeri NOT NULL.

prenume = variabilă de tip caracter, de lungime maximă 30, care reprezintă prenumele unui soldat. Are ca și constrângeri NOT NULL.

data\_inrolare = variabilă de tip dată calendaristică, care reprezintă data înrolării unui soldat. Are ca și constrângeri NOT NULL.

varsta = variabilă de tip întreg, de lungime maximă 2, care reprezintă vârsta unui soldat.

cod\_pluton = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul plutonului din care face parte soldatul. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Pluton. Are ca și constrângeri NOT NULL.

cod\_superior\_direct = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul persoanei care îi este direct superior soldatului. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Soldat sau este NULL, caz în care acel soldat nu are un superior direct.

cod\_echipament = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul echipamentului care îi este acordat soldatului. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Echipament. Are ca și constrângeri NOT NULL.

cod\_rang = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul rangului pe care îl deține soldatul. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Rang. Are ca și constrângeri NOT NULL.

Entitatea Tara are ca atribute:

cod\_tara = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unei țări.

nume\_tara = variabilă de tip caracter, de lungime maximă 30, care reprezintă numele unei țări. Are ca și constrângeri NOT NULL.

cod\_regiune = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul regiunii în care se află țara. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Regiune. Are ca și constrângeri NOT NULL.

Entitatea Medalie are ca atribute:

cod\_medalie = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unei medalii.

nume\_medalie = variabilă de tip caracter, de lungime maximă 30, care reprezintă numele unei medalii. Are ca și constrângeri NOT NULL.

conditie\_dobandire = variabilă de tip caracter, de lungime maximă 15, care include condițiile de dobândire a unei medalii. Are ca valori posibile: “curaj”, “eroism”, “ranit/decedat”.

Entitatea Regiune are ca atribute:

cod\_regiune = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unei regiuni.

nume\_regiune = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele unei regiuni. Are ca și constrângeri NOT NULL.

inamic\_principal = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele țării care este considerată inamicul principal al Statelor Unite ale Americii într-o anumită regiune în Al Doilea Război Mondial. Are ca valori posibile: “Germania”, “Italia”, “Romania+Bulgaria”, “Japonia” sau poate lua valoarea de NULL, caz în care este o regiune în care nu există țări inamice.

Entitatea Echipament are ca atribute:

cod\_echipament = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unui echipament.

nume\_echipament = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele unui echipament. Are ca și constrângeri NOT NULL.

numar\_grenade = variabilă de tip întreg, de lungime maximă 1, care reprezintă numărul de grenade incluse într-un echipament. Are ca și valoare implicită 2.

cod\_arma = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul armei incluse în echipament. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Arma. Are ca și constrângeri NOT NULL.

Entitatea Arma are ca atribute:

cod\_arma = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unei arme.

nume\_model = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele unei arme. Are ca și constrângeri NOT NULL.

calibru = variabilă de tip caracter, de lungime maximă 20, care reprezintă calibrul unei arme. Are ca și constrângeri NOT NULL.

cod\_tara = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul țării în care a fost fabricată arma. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Tara. Are ca și constrângeri NOT NULL.

Entitatea Munitie are ca atribute:

cod\_munitie = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul unei muniții.

nume\_munitie = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele unei muniții. Are ca și constrângeri NOT NULL.

calibru = variabilă de tip caracter, de lungime maximă 20, care reprezintă calibrul unei muniții. Are ca și constrângeri NOT NULL.

numar\_de\_gloante = variabilă de tip întreg, de lungime maximă 4, care reprezintă numărul de gloanțe primit de un soldat care utilizeaza un echipament care include o armă care utilizeaza o muniție. Are ca și constrângeri NOT NULL.

Relația Soldat\_a\_fost\_decorat\_cu\_Medalie are ca atribute:

cod\_medalie = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul medaliei acordate soldatului. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Medalie.

cod\_soldat = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul soldatului care a primit medalia. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Soldat.

data\_acordare = variabilă de tip dată calendaristică care reprezintă data acordării medaliei unui soldat. Are ca și constrândrei NOT NULL.

cod\_soldat și cod\_medalie constituie o cheie primară compusă.

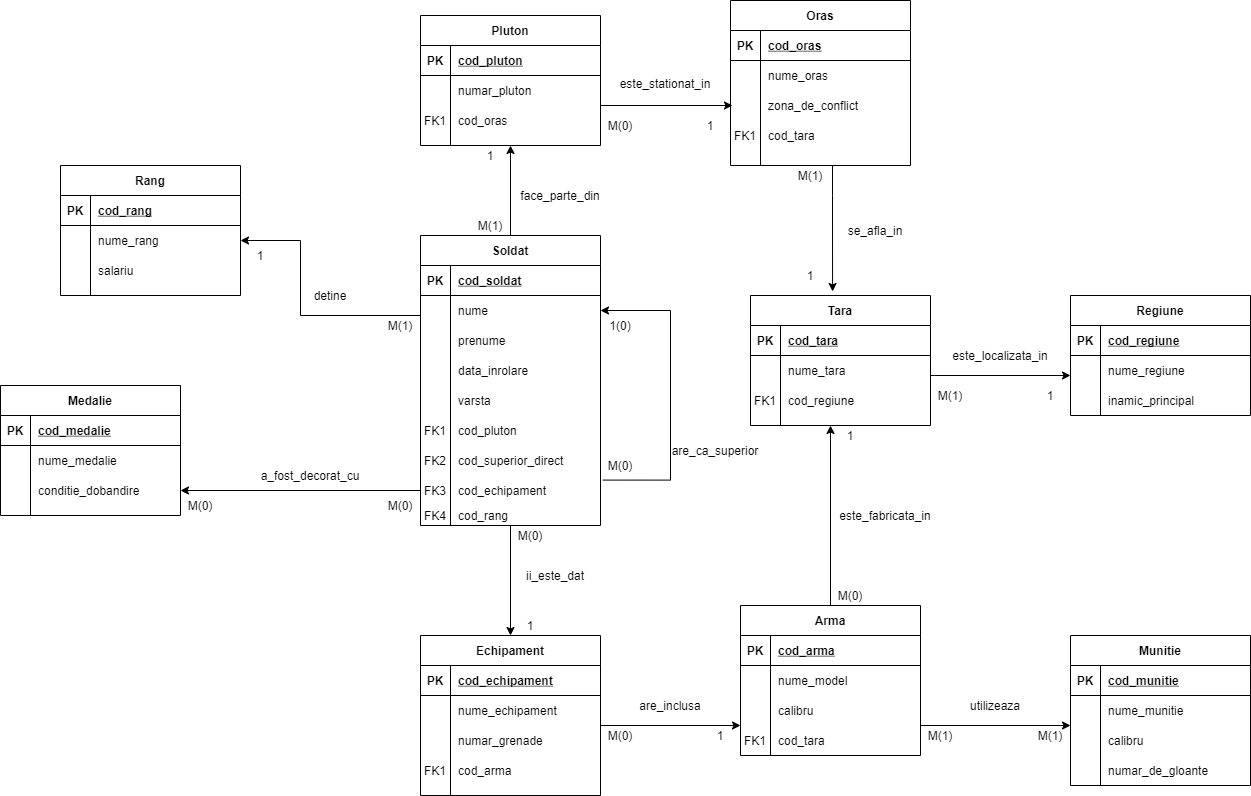
Relația Arma\_utilizeaza\_Munitie are ca atribute:

cod\_arma = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul armei care folosește muniția. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Arma. Are ca și constrângeri NOT NULL.

cod\_munitie = variabilă de tip întreg, de lungime maximă 3, care reprezintă codul muniției care este utilizată de armă. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul Munitie. Are ca și constrângeri NOT NULL.

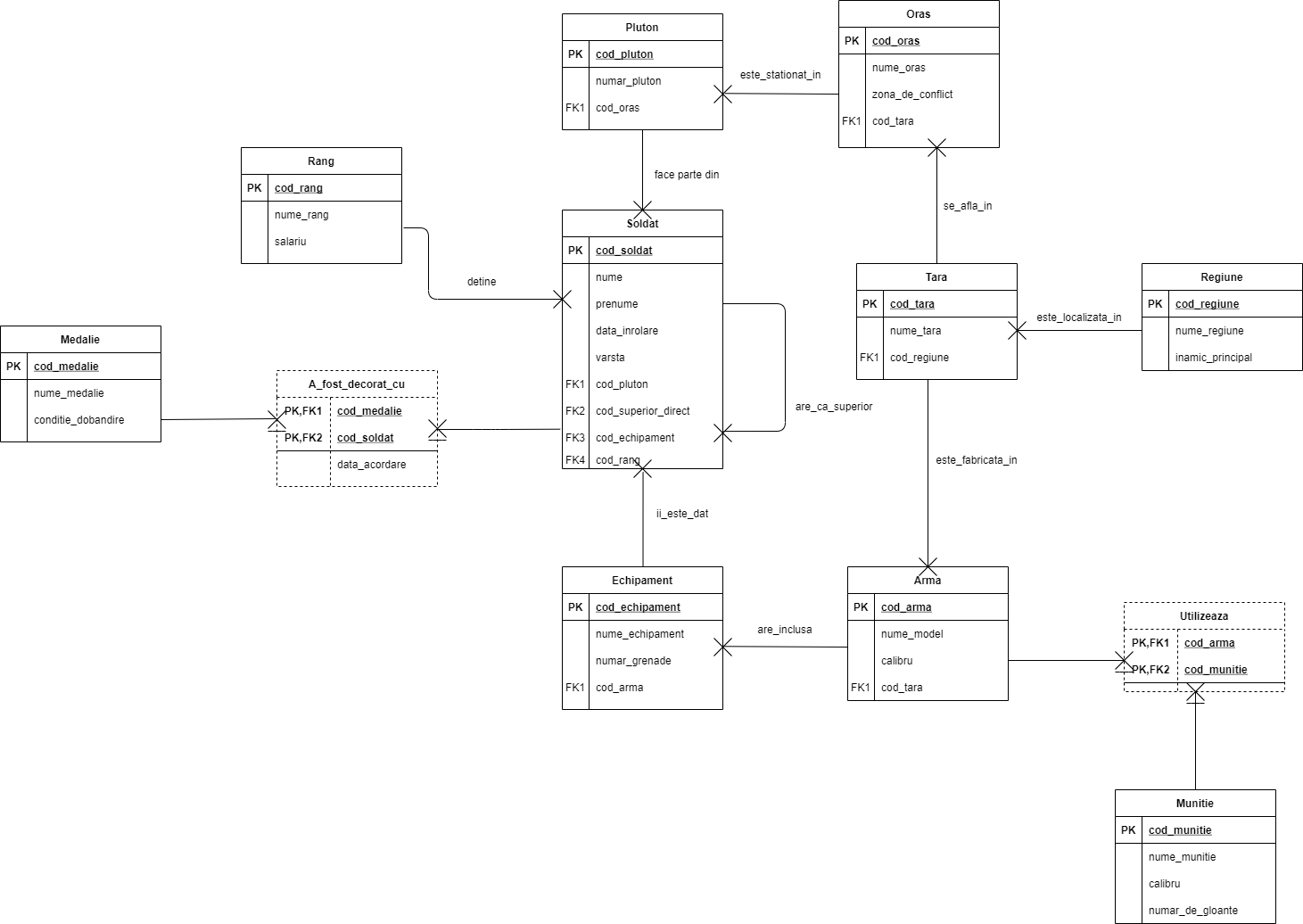
cod\_arma și cod\_munitie constituie o cheie primară compusă.

**Diagrama entitate-relație**



**Fig.1.1.**Diagrama E/R.

**Diagrama conceptuală**



**Fig. 1.2.** Diagrama conceptuală.

**Scheme relaționale:**

**Schemele relaționale** corespunzătoare diagramei conceptuale din figura 1.2. sunt următoarele:

Pluton (cod\_pluton#, numar\_pluton, cod\_oras)

Oras (cod\_oras#, nume\_oras, zona\_de\_conflict, cod\_tara)

Rang (cod\_rang#, nume\_rang, salariu)

Soldat (cod\_soldat#, nume, prenume, data\_inrolare, varsta, cod\_pluton, cod\_superior\_direct, cod\_echipament, cod\_rang)

Tara (cod\_tara#, nume\_tara, cod\_regiune)

Medalie (cod\_medalie#, nume\_medalie, conditie\_dobandire)

Regiune (cod\_regiune#, nume\_regiune, inamic\_principal)

Echipament (cod\_echipament#, nume\_echipament, numar\_grenade, cod\_arma)

Arma (cod\_arma#, nume\_model, calibru, cod\_tara)

Munitie (cod\_munitie#, nume\_munitie, calibru, numar\_de\_gloante)

A\_fost\_decorat\_cu (cod\_medalie#, cod\_soldat#, data\_acordare)

Utilizează(cod\_arma#, cod\_munitie#)

**Normalizarea până la forma normală 3(FN1-FN3):**

Toate relațiile din cadrul modelului de date sunt în formă normală 3(FN3).

Relațiile sunt în prima formă normală deoarece fiecărui atribut îi corespunde o valoare indivizibilă.

**Exemplu de transformare a relației A\_fost\_decorat\_cu în formă Non FN1:**

|  |  |  |
| --- | --- | --- |
| cod\_soldat | nume | medalii |
| 15 | Sofei | Medalia 1, Medalia 2, Medalia 3 |
| 17 | Popescu | Medalia 2, Medalia 4 |

**Forma în FN1:**

|  |  |  |
| --- | --- | --- |
| cod\_soldat | nume | medalie |
| 15 | Sofei | Medalia 1 |
| 15 | Sofei | Medalia 2 |
| 15 | Sofei | Medalia 3 |
| 17 | Popescu | Medalia 2 |
| 17 | Popescu | Medalia 4 |

Relațiile sunt în a doua formă normală deoarece:

* Sunt în FN1.
* Fiecare atribut care nu este cheie (nu participă la cheia primară) este dependent de întreaga cheie primară.

**Exemplu de transformare a relației A\_fost\_decorat\_cu în formă Non FN2:**

|  |  |  |
| --- | --- | --- |
| cod\_soldat | nume | medalie |
| 15 | Sofei | Medalia 1 |
| 15 | Sofei | Medalia 2 |
| 15 | Sofei | Medalia 3 |
| 17 | Popescu | Medalia 2 |
| 17 | Popescu | Medalia 4 |

**Forma în FN2:**

|  |  |
| --- | --- |
| cod\_soldat | medalie |
| 15 | Medalia 1 |
| 15 | Medalia 2 |
| 15 | Medalia 3 |
| 17 | Medalia 2 |
| 17 | Medalia 4 |

Relațiile sunt în a treia formă normală deoarece:

* Sunt în FN2.
* Fiecare atribut care nu este cheie (nu participă la o cheie) depinde direct de cheia primară.

**Exemplu de transformare a relației A\_fost\_decorat\_cu în formă Non FN3:**

|  |  |  |  |
| --- | --- | --- | --- |
| cod\_soldat | medalie | data\_acordare | vremea\_in\_acea\_zi |
| 15 | Medalia 1 | 21.05.2021 | însorit |
| 15 | Medalia 2 | 17.05.2021 | ploaie |
| 15 | Medalia 3 | 13.02.2021 | ninsoare |
| 17 | Medalia 2 | 17.05.2021 | ploaie |
| 17 | Medalia 4 | 21.05.2021 | însorit |

Nu este în formă FN3 deoarece atributul vremea\_in\_acea\_zi depinde direct de data\_acordare, nu depinde direct de cheia principală.

**Forma în FN3:**

|  |  |  |
| --- | --- | --- |
| cod\_soldat | medalie | data\_acordare |
| 15 | Medalia 1 | 21.05.2021 |
| 15 | Medalia 2 | 17.05.2021 |
| 15 | Medalia 3 | 13.02.2021 |
| 17 | Medalia 2 | 17.05.2021 |
| 17 | Medalia 4 | 21.05.2021 |

**Cereri:**

**1.Să se afișeze, pentru soldații staționați în regiunea Europa de Vest și care au litera 'o' în nume sau în prenume, id-ul soldatului, numele, prenumele,**

**data primului salariu(ultima zi din luna în care s-a înrolat),câte zile a fost activ în forțele armate până la terminarea războiului(diferența dintre ziua sfârșitului războiului 02.09.1945 și data înrolării)**

**și dacă este ofițer(maior,capitan,locotenent),subofiter(prim sergent, sergent) sau soldat (caporal, soldat prima clasa, soldat).**

**Să se ordoneze valorile după lungimea numelui și în caz de egalitate după lungimea prenumelui.**

select s.cod\_soldat,s.nume,s.prenume,

last\_day(s.data\_inrolare) as primul\_salariu,

to\_date('02-09-1945','dd-mm-yyyy')-s.data\_inrolare as zile\_la\_activ,

decode(rng.nume\_rang, 'maior','ofiter','capitan','ofiter','locotenent','ofiter',

'prim sergent','subofiter','sergent','subofiter','soldat') as rang

from soldat s,

pluton p,

oras o,

tara t,

regiune r,

rang rng

where s.cod\_pluton = p.cod\_pluton

and p.cod\_oras = o.cod\_oras

and o.cod\_tara = t.cod\_tara

and t.cod\_regiune = r.cod\_regiune

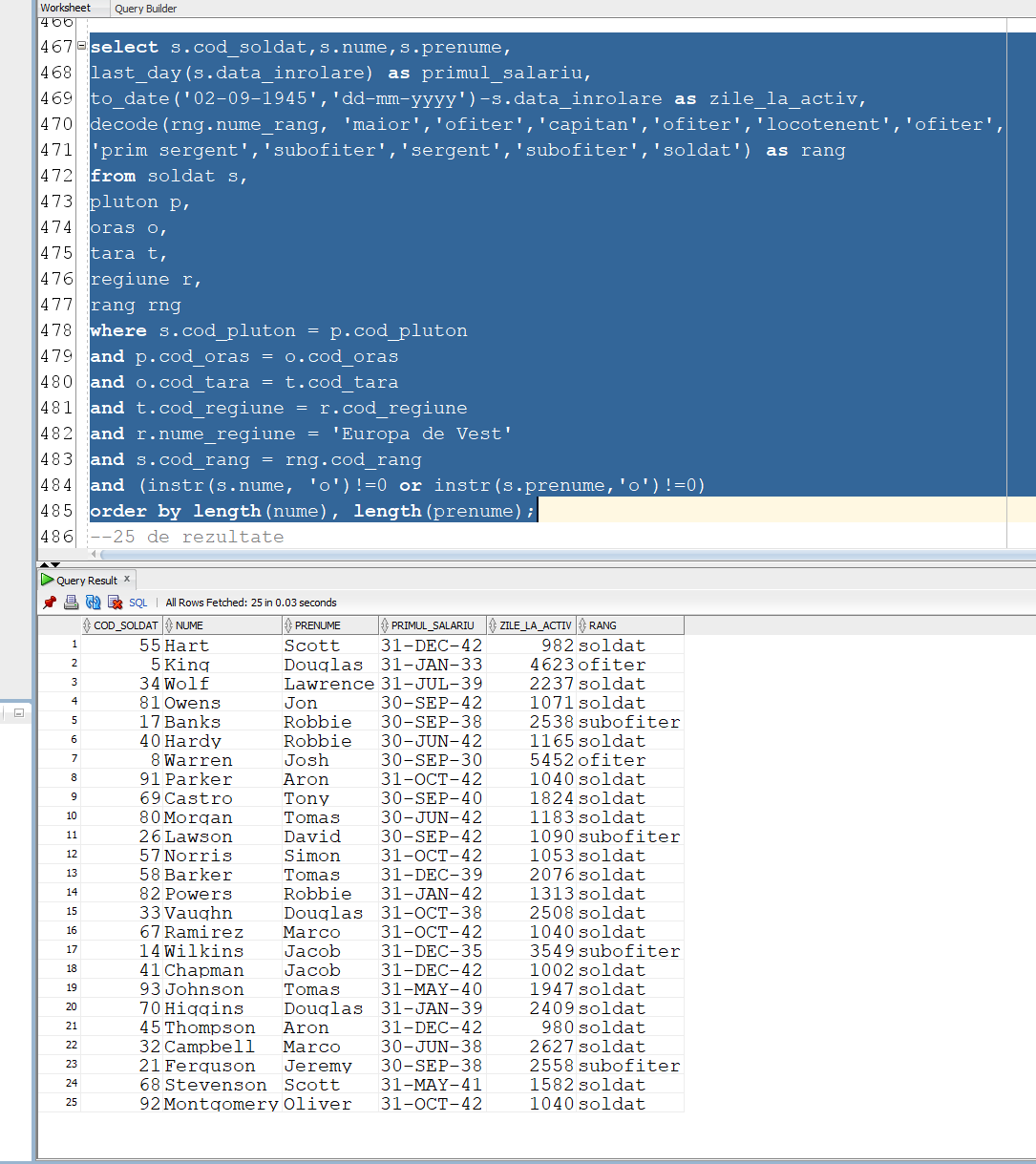
and r.nume\_regiune = 'Europa de Vest'

and s.cod\_rang = rng.cod\_rang

and (instr(s.nume, 'o')!=0 or instr(s.prenume,'o')!=0)

order by length(nume), length(prenume);

--25 de rezultate



**2.Să se afișeze numele, prenumele,arma, tipul de armă(armă automată (cod arma=2,3,7,8,9,10), armă semi-automată(cod arma=1,4,5) sau pușcă cu repetiție(cod arma=6) și salariul,**

**pentru soldații care au ca superior direct un soldat staționat într-o regiune în care inamicul principal este Italia și care au salariul mai mic decât media salariilor rangurilor.**

select s.cod\_soldat,s.nume, s.prenume,

a.nume\_model,

case

when a.cod\_arma=2 or a.cod\_arma=3 or a.cod\_arma=7 or a.cod\_arma=8 or a.cod\_arma=9 or a.cod\_arma=10 then 'arma automata'

when a.cod\_arma=1 or a.cod\_arma=4 or a.cod\_arma=5 then 'arma semi-automata'

else 'pusca cu repetitie'

end as tip\_arma

from soldat s,

echipament e,

arma a,

rang rng

where s.cod\_superior\_direct in(select s2.cod\_soldat

from soldat s2,

pluton p,

oras o,

tara t,

regiune r

where s2.cod\_pluton=p.cod\_pluton

and p.cod\_oras=o.cod\_oras

and o.cod\_tara=t.cod\_tara

and t.cod\_regiune=r.cod\_regiune

and r.inamic\_principal='Italia')

and s.cod\_rang = rng.cod\_rang

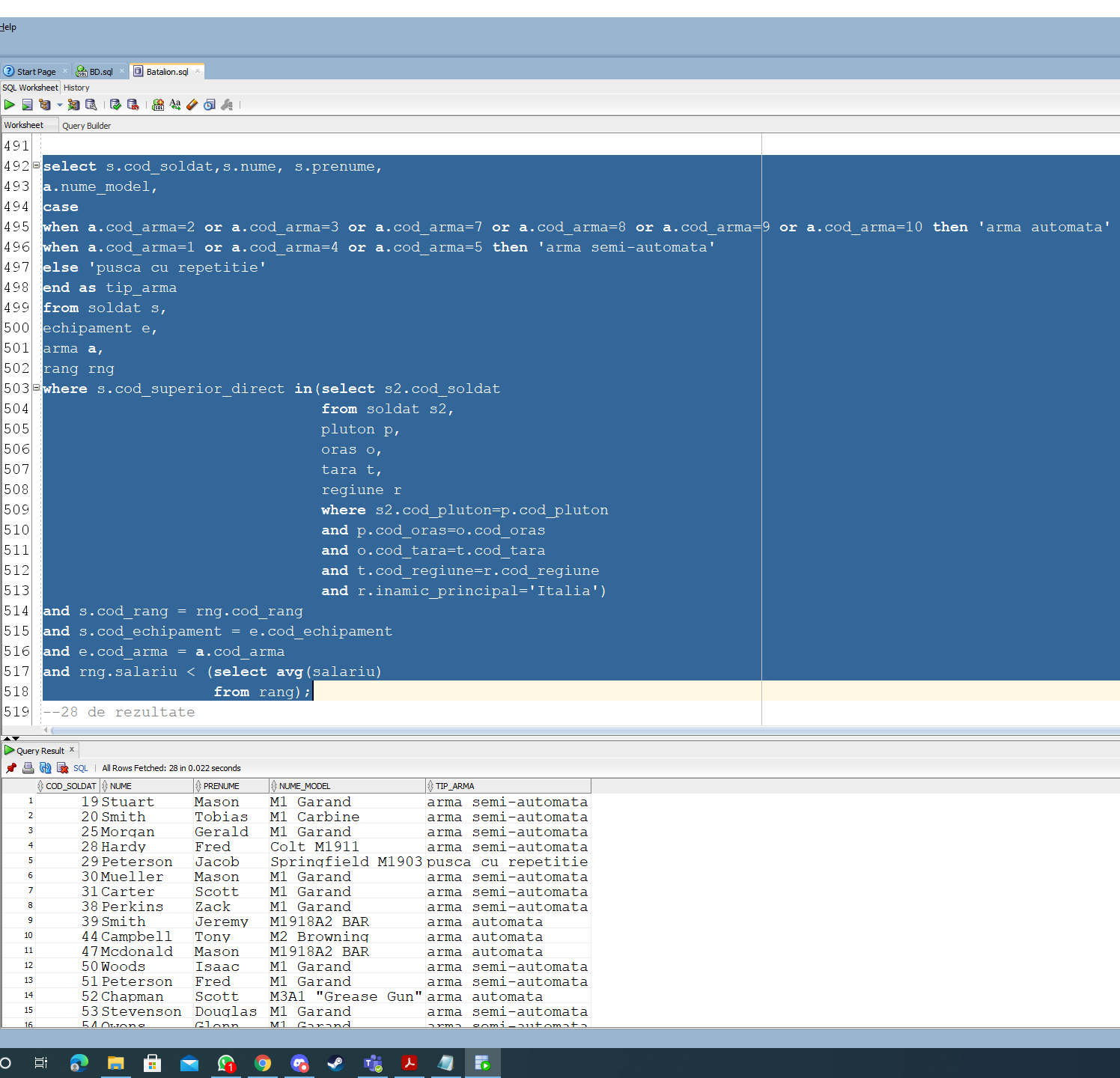
and s.cod\_echipament = e.cod\_echipament

and e.cod\_arma = a.cod\_arma

and rng.salariu < (select avg(salariu)

from rang);

--28 de rezultate



**3.Să se afișeze codul si numărul de subordonați direcți ai soldaților care au cel puțin 2 subordonați direcți.**

**Să se ordoneze valorile descrescător după numărul de subordonați direcți și în caz de egalitate crescător după codul soldatului.**

select s1.cod\_superior\_direct as soldat,

count(s1.cod\_soldat) as numar\_subordonati\_directi

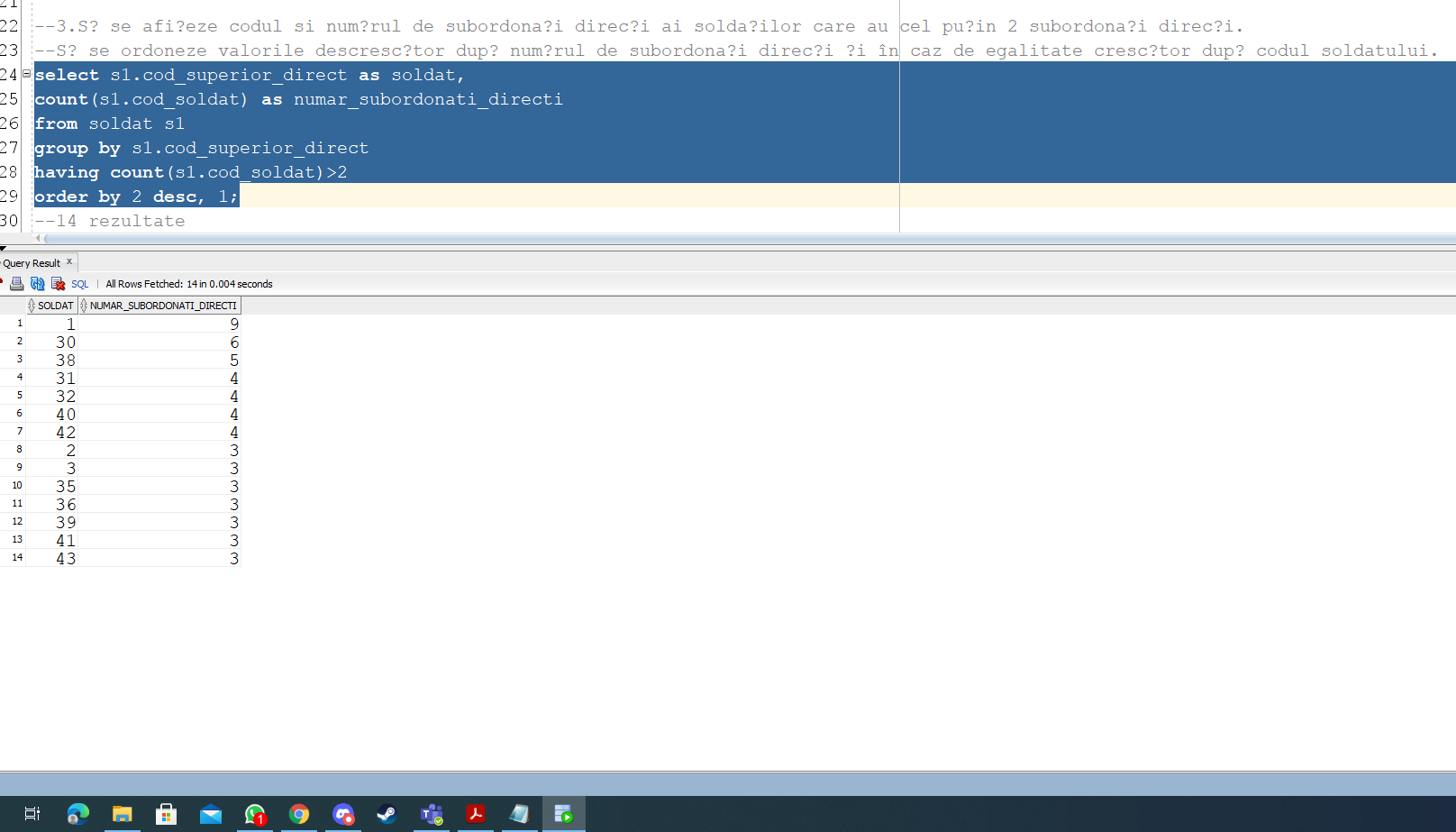
from soldat s1

group by s1.cod\_superior\_direct

having count(s1.cod\_soldat)>2

order by 2 desc, 1;

--14 rezultate



**4.Să se afișeze codul, numele, prenumele, codul superiorului direct(daca nu au se va pune valoarea 0),codul plutonului și salariul pentru soldații care au un salariu mai mare decât media salariilor soldaților din plutonul lor.**

**Să se ordoneze crescător după codul plutonului.**

select s.cod\_soldat, s.nume,s.prenume,

NVL(s.cod\_superior\_direct,0) as cod\_superior\_direct,

p.cod\_pluton, rng.salariu

from soldat s,

pluton p,

rang rng

where s.cod\_pluton=p.cod\_pluton

and s.cod\_rang=rng.cod\_rang

and rng.salariu > (select avg(rng2.salariu)

from soldat s2,

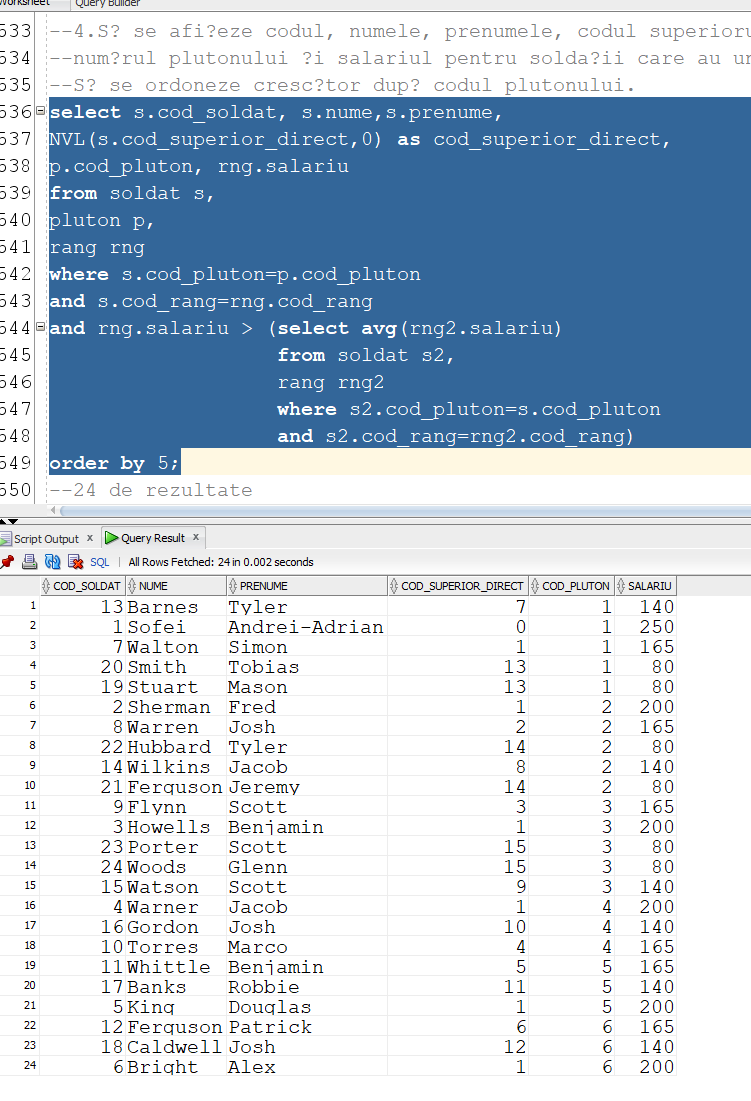
rang rng2

where s2.cod\_pluton=s.cod\_pluton

and s2.cod\_rang=rng2.cod\_rang)

order by 5;

--24 de rezultate



**5.Să se afișeze numele, prenumele și codul plutonului pentru soldații care au ca și armă M1 Garand și fac parte din plutonul cu codul 1.**

with temp(nume\_soldat\_cu\_garand, prenume\_soldat\_cu\_garand, cod\_pluton\_soldat\_cu\_garand) as

(select st.nume,st.prenume,st.cod\_pluton

from soldat st,

echipament et,

arma at

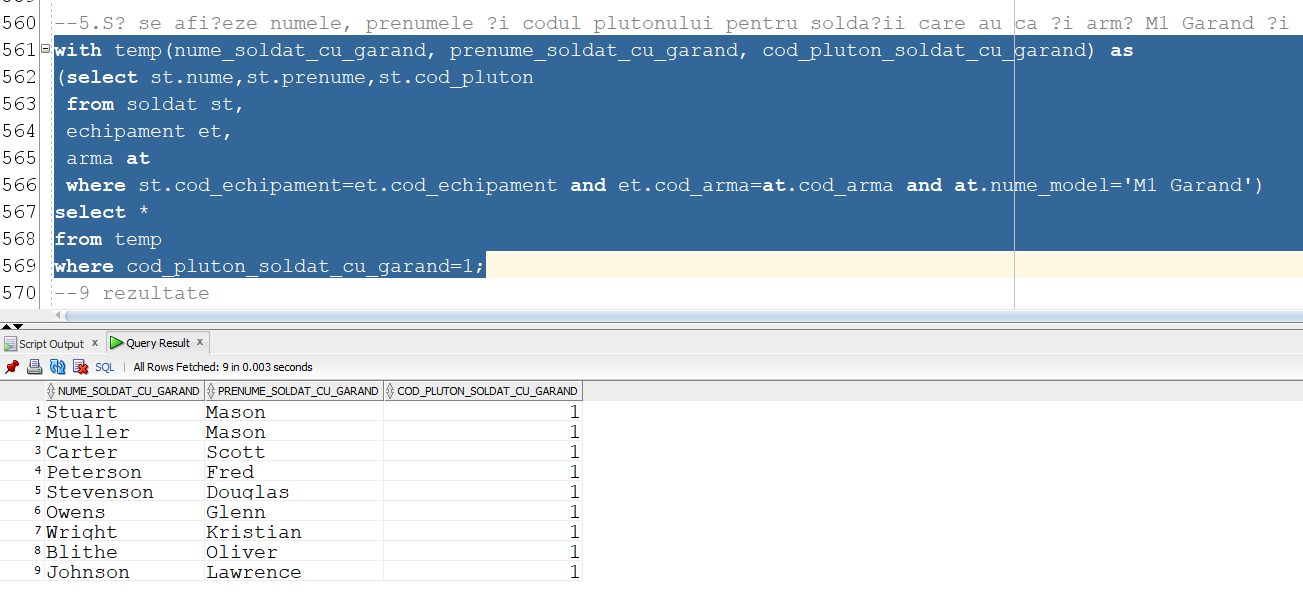
where st.cod\_echipament=et.cod\_echipament and et.cod\_arma=at.cod\_arma and at.nume\_model='M1 Garand')

select \*

from temp

where cod\_pluton\_soldat\_cu\_garand=1;

--9 rezultate



**Actualizari sau suprimari:**

**1.Treceți orașele din Singapore, Indonezia și Belgia ca fiind într-o zonă fără conflict.**

Update Oras

set zona\_de\_conflict='Nu'

where nume\_oras in (select o.nume\_oras

from oras o,

tara t

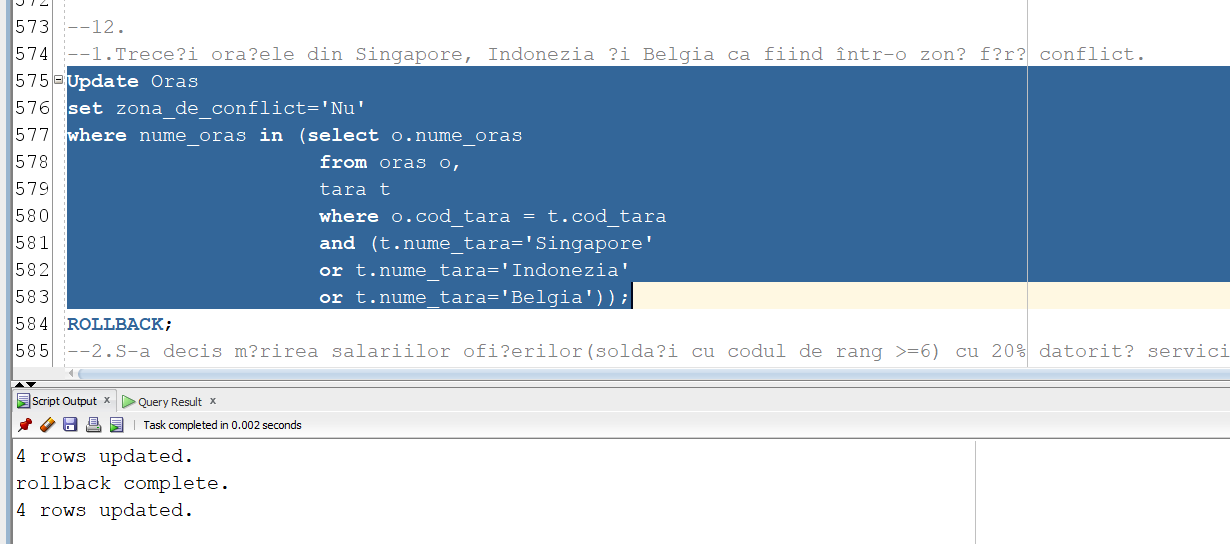
where o.cod\_tara = t.cod\_tara

and (t.nume\_tara='Singapore'

or t.nume\_tara='Indonezia'

or t.nume\_tara='Belgia'));

ROLLBACK;



**2.S-a decis mărirea salariilor ofițerilor(soldați cu codul de rang >=6) cu 20% datorită serviciilor excepționale. Efectuați această actualizare.**

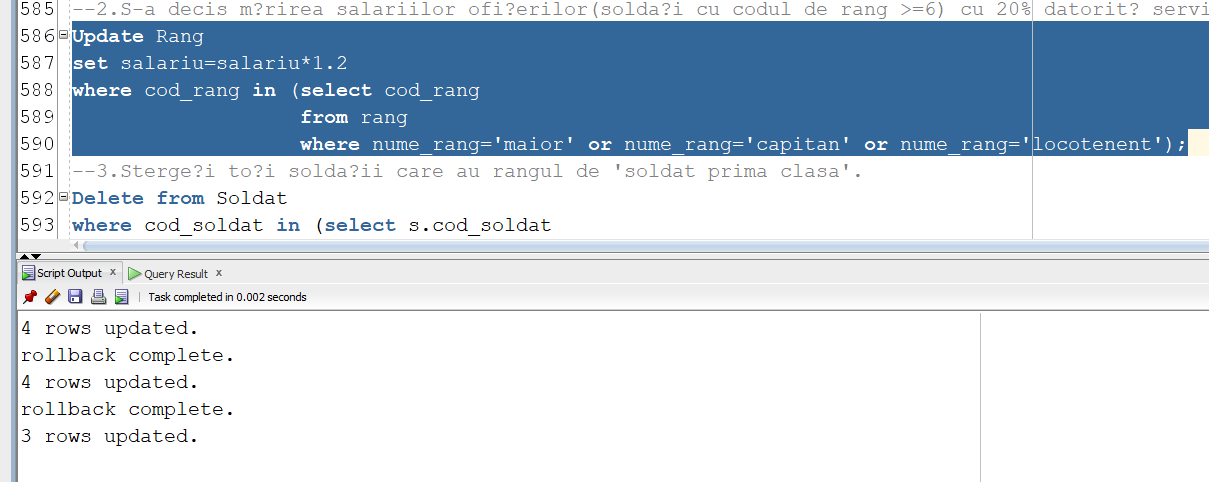
Update Rang

set salariu=salariu\*1.2

where cod\_rang in (select cod\_rang

from rang

where nume\_rang='maior' or nume\_rang='capitan' or nume\_rang='locotenent');



**3.Stergeți toți soldații care au rangul de 'soldat prima clasa'.**

Delete from Soldat

where cod\_soldat in (select s.cod\_soldat

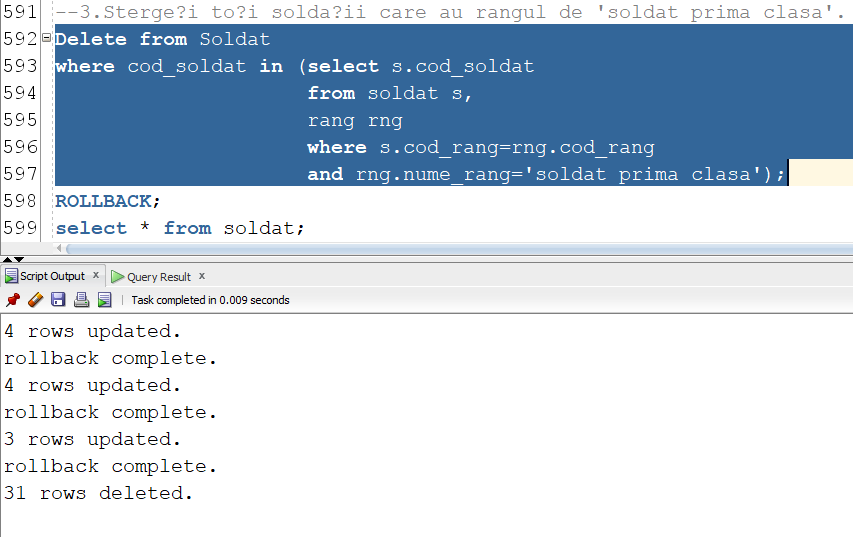
from soldat s,

rang rng

where s.cod\_rang=rng.cod\_rang

and rng.nume\_rang='soldat prima clasa');

ROLLBACK;



**Tabele:**

CREATE TABLE Regiune(

cod\_regiune number(3) primary key,

nume\_regiune varchar2(20) not null,

inamic\_principal VARCHAR2(20) check(inamic\_principal in('Germania','Italia','România+Bulgaria','Japonia',NULL)),

unique(nume\_regiune)

);

CREATE TABLE Tara(

cod\_tara number(3) primary key,

nume\_tara varchar2(30) not null,

cod\_regiune number(3) not null,

foreign key(cod\_regiune) references Regiune(cod\_regiune) on delete CASCADE,

unique(nume\_tara)

);

CREATE TABLE Oras(

cod\_oras number(3) primary key,

nume\_oras varchar2(20) not null,

zona\_de\_conflict varchar2(2) not null check(zona\_de\_conflict in('Da','Nu')),

cod\_tara number(3) not null,

foreign key(cod\_tara) references Tara(cod\_tara) on delete CASCADE,

unique(nume\_oras)

);

CREATE TABLE Pluton(

cod\_pluton number(3) primary key,

numar\_pluton number(2) not null,

cod\_oras number(3) not null,

foreign key(cod\_oras) references Oras(cod\_oras) on delete CASCADE,

unique(numar\_pluton)

);

CREATE TABLE Rang(

cod\_rang number(3) primary key,

nume\_rang varchar2(20) not null check(nume\_rang in('soldat','soldat prima clasa','caporal','sergent','prim sergent','locotenent','capitan','maior')),

salariu number(5) not null,

unique(nume\_rang)

);

CREATE TABLE Medalie(

cod\_medalie number(3) primary key,

nume\_medalie varchar2(30) not null,

conditie\_dobandire varchar2(15) not null check(conditie\_dobandire in('curaj','eroism','ranit/decedat')),

unique(nume\_medalie)

);

CREATE TABLE Munitie(

cod\_munitie number(3) primary key,

nume\_munitie varchar2(20) not null,

calibru varchar2(20) not null,

numar\_de\_gloante number(4) not null,

unique(nume\_munitie)

);

CREATE TABLE Arma(

cod\_arma number(3) primary key,

nume\_model varchar2(20) not null,

calibru varchar2(20) not null,

cod\_tara number(3) not null,

foreign key(cod\_tara) references Tara(cod\_tara) on delete CASCADE,

unique(nume\_model)

);

CREATE TABLE Echipament(

cod\_echipament number(3) primary key,

nume\_echipament varchar2(20) not null,

numar\_grenade number(1) default (2),

cod\_arma number(3) not null,

foreign key(cod\_arma) references Arma(cod\_arma) on delete CASCADE,

unique(nume\_echipament)

);

CREATE TABLE Soldat(

cod\_soldat number(3) primary key,

nume varchar2(20) not null,

prenume varchar2(20) not null,

data\_inrolare date not null,

varsta number(2),

cod\_pluton number(3) not null,

cod\_superior\_direct number(3),

cod\_echipament number(3) not null,

cod\_rang number(3) not null,

foreign key(cod\_pluton) references Pluton(cod\_pluton) on delete CASCADE,

foreign key(cod\_superior\_direct) references Soldat(cod\_soldat) on delete set NULL,

foreign key(cod\_echipament) references Echipament(cod\_echipament) on delete CASCADE,

foreign key(cod\_rang) references Rang(cod\_rang) on delete CASCADE,

unique(nume, prenume)

);

CREATE TABLE Utilizeaza(

cod\_arma number(3),

cod\_munitie number(3),

primary key(cod\_arma, cod\_munitie),

foreign key(cod\_arma) references Arma(cod\_arma) on delete CASCADE,

foreign key(cod\_munitie) references Munitie(cod\_munitie) on delete CASCADE

);

CREATE TABLE A\_fost\_decorat\_cu(

cod\_medalie number(3),

cod\_soldat number(3),

data\_acordare date not null,

primary key(cod\_medalie, cod\_soldat),

foreign key(cod\_medalie) references Medalie(cod\_medalie) on delete CASCADE,

foreign key(cod\_soldat) references Soldat(cod\_soldat) on delete CASCADE

);

/\*stop creare tabele\*/

/\*start secvente\*/

create sequence incrementare\_cod\_regiune

increment by 1

start with 1

nocycle

nocache;

create sequence incrementare\_cod\_tara

increment by 1

start with 1

nocycle

nocache;

create sequence incrementare\_cod\_oras

increment by 1

start with 1

nocycle

nocache;

create sequence incrementare\_cod\_pluton

increment by 1

start with 1

nocycle

nocache;

create sequence incrementare\_cod\_rang

increment by 1

start with 1

nocycle

nocache;

create sequence incrementare\_cod\_medalie

increment by 1

start with 1

nocycle

nocache;

create sequence incrementare\_cod\_munitie

increment by 1

start with 1

nocycle

nocache;

create sequence incrementare\_cod\_arma

increment by 1

start with 1

nocycle

nocache;

create sequence incrementare\_cod\_echipament

increment by 1

start with 1

nocycle

nocache;

create sequence incrementare\_cod\_soldat

increment by 1

start with 1

nocycle

nocache;

/\*start secvente\*/

/\*start inserari\*/

insert into Regiune values(incrementare\_cod\_regiune.nextval,'America de Nord',NULL);

insert into Regiune values(incrementare\_cod\_regiune.nextval,'Africa de Nord','Italia');

insert into Regiune values(incrementare\_cod\_regiune.nextval,'Europa de Vest','Germania');

insert into Regiune values(incrementare\_cod\_regiune.nextval,'Europa de Sud','Italia');

insert into Regiune values(incrementare\_cod\_regiune.nextval,'Peninsula Balcanic?','România+Bulgaria');

insert into Regiune values(incrementare\_cod\_regiune.nextval,'Asia','Japonia');

insert into Tara values(incrementare\_cod\_tara.nextval,'Statele Unite ale Americii',1);

insert into Tara values(incrementare\_cod\_tara.nextval,'Canada',1);

insert into Tara values(incrementare\_cod\_tara.nextval,'Libia',2);

insert into Tara values(incrementare\_cod\_tara.nextval,'Tunisia',2);

insert into Tara values(incrementare\_cod\_tara.nextval,'Algeria',2);

insert into Tara values(incrementare\_cod\_tara.nextval,'Belgia',3);

insert into Tara values(incrementare\_cod\_tara.nextval,'Olanda',3);

insert into Tara values(incrementare\_cod\_tara.nextval,'Fran?a',3);

insert into Tara values(incrementare\_cod\_tara.nextval,'Italia',4);

insert into Tara values(incrementare\_cod\_tara.nextval,'Grecia',5);

insert into Tara values(incrementare\_cod\_tara.nextval,'Filipine',6);

insert into Tara values(incrementare\_cod\_tara.nextval,'Japonia',6);

insert into Tara values(incrementare\_cod\_tara.nextval,'Singapore',6);

insert into Tara values(incrementare\_cod\_tara.nextval,'Indonezia',6);

insert into Tara values(incrementare\_cod\_tara.nextval,'Papua Noua Guinee',6);

insert into Tara values(incrementare\_cod\_tara.nextval,'Regatul Unit',3);

insert into Tara values(incrementare\_cod\_tara.nextval,'Germania',3);

insert into Munitie values(incrementare\_cod\_munitie.nextval,'.303 British','.303',300);

insert into Munitie values(incrementare\_cod\_munitie.nextval,'.45 ACP','.45',300);

insert into Munitie values(incrementare\_cod\_munitie.nextval,'.30 Carbine','.30',160);

insert into Munitie values(incrementare\_cod\_munitie.nextval,'.30-06 Springfield','.30',100);

insert into Munitie values(incrementare\_cod\_munitie.nextval,'.50 BMG','.50',5000);

insert into Munitie values(incrementare\_cod\_munitie.nextval,'9mm Parabellum','9mm',360);

insert into Medalie values(incrementare\_cod\_medalie.nextval,'Medal of Honor','curaj');

insert into Medalie values(incrementare\_cod\_medalie.nextval,'Distinguished Service Cross','eroism');

insert into Medalie values(incrementare\_cod\_medalie.nextval,'Silver Star','curaj');

insert into Medalie values(incrementare\_cod\_medalie.nextval,'Bronze Star','curaj');

insert into Medalie values(incrementare\_cod\_medalie.nextval,'Purple Heart','ranit/decedat');

insert into Rang values(incrementare\_cod\_rang.nextval,'soldat',50);

insert into Rang values(incrementare\_cod\_rang.nextval,'soldat prima clasa',55);

insert into Rang values(incrementare\_cod\_rang.nextval,'caporal',65);

insert into Rang values(incrementare\_cod\_rang.nextval,'sergent',80);

insert into Rang values(incrementare\_cod\_rang.nextval,'prim sergent',140);

insert into Rang values(incrementare\_cod\_rang.nextval,'locotenent',165);

insert into Rang values(incrementare\_cod\_rang.nextval,'capitan',200);

insert into Rang values(incrementare\_cod\_rang.nextval,'maior',250);

insert into Arma values(incrementare\_cod\_arma.nextval,'Colt M1911','.45',1);

insert into Arma values(incrementare\_cod\_arma.nextval,'Thompson M1A1','.45',1);

insert into Arma values(incrementare\_cod\_arma.nextval,'M3A1 "Grease Gun"','.45',1);

insert into Arma values(incrementare\_cod\_arma.nextval,'M1 Carbine','.30',1);

insert into Arma values(incrementare\_cod\_arma.nextval,'M1 Garand','.30',1);

insert into Arma values(incrementare\_cod\_arma.nextval,'Springfield M1903','.30',1);

insert into Arma values(incrementare\_cod\_arma.nextval,'M1918A2 BAR','.30',1);

insert into Arma values(incrementare\_cod\_arma.nextval,'M2 Browning','.50',1);

insert into Arma values(incrementare\_cod\_arma.nextval,'Sten','9mm',16);

insert into Arma values(incrementare\_cod\_arma.nextval,'Bren','.303',16);

insert into Echipament values(incrementare\_cod\_echipament.nextval,'Medic',0,1);

insert into Echipament values(incrementare\_cod\_echipament.nextval,'Lunetist',1,6);

insert into Echipament values(incrementare\_cod\_echipament.nextval,'Mitralior',0,8);

insert into Echipament values(incrementare\_cod\_echipament.nextval,'Infanterist v1',2,5);

insert into Echipament values(incrementare\_cod\_echipament.nextval,'Infanterist v2',2,4);

insert into Echipament values(incrementare\_cod\_echipament.nextval,'Infanterist v3',3,3);

insert into Echipament values(incrementare\_cod\_echipament.nextval,'Infanterist v4',3,7);

insert into Echipament values(incrementare\_cod\_echipament.nextval,'Infanterist v5',3,2);

insert into Utilizeaza values(1,2);

insert into Utilizeaza values(2,2);

insert into Utilizeaza values(3,2);

insert into Utilizeaza values(3,6);

insert into Utilizeaza values(4,3);

insert into Utilizeaza values(5,4);

insert into Utilizeaza values(6,4);

insert into Utilizeaza values(7,4);

insert into Utilizeaza values(8,5);

insert into Utilizeaza values(9,6);

insert into Utilizeaza values(10,1);

insert into Oras values(incrementare\_cod\_oras.nextval,'Washington','Nu',1);

insert into Oras values(incrementare\_cod\_oras.nextval,'New York','Nu',1);

insert into Oras values(incrementare\_cod\_oras.nextval,'Ottawa','Nu',2);

insert into Oras values(incrementare\_cod\_oras.nextval,'Quebec','Nu',2);

insert into Oras values(incrementare\_cod\_oras.nextval,'Tripoli','Da',3);

insert into Oras values(incrementare\_cod\_oras.nextval,'Tunis','Da',4);

insert into Oras values(incrementare\_cod\_oras.nextval,'Algiers','Da',5);

insert into Oras values(incrementare\_cod\_oras.nextval,'Bruxel','Da',6);

insert into Oras values(incrementare\_cod\_oras.nextval,'Liege','Da',6);

insert into Oras values(incrementare\_cod\_oras.nextval,'Amsterdam','Da',7);

insert into Oras values(incrementare\_cod\_oras.nextval,'Eindhoven','Da',7);

insert into Oras values(incrementare\_cod\_oras.nextval,'Paris','Nu',8);

insert into Oras values(incrementare\_cod\_oras.nextval,'Caen','Da',8);

insert into Oras values(incrementare\_cod\_oras.nextval,'Napoli','Da',9);

insert into Oras values(incrementare\_cod\_oras.nextval,'Roma','Da',9);

insert into Oras values(incrementare\_cod\_oras.nextval,'Salonic','Da',10);

insert into Oras values(incrementare\_cod\_oras.nextval,'Manila','Da',11);

insert into Oras values(incrementare\_cod\_oras.nextval,'Okinawa','Da',12);

insert into Oras values(incrementare\_cod\_oras.nextval,'Singapore','Da',13);

insert into Oras values(incrementare\_cod\_oras.nextval,'Jakarta','Da',14);

insert into Oras values(incrementare\_cod\_oras.nextval,'Port Moresby','Nu',15);

insert into Oras values(incrementare\_cod\_oras.nextval,'Londra','Nu',16);

insert into Oras values(incrementare\_cod\_oras.nextval,'Cologne','Da',17);

insert into Oras values(incrementare\_cod\_oras.nextval,'Frankfurt','Da',17);

insert into Pluton values(incrementare\_cod\_pluton.nextval,11,6);

insert into Pluton values(incrementare\_cod\_pluton.nextval,12,11);

insert into Pluton values(incrementare\_cod\_pluton.nextval,13,16);

insert into Pluton values(incrementare\_cod\_pluton.nextval,21,14);

insert into Pluton values(incrementare\_cod\_pluton.nextval,22,24);

insert into Pluton values(incrementare\_cod\_pluton.nextval,23,18);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Sofei','Andrei-Adrian',to\_date('16-04-1920','dd-mm-yyyy'),51,1,NULL,8,8);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Sherman','Fred',to\_date('08-05-1927','dd-mm-yyyy'),47,2,1,5,7);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Howells','Benjamin',to\_date('28-10-1929','dd-mm-yyyy'),37,3,1,6,7);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Warner','Jacob',to\_date('01-01-1933','dd-mm-yyyy'),41,4,1,7,7);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'King','Douglas',to\_date('05-01-1933','dd-mm-yyyy'),39,5,1,8,7);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Bright','Alex',to\_date('01-06-1934','dd-mm-yyyy'),44,6,1,5,7);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Walton','Simon',to\_date('29-04-1930','dd-mm-yyyy'),41,1,1,8,6);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Warren','Josh',to\_date('29-09-1930','dd-mm-yyyy'),36,2,2,7,6);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Flynn','Scott',to\_date('10-07-1931','dd-mm-yyyy'),34,3,3,8,6);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Torres','Marco',to\_date('22-07-1932','dd-mm-yyyy'),33,4,4,5,6);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Whittle','Benjamin',to\_date('05-06-1933','dd-mm-yyyy'),29,5,5,4,6);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Ferguson','Patrick',to\_date('09-11-1933','dd-mm-yyyy'),27,6,6,5,6);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Barnes','Tyler',to\_date('22-02-1934','dd-mm-yyyy'),28,1,7,7,5);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Wilkins','Jacob',to\_date('15-12-1935','dd-mm-yyyy'),31,2,8,8,5);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Watson','Scott',to\_date('11-04-1937','dd-mm-yyyy'),30,3,9,4,5);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Gordon','Josh',to\_date('06-05-1937','dd-mm-yyyy'),28,4,10,5,5);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Banks','Robbie',to\_date('21-09-1938','dd-mm-yyyy'),29,5,11,5,5);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Caldwell','Josh',to\_date('27-04-1939','dd-mm-yyyy'),27,6,12,4,5);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Stuart','Mason',to\_date('10-07-1935','dd-mm-yyyy'),27,1,13,4,4);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Smith','Tobias',to\_date('25-07-1937','dd-mm-yyyy'),26,1,13,5,4);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Ferguson','Jeremy',to\_date('01-09-1938','dd-mm-yyyy'),28,2,14,7,4);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Hubbard','Tyler',to\_date('09-03-1939','dd-mm-yyyy'),31,2,14,8,4);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Porter','Scott',to\_date('30-05-1939','dd-mm-yyyy'),23,3,15,6,4);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Woods','Glenn',to\_date('28-04-1941','dd-mm-yyyy'),24,3,15,5,4);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Morgan','Gerald',to\_date('20-10-1941','dd-mm-yyyy'),23,4,16,4,4);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Lawson','David',to\_date('08-09-1942','dd-mm-yyyy'),22,5,17,4,4);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Wagner','Tony',to\_date('25-11-1942','dd-mm-yyyy'),26,6,18,4,4);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Hardy','Fred',to\_date('09-02-1938','dd-mm-yyyy'),23,1,1,1,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Peterson','Jacob',to\_date('31-05-1938','dd-mm-yyyy'),21,1,1,2,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Mueller','Mason',to\_date('05-06-1938','dd-mm-yyyy'),20,1,19,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Carter','Scott',to\_date('17-06-1938','dd-mm-yyyy'),24,1,20,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Campbell','Marco',to\_date('24-06-1938','dd-mm-yyyy'),22,2,21,7,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Vaughn','Douglas',to\_date('21-10-1938','dd-mm-yyyy'),23,2,21,5,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Wolf','Lawrence',to\_date('19-07-1939','dd-mm-yyyy'),21,2,22,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Walters','Tobias',to\_date('11-09-1939','dd-mm-yyyy'),22,3,23,5,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Bush','Patrick',to\_date('23-05-1940','dd-mm-yyyy'),24,3,23,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Graves','Glenn',to\_date('27-10-1940','dd-mm-yyyy'),26,3,24,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Perkins','Zack',to\_date('13-02-1941','dd-mm-yyyy'),24,4,25,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Smith','Jeremy',to\_date('08-08-1941','dd-mm-yyyy'),27,4,25,7,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Hardy','Robbie',to\_date('25-06-1942','dd-mm-yyyy'),25,5,26,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Chapman','Jacob',to\_date('05-12-1942','dd-mm-yyyy'),20,5,26,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Daniels','Melvin',to\_date('19-09-1941','dd-mm-yyyy'),23,6,27,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Reyes','Gerald',to\_date('06-05-1942','dd-mm-yyyy'),26,6,27,4,3);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Campbell','Tony',to\_date('24-10-1942','dd-mm-yyyy'),22,1,1,3,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Thompson','Aron',to\_date('27-12-1942','dd-mm-yyyy'),19,2,2,1,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Fernandez','Lawrence',to\_date('26-11-1940','dd-mm-yyyy'),21,3,3,1,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Mcdonald','Mason',to\_date('07-09-1939','dd-mm-yyyy'),23,1,30,7,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Burns','David',to\_date('03-06-1941','dd-mm-yyyy'),20,2,2,3,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Morgan','Zack',to\_date('07-09-1940','dd-mm-yyyy'),21,3,3,3,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Woods','Isaac',to\_date('27-12-1942','dd-mm-yyyy'),18,4,38,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Peterson','Fred',to\_date('08-01-1939','dd-mm-yyyy'),24,1,30,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Chapman','Scott',to\_date('18-03-1940','dd-mm-yyyy'),22,1,30,6,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Stevenson','Douglas',to\_date('11-03-1941','dd-mm-yyyy'),23,1,31,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Owens','Glenn',to\_date('27-09-1942','dd-mm-yyyy'),21,1,31,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Hart','Scott',to\_date('25-12-1942','dd-mm-yyyy'),20,2,32,5,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Hardy','David',to\_date('11-03-1942','dd-mm-yyyy'),18,2,32,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Norris','Simon',to\_date('15-10-1942','dd-mm-yyyy'),19,2,33,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Barker','Tomas',to\_date('27-12-1939','dd-mm-yyyy'),24,2,34,5,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Peterson','Melvin',to\_date('04-09-1940','dd-mm-yyyy'),25,3,35,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Castro','Josh',to\_date('28-10-1942','dd-mm-yyyy'),23,3,35,6,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Higgins','Robbie',to\_date('27-12-1942','dd-mm-yyyy'),20,3,36,5,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Klein','Mason',to\_date('15-10-1941','dd-mm-yyyy'),21,3,37,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Cunningham','Jon',to\_date('07-09-1940','dd-mm-yyyy'),22,4,38,5,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Smith','Isaac',to\_date('04-05-1941','dd-mm-yyyy'),21,4,38,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Hansen','Aron',to\_date('11-03-1943','dd-mm-yyyy'),19,4,39,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Johnson','Zack',to\_date('15-10-1941','dd-mm-yyyy'),19,4,39,6,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Ramirez','Marco',to\_date('28-10-1942','dd-mm-yyyy'),18,5,40,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Stevenson','Scott',to\_date('04-05-1941','dd-mm-yyyy'),20,5,40,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Castro','Tony',to\_date('04-09-1940','dd-mm-yyyy'),23,5,41,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Higgins','Douglas',to\_date('28-01-1939','dd-mm-yyyy'),26,5,41,5,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Matthews','Gerald',to\_date('07-06-1942','dd-mm-yyyy'),24,6,42,5,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Houston','Alex',to\_date('27-12-1939','dd-mm-yyyy'),26,6,42,6,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Frost','Zack',to\_date('04-05-1940','dd-mm-yyyy'),25,6,43,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Montgomery','David',to\_date('07-06-1942','dd-mm-yyyy'),21,6,43,4,2);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Wright','Kristian',to\_date('07-09-1939','dd-mm-yyyy'),21,1,30,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Blithe','Oliver',to\_date('04-09-1940','dd-mm-yyyy'),23,1,30,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Klein','Jon',to\_date('11-03-1941','dd-mm-yyyy'),20,1,30,3,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Cunningham','Stefan',to\_date('07-06-1942','dd-mm-yyyy'),21,1,31,5,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Johnson','Lawrence',to\_date('04-05-1940','dd-mm-yyyy'),22,1,31,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Morgan','Tomas',to\_date('07-06-1942','dd-mm-yyyy'),22,2,32,6,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Owens','Jon',to\_date('27-09-1942','dd-mm-yyyy'),21,2,32,5,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Powers','Robbie',to\_date('28-01-1942','dd-mm-yyyy'),19,2,33,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Matthews','Zack',to\_date('28-10-1942','dd-mm-yyyy'),20,2,34,6,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Wright','Tony',to\_date('15-10-1941','dd-mm-yyyy'),24,3,35,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Todd','Mason',to\_date('04-05-1940','dd-mm-yyyy'),19,3,36,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Turner','Alex',to\_date('28-10-1942','dd-mm-yyyy'),20,3,36,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Ryan','Stefan',to\_date('07-06-1942','dd-mm-yyyy'),21,3,37,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Daniel','Charles',to\_date('27-09-1942','dd-mm-yyyy'),21,4,38,6,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Knight','Alan',to\_date('28-10-1942','dd-mm-yyyy'),21,4,38,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'White','Kristian',to\_date('28-01-1939','dd-mm-yyyy'),23,4,39,5,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Parker','Aron',to\_date('28-10-1942','dd-mm-yyyy'),22,5,40,6,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Montgomery','Oliver',to\_date('28-10-1942','dd-mm-yyyy'),23,5,40,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Johnson','Tomas',to\_date('04-05-1940','dd-mm-yyyy'),19,5,41,6,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Owens','David',to\_date('07-06-1942','dd-mm-yyyy'),18,6,42,5,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Woods','Mason',to\_date('15-10-1941','dd-mm-yyyy'),2,6,42,4,1);

insert into Soldat values(incrementare\_cod\_soldat.nextval,'Ramirez','Tony',to\_date('27-09-1942','dd-mm-yyyy'),20,6,43,4,1);

insert into A\_fost\_decorat\_cu values(1,1,to\_date('09-06-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(1,14,to\_date('24-03-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(1,33,to\_date('12-05-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(2,1,to\_date('08-02-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(2,26,to\_date('28-09-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(2,7,to\_date('12-05-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(2,5,to\_date('30-11-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(3,16,to\_date('07-01-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(3,32,to\_date('14-07-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(3,47,to\_date('08-02-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(3,89,to\_date('09-06-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(3,54,to\_date('15-08-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(3,23,to\_date('08-08-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,46,to\_date('24-03-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,37,to\_date('28-09-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,68,to\_date('30-11-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,39,to\_date('17-12-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,62,to\_date('19-03-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,19,to\_date('08-02-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,6,to\_date('07-01-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,86,to\_date('27-06-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,71,to\_date('27-06-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(4,44,to\_date('09-06-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,13,to\_date('14-07-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,87,to\_date('15-08-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,52,to\_date('30-11-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,74,to\_date('08-08-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,42,to\_date('01-05-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,18,to\_date('19-03-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,19,to\_date('15-08-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,26,to\_date('28-09-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,39,to\_date('07-01-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,65,to\_date('12-05-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,48,to\_date('17-12-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,95,to\_date('27-06-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,57,to\_date('08-08-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,64,to\_date('08-02-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,35,to\_date('17-03-1945','dd-mm-yyyy'));

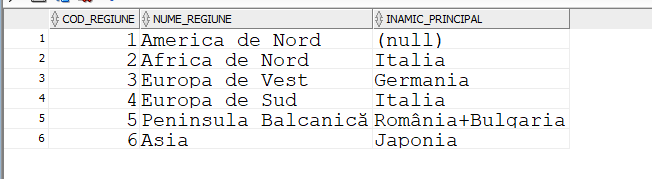
insert into A\_fost\_decorat\_cu values(5,3,to\_date('17-03-1945','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,79,to\_date('09-06-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,60,to\_date('14-07-1944','dd-mm-yyyy'));

insert into A\_fost\_decorat\_cu values(5,20,to\_date('19-03-1945','dd-mm-yyyy'));

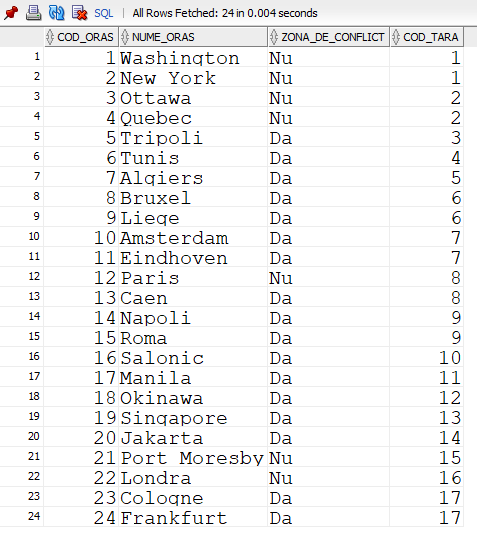
**Regiune:**



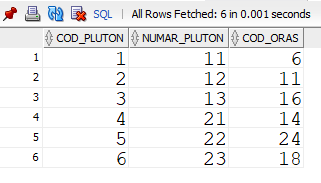
**Tara:**



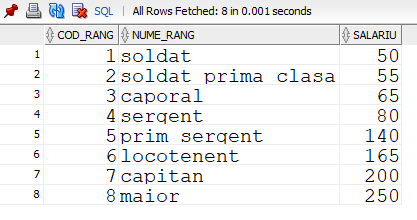
**Oras:**



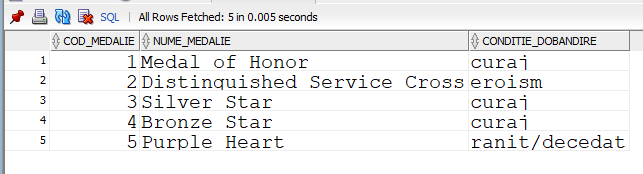
**Pluton:**



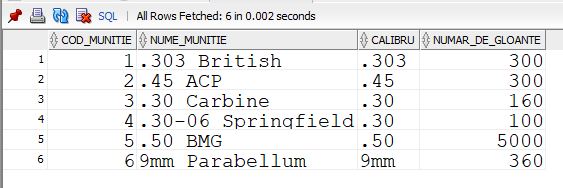
**Rang:**



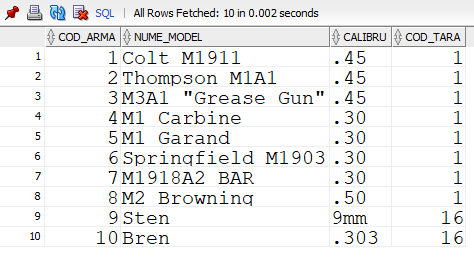
**Medalie:**



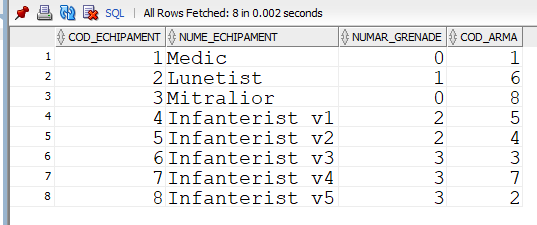
**Munitie:**



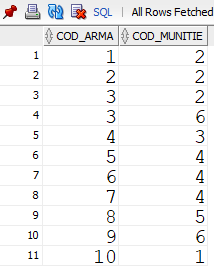
**Arma:**



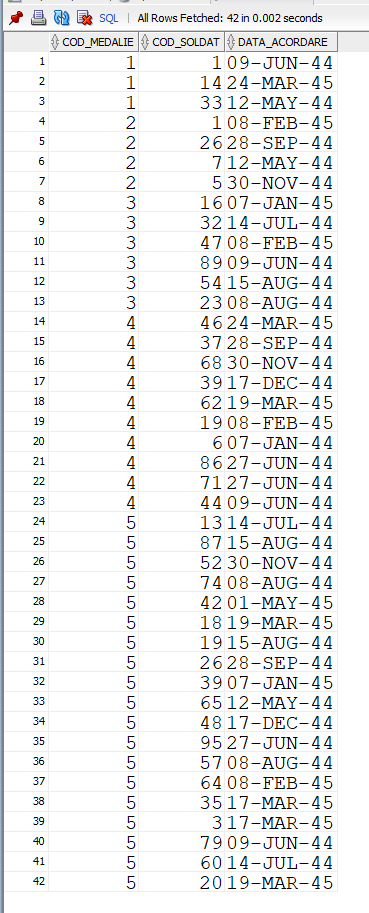
**Echipament:**



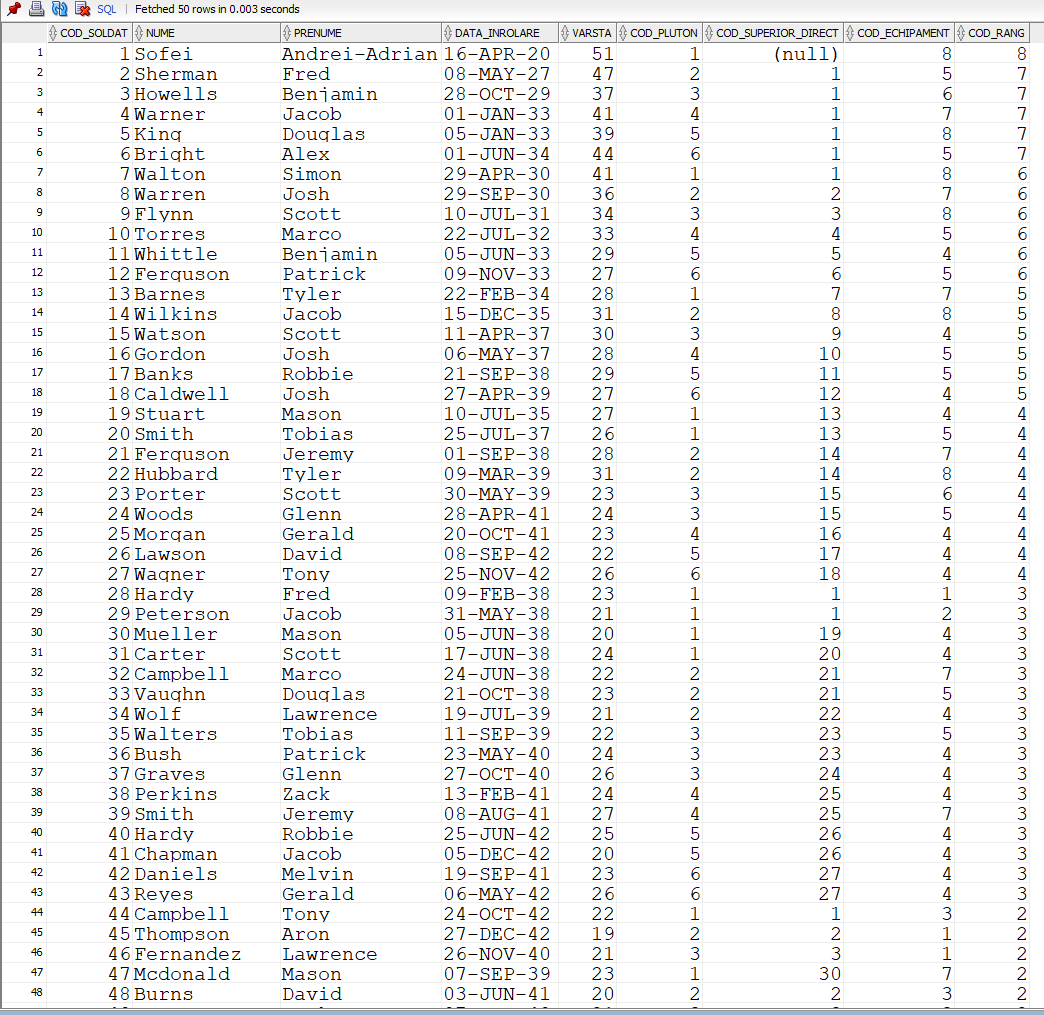
**Utilizeaza:**

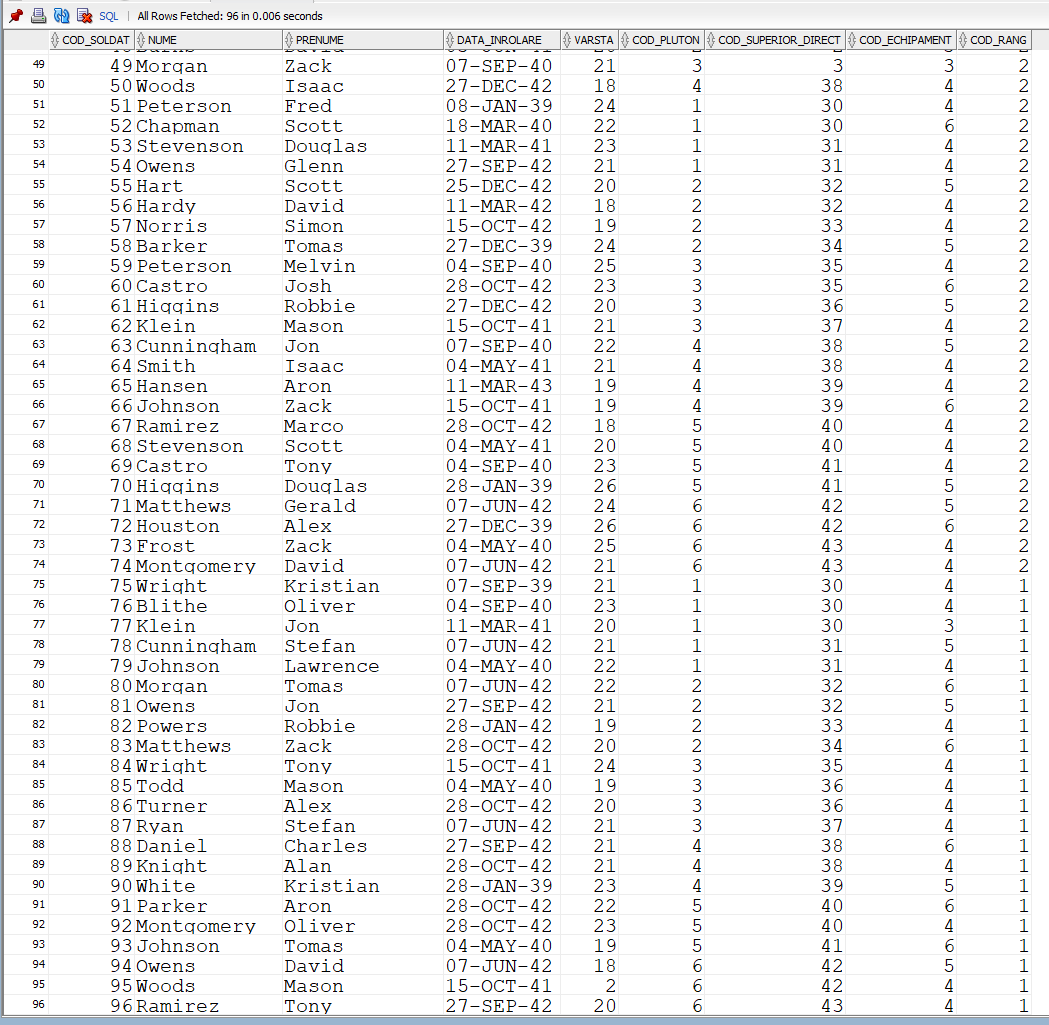


**A\_fost\_decorat\_cu:**



**Soldat:**





**--6.**

--Afisati cati soldati fac parte din plutonul X si sunt ofiteri (maior,capitan,locotenent)

--folosind 2 tabele in care tinem soldatii din plutonul 1 respectiv ofiterii)

CREATE OR REPLACE FUNCTION nr\_ofiteri\_plutonul\_x

(pluton pluton.cod\_pluton%type)

RETURN NUMBER

IS

nr\_ofiteri\_din\_pluton number(3) :=0;

nr\_soldati\_din\_pluton number(3) :=0;

--tablou indexat

type lista\_soldati is table of soldat%rowtype index by pls\_integer;

soldati\_plutonul\_x lista\_soldati;

--tablou imbricat

type lista\_ofiteri is table of soldat%rowtype;

ofiteri lista\_ofiteri := lista\_ofiteri();

BEGIN

select count(\*) into nr\_soldati\_din\_pluton

from soldat

where cod\_pluton = pluton;

select \* bulk collect into soldati\_plutonul\_x

from soldat

where cod\_pluton = pluton;

select \* bulk collect into ofiteri

from soldat

where cod\_rang = 6 or cod\_rang = 7 or cod\_rang = 8;

for i in soldati\_plutonul\_x.FIRST..soldati\_plutonul\_x.LAST loop

for j in ofiteri.FIRST..ofiteri.LAST loop

if soldati\_plutonul\_x(i).cod\_soldat = ofiteri(j).cod\_soldat then

nr\_ofiteri\_din\_pluton := nr\_ofiteri\_din\_pluton + 1;

end if;

end loop;

end loop;

return nr\_ofiteri\_din\_pluton;

EXCEPTION

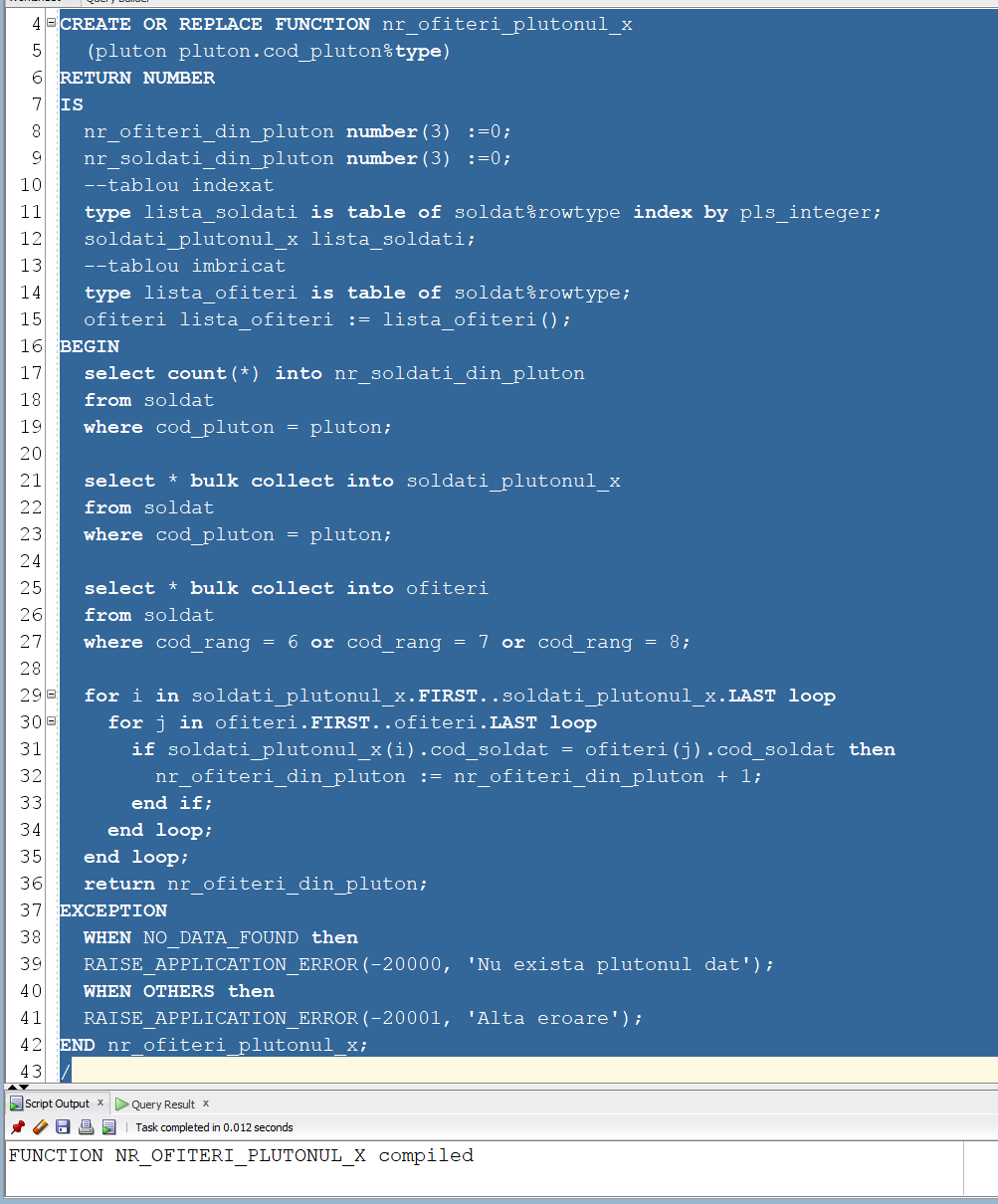
WHEN NO\_DATA\_FOUND then

RAISE\_APPLICATION\_ERROR(-20000, 'Nu exista plutonul dat');

WHEN OTHERS then

RAISE\_APPLICATION\_ERROR(-20001, 'Alta eroare');

END nr\_ofiteri\_plutonul\_x;

/

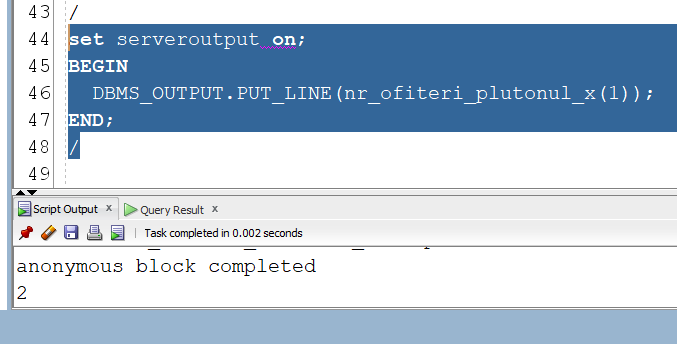
set serveroutput on;

BEGIN

DBMS\_OUTPUT.PUT\_LINE(nr\_ofiteri\_plutonul\_x(1));

END;

/



**--7.**

--Folosind un cursor afisati pentru fiecare pluton

--cati soldati are si media de varsta pe fiecare pluton

CREATE OR REPLACE PROCEDURE detalii\_plutoane

IS

plutonul pluton.cod\_pluton%TYPE;

nr\_soldati\_pluton number(5);

varsta\_medie float(10);

CURSOR cursor\_detalii\_plutoane is

select p.cod\_pluton codul\_plutonului, count(s.cod\_soldat) nr\_soldati, avg(s.varsta) medie\_varsta

from pluton p, soldat s

where p.cod\_pluton = s.cod\_pluton(+)

group by p.cod\_pluton;

BEGIN

OPEN cursor\_detalii\_plutoane;

loop

fetch cursor\_detalii\_plutoane into plutonul,nr\_soldati\_pluton,varsta\_medie;

exit when cursor\_detalii\_plutoane%NOTFOUND;

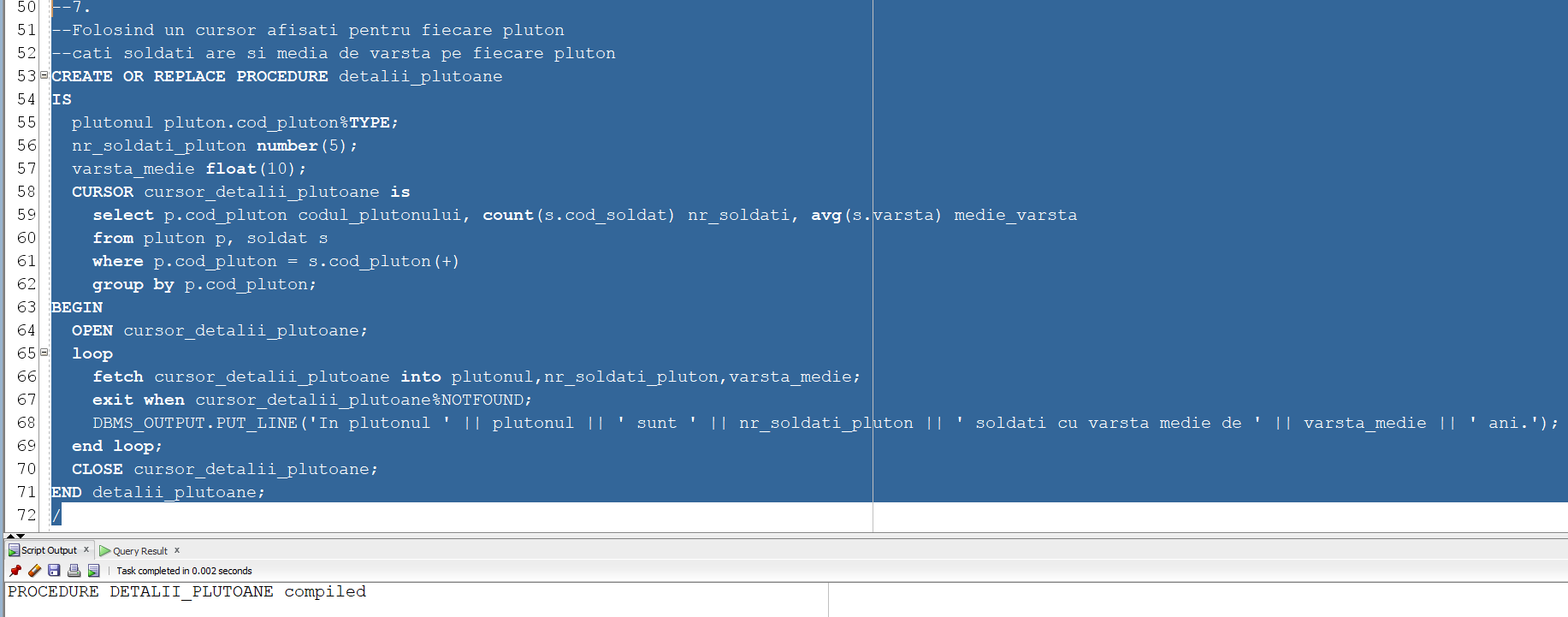
DBMS\_OUTPUT.PUT\_LINE('In plutonul ' || plutonul || ' sunt ' || nr\_soldati\_pluton || ' soldati cu varsta medie de ' || varsta\_medie || ' ani.');

end loop;

CLOSE cursor\_detalii\_plutoane;

END detalii\_plutoane;

/



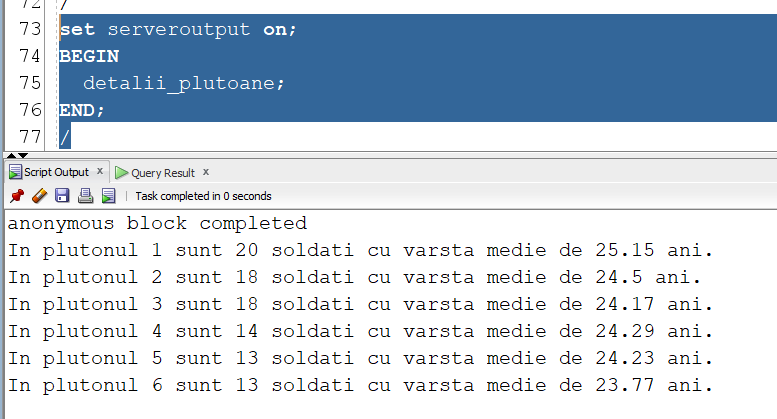
set serveroutput on;

BEGIN

detalii\_plutoane;

END;

/



**--8.**

--Afisati arma pe care o foloseste soldatul cu un prenume dat

CREATE OR REPLACE FUNCTION ce\_arma\_foloseste

(prenume\_soldat soldat.prenume%TYPE DEFAULT 'Andrei-Adrian')

RETURN VARCHAR2

IS

cod\_soldat\_dat soldat.cod\_soldat%TYPE;

nume\_arma arma.nume\_model%TYPE;

BEGIN

select s.cod\_soldat into cod\_soldat\_dat

from soldat s

where s.prenume = prenume\_soldat;

select a.nume\_model into nume\_arma

from soldat s, echipament e, arma a

where s.prenume = prenume\_soldat and

s.cod\_echipament = e.cod\_echipament and

e.cod\_arma = a.cod\_arma;

return nume\_arma;

EXCEPTION

WHEN NO\_DATA\_FOUND then

RAISE\_APPLICATION\_ERROR(-20000, 'Nu exista niciun soldat cu prenumele dat');

WHEN TOO\_MANY\_ROWS then

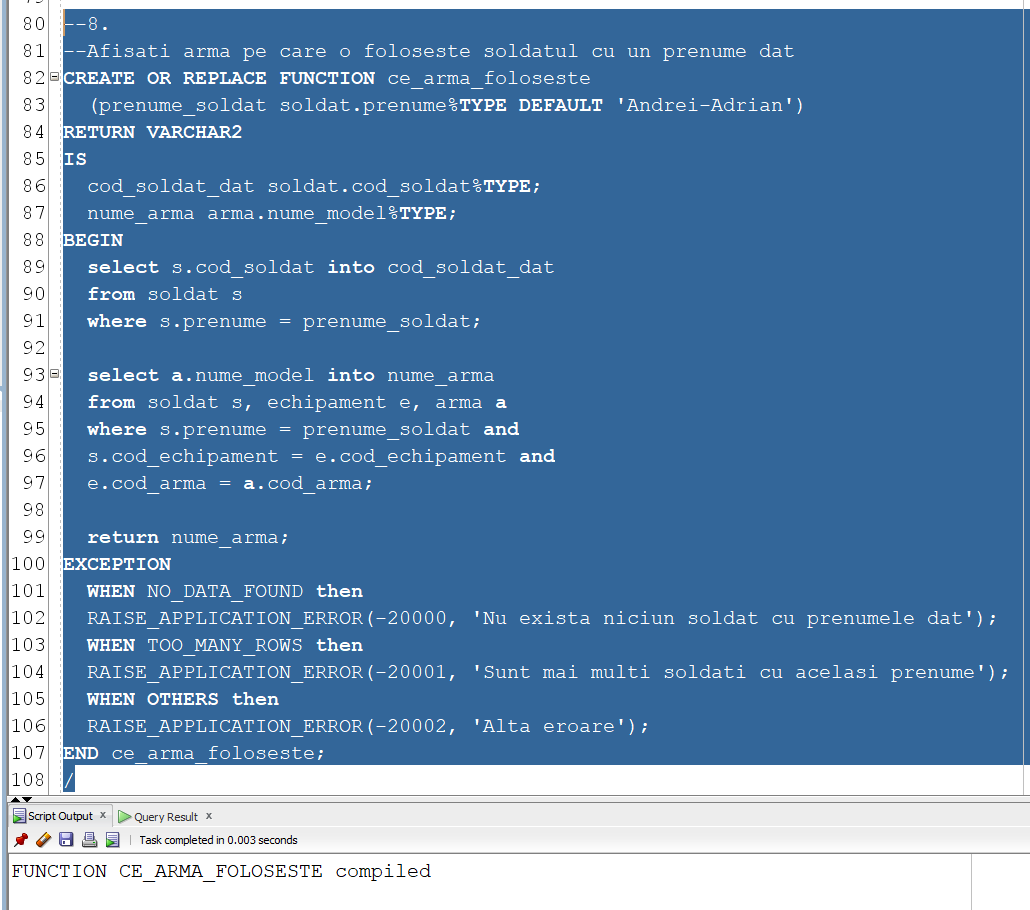
RAISE\_APPLICATION\_ERROR(-20001, 'Sunt mai multi soldati cu acelasi prenume');

WHEN OTHERS then

RAISE\_APPLICATION\_ERROR(-20002, 'Alta eroare');

END ce\_arma\_foloseste;

/



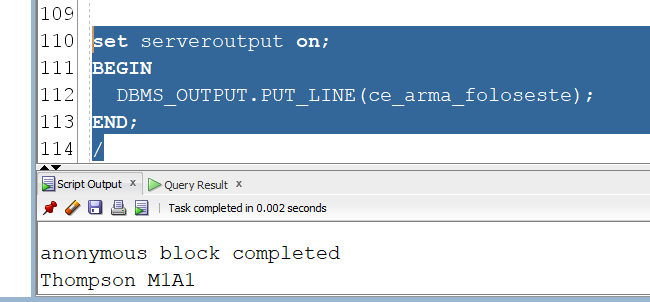
set serveroutput on;

BEGIN

DBMS\_OUTPUT.PUT\_LINE(ce\_arma\_foloseste);

END;

/

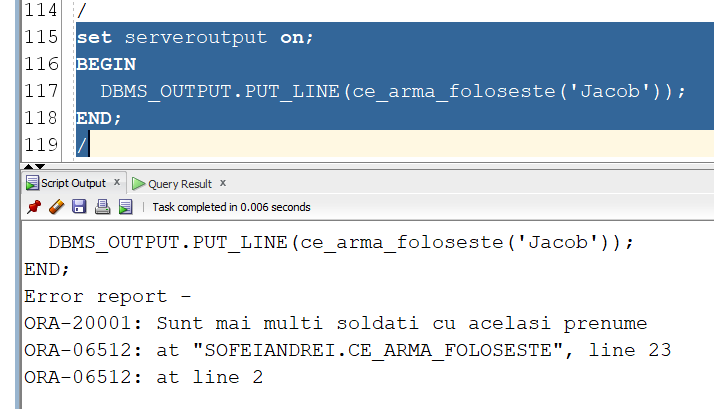
set serveroutput on;

BEGIN

DBMS\_OUTPUT.PUT\_LINE(ce\_arma\_foloseste('Jacob'));

END;

/



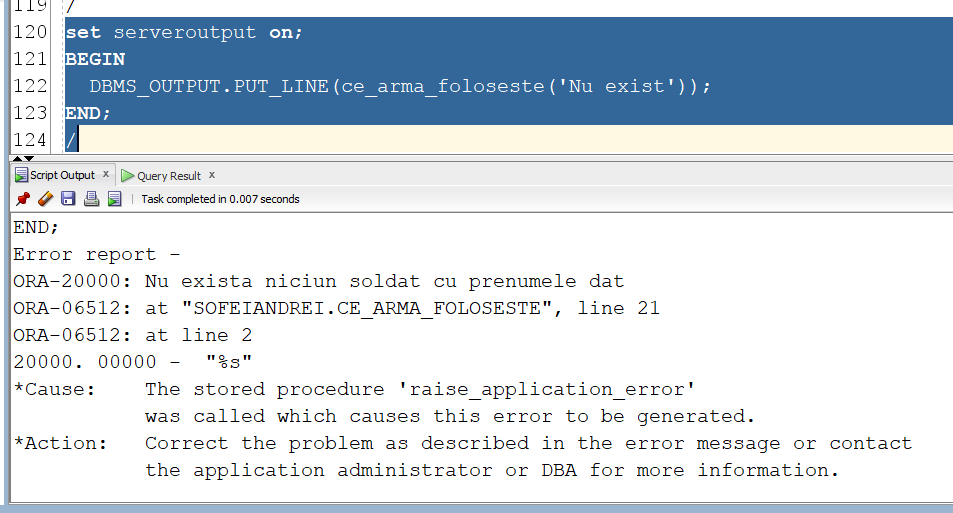
set serveroutput on;

BEGIN

DBMS\_OUTPUT.PUT\_LINE(ce\_arma\_foloseste('Nu exist'));

END;

/



**--9.**

--Afisati in ce regiune este stationat soldatul care foloseste echipamentul cu codul dat

CREATE OR REPLACE PROCEDURE regiune\_soldat

(echipament\_dat echipament.cod\_echipament%TYPE)

IS

cod\_soldat\_dat soldat.cod\_soldat%TYPE;

regiunea regiune.nume\_regiune%TYPE;

BEGIN

select s.cod\_soldat into cod\_soldat\_dat

from soldat s

where s.cod\_echipament = echipament\_dat;

select r.nume\_regiune into regiunea

from soldat s, pluton p, oras o, tara t, regiune r

where s.cod\_echipament = echipament\_dat and

s.cod\_pluton = p.cod\_pluton and

p.cod\_oras = o.cod\_oras and

o.cod\_tara = t.cod\_tara and

t.cod\_regiune = r.cod\_regiune;

DBMS\_OUTPUT.PUT\_LINE(regiunea);

EXCEPTION

WHEN NO\_DATA\_FOUND then

RAISE\_APPLICATION\_ERROR(-20000, 'Nu exista niciu soldat care foloseste echipamentul cu codul dat');

WHEN TOO\_MANY\_ROWS then

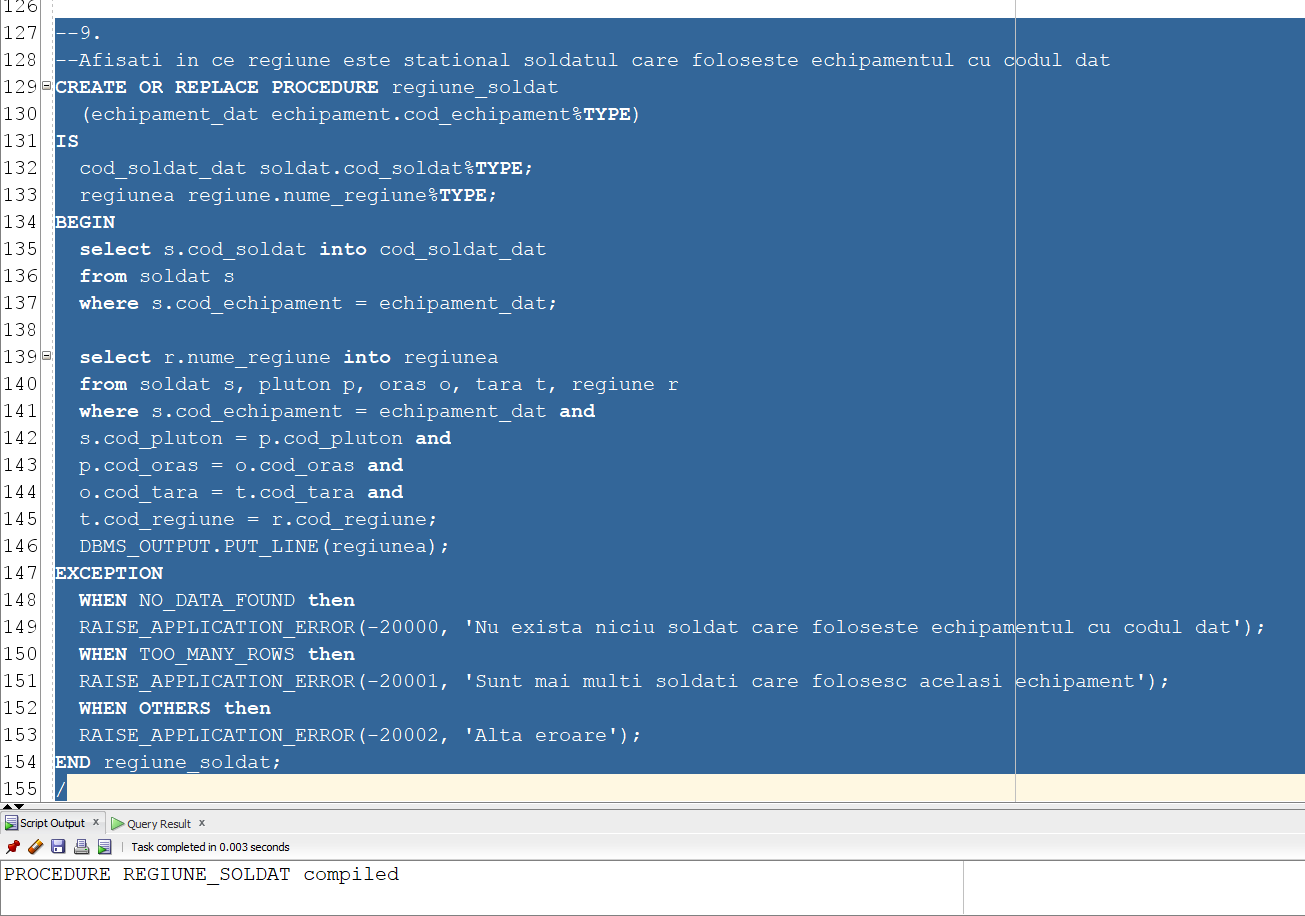
RAISE\_APPLICATION\_ERROR(-20001, 'Sunt mai multi soldati care folosesc acelasi echipament');

WHEN OTHERS then

RAISE\_APPLICATION\_ERROR(-20002, 'Alta eroare');

END regiune\_soldat;

/



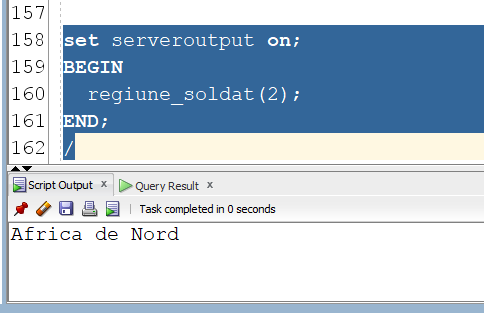
set serveroutput on;

BEGIN

regiune\_soldat(2);

END;

/



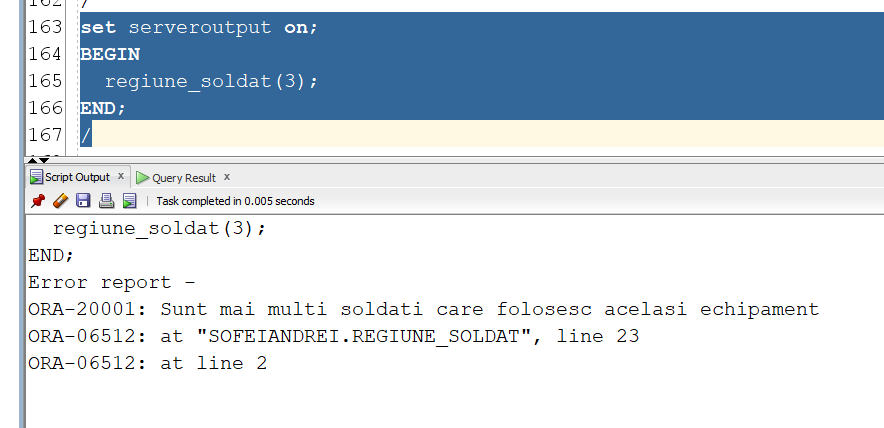
set serveroutput on;

BEGIN

regiune\_soldat(3);

END;

/



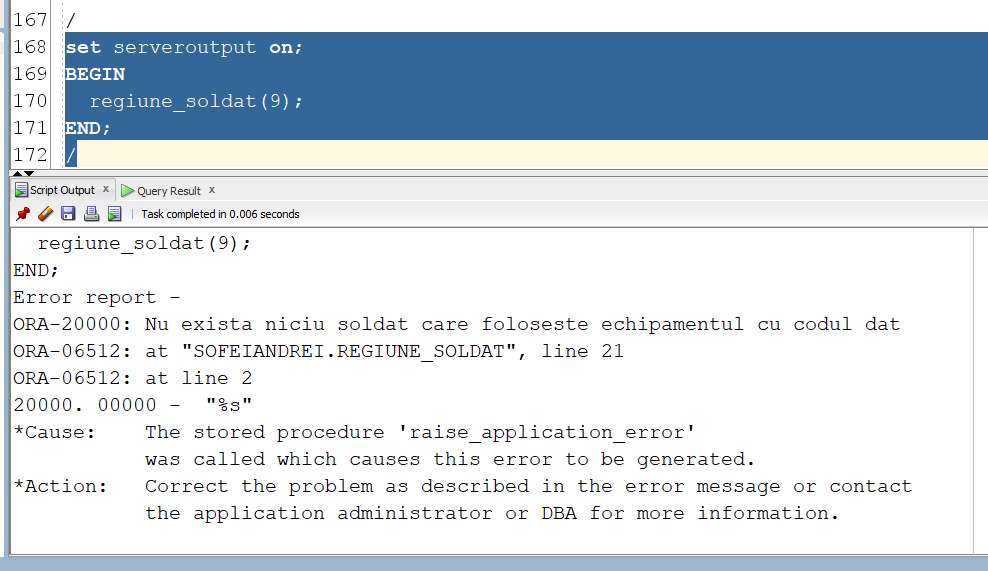
set serveroutput on;

BEGIN

regiune\_soldat(9);

END;

/



**--10**

--Trigger care sa nu permita modificarea datelor din tabelul soldat daca sunt efectuate in week-end(sambata/duminica) sau joia

CREATE OR REPLACE TRIGGER trigger\_sambata\_si\_duminica

BEFORE INSERT OR UPDATE OR DELETE ON soldat

BEGIN

if TO\_CHAR(SYSDATE, 'D') = 1 then

RAISE\_APPLICATION\_ERROR(-20001,'tabelul SOLDAT nu poate fi actualizat duminica');

elsif TO\_CHAR(SYSDATE, 'D') = 7 then

RAISE\_APPLICATION\_ERROR(-20001,'tabelul SOLDAT nu poate fi actualizat sambata');

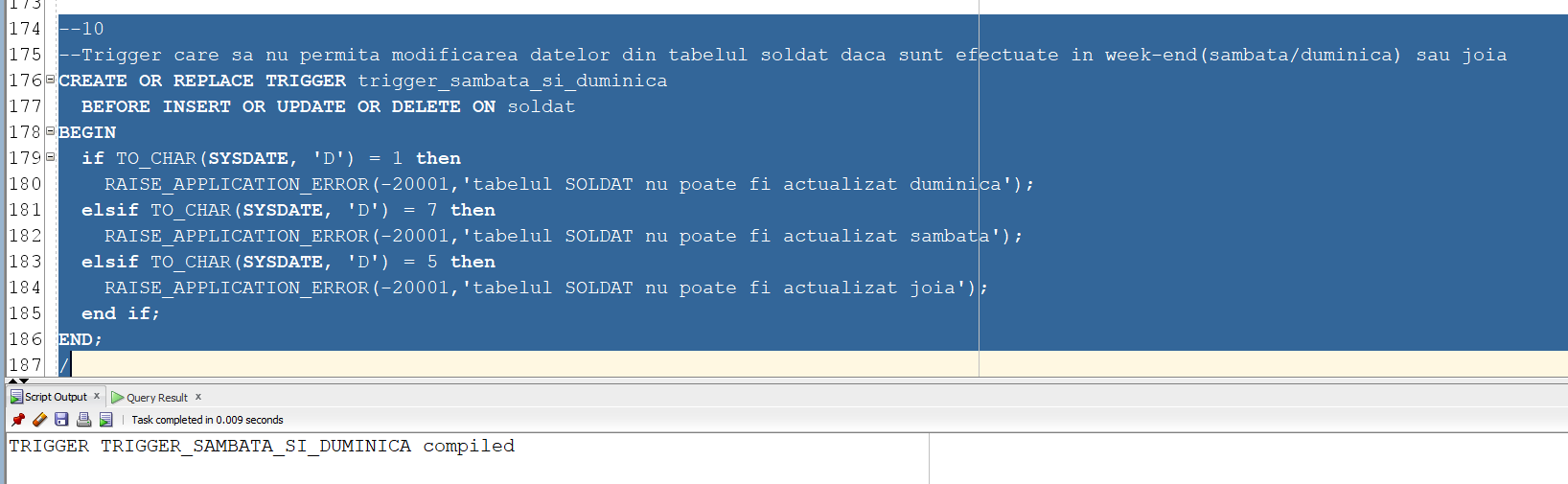
elsif TO\_CHAR(SYSDATE, 'D') = 5 then

RAISE\_APPLICATION\_ERROR(-20001,'tabelul SOLDAT nu poate fi actualizat joia');

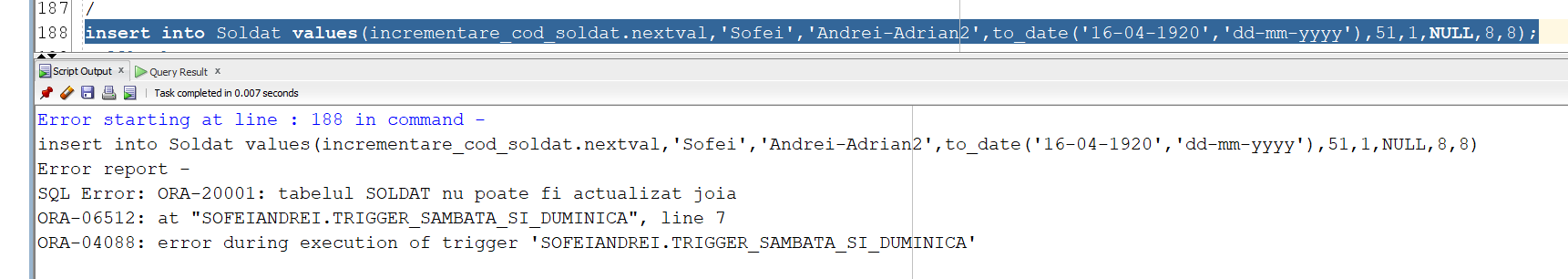
end if;

END;

/



insert into Soldat values(incrementare\_cod\_soldat.nextval,'Sofei','Andrei-Adrian2',to\_date('16-04-1920','dd-mm-yyyy'),51,1,NULL,8,8);

DROP TRIGGER trigger\_sambata\_si\_duminica;

**--11**

--Trigger care sa nu permita micsorarea varstei unui soldat

CREATE OR REPLACE TRIGGER trigger\_micsorare\_varsta

BEFORE UPDATE OF varsta ON soldat

FOR EACH ROW

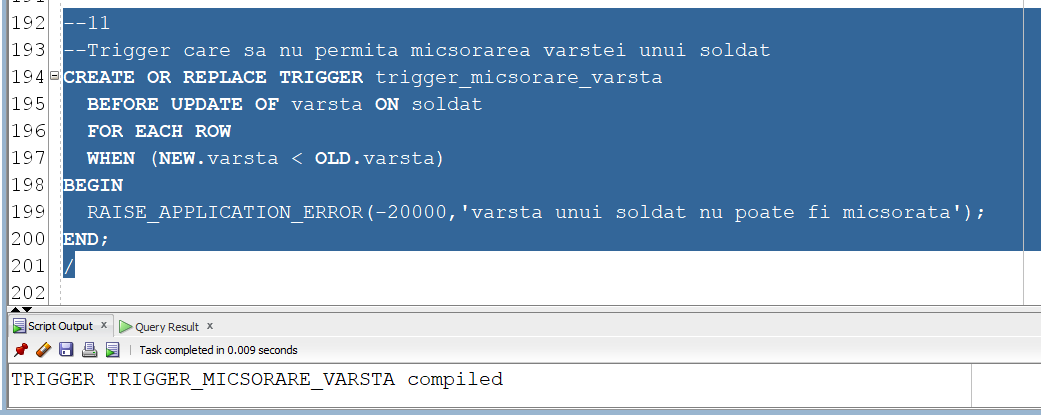
WHEN (NEW.varsta < OLD.varsta)

BEGIN

RAISE\_APPLICATION\_ERROR(-20000,'varsta unui soldat nu poate fi micsorata');

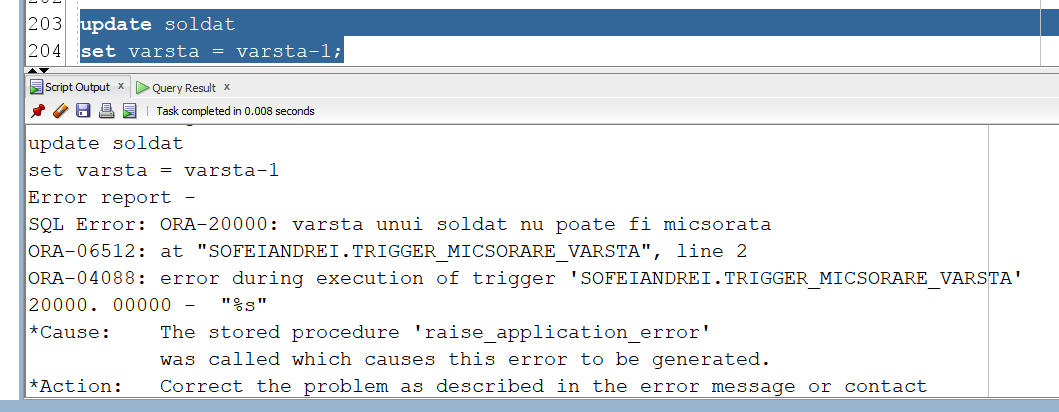
END;

/



update soldat

set varsta = varsta-1;



DROP TRIGGER trigger\_micsorare\_varsta;

**--12**

--Trigger care afiseaza date despre instructiunile apelate

CREATE OR REPLACE TRIGGER trigger\_date\_utilizator

AFTER CREATE OR DROP OR ALTER ON SCHEMA

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Utilizator: ' || SYS.LOGIN\_USER);

DBMS\_OUTPUT.PUT\_LINE('Comanda Folosita: ' || SYS.SYSEVENT);

DBMS\_OUTPUT.PUT\_LINE('Numele Tabelului Afectat: ' || SYS.DICTIONARY\_OBJ\_NAME);

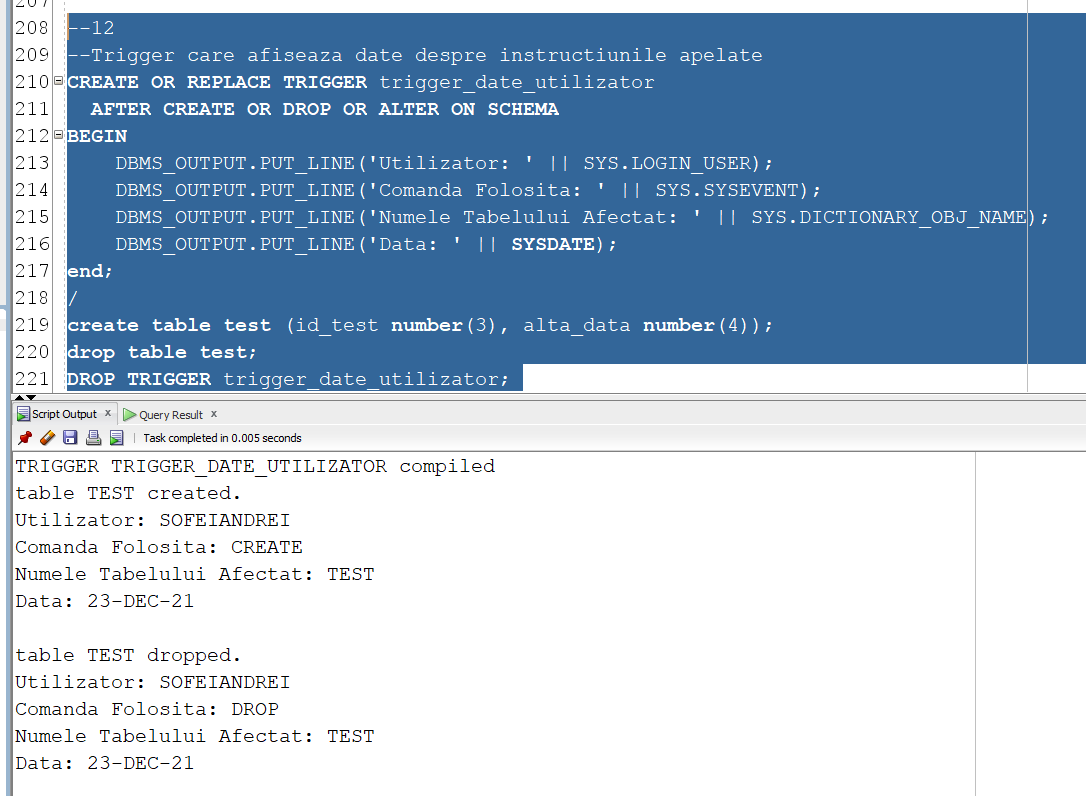
DBMS\_OUTPUT.PUT\_LINE('Data: ' || SYSDATE);

end;

/

create table test (id\_test number(3), alta\_data number(4));

drop table test;



DROP TRIGGER trigger\_date\_utilizator;

13.

CREATE OR REPLACE PACKAGE pachet\_simplu AS

FUNCTION nr\_ofiteri\_plutonul\_x\_pct(pluton pluton.cod\_pluton%type)

RETURN NUMBER;

PROCEDURE detalii\_plutoane\_pct;

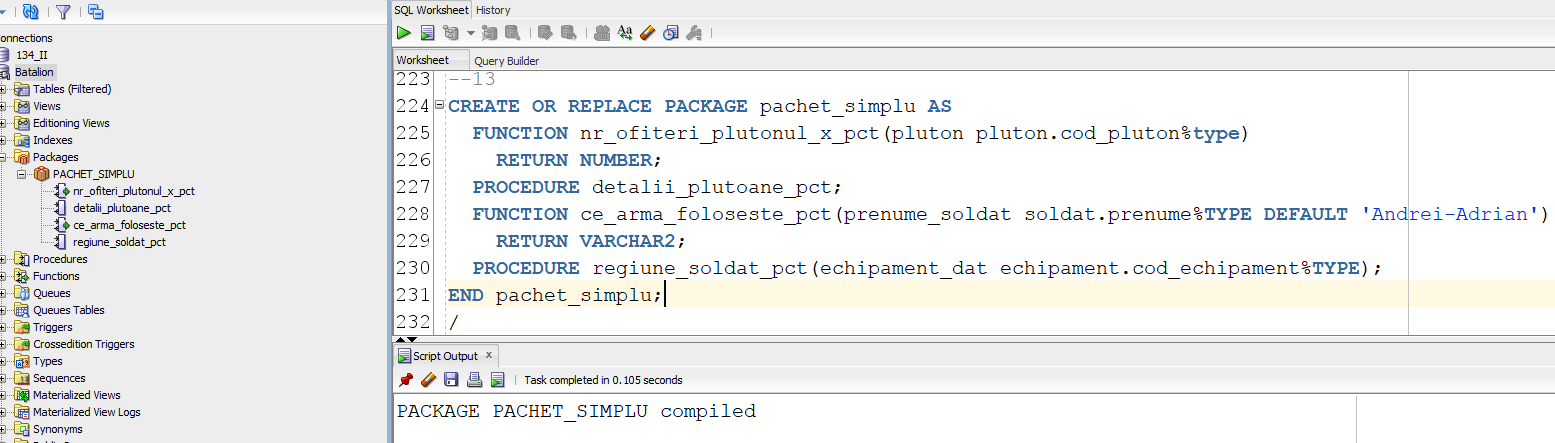
FUNCTION ce\_arma\_foloseste\_pct(prenume\_soldat soldat.prenume%TYPE DEFAULT 'Andrei-Adrian')

RETURN VARCHAR2;

PROCEDURE regiune\_soldat\_pct(echipament\_dat echipament.cod\_echipament%TYPE);

END pachet\_simplu;

/

CREATE OR REPLACE PACKAGE BODY pachet\_simplu AS

FUNCTION nr\_ofiteri\_plutonul\_x\_pct(pluton pluton.cod\_pluton%type)

RETURN NUMBER

IS

nr\_ofiteri\_din\_pluton number(3) :=0;

nr\_soldati\_din\_pluton number(3) :=0;

--tablou indexat

type lista\_soldati is table of soldat%rowtype index by pls\_integer;

soldati\_plutonul\_x lista\_soldati;

--tablou imbricat

type lista\_ofiteri is table of soldat%rowtype;

ofiteri lista\_ofiteri := lista\_ofiteri();

BEGIN

select count(\*) into nr\_soldati\_din\_pluton

from soldat

where cod\_pluton = pluton;

select \* bulk collect into soldati\_plutonul\_x

from soldat

where cod\_pluton = pluton;

select \* bulk collect into ofiteri

from soldat

where cod\_rang = 6 or cod\_rang = 7 or cod\_rang = 8;

for i in soldati\_plutonul\_x.FIRST..soldati\_plutonul\_x.LAST loop

for j in ofiteri.FIRST..ofiteri.LAST loop

if soldati\_plutonul\_x(i).cod\_soldat = ofiteri(j).cod\_soldat then

nr\_ofiteri\_din\_pluton := nr\_ofiteri\_din\_pluton + 1;

end if;

end loop;

end loop;

return nr\_ofiteri\_din\_pluton;

EXCEPTION

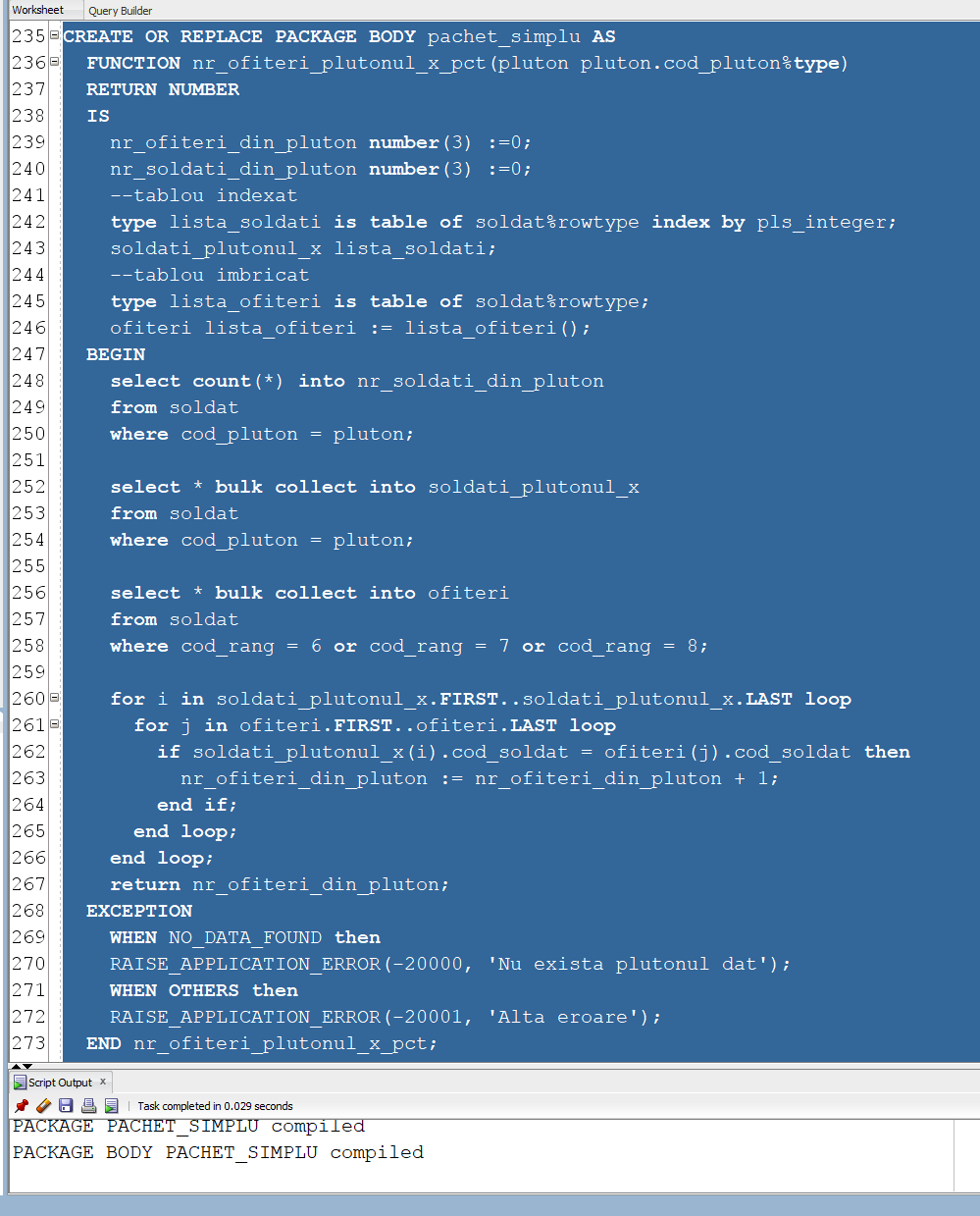
WHEN NO\_DATA\_FOUND then

RAISE\_APPLICATION\_ERROR(-20000, 'Nu exista plutonul dat');

WHEN OTHERS then

RAISE\_APPLICATION\_ERROR(-20001, 'Alta eroare');

END nr\_ofiteri\_plutonul\_x\_pct;



PROCEDURE detalii\_plutoane\_pct

IS

plutonul pluton.cod\_pluton%TYPE;

nr\_soldati\_pluton number(5);

varsta\_medie float(10);

CURSOR cursor\_detalii\_plutoane is

select p.cod\_pluton codul\_plutonului, count(s.cod\_soldat) nr\_soldati, avg(s.varsta) medie\_varsta

from pluton p, soldat s

where p.cod\_pluton = s.cod\_pluton(+)

group by p.cod\_pluton;

BEGIN

OPEN cursor\_detalii\_plutoane;

loop

fetch cursor\_detalii\_plutoane into plutonul,nr\_soldati\_pluton,varsta\_medie;

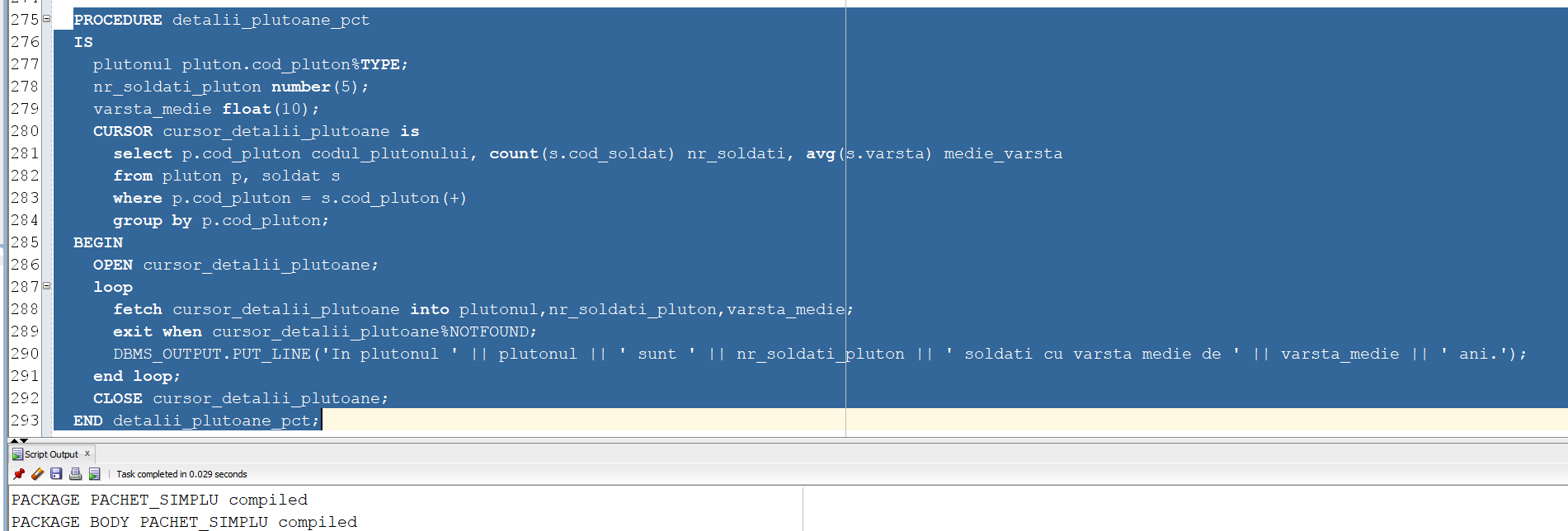
exit when cursor\_detalii\_plutoane%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('In plutonul ' || plutonul || ' sunt ' || nr\_soldati\_pluton || ' soldati cu varsta medie de ' || varsta\_medie || ' ani.');

end loop;

CLOSE cursor\_detalii\_plutoane;

END detalii\_plutoane\_pct;



FUNCTION ce\_arma\_foloseste\_pct(prenume\_soldat soldat.prenume%TYPE DEFAULT 'Andrei-Adrian')

RETURN VARCHAR2

IS

cod\_soldat\_dat soldat.cod\_soldat%TYPE;

nume\_arma arma.nume\_model%TYPE;

BEGIN

select s.cod\_soldat into cod\_soldat\_dat

from soldat s

where s.prenume = prenume\_soldat;

select a.nume\_model into nume\_arma

from soldat s, echipament e, arma a

where s.prenume = prenume\_soldat and

s.cod\_echipament = e.cod\_echipament and

e.cod\_arma = a.cod\_arma;

return nume\_arma;

EXCEPTION

WHEN NO\_DATA\_FOUND then

RAISE\_APPLICATION\_ERROR(-20000, 'Nu exista niciun soldat cu prenumele dat');

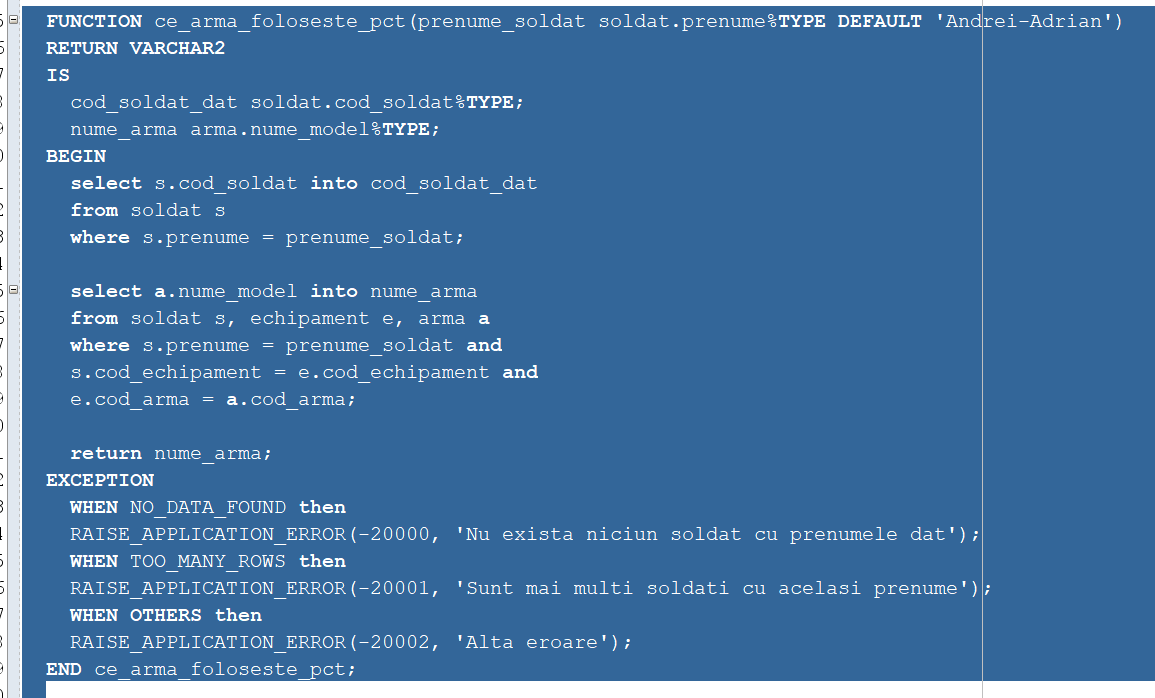
WHEN TOO\_MANY\_ROWS then

RAISE\_APPLICATION\_ERROR(-20001, 'Sunt mai multi soldati cu acelasi prenume');

WHEN OTHERS then

RAISE\_APPLICATION\_ERROR(-20002, 'Alta eroare');

END ce\_arma\_foloseste\_pct;



PROCEDURE regiune\_soldat\_pct(echipament\_dat echipament.cod\_echipament%TYPE)

IS

cod\_soldat\_dat soldat.cod\_soldat%TYPE;

regiunea regiune.nume\_regiune%TYPE;

BEGIN

select s.cod\_soldat into cod\_soldat\_dat

from soldat s

where s.cod\_echipament = echipament\_dat;

select r.nume\_regiune into regiunea

from soldat s, pluton p, oras o, tara t, regiune r

where s.cod\_echipament = echipament\_dat and

s.cod\_pluton = p.cod\_pluton and

p.cod\_oras = o.cod\_oras and

o.cod\_tara = t.cod\_tara and

t.cod\_regiune = r.cod\_regiune;

DBMS\_OUTPUT.PUT\_LINE(regiunea);

EXCEPTION

WHEN NO\_DATA\_FOUND then

RAISE\_APPLICATION\_ERROR(-20000, 'Nu exista niciu soldat care foloseste echipamentul cu codul dat');

WHEN TOO\_MANY\_ROWS then

RAISE\_APPLICATION\_ERROR(-20001, 'Sunt mai multi soldati care folosesc acelasi echipament');

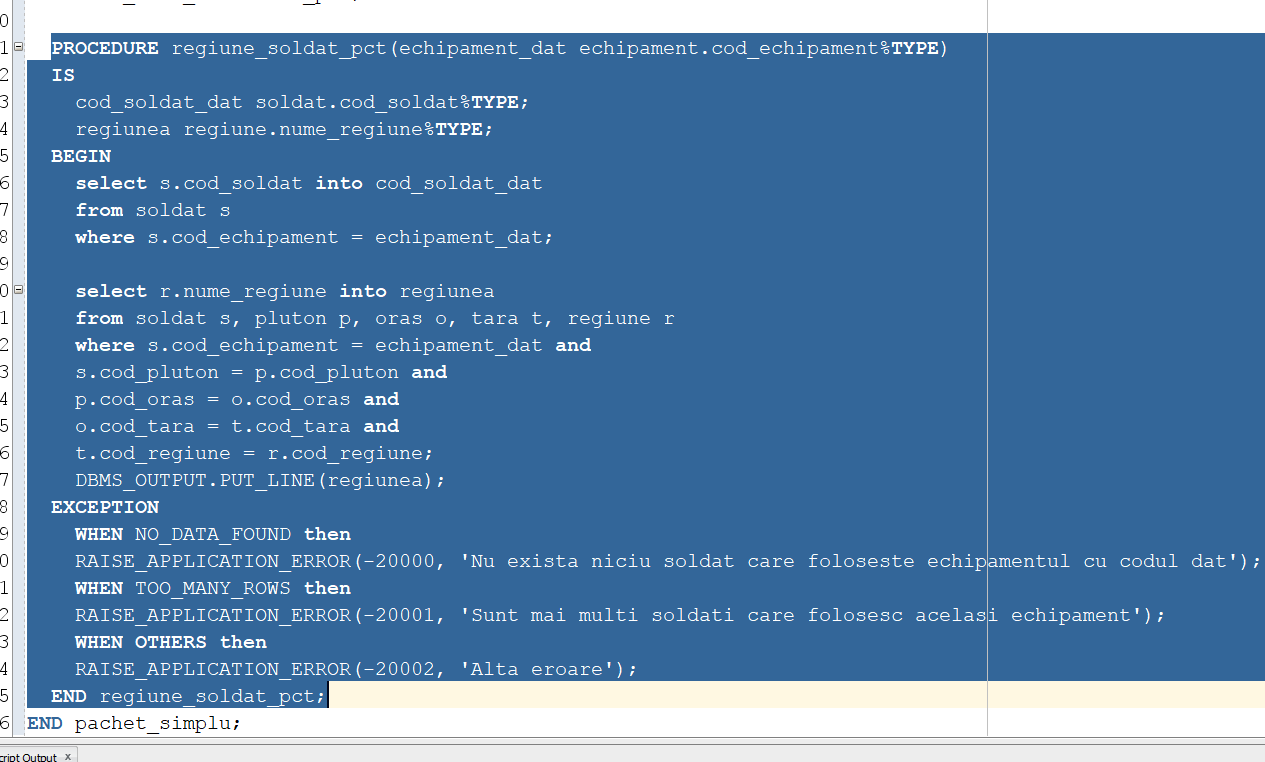
WHEN OTHERS then

RAISE\_APPLICATION\_ERROR(-20002, 'Alta eroare');

END regiune\_soldat\_pct;

END pachet\_simplu;

/

BEGIN

DBMS\_OUTPUT.PUT\_LINE(pachet\_simplu.nr\_ofiteri\_plutonul\_x\_pct(1));

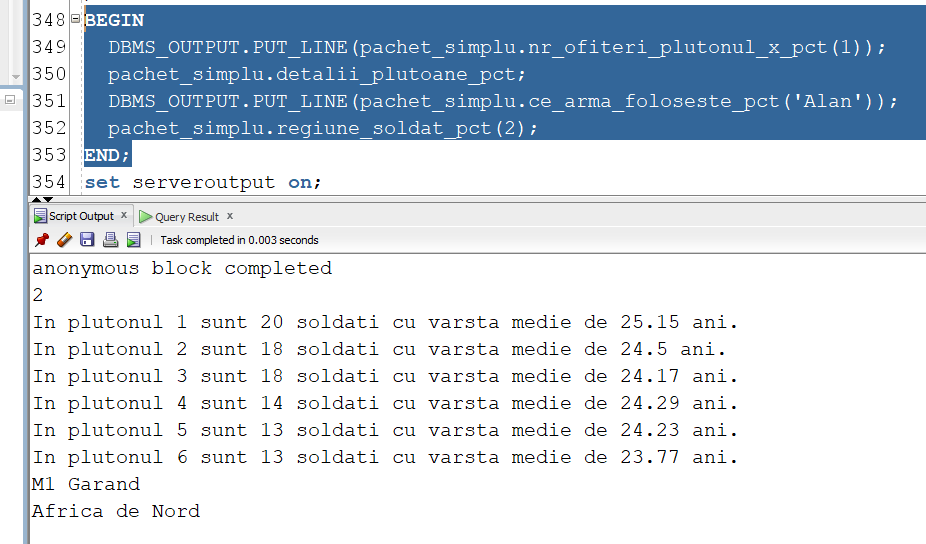
pachet\_simplu.detalii\_plutoane\_pct;

DBMS\_OUTPUT.PUT\_LINE(pachet\_simplu.ce\_arma\_foloseste\_pct('Alan'));

pachet\_simplu.regiune\_soldat\_pct(2);

END;

/



**--14**

--Tip Date Tablou Imbricat in care tin minte indicii plutoanelor

--Tip Date Record in care tinem minte informatii despre perechea rezultatelor la functie

--Tip Date Record in care tinem minte informatii despre toti soldatii din plutonul respectiv(cod\_soldat, nume, prenume, cod\_superior)

--Tip Date Tablou Imbricat In care pentru fiecare pluton tinem minte detalii despre toti soldatii

--Tip Date Tablou Imbricat in care tinem minte rezultatele pentru fiecare apel al functiei Aflare Cel Mai Mare Superior ca record pereche(soldat, superior maximi)

--Functie Aflare Pluton

--Functie Aflare numar de medalii

--Functie Aflare Superior direct (SE TOT APELEAZA in functia complexa PANA CAND SE AJUNGE LA cineva care nu are superior direct)

--Procedura inserare coduri plutoane in vector;

--Procedura inserare date in tabelul detalii\_plutoane care isi creeaza un tabel local de tipul "detalii\_soldati"

--Functie Complexa Aflare superior maxim al unui soldat dat dupa Id care sa fie in acelasi pluton

--cu el si sa se afiseze cate medalii a castigat si insereaza intr-un tabel rezultatele

CREATE OR REPLACE PACKAGE pachet\_complex AS

TYPE pereche\_rezultat is record

(cod\_soldat soldat.cod\_soldat%type,

nume\_soldat soldat.nume%type,

prenume\_soldat soldat.prenume%type,

cod\_superior soldat.cod\_soldat%type,

nume\_superior soldat.nume%type,

prenume\_superior soldat.prenume%type);

TYPE rezultate\_cmms IS TABLE OF pereche\_rezultat;

rezultate rezultate\_cmms:= rezultate\_cmms();

TYPE date\_importante\_soldat is record

(cod\_soldat soldat.cod\_soldat%type,

nume\_soldat soldat.nume%type,

prenume\_soldat soldat.prenume%type,

cod\_superior soldat.cod\_soldat%type,

numar\_medalii NUMBER);

TYPE detalii\_soldati IS TABLE OF date\_importante\_soldat INDEX BY PLS\_INTEGER;

TYPE detalii\_pluton IS record

(cod\_pluton pluton.cod\_pluton%type,

soldati detalii\_soldati);

TYPE tabel\_detalii\_plutoane IS TABLE OF detalii\_pluton INDEX BY PLS\_INTEGER;

detalii\_plutoane tabel\_detalii\_plutoane;

TYPE cod\_plut IS TABLE OF NUMBER;

coduri\_plutoane cod\_plut :=cod\_plut();

PROCEDURE inserare\_coduri\_plut;

PROCEDURE inserare\_det\_plutoane;

FUNCTION aflare\_pluton(id\_soldat soldat.cod\_soldat%type) RETURN NUMBER;

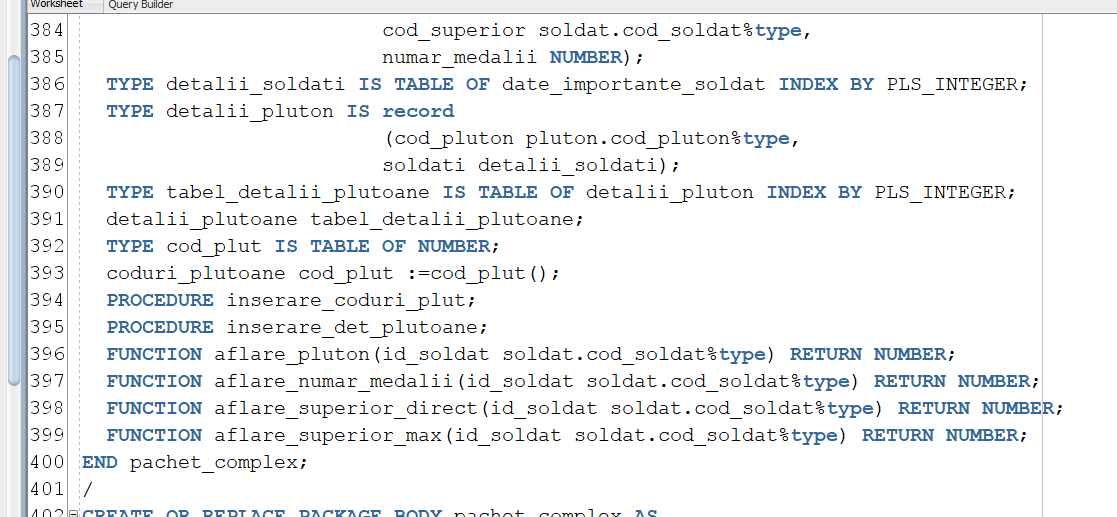
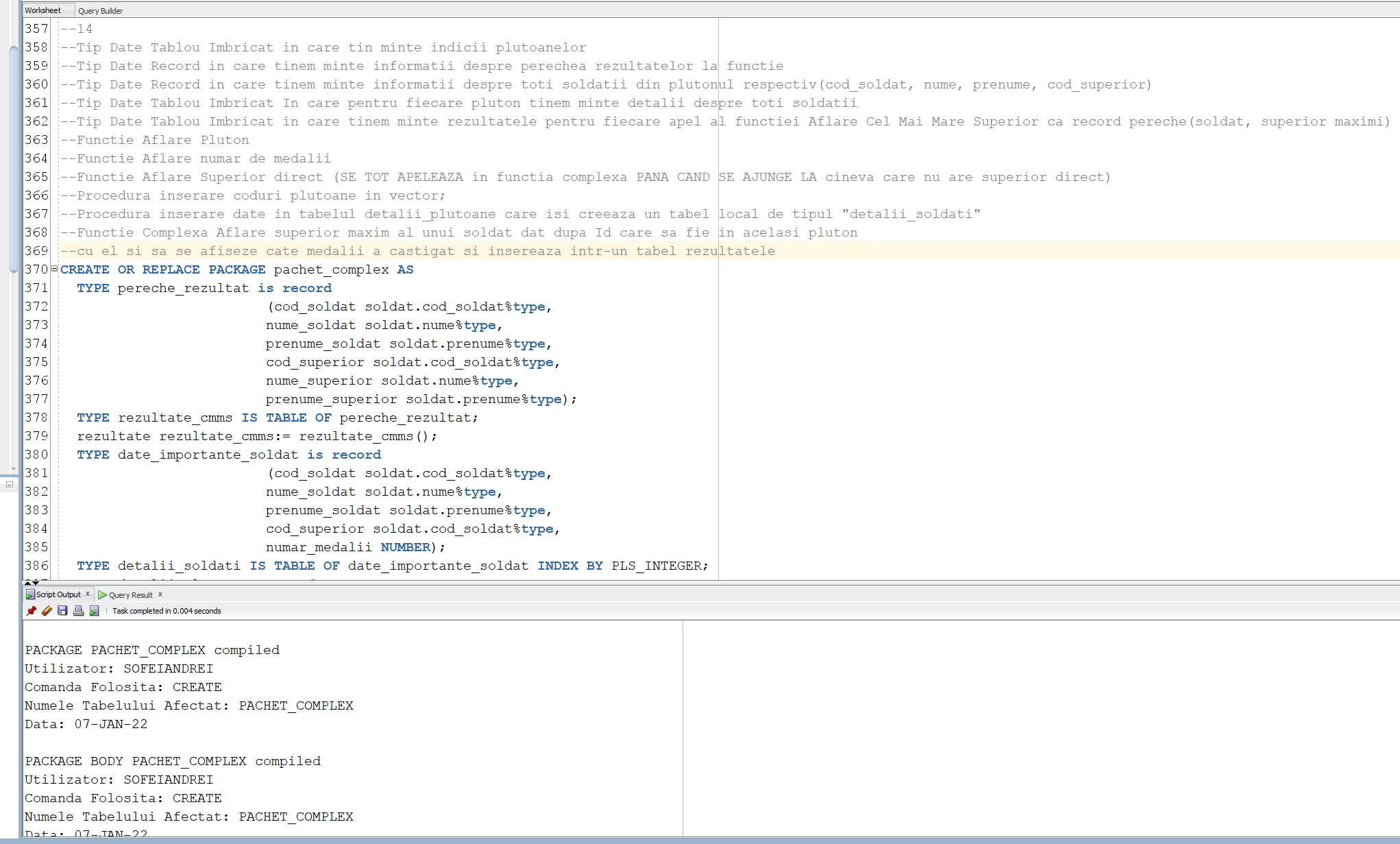
FUNCTION aflare\_numar\_medalii(id\_soldat soldat.cod\_soldat%type) RETURN NUMBER;

FUNCTION aflare\_superior\_direct(id\_soldat soldat.cod\_soldat%type) RETURN NUMBER;

FUNCTION aflare\_superior\_max(id\_soldat soldat.cod\_soldat%type) RETURN NUMBER;

END pachet\_complex;

/



CREATE OR REPLACE PACKAGE BODY pachet\_complex AS

PROCEDURE inserare\_coduri\_plut

IS

BEGIN

select cod\_pluton

bulk collect into coduri\_plutoane

from pluton;

END inserare\_coduri\_plut;

PROCEDURE inserare\_det\_plutoane

IS

det\_sold detalii\_soldati;

dat\_imp\_sold date\_importante\_soldat;

nr\_sold\_din\_plut number(3);

BEGIN

for i in coduri\_plutoane.first..coduri\_plutoane.last loop

select s.cod\_soldat, s.nume, s.prenume, s.cod\_superior\_direct, aflare\_numar\_medalii(s.cod\_soldat)

bulk collect into det\_sold

from soldat s

where cod\_pluton = coduri\_plutoane(i);

detalii\_plutoane(i).cod\_pluton := coduri\_plutoane(i);

detalii\_plutoane(i).soldati := det\_sold;

end loop;

END inserare\_det\_plutoane;

FUNCTION aflare\_pluton(id\_soldat soldat.cod\_soldat%type)

RETURN NUMBER

IS

rezultat NUMBER(2);

BEGIN

select cod\_pluton

into rezultat

from soldat s

where id\_soldat = s.cod\_soldat;

return rezultat;

EXCEPTION

WHEN NO\_DATA\_FOUND then

RAISE\_APPLICATION\_ERROR(-20000, 'Nu exista soldatul dat');

WHEN OTHERS then

RAISE\_APPLICATION\_ERROR(-20001, 'Alta eroare');

END aflare\_pluton;

FUNCTION aflare\_numar\_medalii(id\_soldat soldat.cod\_soldat%type)

RETURN NUMBER

IS

rezultat NUMBER(2);

BEGIN

select count(\*)

into rezultat

from a\_fost\_decorat\_cu

where cod\_soldat=id\_soldat;

return rezultat;

END aflare\_numar\_medalii;

FUNCTION aflare\_superior\_direct(id\_soldat soldat.cod\_soldat%type)

RETURN NUMBER

IS

rezultat NUMBER(3);

BEGIN

select cod\_superior\_direct

into rezultat

from soldat

where cod\_soldat=id\_soldat;

return rezultat;

END aflare\_superior\_direct;

FUNCTION aflare\_superior\_max(id\_soldat soldat.cod\_soldat%type)

RETURN NUMBER

IS

de\_inserat pereche\_rezultat;

soldat\_curent NUMBER(3);

pluton\_curent NUMBER(2);

superior NUMBER(3);

ok NUMBER(1);

numar\_medalii NUMBER(2);

BEGIN

select cod\_soldat, nume, prenume

into de\_inserat.cod\_soldat, de\_inserat.nume\_soldat, de\_inserat.prenume\_soldat

from soldat

where cod\_soldat = id\_soldat;

de\_inserat.cod\_superior:=de\_inserat.cod\_soldat;

de\_inserat.nume\_superior:=de\_inserat.nume\_soldat;

de\_inserat.prenume\_superior:=de\_inserat.prenume\_soldat;

ok:=1;

soldat\_curent := id\_soldat;

pluton\_curent := aflare\_pluton(id\_soldat);

for i in detalii\_plutoane.first..detalii\_plutoane.last loop

if detalii\_plutoane(i).cod\_pluton = pluton\_curent then

while ok=1 loop

ok:=0;

superior:=aflare\_superior\_direct(soldat\_curent);

for j in detalii\_plutoane(i).soldati.first..detalii\_plutoane(i).soldati.last loop

if detalii\_plutoane(i).soldati(j).cod\_soldat = superior then

ok:=1;

soldat\_curent := superior;

de\_inserat.cod\_superior:=detalii\_plutoane(i).soldati(j).cod\_soldat;

de\_inserat.nume\_superior:=detalii\_plutoane(i).soldati(j).nume\_soldat;

de\_inserat.prenume\_superior:=detalii\_plutoane(i).soldati(j).prenume\_soldat;

numar\_medalii:=detalii\_plutoane(i).soldati(j).numar\_medalii;

end if;

end loop;

end loop;

end if;

end loop;

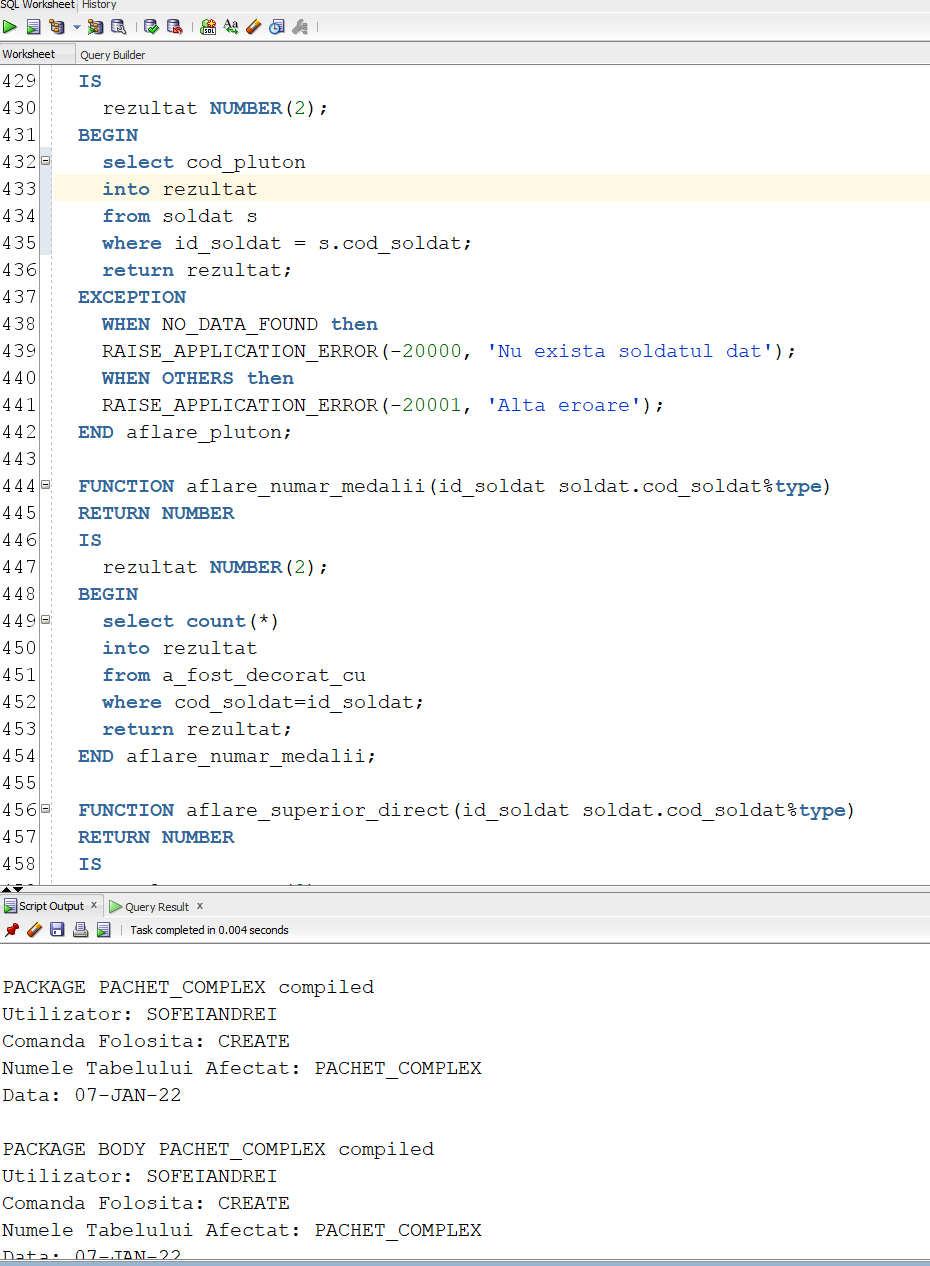
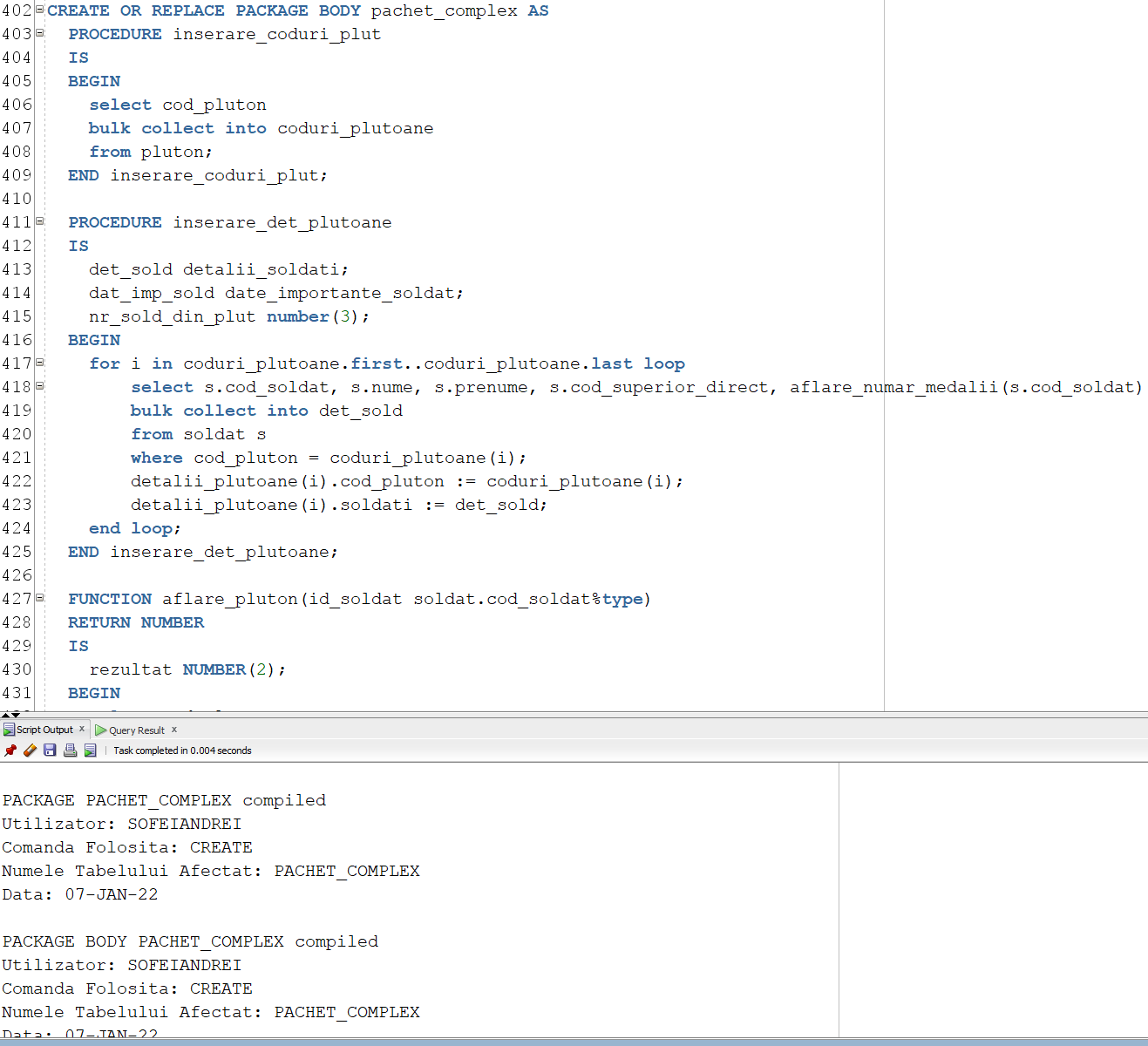
DBMS\_OUTPUT.PUT\_LINE(numar\_medalii);

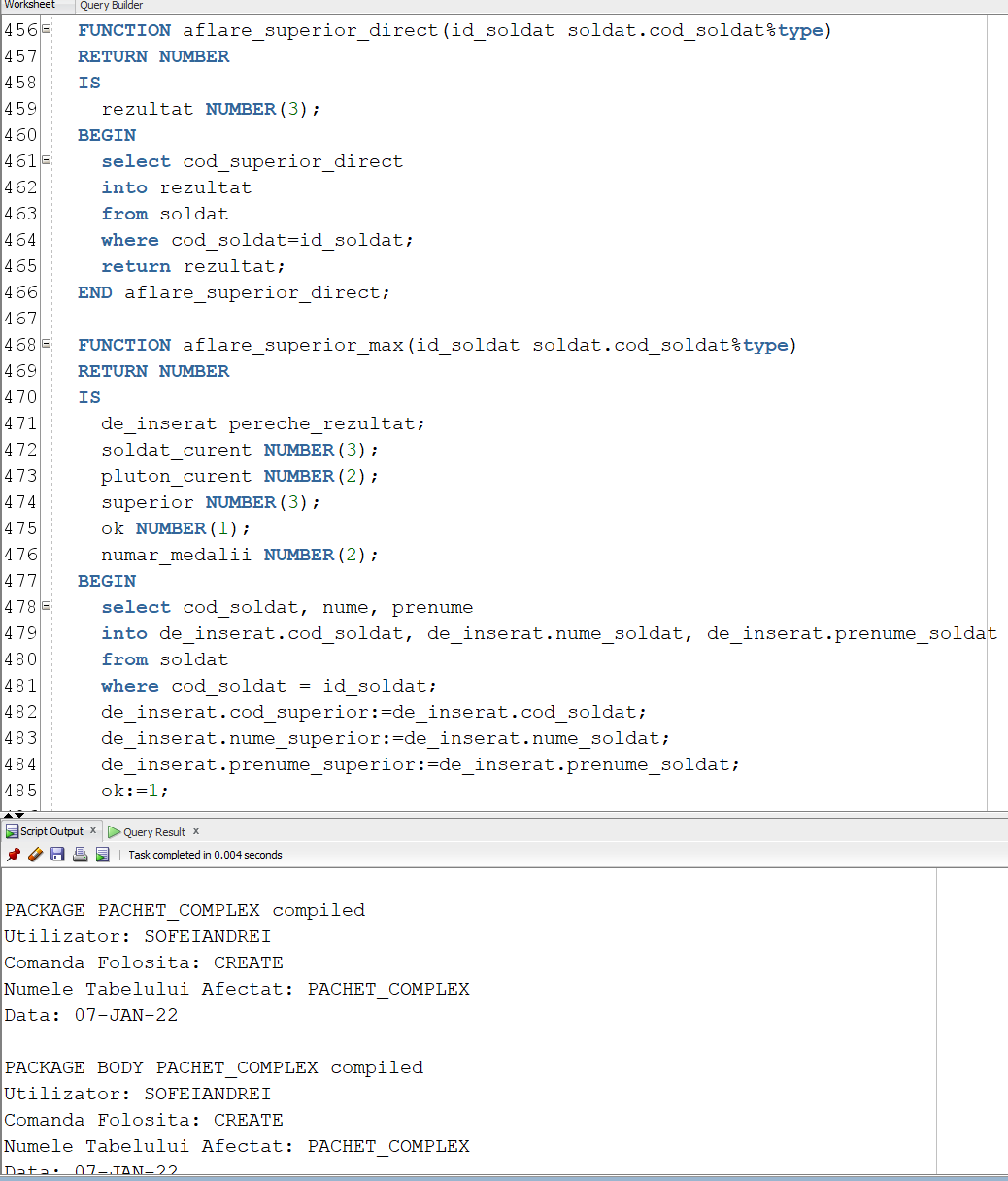
return soldat\_curent;

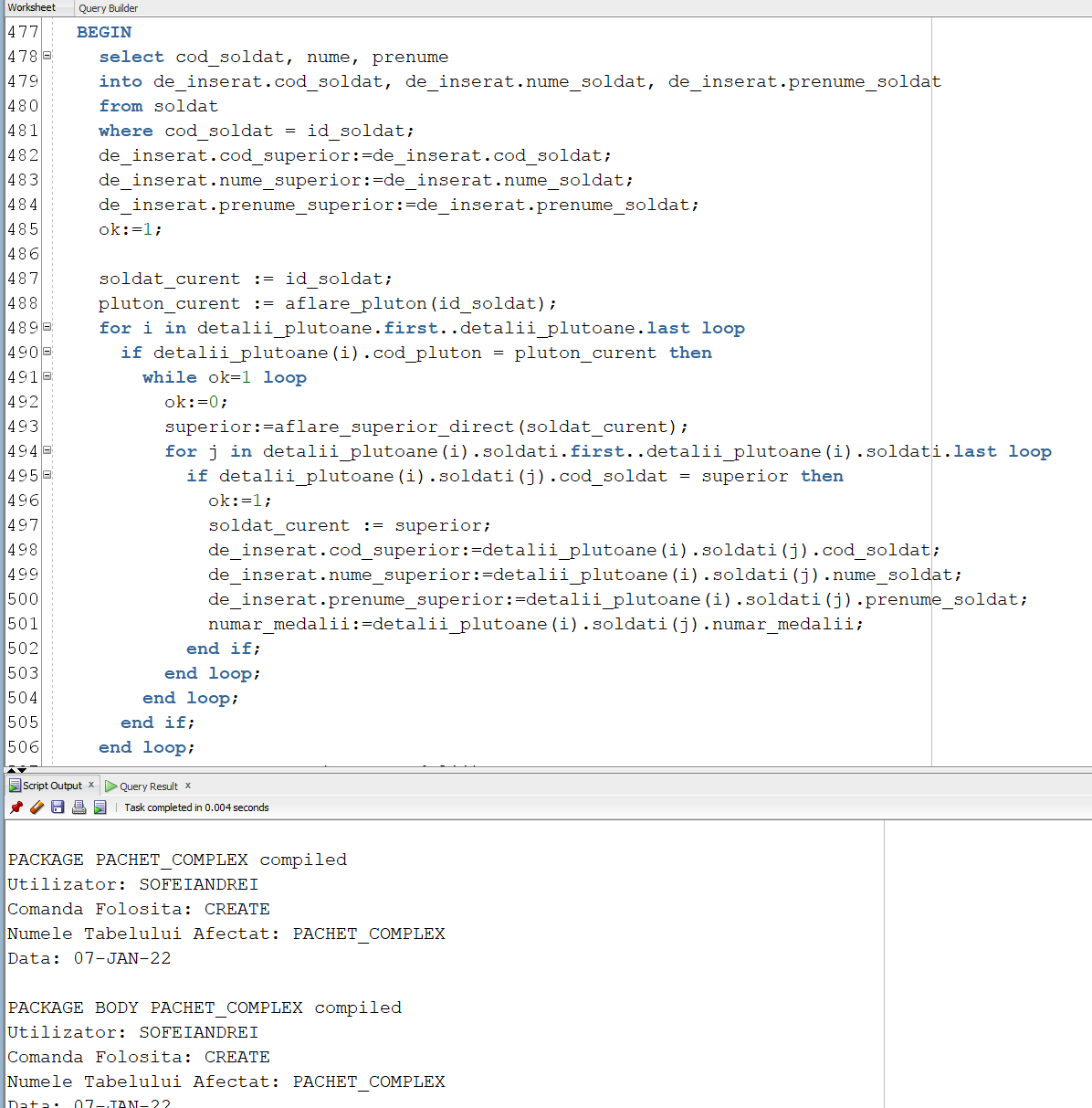
END aflare\_superior\_max;

END pachet\_complex;

/







BEGIN

DBMS\_OUTPUT.PUT\_LINE(pachet\_complex.aflare\_pluton(15));

pachet\_complex.inserare\_coduri\_plut;

pachet\_complex.inserare\_det\_plutoane;

DBMS\_OUTPUT.PUT\_LINE(pachet\_complex.coduri\_plutoane(5));

DBMS\_OUTPUT.PUT\_LINE(pachet\_complex.aflare\_numar\_medalii(33));

DBMS\_OUTPUT.PUT\_LINE(pachet\_complex.aflare\_superior\_direct(41));

DBMS\_OUTPUT.PUT\_LINE(pachet\_complex.detalii\_plutoane(1).soldati(4).nume\_soldat);

DBMS\_OUTPUT.PUT\_LINE(pachet\_complex.aflare\_superior\_max(30));

END;

/

