

Proiect – Baze de Date

Model de bază de date a unei platforme online de bilete pentru evenimente

1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare

Tema pe care am ales-o pentru acest proiect este *baza de date a unei platforme ce se ocupă cu distribuirea de bilete pentru diferite spectacole și evenimente*. Modelul de date va centraliza programul pentru mai multe locații care găzduiesc aceste evenimente și va permite atât rezervarea de locuri, cât și cumpărarea de bilete, facilitând procesul de planificare a participării în timpul liber la astfel de activități. Spectacolele din baza de date pot fi atât spectacole de teatru, cât și filme sau concerte. Spectacolele se desfășoară în cadrul unor evenimente, pentru care se cunosc organizatorii și locația exactă, incluzând orașul, clădirea și sala. Fiecare sală dispune de mai multe locuri, asociate unor bilete individuale fiecărui eveniment. Persoanele care vor să folosească platforma trebuie să se înscrie ca utilizatori, după care pot efectua rezervări pentru orice număr de bilete ale oricărui eveniment, pe care le pot și plăti online eventual. Plata ține cont și de reducerile posibile. Fiind digitalizat întregul proces de rezervare, nu este nevoie de angajați care să gestioneze vânzările.

2. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului

Modelul va funcționa după următoarele restricții:

- un spectacol trebuie să aibă loc în cadrul a cel puțin unui eveniment
- în fiecare oraș se află cel puțin o locație, care are cel puțin o sală, care are cel puțin un loc
- fiecare bilet poate fi rezervat și eventual platit de un singur utilizator
- biletele nu pot fi achiziționate fizic de un cumpărător care nu e trecut în baza de date ca utilizator
- dar un utilizator poate exista în baza de date fără a fi plasat încă o rezervare
- organizatorul unui eveniment este unic, se consideră că indiferent dacă este vorba de o echipă care se ocupă de organizare, persoana de contact este unică
- organizatorul poate să se ocupe de mai multe evenimente
- un event se desfășoară în o sală și numai una
- o rezervare poate cuprinde oricâte bilete, dar ele trebuie să fie ale aceleiași eveniment
- lista de spectacole și a evenimentelor aferente lor trebuie să fie complete
- desfășurarea evenimentului la o locație nu e condiționată de tipul locației

3. Descrierea entităților, incluzând precizarea cheii primare

Acest model de date al unei platforme online folosește ca entități structurile SPECTACOL, FILM, TEATRU, CONCERT, ORGANIZATOR, EVENIMENT, SALA, LOCATIE, ORAS, REZERVARE, BILET, LOC, UTILIZATOR, PLATA, REDUCERE.

Dintre acestea, o parte sunt independente(SPECTACOL, ORGANIZATOR, UTILIZATOR, REDUCERE, ORAS, EVENIMENT, LOCATIE, REZERVARE, PLATA), o parte sunt dependente(SALA, BILET, LOC), iar restul sunt subentități(FILM, TEATRU, CONCERT).

SPECTACOL = (cheia primară: id_spectacol) ideea, conceptul unei activități ce urmează să se materializeze în cadrul unui eveniment

FILM = (cheia primară: id_spectacol) subentitate a entității SPECTACOL care se referă la un tip de conținut vizual ce urmează să fie proiectat unei audiențe

TEATRU = (cheia primară: id_spectacol) subentitate a entității SPECTACOL care se referă la un tip de conținut artistic care urmează să fie jucat în fața unei audiențe

CONCERT = (cheia primară: id_spectacol) subentitate a entității SPECTACOL care se referă la un tip de conținut artistic care urmează să fie cântat în fața unei audiențe

ORGANIZATOR = (cheia primară: id_organizator) persoană fizică de contact pentru organizarea evenimentelor

EVENIMENT = (cheia primară: id_eveniment) expunere concretă a unui spectacol la o anumită dată și oră

ORAS = (cheia primară: id_oras) unitate administrativă a țării în care urmează să se desfășoare un eveniment

LOCATIE = (cheia primară: id_locatie) locul precis determinat al unui oraș care are capacitatea de a găzdui un eveniment

SALA = (cheia primară: id_locatie, id_sala) entitate dependentă de LOCATIE, reprezentând un compartiment al unei locații rezervat pentru un eveniment

LOC = (cheia primară: id_locatie, id_sala, id_loc) entitate dependentă de SALA și LOCATIE, reprezentând un spațiu concret al unei săli a unei locații care poate fi asociat unei persoane fizice pentru participarea ei la eveniment

BILET = (cheia primară: id_bilet, id_eveniment) entitate dependentă de EVENIMENT, reprezentând un document pus în vânzare, ce atestă asocierea unei persoane fizice cu un loc pe care îl poate ocupa în cadrul unui eveniment, după cumpărare

UTILIZATOR = (cheia primară: id_utilizator) persoană fizică ce își oferă datele spre a achiziționa cel puțin un bilet la cel puțin un eveniment

REZERVARE = (cheia primară: id_rezervare) set de informații care atestă asocierea unei persoane fizice cu unul sau mai multe bilete spre o viitoare achiziție

PLATA = (cheia primară: id_plata) set de date primit odată cu achiziționarea unui bilet legat de tranzacția efectuată

REDUCERE = (cheia primară: id_reducere) ofertă de scădere a prețului plătit, oferită de platformă sub anumite condiții, pentru care utilizatorii pot opta în cadrul unei plăți

4. Descrierea relațiilor, incluzând precizarea cardinalității acestora

Relațiile din model indică cum interacționează entitățile, arătând atât legătura dintre ele, cât și cardinalitățile minime și maxime.

SPECTACOL_is_a_FILM = relația care leagă entitățile SPECTACOL și FILM, marcând legătura dintre acestea(un spectacol poate fi un film)

-> cardinalitatea minimă este de 1:0(nu este neaparat ca unul dintre spectacole să fie un film)

-> cardinalitatea maximă este de 1:1(un spectacol poate fi cel mult un film)

SPECTACOL_is_a_TEATRU = relația care leagă entitățile SPECTACOL și TEATRU, marcând legătura dintre acestea(un spectacol poate fi un spectacol de teatru)

-> cardinalitatea minimă este de 1:0(nu este neaparat ca unul dintre spectacole să fie un spectacol de teatru)

-> cardinalitatea maximă este de 1:1(un spectacol poate fi cel mult un spectacol de teatru)

SPECTACOL_is_a_CONCERT = relația care leagă entitățile SPECTACOL și CONCERT, marcând legătura dintre acestea(un spectacol poate fi un concert)

-> cardinalitatea minimă este de 1:0(nu este neaparat ca un spectacol să fie un concert)

-> cardinalitatea maximă este de 1:1(un spectacol poate fi cel mult un concert)

EVENIMENT_prezinta_SPECTACOL = relația care leagă entitățile EVENIMENT și SPECTACOL, marcând legătura dintre acestea(un eveniment prezintă un spectacol)

-> cardinalitatea minimă este de 1:1(cel puțin un eveniment trebuie să prezinte un spectacol și cel puțin un spectacol trebuie să fie prezentat în cadrul unui eveniment)

-> cardinalitatea maximă este de M:1(mai multe evenimente pot prezenta un spectacol, dar un eveniment nu poate prezenta mai mult de un spectacol)

ORGANIZATOR_organizeaza_EVENIMENT = relația care leagă entitățile ORGANIZATOR și EVENIMENT, marcând legătura dintre acestea(un organizator organizează un eveniment)

-> cardinalitatea minimă este 1:1(un organizator poate organiza cel puțin un eveniment, iar un eveniment trebuie să fie organizat de cel puțin un organizator)

-> cardinalitatea maximă este 1:M(un organizator poate organiza mai multe evenimente, dar un eveniment nu poate fi organizat de mai mulți organizatori)

EVENTIMENT_are_la_dispozitie_SALA = relația care leagă entitățile EVENTIMENT și SALA, marcând legătura dintre acestea(un eveniment are la dispoziție o sală în care să se desfășoare)

-> cardinalitatea minimă este 0:1(pot exista săli în care să nu se desfășoare evenimente, dar fiecare eveniment trebuie să aibă la dispoziție măcar o sală)

-> cardinalitatea maximă este M:1(mai multe evenimente se pot desfășura în aceeași sală - la momente diferite de timp, dar un eveniment nu poate ocupa mai mult de o sală)

EVENTIMENT_pune_la_dispozitie_BILET = relația care leagă entitățile EVENTIMENT și BILET, marcând legătura dintre acestea(un eveniment pune la dispoziție bilete pentru clienți)

-> cardinalitatea minimă este 1:1 (un eveniment trebuie să puna la dispoziție minim un bilet, iar un bilet trebuie să fie pus la dispoziție de cel puțin un eveniment)

-> cardinalitatea maximă este 1:M (un eveniment poate pune la dispoziție mai multe bilete, dar fiecare bilet este pus la dispoziție de cel mult un eveniment)

REZERVARE_include_BILET = relația care leagă entitățile REZERVARE și BILET, marcând legătura dintre acestea(o rezervare include bilete)

-> cardinalitatea minimă este 0:1(o rezervare trebuie să includă minim un bilet, dar un bilet poate să nu fie inclus în nicio rezervare)

-> cardinalitatea maximă este 1:M(o rezervare poate include mai multe bilete, dar fiecare bilet este inclus în cel mult o rezervare)

PLATA_confirma_REZERVARE = relația care leagă entitățile REZERVARE și PLATA, marcând legătura dintre acestea(o plată confirmă o rezervare)

-> cardinalitatea minimă este 0:1(o rezervare poate să nu fie confirmată de nicio plată)

-> cardinalitatea maximă este 1:1(o rezervare poate fi confirmată de cel mult o plată, iar o plată confirmă cel mult o rezervare)

PLATA_implica_REDUCERE = relația care leagă entitățile PLATA și REDUCERE, marcând legătura dintre acestea(o plată poate implica o reducere)

-> cardinalitatea minimă este 0:0 (se poate ca nicio plată să nu implice nicio reducere)

-> cardinalitatea maximă este M:M(o plată poate implica mai multe reduceri, iar o reducere poate fi implicată în mai multe plăți)

UTILIZATOR_plaseaza_REZERVARE = relația care leagă entitățile ORGANIZATOR și EVENIMENT, marcând legatura dintre acestea(un utilizator poate să plaseze o rezervare)

-> cardinalitatea minimă este de 1:0(un utilizator nu trebuie să plaseze nicio rezervare, dar o rezervare trebuie să fie plasată de un utilizator)

-> cardinalitatea maximă este de 1:M(un utilizator poate plasa mai multe rezervări, dar o rezervare este plasată de cel mult un utilizator)

BILET_corespunde_LOC = relația care leagă entitățile LOC și BILET, marcând legatura dintre acestea(un bilet corespunde unui loc)

-> cardinalitatea minimă este 0:1(un loc poate să nu corespundă niciunui bilet, dar un bilet trebuie să corespundă unui loc)

-> cardinalitatea maximă este 1:M(un loc poate corespunde mai multor bilete – pentru evenimente diferite, dar un bilet corespunde cel mult unui loc)

SALA_cuprinde_LOC = relația care leagă entitățile SALA și LOC, marcând legatura dintre acestea(o sală trebuie să cuprindă un loc)

-> cardinalitatea minimă este de 1:1(o sala trebuie să conțină măcar un loc, iar un loc trebuie să se afle în cel puțin o sală)

-> cardinalitatea maximă este de 1:M(o sala poate conține mai multe locuri, dar un loc este cuprins în cel mult o sală)

LOCATIA_dispune_de_SALA = relația care leagă entitățile LOCATIE și SALA, marcând legatura dintre acestea(o locatie trebuie să dispună de o sală)

-> cardinalitatea minimă este 1:1(o locatie trebuie să dispună de măcar o sală, iar o sală trebuie să fie la dispoziție de o locație)

-> cardinalitatea maximă este 1:M(o locație poate dispune de mai multe săli, dar o sală trebuie să aparțină cel mult unei locații)

ORAS_contine_LOCATIE = relația care leagă entitățile ORAS și LOCATIE, marcând legatura dintre acestea(un oraș trebuie să conțină o locație)

-> cardinalitatea minimă este 1:1(un oraș trebuie să conțină cel puțin o locație, iar o locație trebuie să corespundă unui oraș)

-> cardinalitatea maximă este 1:M(un oraș poate conține mai multe locații, însă o locație aparține cel mult unui oraș)

5. Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor

Atributele entității independente SPECTACOL sunt:

id_spectacol = cheie primară, variabilă de tip întreg, reprezintă numărul de identificare al unui spectacol

tip = șir de caractere de lungime maximă 20, reprezintă tipul spectacolului(film, teatru sau concert), valoarea nu poate fi nulă

nume = șir de caractere de lungime maximă 100, reprezintă numele spectacolului, valoarea nu poate fi nulă

Atributele subentității FILM sunt:

id_spectacol = cheie primară, dar și externă, variabilă de tip întreg, reprezintă numărul de identificare al filmului și corespunde unei valori a cheii primare din tabelul SPECTACOL

durata = variabilă de tip întreg, arată, în minute, cât durează filmul

limba = șir de caractere de lungime maximă 20, reprezintă limba în care este dublat/realizat filmul, valoarea implicită este nulă

data_lansarii = variabilă de tip „date”, reprezentând date calendaristice valide, mai exact data lansării filmului, valoarea implicită este nulă

tara = șir de caractere de lungime maximă 20, reprezintă țara în care a fost realizat filmul, valoarea implicită este nulă

gen = șir de caractere de lungime maximă 20, reprezintă tipul filmului, valoarea implicită este nulă

Atributele subentității TEATRU sunt:

id_spectacol = cheie primară, dar și externă, variabilă de tip întreg, reprezintă numărul de identificare a piesei de teatru și corespunde unei valori a cheii primare din tabelul SPECTACOL

autor = șir de caractere de lungime maximă 50, reprezentând numele autorului piesei de teatru, valoarea implicită este nulă

regizor = șir de caractere de lungime maximă 50, reprezentând numele regizorului piesei de teatru, valoarea implicită este nulă

scenograf = șir de caractere de lungime maximă 50, reprezentând numele scenografului piesei de teatru, valoarea implicită este nulă

Atributele subentității CONCERT sunt:

id_spectacol = cheie primară, dar și externă, variabilă de tip întreg, reprezintă numărul de identificare a concertului și corespunde unei valori a cheii primare din tabelul SPECTACOL

artist = șir de caractere de lungime maximă 50, reprezentând numele artistului ce cântă la concert, valoarea implicită este nulă

invitati = șir de caractere de lungime maximă 100, reprezentând numele invitaților la concert, valoarea implicită este nulă

gen = șir de caractere de lungime maximă 50, reprezentând genul de muzică cântat la concert, valoarea implicită este nulă

Atributele entității independente ORGANIZATOR sunt:

id_organizator = cheie primară, variabilă de tip întreg, reprezintă numărul de identificare al organizatorului

nume = șir de caractere de lungime maximă 50, reprezentând numele organizatorului evenimentului, valoarea nu poate fi nulă

email = șir de caractere de lungime maximă 50, reprezentând adresa de e-mail a organizatorului evenimentului, valoarea nu poate fi nulă

nr_telefon = șir de caractere de lungime maximă 20, reprezentând numărul de telefon al organizatorului evenimentului, valoarea nu poate fi nulă

Atributele entității independente ORAȘ:

id_oras = cheie primară, variabilă de tip întreg, reprezintă numărul de identificare al orașului

nume = șir de caractere de lungime maximă 50, reprezentând numele orașului, valoarea nu poate fi nulă

localitate = șir de caractere de lungime maximă 50, reprezentând localitatea din care face parte orașul, valoarea nu poate fi nulă

Atributele entității independente LOCAȚIE:

id_locatie = cheie primară, variabilă de tip întreg, reprezintă numărul de identificare al locației

id_oras = variabilă de tip întreg, reprezintă numărul de identificare al orașului în care se află locația, valoarea nu poate fi nulă

nume = șir de caractere de lungime maximă 50, reprezentând numele locației, valoarea nu poate fi nulă

nr_sali = variabilă de tip întreg, reprezintă numărul de săli din locație care pot fi puse la dispoziția evenimentelor, valoarea nu poate fi nulă

adresa = șir de caractere de lungime maximă 50, reprezentând adresa la care se află locația, valoarea nu poate fi nulă

Atributele entității dependențe SALA:

id_sala = atribut care face parte din cheia primară compusă, variabilă de tip întreg, reprezintă numărul de identificare al sălii, valoarea nu poate fi nulă

id_locatie = cheie externă și atribut care face parte din cheia primară compusă, variabilă de tip întreg, reprezintă numărul de identificare al locației, valoarea nu poate fi nulă

nume = șir de caractere de lungime maximă 50, reprezentând numele sălii, valoarea nu poate fi nulă

nr_locuri = variabilă de tip întreg, reprezentând numărul de locuri din sala, valoare implicită e nulă

Atribute entității independente EVENIMENT:

id_eveniment = cheie primară, variabilă de tip întreg, reprezintă numărul de identificare al evenimentului

id_spectacol = cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al spectacolului care are loc în cadrul evenimentului, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul SPECTACOL

id_locatie = cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al locației unde se desfășoară evenimentul, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul SALA

id_sala = cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al sălii unde se desfășoară evenimentul, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul SALA

id_organizator = cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al organizatorului, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul ORGANIZATOR

data_eveniment = variabilă de tip „date”, reprezentând date calendaristice valide, mai exact dată evenimentului, valoarea nu poate fi nulă

ora_start = șir de caractere de lungime maximă 10, reprezentând ora de începere a evenimentului, valoarea nu poate fi nulă

ora_stop = șir de caractere de lungime maximă 10, reprezentând ora de terminare a evenimentului, valoarea nu poate fi nulă

Atributele entității dependente LOC:

id_loc = atribut care face parte din cheia primară compusă, variabilă de tip întreg, reprezintă numărul de identificare al locului, valoarea nu poate fi nulă

id_sala = cheie externă și atribut care face parte din cheia primară compusă, variabilă de tip întreg, reprezintă numărul de identificare al sălii, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul SALA

id_locatie = cheie externă și atribut care face parte din cheia primară compusă, variabilă de tip întreg, reprezintă numărul de identificare al locului, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul SALA

tip = șir de caractere de lungime maximă 50, reprezentând tipul locului, valoarea nu poate fi nulă

Atributele entității independente UTILIZATOR:

id_utilizator = cheie primară, variabilă de tip întreg, reprezintă numărul de identificare al utilizatorului

nume = șir de caractere de lungime maximă 50, reprezentând numele utilizatorului, valoarea nu poate fi nulă

parolă = șir de caractere de lungime maximă 50, reprezentând parola utilizatorului, valoarea nu poate fi nulă

email = șir de caractere de lungime maximă 50, reprezentând adresa de e-mail a utilizatorului, pe care o va folosi pentru autentificarea pe site, valoarea nu poate fi nulă

nr_telefon = șir de caractere de lungime maximă 20, reprezentând numărul de telefon al utilizatorului, valoarea nu poate fi nulă

Atributele entității independente REZERVARE:

id_rezervare = cheie primară, variabilă de tip întreg, reprezentând numărul de identificare a rezervarii

id_utilizator = cheie externă, variabilă de tip întreg, reprezentând numărul de identificare al utilizatorului, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul UTILIZATOR

data_rezervare = variabilă de tip „date”, reprezentând date calendaristice valide, mai exact dată rezervarii, valoarea nu poate fi nulă

nr_bilete = variabilă de tip întreg, reprezintă numărul de bilete al rezervarii, valoarea nu poate fi nulă

stare = variabilă de tip întreg, reprezintă starea rezervarii (0 - neplătită, 1 - plătită), valoarea nu poate fi nulă

Atributele entității dependențe BILET:

id_bilet = cheie primară, variabilă de tip întreg, reprezentând numărul de identificare al biletului

id_eveniment = cheie primară și cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al evenimentului, și corespunde unei valori a cheii primare din tabelul EVENIMENT

id_loc = cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al locului, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul LOC

id_sala = cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al sălii, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul LOC

id_locatie = cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al locației, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul LOC

id_rezervare = cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al rezervării, valoarea implicită e nulă, și corespunde unei valori a cheii primare din tabelul REZERVARE

pret = variabilă de tip întreg, reprezintă prețul biletului, valoarea nu poate fi nulă

stare = variabilă de tip întreg, reprezintă starea biletului (0 - nerezervat, 1 - rezervat), valoarea nu poate fi nulă

Atributele entității independente PLATA:

id_plata = cheie primară, variabilă de tip întreg, reprezintă numărul de identificare al plății

suma = variabilă de tip întreg, reprezintă suma plății, valoarea nu poate fi nulă

data_plata = variabilă de tip „date”, reprezentând date calendaristice valide, mai exact dată efectuării plății, valoarea nu poate fi nulă

id_rezervare = cheie externă, variabilă de tip întreg, reprezintă numărul de identificare al rezervării, valoarea nu poate fi nulă, și corespunde unei valori a cheii primare din tabelul REZERVARE

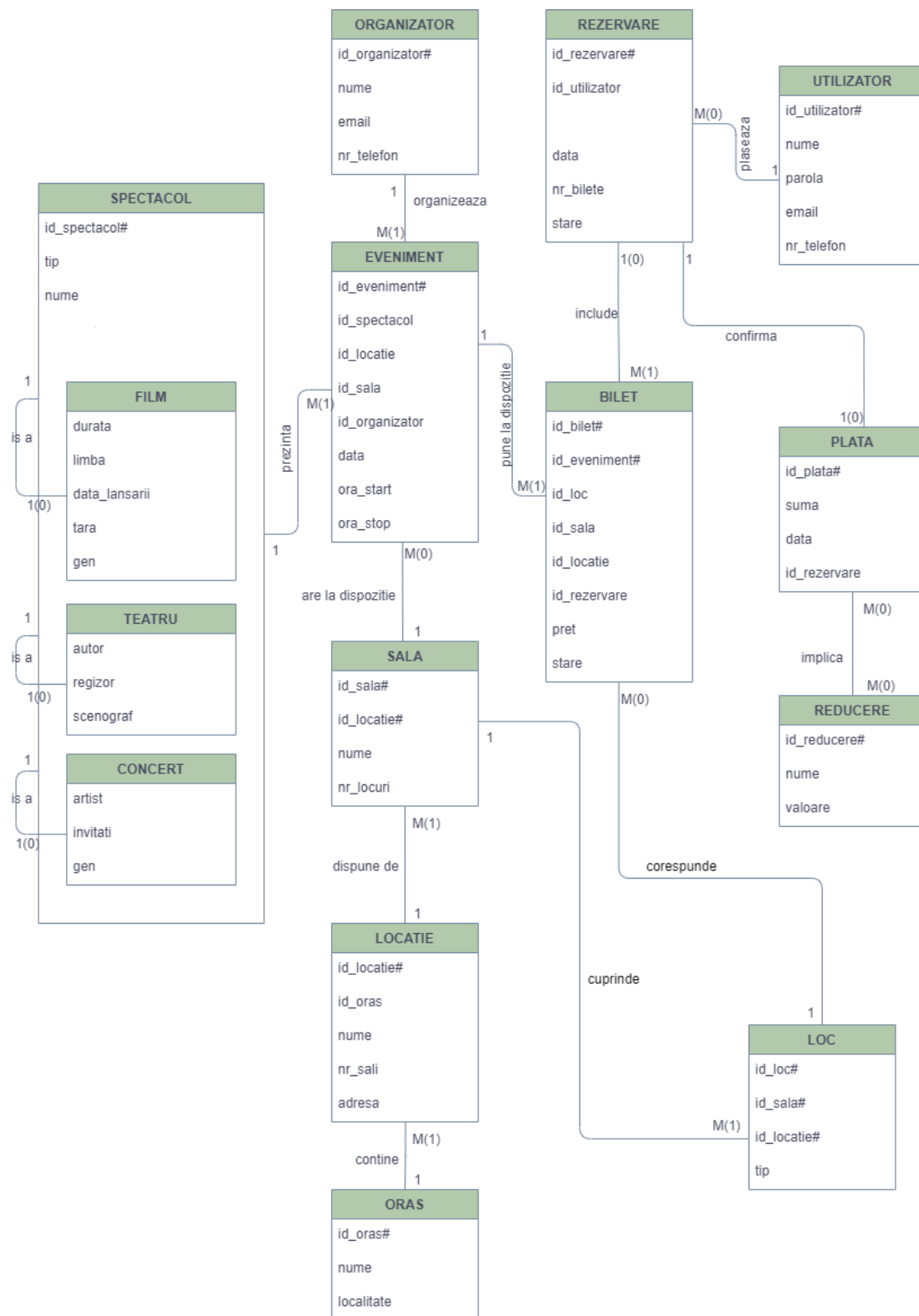
Atributele entității independente REDUCERE:

id_reducere = cheie primară, variabilă de tip întreg, reprezintă numărul de identificare al reducerii

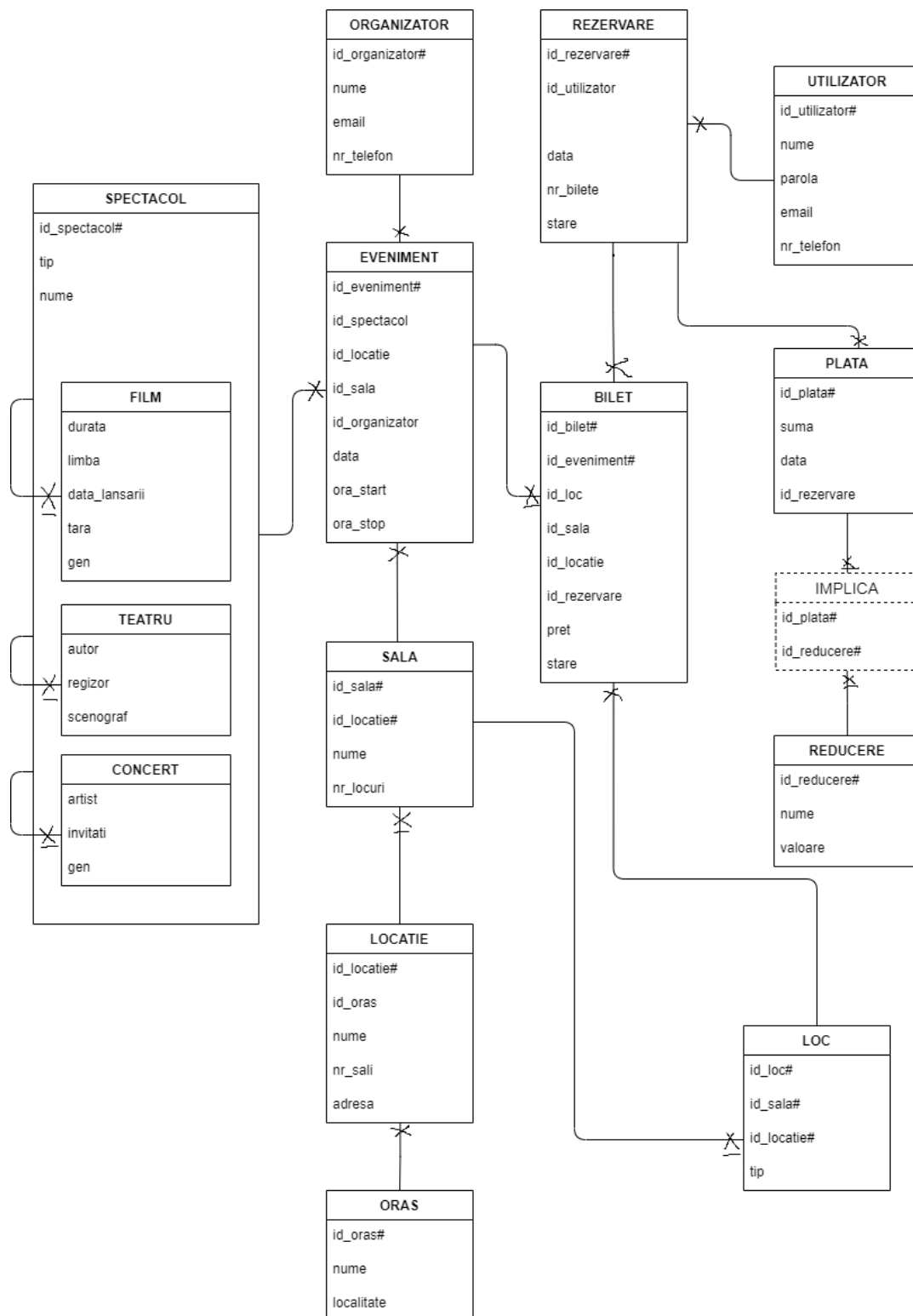
nume = șir de caractere de lungime maximă 50, reprezentând numele reducerii, valoarea nu poate fi nulă

valoare = variabilă de tip întreg, reprezintă valoarea reducerii, valoarea nu poate fi nulă

6. Realizarea diagramei ER corespunzătoare descrierii de la punctele 3-5



7. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. Diagrama conceptuală obținută trebuie să conțină minimum 6 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.



8. Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7

SPECTACOL(id_spectacol#, tip, nume)

FILM(id_spectacol#, durata, limba, data_lansarii, tara, gen)

TEATRU(id_spectacol#, autor, regizor, scenograf)

CONCERT(id_spectacol#, artist, invitati, gen)

ORGANIZATOR(id_organizator#, nume, email, nr_telefon)

EVENIMENT(id_eveniment#, id_spectacol, id_locatie, id_sala, id_organizator, data_eveniment, ora_start, ora_stop)

SALA(id_sala#, id_locatie#, nume, nr_locuri)

LOCATIE(id_locatie#, id_oras, nume, nr_sali, adresa)

ORAS(id_oras#, nume, localitate)

REZERVARE(id_rezervare#, id_utilizator, data_rezervare, nr_bilete, stare)

BILET(id_bilet#, id_eveniment#, id_loc, id_sala, id_locatie, id_rezervare, pret, stare)

LOC(id_loc#, id_sala#, id_locatie#, tip)

UTILIZATOR(id_utilizator#, nume, parola, email, nr_telefon)

PLATA(id_plata#, suma, data_plata, id_rezervare)

IMPLICA(id_plata#, id_reducere#)

REDUCERE(id_reducere#, nume, valoare)

9. Realizarea normalizării până la forma normală 3 (FN1-FN3).

O posibila formă a schemei care nu se încadrează în FN1 este:

SPECTACOL(id_spectacol#, tip, nume)

FILM(id_spectacol#, durata, limba, data_lansarii, tara, gen)

TEATRU(id_spectacol#, autor, regizor, scenograf)

CONCERT(id_spectacol#, artist, invitati, gen)

EVENIMENT(id_eveniment#, id_spectacol, id_locatie, id_sala, id_organizator#, nume_organizator, data_eveniment, ora_start, ora_stop)

SALA(id_sala#, id_locatie#, nume, nr_locuri)

LOCATIE(id_locatie#, nume_oras, localitate)

REZERVARE(id_rezervare#, id_utilizator, data_rezervare, nr_bilete, stare)

BILET(id_bilet#, id_eveniment#, id_loc, id_sala, id_locatie, id_rezervare, pret, stare)

LOC(id_loc#, id_sala#, id_locatie#, tip)

UTILIZATOR(id_utilizator#, nume, parola, email, nr_telefon)

PLATA(id_plata#, id_reducere#, suma, data_plata, id_rezervare)

REDUCERE(id_reducere#, nume, valoare)

Această variantă nu se încadrează în FN1 deoarece id_reducere este cheie primară a entității PLATA, deși o plată poate implica mai multe reduceri. Pentru a aduce modelul în FN1, avem nevoie de tabelul asociativ IMPLICA:

SPECTACOL(id_spectacol#, tip, nume)

FILM(id_spectacol#, durata, limba, data_lansarii, tara, gen)

TEATRU(id_spectacol#, autor, regizor, scenograf)

CONCERT(id_spectacol#, artist, invitati, gen)

EVENIMENT(id_eveniment#, id_spectacol, id_locatie, id_sala, id_organizator#, nume_organizator, data_eveniment, ora_start, ora_stop)

SALA(id_sala#, id_locatie#, nume, nr_locuri)

LOCATIE(id_locatie#, nume_oras, localitate)

REZERVARE(id_rezervare#, id_utilizator, data_rezervare, nr_bilete, stare)

BILET(id_bilet#, id_eveniment#, id_loc, id_sala, id_locatie, id_rezervare, pret, stare)

LOC(id_loc#, id_sala#, id_locatie#, tip)

UTILIZATOR(id_utilizator#, nume, parola, email, nr_telefon)

PLATA(id_plata#, id_reducere#, suma, data_plata, id_rezervare)

IMPLICA(id_plata#, id_reducere#)

REDUCERE(id_reducere#, nume, valoare)

Această variantă nu se încadrează în FN2 deoarece atributul `nume_organizator` al entității `EVENTIMENT` nu depinde funcțional de totalitatea cheii primare, adică `id_eventiment` și `id_organizator`, ci numai de `id_organizator`. Pentru a aduce modelul în FN2, trebuie creată entitatea `ORGANIZATOR`:

`SPECTACOL(id_spectacol#, tip, nume)`

`FILM(id_spectacol#, durata, limba, data_lansarii, tara, gen)`

`TEATRU(id_spectacol#, autor, regizor, scenograf)`

`CONCERT(id_spectacol#, artist, invitati, gen)`

`ORGANIZATOR(id_organizator#, nume, email, nr_telefon)`

`EVENTIMENT(id_eventiment#, id_spectacol, id_locatie, id_sala, id_organizator, data_eventiment, ora_start, ora_stop)`

`SALA(id_sala#, id_locatie#, nume, nr_locuri)`

`LOCATIE(id_locatie#, nume_oras, localitate)`

`REZERVARE(id_rezervare#, id_utilizator, data_rezervare, nr_bilete, stare)`

`BILET(id_bilet#, id_eventiment#, id_loc, id_sala, id_locatie, id_rezervare, pret, stare)`

`LOC(id_loc#, id_sala#, id_locatie#, tip)`

`UTILIZATOR(id_utilizator#, nume, parola, email, nr_telefon)`

`PLATA(id_plata#, suma, data_plata, id_rezervare)`

`IMPLICA(id_plata#, id_reducere#)`

`REDUCERE(id_reducere#, nume, valoare)`

Această variantă nu se încadrează în FN3 deoarece toate atributele non-chei ale entității `LOCATIE` sunt mutual dependente, deci le putem organiza într-o nouă entitate, `ORAS`:

`SPECTACOL(id_spectacol#, tip, nume)`

`FILM(id_spectacol#, durata, limba, data_lansarii, tara, gen)`

`TEATRU(id_spectacol#, autor, regizor, scenograf)`

`CONCERT(id_spectacol#, artist, invitati, gen)`

`ORGANIZATOR(id_organizator#, nume, email, nr_telefon)`

EVENIMENT(id_eveniment#, id_spectacol, id_locatie, id_sala, id_organizator, data_eveniment, ora_start, ora_stop)

SALA(id_sala#, id_locatie#, nume, nr_locuri)

LOCATIE(id_locatie#, id_oras, nume, nr_sali, adresa)

ORAS(id_oras#, nume, localitate)

REZERVARE(id_rezervare#, id_utilizator, data_rezervare, nr_bilete, stare)

BILET(id_bilet#, id_eveniment#, id_loc, id_sala, id_locatie, id_rezervare, pret, stare)

LOC(id_loc#, id_sala#, id_locatie#, tip)

UTILIZATOR(id_utilizator#, nume, parola, email, nr_telefon)

PLATA(id_plata#, suma, data_plata, id_rezervare)

IMPLICA(id_plata#, id_reducere#)

REDUCERE(id_reducere#, nume, valoare)

10. Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele (punctul 11).

CREATE SEQUENCE secv_utilizator

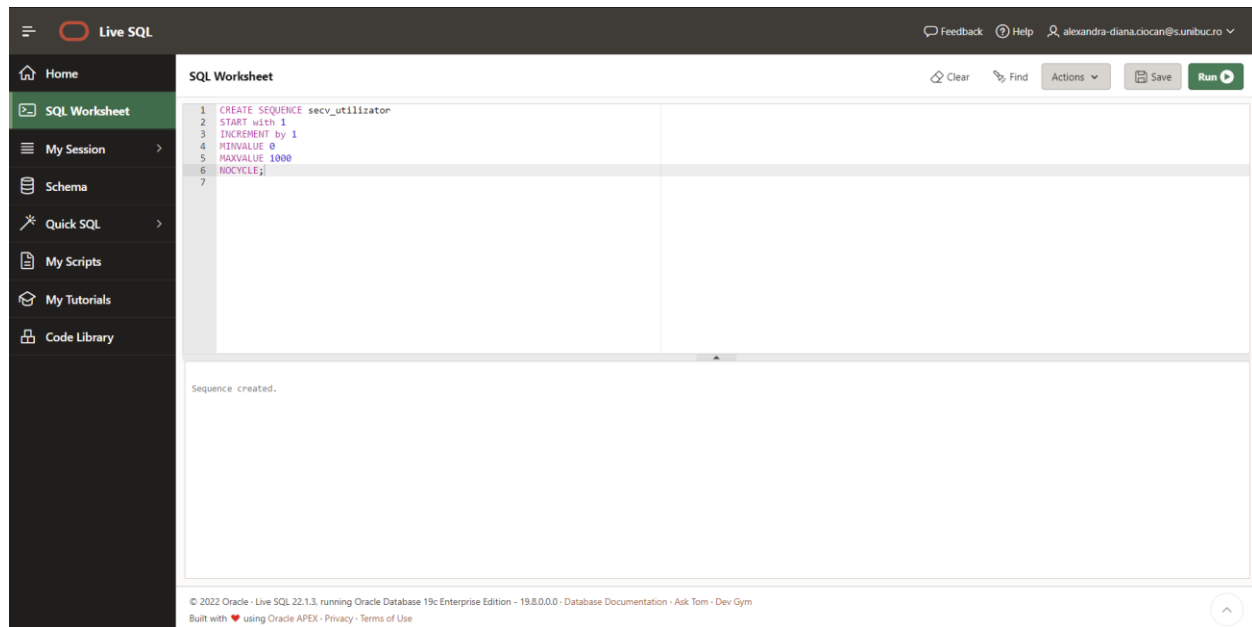
START with 1

INCREMENT by 1

MINVALUE 0

MAXVALUE 1000

NOCYCLE;



11. Crearea tabelor în SQL și inserarea de date coerente în fiecare dintre acestea (minimum 5 înregistrări în fiecare tabel neasociativ; minimum 10 înregistrări în tabelele asociative).

```
CREATE TABLE SPECTACOL(
id_spectacol int primary key,
tip varchar2(20) not null,
nume varchar2(100) not null
);
```

```
CREATE TABLE FILM(
id_spectacol int primary key,
durata int,
limba varchar2(20),
data_lansarii date,
tara varchar2(20),
gen varchar2(20),
constraint fk_spectacolfilm foreign key (id_spectacol) references spectacol(id_spectacol)
```

);

CREATE TABLE TEATRU(

id_spectacol int primary key,

autor varchar2(50),

regizor varchar2(50),

scenograf varchar2(50),

constraint fk_spectacolteatru foreign key (id_spectacol) references spectacol(id_spectacol)

);

CREATE TABLE CONCERT(

id_spectacol int primary key,

artist varchar2(50),

invitati varchar2(100),

gen varchar2(50),

constraint fk_spectacolconcert foreign key (id_spectacol) references spectacol(id_spectacol)

);

CREATE TABLE ORGANIZATOR(

id_organizator int primary key,

nume varchar2(50) not null,

email varchar2(50) not null,

nr_telefon varchar2(20) not null

);

CREATE TABLE ORAS(

id_oras int primary key,

nume varchar2(50) not null,

localitate varchar2(50) not null

);

CREATE TABLE LOCATIE(

id_locatie int primary key,

id_oras int not null,

nume varchar2(50) not null,

nr_sali int not null,

adresa varchar2(100) not null

);

CREATE TABLE SALA(

id_sala int not null,

id_locatie int not null,

nume varchar2(50) not null,

nr_locuri int,

primary key (id_sala, id_locatie),

constraint fk_locatiesala foreign key (id_locatie) references locatie(id_locatie)

);

CREATE TABLE EVENIMENT(

id_eveniment int primary key,

id_spectacol int not null,

id_locatie int not null,

id_sala int not null,

id_organizator int not null,

data_eveniment date not null,

ora_start varchar2(10) not null,

ora_stop varchar2(10) not null,

```
constraint fk_spectacoleveniment foreign key (id_spectacol) references spectacol(id_spectacol),  
constraint fk_salaeveniment foreign key (id_sala, id_locatie) references sala(id_sala, id_locatie),  
constraint fk_organizatoreveniment foreign key (id_organizator) references organizator(id_organizator)  
);
```

```
CREATE TABLE LOC(  
  
id_loc int not null,  
  
id_sala int not null,  
  
id_locatie int not null,  
  
tip varchar2(50) not null,  
  
primary key (id_loc, id_sala, id_locatie),  
  
constraint fk_salaloc foreign key (id_sala, id_locatie) references sala(id_sala, id_locatie)  
  
);
```

```
CREATE TABLE UTILIZATOR(  
  
id_utilizator int primary key,  
  
nume varchar2(50) not null,  
  
parola varchar2(50) not null,  
  
email varchar2(50) not null,  
  
nr_telefon varchar2(20) not null  
  
);
```

```
CREATE TABLE REZERVARE(  
  
id_rezervare int primary key,  
  
id_utilizator int not null,  
  
data_rezervare date not null,  
  
nr_bilete int not null,  
  
stare int not null,
```

```
constraint fk_utilizatorrezervare foreign key (id_utilizator) references utilizator(id_utilizator)
);

CREATE TABLE BILET(
id_bilet int,
id_eveniment int,
id_loc int not null,
id_sala int not null,
id_locatie int not null,
id_rezervare int,
pret int not null,
stare int not null,
primary key (id_bilet, id_eveniment),
constraint fk_locbilet foreign key (id_loc, id_sala, id_locatie) references loc(id_loc, id_sala, id_locatie),
constraint fk_evenimentbilet foreign key (id_eveniment) references eveniment(id_eveniment),
constraint fk_rezervarebilet foreign key (id_rezervare) references rezervare(id_rezervare)
);

CREATE TABLE PLATA(
id_plata int primary key,
suma int not null,
data_plata date not null,
id_rezervare int not null,
constraint fk_rezervareplata foreign key (id_rezervare) references rezervare(id_rezervare)
);

CREATE TABLE REDUCERE(
id_reducere int primary key,
```

nume varchar2(50) not null,

valoare int not null

);

CREATE TABLE IMPLICA(

id_plata int,

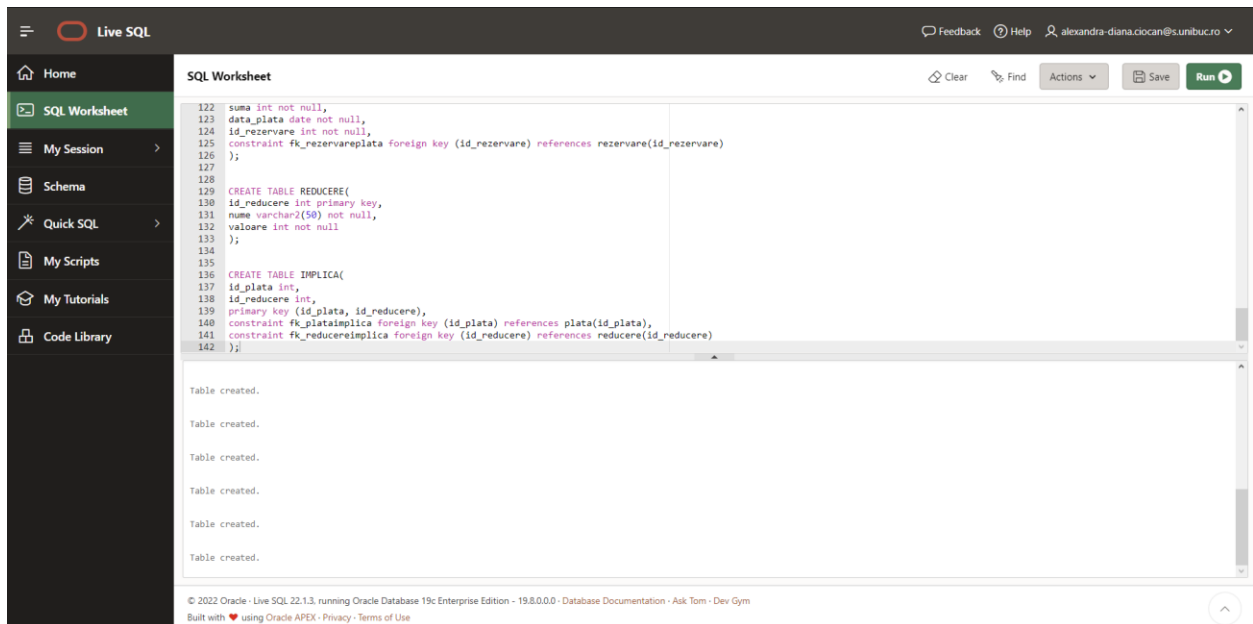
id_reducere int,

primary key (id_plata, id_reducere),

constraint fk_plataimplica foreign key (id_plata) references plata(id_plata),

constraint fk_reducereimplica foreign key (id_reducere) references reducere(id_reducere)

);



INSERT INTO spectacol VALUES (1, 'film', 'Omul cu umbra');

INSERT INTO spectacol VALUES (2, 'film', 'The Shawshank Redemption');

INSERT INTO spectacol VALUES (3, 'film', 'The Godfather');

INSERT INTO spectacol VALUES (4, 'film', 'The Dark Knight');

INSERT INTO spectacol VALUES (5, 'film', 'Pulp Fiction');

INSERT INTO spectacol VALUES (6, 'film', 'Amelie');

INSERT INTO spectacol VALUES (7, 'film', 'Fight Club');

INSERT INTO spectacol VALUES (8, 'film', 'The Passion of Joan of Arc');

INSERT INTO spectacol VALUES (9, 'film', 'Forrest Gump');

INSERT INTO spectacol VALUES (10, 'film', 'La Vie en Rose');

INSERT INTO spectacol VALUES (11, 'film', 'Amintiri din Epoca de Aur');

INSERT INTO spectacol VALUES (12, 'film', 'Filantropica');

INSERT INTO spectacol VALUES (13, 'teatru', 'Long Days Journey Into Night');

INSERT INTO spectacol VALUES (14, 'teatru', 'Hamlet');

INSERT INTO spectacol VALUES (15, 'teatru', 'Whos Afraid of Virginia Woolf?');

INSERT INTO spectacol VALUES (16, 'teatru', 'Death of a Salesman ');

INSERT INTO spectacol VALUES (17, 'teatru', 'Oedipus Rex');

INSERT INTO spectacol VALUES (18, 'teatru', 'Angels in America');

INSERT INTO spectacol VALUES (19, 'teatru', 'The Glass Menagerie');

INSERT INTO spectacol VALUES (20, 'teatru', 'Look Back in Anger');

INSERT INTO spectacol VALUES (21, 'concert', 'Holograf @ Sala Polivalenta');

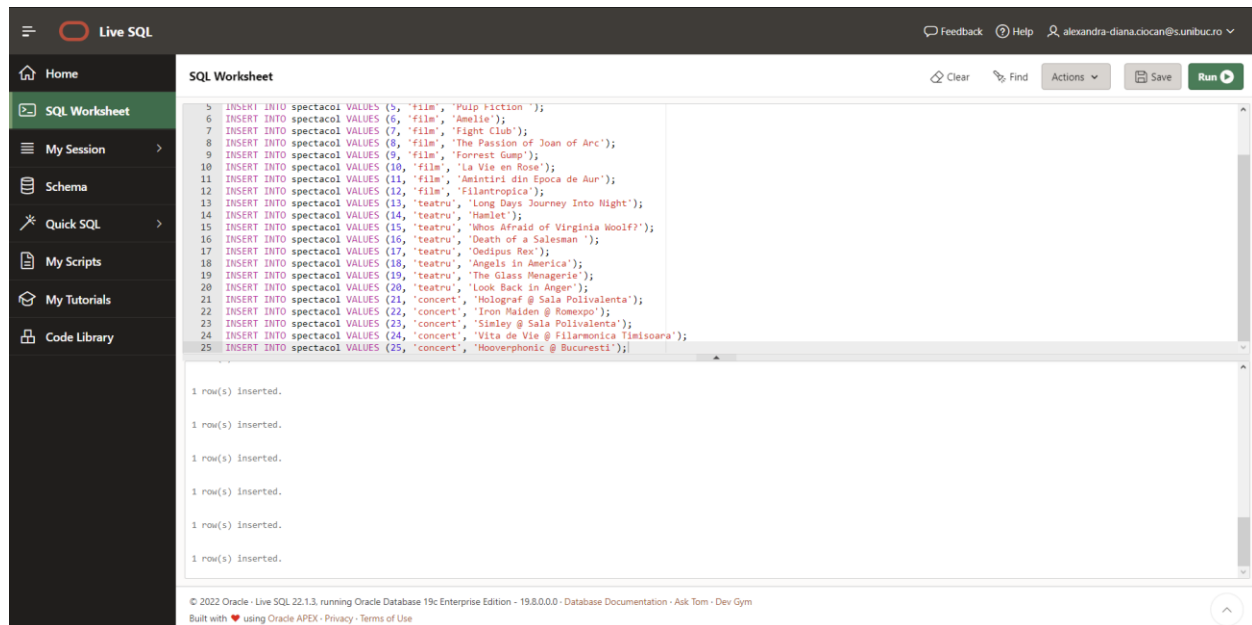
INSERT INTO spectacol VALUES (22, 'concert', 'Iron Maiden @ Romexpo');

INSERT INTO spectacol VALUES (23, 'concert', 'Simley @ Sala Polivalenta');

INSERT INTO spectacol VALUES (24, 'concert', 'Vita de Vie @ Filarmonica Timisoara');

INSERT INTO spectacol VALUES (25, 'concert', 'Hooverphonic @ Bucuresti');

Ciocan Alexandra-Diana, grupa 132



INSERT INTO film VALUES (1, 136, 'Romana', to_date('1997-04-17', 'yyyy-mm-dd'), 'Romania', 'Documentar');

INSERT INTO film VALUES (2, 110, 'Engleza', to_date('2011-09-14', 'yyyy-mm-dd'), 'Marea Britanie', 'Thriller');

INSERT INTO film VALUES (3, 140, 'Engleza', to_date('1998-09-10', 'yyyy-mm-dd'), 'SUA', 'Thriller');

INSERT INTO film VALUES (4, 109, 'Engleza', to_date('2015-07-16', 'yyyy-mm-dd'), 'SUA', 'Actiune');

INSERT INTO film VALUES (5, 138, 'Engleza', to_date('1999-07-06', 'yyyy-mm-dd'), 'SUA', 'Actiune');

INSERT INTO film VALUES (6, 119, 'Franceza', to_date('2004-07-19', 'yyyy-mm-dd'), 'Franta', 'Drama');

INSERT INTO film VALUES (7, 131, 'Engleza', to_date('1998-09-20', 'yyyy-mm-dd'), 'SUA', 'Actiune');

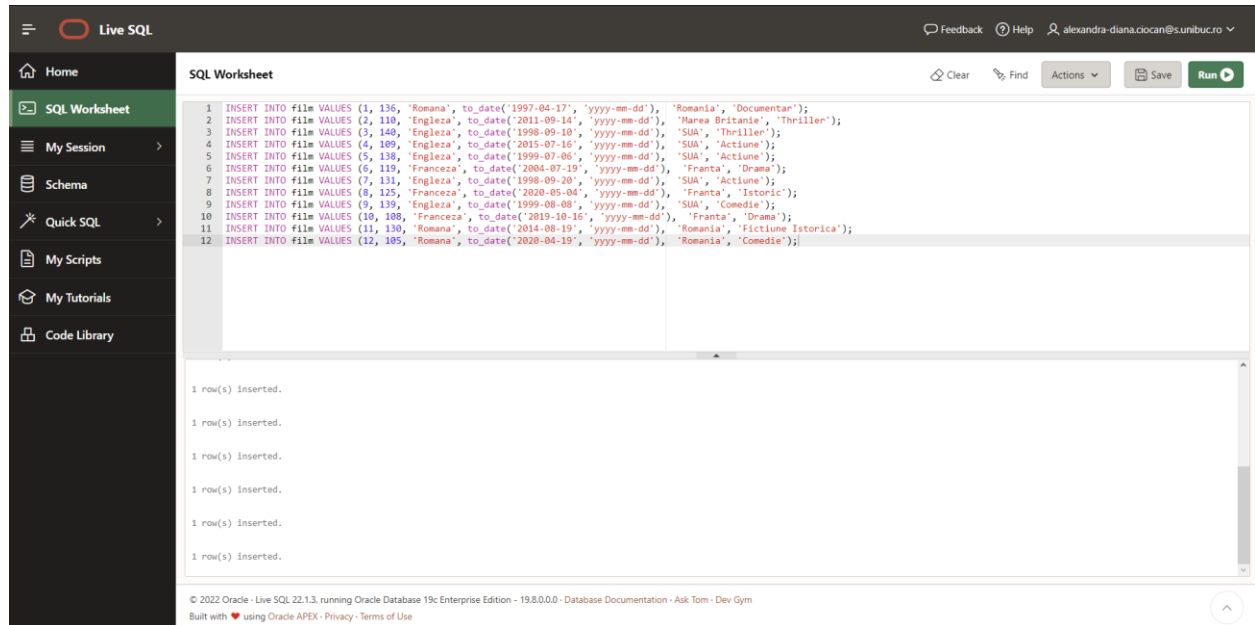
INSERT INTO film VALUES (8, 125, 'Franceza', to_date('2020-05-04', 'yyyy-mm-dd'), 'Franta', 'Istoric');

INSERT INTO film VALUES (9, 139, 'Engleza', to_date('1999-08-08', 'yyyy-mm-dd'), 'SUA', 'Comedie');

INSERT INTO film VALUES (10, 108, 'Franceza', to_date('2019-10-16', 'yyyy-mm-dd'), 'Franta', 'Drama');

INSERT INTO film VALUES (11, 130, 'Romana', to_date('2014-08-19', 'yyyy-mm-dd'), 'Romania', 'Fictiune Istoric');

INSERT INTO film VALUES (12, 105, 'Romana', to_date('2020-04-19', 'yyyy-mm-dd'), 'Romania', 'Comedie');



The screenshot shows the Live SQL interface. On the left is a sidebar with navigation links: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains 12 lines of SQL code. Below the code, the execution results are displayed, showing '1 row(s) inserted.' for each of the 12 statements. At the bottom, a footer indicates the version: '© 2022 Oracle - Live SQL 22.1.3, running Oracle Database 19c Enterprise Edition - 19.0.0.0.0 - Database Documentation - Ask Tom - Dev Gym'.

```
1 INSERT INTO file VALUES (1, 136, 'Romana', to_date('1997-04-17', 'yyyy-mm-dd'), 'Romania', 'Documentar');
2 INSERT INTO file VALUES (2, 110, 'Engleza', to_date('2011-09-14', 'yyyy-mm-dd'), 'Marea Britanie', 'Thriller');
3 INSERT INTO file VALUES (3, 140, 'Engleza', to_date('1998-09-10', 'yyyy-mm-dd'), 'SUA', 'Thriller');
4 INSERT INTO file VALUES (4, 109, 'Engleza', to_date('2015-07-16', 'yyyy-mm-dd'), 'SUA', 'Actiune');
5 INSERT INTO file VALUES (5, 136, 'Engleza', to_date('1999-07-06', 'yyyy-mm-dd'), 'SUA', 'Actiune');
6 INSERT INTO file VALUES (6, 119, 'Franceza', to_date('2004-07-19', 'yyyy-mm-dd'), 'Franta', 'Drama');
7 INSERT INTO file VALUES (7, 131, 'Engleza', to_date('1998-09-20', 'yyyy-mm-dd'), 'SUA', 'Actiune');
8 INSERT INTO file VALUES (8, 125, 'Franceza', to_date('2020-05-04', 'yyyy-mm-dd'), 'Franta', 'Istoric');
9 INSERT INTO file VALUES (9, 139, 'Engleza', to_date('1999-08-08', 'yyyy-mm-dd'), 'SUA', 'Comedie');
10 INSERT INTO file VALUES (10, 108, 'Franceza', to_date('2019-10-16', 'yyyy-mm-dd'), 'Franta', 'Drama');
11 INSERT INTO file VALUES (11, 130, 'Romana', to_date('2014-08-19', 'yyyy-mm-dd'), 'Romania', 'Fictiune Istoric');
12 INSERT INTO file VALUES (12, 105, 'Romana', to_date('2020-04-19', 'yyyy-mm-dd'), 'Romania', 'Comedie');
```

1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.

© 2022 Oracle - Live SQL 22.1.3, running Oracle Database 19c Enterprise Edition - 19.0.0.0.0 - Database Documentation - Ask Tom - Dev Gym
Built with ♥ using Oracle APEX - Privacy - Terms of Use

INSERT INTO teatru VALUES (13, 'Eugene ONeill', 'Radu Afrim', 'Elena Fortu');

INSERT INTO teatru VALUES (14, 'William Shakespeare', 'Andrei Serban', 'Petrica Ionescu');

INSERT INTO teatru VALUES (15, 'Edward Albee', 'Victor Ioan Frunza', 'Livia Ante');

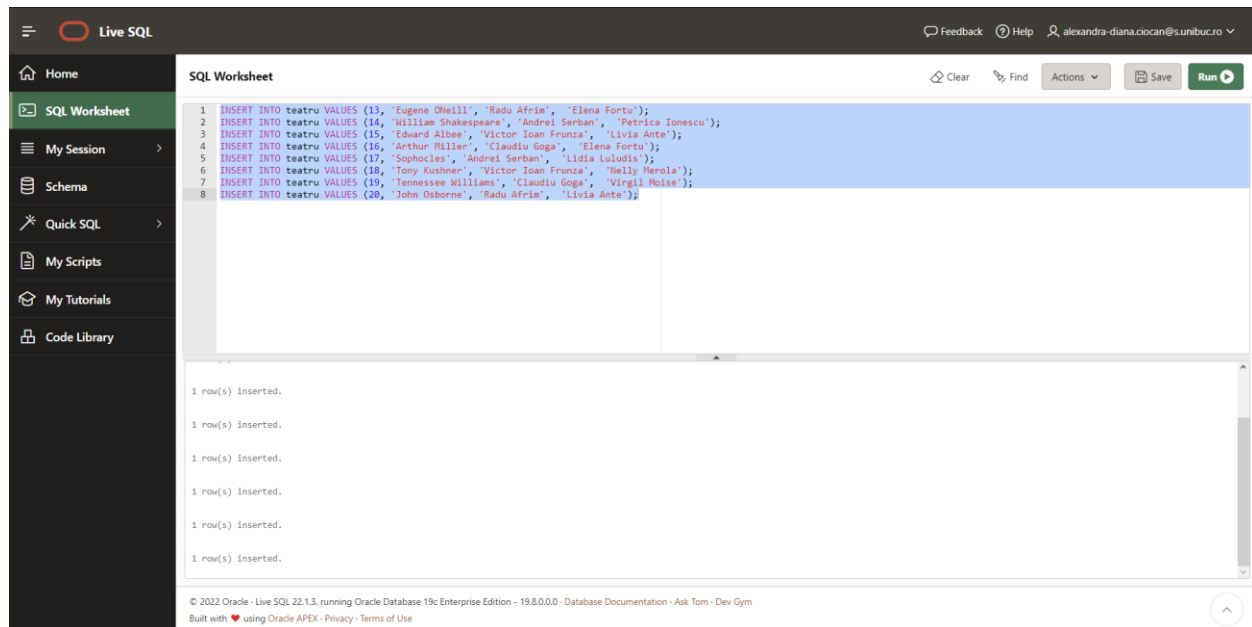
INSERT INTO teatru VALUES (16, 'Arthur Miller', 'Claudiu Goga', 'Elena Fortu');

INSERT INTO teatru VALUES (17, 'Sophocles', 'Andrei Serban', 'Lidia Luludis');

INSERT INTO teatru VALUES (18, 'Tony Kushner', 'Victor Ioan Frunza', 'Nelly Merola');

INSERT INTO teatru VALUES (19, 'Tennessee Williams', 'Claudiu Goga', 'Virgil Moise');

INSERT INTO teatru VALUES (20, 'John Osborne', 'Radu Afrim', 'Livia Ante');



```
1 INSERT INTO teatru VALUES (13, 'Eugene O'Neill', 'Radu Afrim', 'Elena Fortu');
2 INSERT INTO teatru VALUES (14, 'William Shakespeare', 'Andrei Serban', 'Petrica Ionescu');
3 INSERT INTO teatru VALUES (15, 'Edward Albee', 'Victor Ioan Frunza', 'Livia Ante');
4 INSERT INTO teatru VALUES (16, 'Arthur Miller', 'Claudiu Goga', 'Elena Fortu');
5 INSERT INTO teatru VALUES (17, 'Sophocles', 'Andrei Serban', 'Lidia Iulidis');
6 INSERT INTO teatru VALUES (18, 'Tony Kushner', 'Victor Ioan Frunza', 'Nelly Merola');
7 INSERT INTO teatru VALUES (19, 'Tennessee Williams', 'Claudiu Goga', 'Virgil Moise');
8 INSERT INTO teatru VALUES (20, 'John Osborne', 'Radu Afrim', 'Livia Ante');
```

1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.

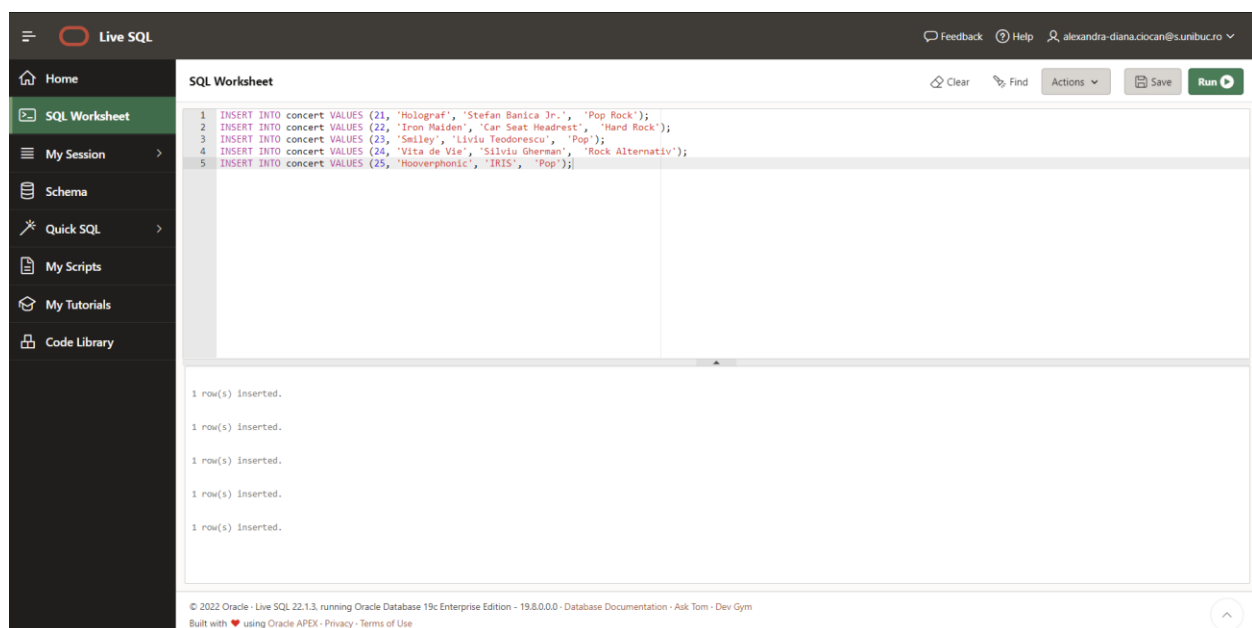
INSERT INTO concert VALUES (21, 'Holograf', 'Stefan Banica Jr.', 'Pop Rock');

INSERT INTO concert VALUES (22, 'Iron Maiden', 'Car Seat Headrest', 'Hard Rock');

INSERT INTO concert VALUES (23, 'Smiley', 'Livia Teodorescu', 'Pop');

INSERT INTO concert VALUES (24, 'Vita de Vie', 'Silviu Gherman', 'Rock Alternativ');

INSERT INTO concert VALUES (25, 'Hooverphonic', 'IRIS', 'Pop');



```
1 INSERT INTO concert VALUES (21, 'Holograf', 'Stefan Banica Jr.', 'Pop Rock');
2 INSERT INTO concert VALUES (22, 'Iron Maiden', 'Car Seat Headrest', 'Hard Rock');
3 INSERT INTO concert VALUES (23, 'Smiley', 'Livia Teodorescu', 'Pop');
4 INSERT INTO concert VALUES (24, 'Vita de Vie', 'Silviu Gherman', 'Rock Alternativ');
5 INSERT INTO concert VALUES (25, 'Hooverphonic', 'IRIS', 'Pop');
```

1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.

INSERT INTO organizator VALUES (1, 'Cornel Popescu', 'cornelpopescu@hotmail.com', '0784819518');

```
INSERT INTO organizator VALUES (2, 'Ionel Teodor', 'ionel_tteo@yahoo.com', '0725100212');

INSERT INTO organizator VALUES (3, 'Andrei Gilcea', 'gilcea_andrei31@gmail.com', '0774333944');

INSERT INTO organizator VALUES (4, 'Rares Pauna', 'pauna_rar3@hotmail.com', '0769834565');

INSERT INTO organizator VALUES (5, 'Cosmin Borcea', 'cosmin@borcea_events.com', '0713528122');

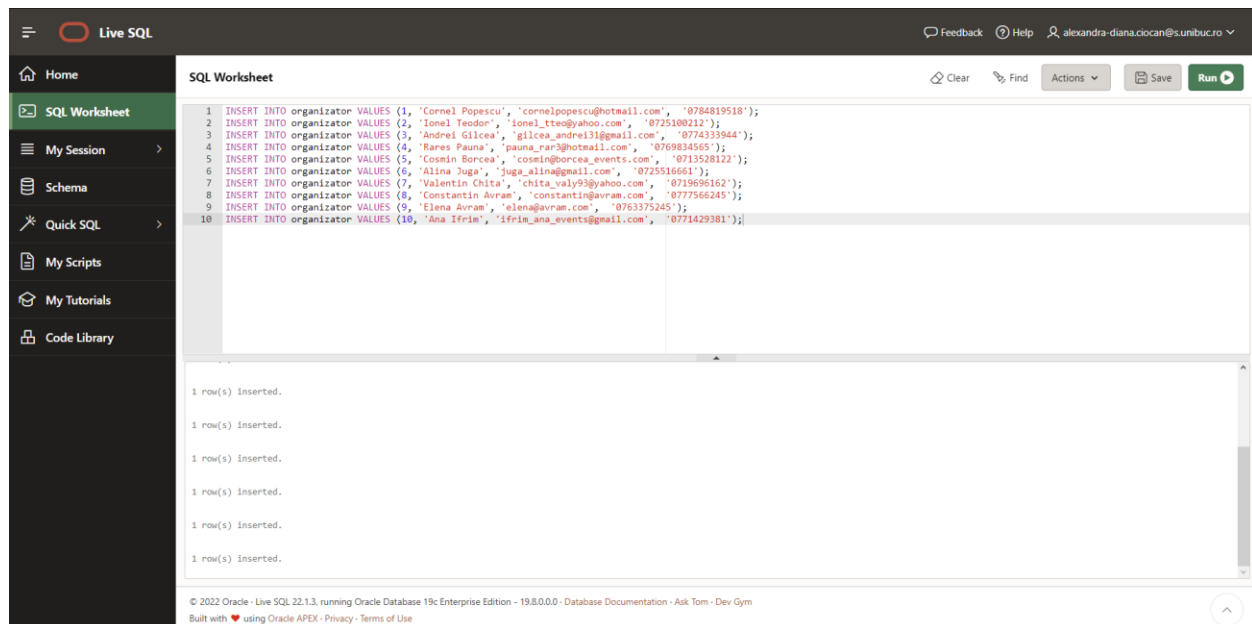
INSERT INTO organizator VALUES (6, 'Alina Juga', 'juga_alina@gmail.com', '0725516661');

INSERT INTO organizator VALUES (7, 'Valentin Chita', 'chita_valy93@yahoo.com', '0719696162');

INSERT INTO organizator VALUES (8, 'Constantin Avram', 'constantin@avram.com', '0777566245');

INSERT INTO organizator VALUES (9, 'Elena Avram', 'elena@avram.com', '0763375245');

INSERT INTO organizator VALUES (10, 'Ana Ifrim', 'ifrim_ana_events@gmail.com', '0771429381');
```



```
INSERT INTO oras VALUES (1, 'Bucuresti', 'Bucuresti');

INSERT INTO oras VALUES (2, 'Oradea', 'Bihor');

INSERT INTO oras VALUES (3, 'Arad', 'Banat');

INSERT INTO oras VALUES (4, 'Timisoara', 'Timis');

INSERT INTO oras VALUES (5, 'Craiova', 'Dolj');

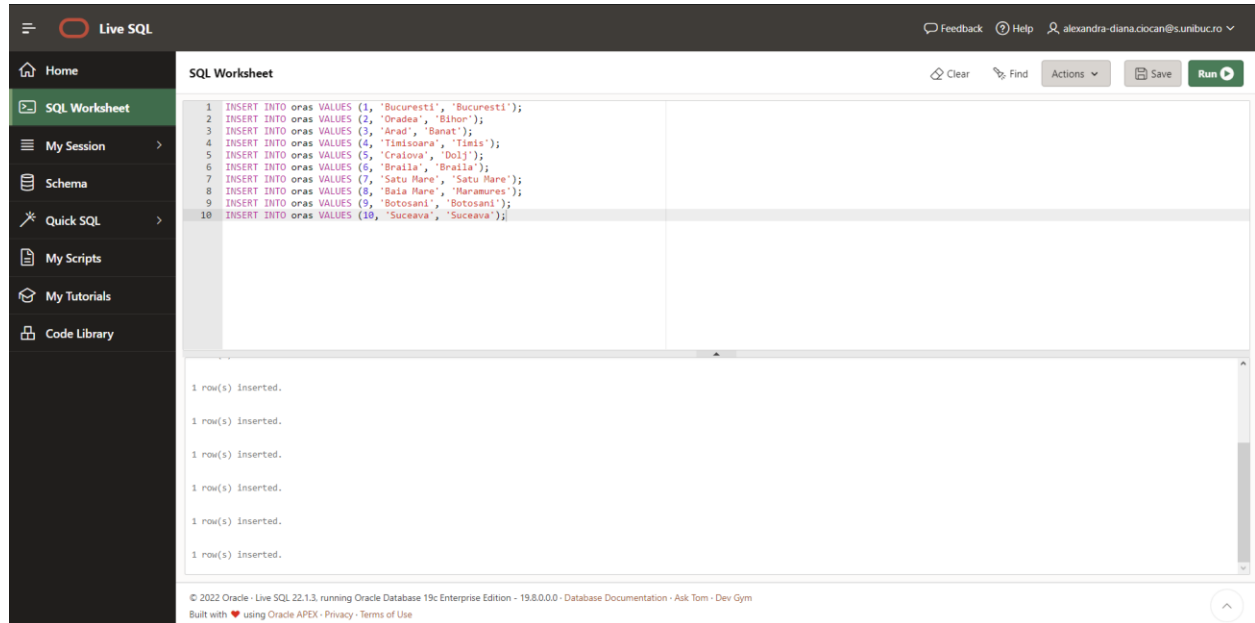
INSERT INTO oras VALUES (6, 'Braila', 'Braila');

INSERT INTO oras VALUES (7, 'Satu Mare', 'Satu Mare');
```

INSERT INTO oras VALUES (8, 'Baia Mare', 'Maramures');

INSERT INTO oras VALUES (9, 'Botosani', 'Botosani');

INSERT INTO oras VALUES (10, 'Suceava', 'Suceava');



INSERT INTO locatie VALUES (1, 1, 'Great Bucharest Halls', 3, 'Str. Foisorului, 12-16');

INSERT INTO locatie VALUES (2, 2, 'Tuned@Arad', 2, 'Bd. Unirii 24');

INSERT INTO locatie VALUES (3, 3, 'Sera Cocorului', 4, 'Str. Cocorului 35');

INSERT INTO locatie VALUES (4, 4, 'Crystal Palace', 2, 'Bd. Iancu de Hendoara 120');

INSERT INTO locatie VALUES (5, 5, 'The Lions Hall', 4, 'Str. Luigi Cazzavillan 20');

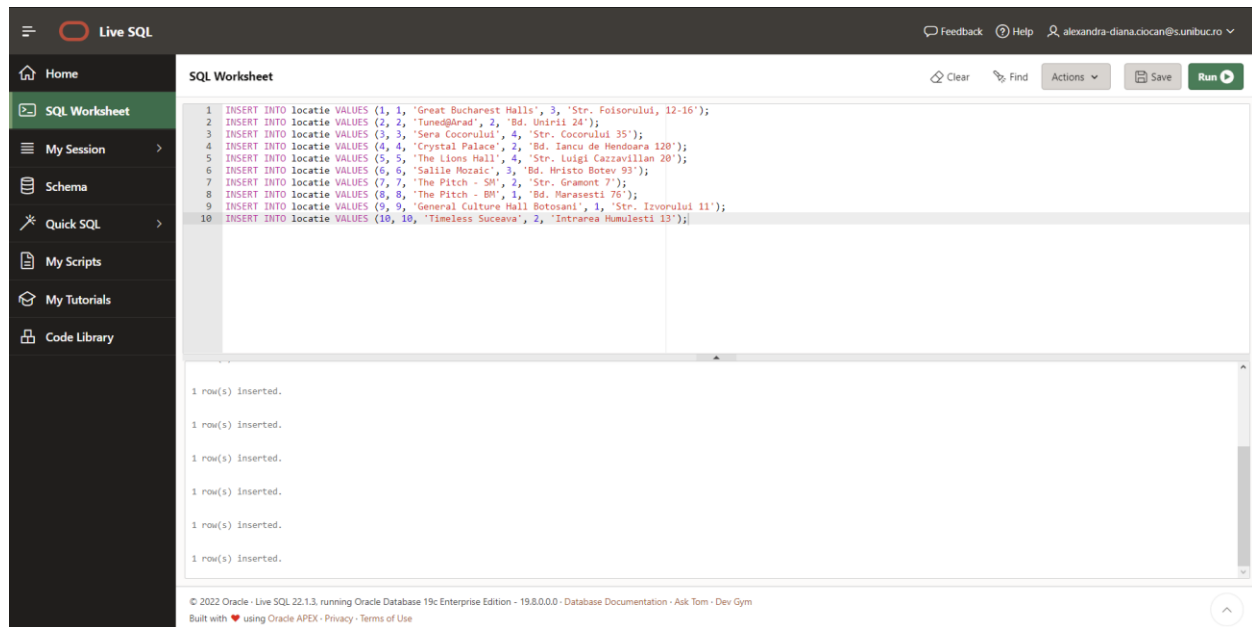
INSERT INTO locatie VALUES (6, 6, 'Salile Mozaic', 3, 'Bd. Hristo Botev 93');

INSERT INTO locatie VALUES (7, 7, 'The Pitch - SM', 2, 'Str. Gramont 7');

INSERT INTO locatie VALUES (8, 8, 'The Pitch - BM', 1, 'Bd. Marasesti 76');

INSERT INTO locatie VALUES (9, 9, 'General Culture Hall Botosani', 1, 'Str. Izvorului 11');

INSERT INTO locatie VALUES (10, 10, 'Timeless Suceava', 2, 'Intrarea Humulesti 13');



INSERT INTO sala VALUES (1, 1, 'I. L. Caragiale', 17);

INSERT INTO sala VALUES (2, 1, 'Mihai Viteazu', 10);

INSERT INTO sala VALUES (3, 1, 'Ion Barbu', 12);

INSERT INTO sala VALUES (1, 2, 'Claudia Boghicevici', 6);

INSERT INTO sala VALUES (2, 2, 'Constantin Brancusi', 6);

INSERT INTO sala VALUES (1, 3, 'Henri Coanda', 16);

INSERT INTO sala VALUES (2, 3, 'Nicolae Iorga', 6);

INSERT INTO sala VALUES (3, 3, 'George Enescu', 8);

INSERT INTO sala VALUES (4, 3, 'Ion I.C. Bratianu', 9);

INSERT INTO sala VALUES (1, 4, 'Emil Cioran', 12);

INSERT INTO sala VALUES (2, 4, 'Petre tutea', 3);

INSERT INTO sala VALUES (1, 5, 'Nichita Stanescu', 4);

INSERT INTO sala VALUES (2, 5, 'Lucian Blaga', 3);

INSERT INTO sala VALUES (3, 5, 'Titu Maiorescu', 13);

INSERT INTO sala VALUES (4, 5, 'Nicolae Balcescu', 8);

INSERT INTO sala VALUES (1, 6, 'Marin Preda', 15);

Ciocan Alexandra-Diana, grupa 132

INSERT INTO sala VALUES (2, 6, 'Grigore C. Moisil', 15);

INSERT INTO sala VALUES (3, 6, 'Anghel Saligny', 3);

INSERT INTO sala VALUES (1, 7, 'Sergiu Nicolaescu', 1);

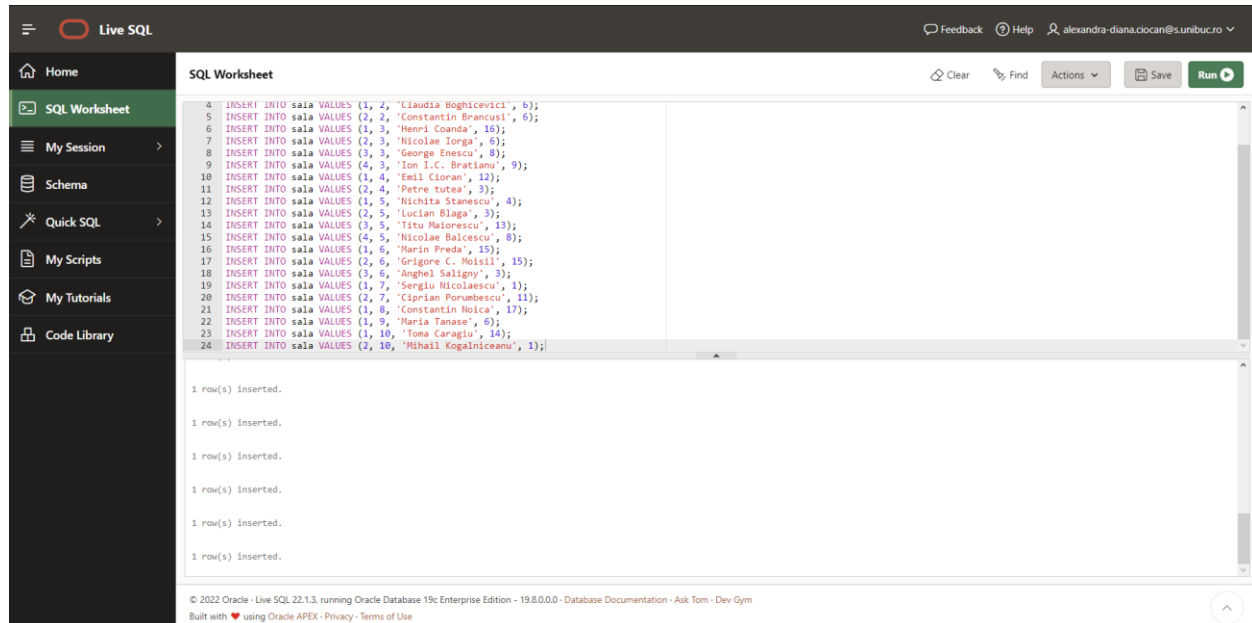
INSERT INTO sala VALUES (2, 7, 'Ciprian Porumbescu', 11);

INSERT INTO sala VALUES (1, 8, 'Constantin Noica', 17);

INSERT INTO sala VALUES (1, 9, 'Maria Tanase', 6);

INSERT INTO sala VALUES (1, 10, 'Toma Caragiu', 14);

INSERT INTO sala VALUES (2, 10, 'Mihail Kogalniceanu', 1);



INSERT INTO loc VALUES (1, 1, 1, 'sala');

INSERT INTO loc VALUES (2, 1, 1, 'sala');

INSERT INTO loc VALUES (3, 1, 1, 'sala');

INSERT INTO loc VALUES (4, 1, 1, 'sala');

INSERT INTO loc VALUES (5, 1, 1, 'sala');

INSERT INTO loc VALUES (6, 1, 1, 'sala');

INSERT INTO loc VALUES (7, 1, 1, 'sala');

INSERT INTO loc VALUES (8, 1, 1, 'sala');

```
INSERT INTO loc VALUES (9, 1, 1, 'loja');  
INSERT INTO loc VALUES (10, 1, 1, 'loja');  
INSERT INTO loc VALUES (11, 1, 1, 'loja');  
INSERT INTO loc VALUES (12, 1, 1, 'loja');  
INSERT INTO loc VALUES (13, 1, 1, 'loja');  
INSERT INTO loc VALUES (14, 1, 1, 'loja');  
INSERT INTO loc VALUES (15, 1, 1, 'loja');  
INSERT INTO loc VALUES (16, 1, 1, 'loja');  
INSERT INTO loc VALUES (17, 1, 1, 'loja');  
INSERT INTO loc VALUES (1, 2, 1, 'sala');  
INSERT INTO loc VALUES (2, 2, 1, 'sala');  
INSERT INTO loc VALUES (3, 2, 1, 'sala');  
INSERT INTO loc VALUES (4, 2, 1, 'sala');  
INSERT INTO loc VALUES (5, 2, 1, 'sala');  
INSERT INTO loc VALUES (6, 2, 1, 'sala');  
INSERT INTO loc VALUES (7, 2, 1, 'sala');  
INSERT INTO loc VALUES (8, 2, 1, 'sala');  
INSERT INTO loc VALUES (9, 2, 1, 'sala');  
INSERT INTO loc VALUES (10, 2, 1, 'sala');  
INSERT INTO loc VALUES (1, 3, 1, 'sala');  
INSERT INTO loc VALUES (2, 3, 1, 'sala');  
INSERT INTO loc VALUES (3, 3, 1, 'sala');  
INSERT INTO loc VALUES (4, 3, 1, 'sala');  
INSERT INTO loc VALUES (5, 3, 1, 'sala');  
INSERT INTO loc VALUES (6, 3, 1, 'sala');
```

```
INSERT INTO loc VALUES (7, 3, 1, 'sala');  
INSERT INTO loc VALUES (8, 3, 1, 'loja');  
INSERT INTO loc VALUES (9, 3, 1, 'loja');  
INSERT INTO loc VALUES (10, 3, 1, 'loja');  
INSERT INTO loc VALUES (11, 3, 1, 'loja');  
INSERT INTO loc VALUES (12, 3, 1, 'loja');  
INSERT INTO loc VALUES (1, 1, 2, 'normal');  
INSERT INTO loc VALUES (2, 1, 2, 'normal');  
INSERT INTO loc VALUES (3, 1, 2, 'normal');  
INSERT INTO loc VALUES (4, 1, 2, 'normal');  
INSERT INTO loc VALUES (5, 1, 2, 'normal');  
INSERT INTO loc VALUES (6, 1, 2, 'normal');  
INSERT INTO loc VALUES (1, 2, 2, 'loja');  
INSERT INTO loc VALUES (2, 2, 2, 'loja');  
INSERT INTO loc VALUES (3, 2, 2, 'loja');  
INSERT INTO loc VALUES (4, 2, 2, 'loja');  
INSERT INTO loc VALUES (5, 2, 2, 'loja');  
INSERT INTO loc VALUES (6, 2, 2, 'loja');  
INSERT INTO loc VALUES (1, 1, 3, 'loja');  
INSERT INTO loc VALUES (2, 1, 3, 'loja');  
INSERT INTO loc VALUES (3, 1, 3, 'loja');  
INSERT INTO loc VALUES (4, 1, 3, 'loja');  
INSERT INTO loc VALUES (5, 1, 3, 'loja');  
INSERT INTO loc VALUES (6, 1, 3, 'loja');  
INSERT INTO loc VALUES (7, 1, 3, 'loja');
```



```
INSERT INTO loc VALUES (8, 1, 3, 'sala');
INSERT INTO loc VALUES (9, 1, 3, 'sala');
INSERT INTO loc VALUES (10, 1, 3, 'sala');
INSERT INTO loc VALUES (11, 1, 3, 'sala');
INSERT INTO loc VALUES (12, 1, 3, 'sala');
INSERT INTO loc VALUES (13, 1, 3, 'sala');
INSERT INTO loc VALUES (14, 1, 3, 'sala');
INSERT INTO loc VALUES (15, 1, 3, 'sala');
INSERT INTO loc VALUES (16, 1, 3, 'sala');
INSERT INTO loc VALUES (1, 2, 3, 'normal');
INSERT INTO loc VALUES (2, 2, 3, 'normal');
INSERT INTO loc VALUES (3, 2, 3, 'normal');
INSERT INTO loc VALUES (4, 2, 3, 'normal');
INSERT INTO loc VALUES (5, 2, 3, 'normal');
INSERT INTO loc VALUES (6, 2, 3, 'normal');
INSERT INTO loc VALUES (1, 3, 3, 'normal');
INSERT INTO loc VALUES (2, 3, 3, 'normal');
INSERT INTO loc VALUES (3, 3, 3, 'normal');
INSERT INTO loc VALUES (4, 3, 3, 'normal');
INSERT INTO loc VALUES (5, 3, 3, 'normal');
INSERT INTO loc VALUES (6, 3, 3, 'normal');
INSERT INTO loc VALUES (7, 3, 3, 'normal');
INSERT INTO loc VALUES (8, 3, 3, 'normal');
INSERT INTO loc VALUES (1, 4, 3, 'normal');
INSERT INTO loc VALUES (2, 4, 3, 'normal');
```

```
INSERT INTO loc VALUES (3, 4, 3, 'normal');
INSERT INTO loc VALUES (4, 4, 3, 'normal');
INSERT INTO loc VALUES (5, 4, 3, 'normal');
INSERT INTO loc VALUES (6, 4, 3, 'normal');
INSERT INTO loc VALUES (7, 4, 3, 'normal');
INSERT INTO loc VALUES (8, 4, 3, 'normal');
INSERT INTO loc VALUES (9, 4, 3, 'normal');
INSERT INTO loc VALUES (1, 1, 4, 'normal');
INSERT INTO loc VALUES (2, 1, 4, 'normal');
INSERT INTO loc VALUES (3, 1, 4, 'normal');
INSERT INTO loc VALUES (4, 1, 4, 'normal');
INSERT INTO loc VALUES (5, 1, 4, 'normal');
INSERT INTO loc VALUES (6, 1, 4, 'normal');
INSERT INTO loc VALUES (7, 1, 4, 'normal');
INSERT INTO loc VALUES (8, 1, 4, 'normal');
INSERT INTO loc VALUES (9, 1, 4, 'normal');
INSERT INTO loc VALUES (10, 1, 4, 'normal');
INSERT INTO loc VALUES (11, 1, 4, 'normal');
INSERT INTO loc VALUES (12, 1, 4, 'normal');
INSERT INTO loc VALUES (1, 2, 4, 'normal');
INSERT INTO loc VALUES (2, 2, 4, 'normal');
INSERT INTO loc VALUES (3, 2, 4, 'normal');
INSERT INTO loc VALUES (1, 1, 5, 'sala');
INSERT INTO loc VALUES (2, 1, 5, 'sala');
INSERT INTO loc VALUES (3, 1, 5, 'sala');
```

```
INSERT INTO loc VALUES (4, 1, 5, 'sala');
```

```
INSERT INTO loc VALUES (1, 2, 5, 'sala');
```

```
INSERT INTO loc VALUES (2, 2, 5, 'sala');
```

```
INSERT INTO loc VALUES (3, 2, 5, 'sala');
```

```
INSERT INTO loc VALUES (1, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (2, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (3, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (4, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (5, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (6, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (7, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (8, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (9, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (10, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (11, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (12, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (13, 3, 5, 'sala');
```

```
INSERT INTO loc VALUES (1, 4, 5, 'loja');
```

```
INSERT INTO loc VALUES (2, 4, 5, 'loja');
```

```
INSERT INTO loc VALUES (3, 4, 5, 'loja');
```

```
INSERT INTO loc VALUES (4, 4, 5, 'loja');
```

```
INSERT INTO loc VALUES (5, 4, 5, 'loja');
```

```
INSERT INTO loc VALUES (6, 4, 5, 'loja');
```

```
INSERT INTO loc VALUES (7, 4, 5, 'loja');
```

```
INSERT INTO loc VALUES (8, 4, 5, 'loja');
```

```
INSERT INTO loc VALUES (1, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (2, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (3, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (4, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (5, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (6, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (7, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (8, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (9, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (10, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (11, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (12, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (13, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (14, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (15, 1, 6, 'loja');
```

```
INSERT INTO loc VALUES (1, 2, 6, 'normal');
```

```
INSERT INTO loc VALUES (2, 2, 6, 'normal');
```

```
INSERT INTO loc VALUES (3, 2, 6, 'normal');
```

```
INSERT INTO loc VALUES (4, 2, 6, 'normal');
```

```
INSERT INTO loc VALUES (5, 2, 6, 'normal');
```

```
INSERT INTO loc VALUES (6, 2, 6, 'normal');
```

```
INSERT INTO loc VALUES (7, 2, 6, 'normal');
```

```
INSERT INTO loc VALUES (8, 2, 6, 'normal');
```

```
INSERT INTO loc VALUES (9, 2, 6, 'normal');
```

```
INSERT INTO loc VALUES (10, 2, 6, 'normal');
```

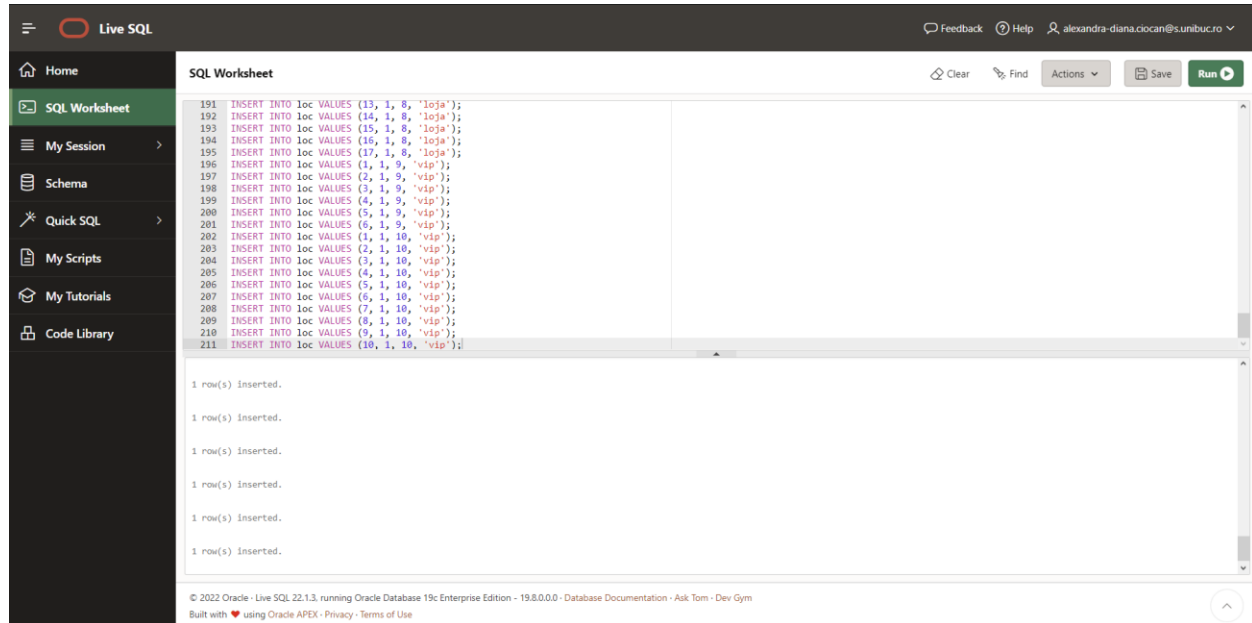
```
INSERT INTO loc VALUES (11, 2, 6, 'normal');
INSERT INTO loc VALUES (12, 2, 6, 'normal');
INSERT INTO loc VALUES (13, 2, 6, 'normal');
INSERT INTO loc VALUES (14, 2, 6, 'normal');
INSERT INTO loc VALUES (15, 2, 6, 'normal');
INSERT INTO loc VALUES (1, 3, 6, 'sala');
INSERT INTO loc VALUES (2, 3, 6, 'sala');
INSERT INTO loc VALUES (3, 3, 6, 'sala');
INSERT INTO loc VALUES (1, 1, 7, 'sala');
INSERT INTO loc VALUES (1, 2, 7, 'sala');
INSERT INTO loc VALUES (2, 2, 7, 'sala');
INSERT INTO loc VALUES (3, 2, 7, 'sala');
INSERT INTO loc VALUES (4, 2, 7, 'sala');
INSERT INTO loc VALUES (5, 2, 7, 'sala');
INSERT INTO loc VALUES (6, 2, 7, 'sala');
INSERT INTO loc VALUES (7, 2, 7, 'sala');
INSERT INTO loc VALUES (8, 2, 7, 'sala');
INSERT INTO loc VALUES (9, 2, 7, 'sala');
INSERT INTO loc VALUES (10, 2, 7, 'sala');
INSERT INTO loc VALUES (11, 2, 7, 'sala');
INSERT INTO loc VALUES (1, 1, 8, 'loja');
INSERT INTO loc VALUES (2, 1, 8, 'loja');
INSERT INTO loc VALUES (3, 1, 8, 'loja');
INSERT INTO loc VALUES (4, 1, 8, 'loja');
INSERT INTO loc VALUES (5, 1, 8, 'loja');
```

```
INSERT INTO loc VALUES (6, 1, 8, 'loja');
INSERT INTO loc VALUES (7, 1, 8, 'loja');
INSERT INTO loc VALUES (8, 1, 8, 'loja');
INSERT INTO loc VALUES (9, 1, 8, 'loja');
INSERT INTO loc VALUES (10, 1, 8, 'loja');
INSERT INTO loc VALUES (11, 1, 8, 'loja');
INSERT INTO loc VALUES (12, 1, 8, 'loja');
INSERT INTO loc VALUES (13, 1, 8, 'loja');
INSERT INTO loc VALUES (14, 1, 8, 'loja');
INSERT INTO loc VALUES (15, 1, 8, 'loja');
INSERT INTO loc VALUES (16, 1, 8, 'loja');
INSERT INTO loc VALUES (17, 1, 8, 'loja');
INSERT INTO loc VALUES (1, 1, 9, 'vip');
INSERT INTO loc VALUES (2, 1, 9, 'vip');
INSERT INTO loc VALUES (3, 1, 9, 'vip');
INSERT INTO loc VALUES (4, 1, 9, 'vip');
INSERT INTO loc VALUES (5, 1, 9, 'vip');
INSERT INTO loc VALUES (6, 1, 9, 'vip');
INSERT INTO loc VALUES (1, 1, 10, 'vip');
INSERT INTO loc VALUES (2, 1, 10, 'vip');
INSERT INTO loc VALUES (3, 1, 10, 'vip');
INSERT INTO loc VALUES (4, 1, 10, 'vip');
INSERT INTO loc VALUES (5, 1, 10, 'vip');
INSERT INTO loc VALUES (6, 1, 10, 'vip');
INSERT INTO loc VALUES (7, 1, 10, 'vip');
```

INSERT INTO loc VALUES (8, 1, 10, 'vip');

INSERT INTO loc VALUES (9, 1, 10, 'vip');

INSERT INTO loc VALUES (10, 1, 10, 'vip');



INSERT INTO eveniment VALUES (1, 9, 7, 1, 9, to_date('2022-12-23', 'yyyy-mm-dd'), '19:30', '21:00');

INSERT INTO eveniment VALUES (2, 5, 8, 1, 5, to_date('2022-07-01', 'yyyy-mm-dd'), '17:00', '19:30');

INSERT INTO eveniment VALUES (3, 7, 4, 2, 7, to_date('2022-08-06', 'yyyy-mm-dd'), '14:30', '17:30');

INSERT INTO eveniment VALUES (4, 25, 9, 1, 5, to_date('2022-08-13', 'yyyy-mm-dd'), '16:15', '19:00');

INSERT INTO eveniment VALUES (5, 7, 8, 1, 7, to_date('2022-10-28', 'yyyy-mm-dd'), '13:00', '15:00');

INSERT INTO eveniment VALUES (6, 10, 2, 2, 10, to_date('2022-07-03', 'yyyy-mm-dd'), '18:00', '20:30');

INSERT INTO eveniment VALUES (7, 10, 9, 1, 10, to_date('2022-08-24', 'yyyy-mm-dd'), '12:00', '16:00');

INSERT INTO eveniment VALUES (8, 12, 5, 1, 2, to_date('2022-06-29', 'yyyy-mm-dd'), '13:00', '15:00');

INSERT INTO eveniment VALUES (9, 20, 7, 2, 10, to_date('2022-10-04', 'yyyy-mm-dd'), '19:15', '21:15');

INSERT INTO eveniment VALUES (10, 21, 4, 2, 1, to_date('2022-08-11', 'yyyy-mm-dd'), '20:00', '22:30');

INSERT INTO eveniment VALUES (11, 14, 1, 3, 4, to_date('2022-10-24', 'yyyy-mm-dd'), '21:00', '23:00');

INSERT INTO eveniment VALUES (12, 15, 7, 2, 5, to_date('2022-10-25', 'yyyy-mm-dd'), '20:30', '23:30');

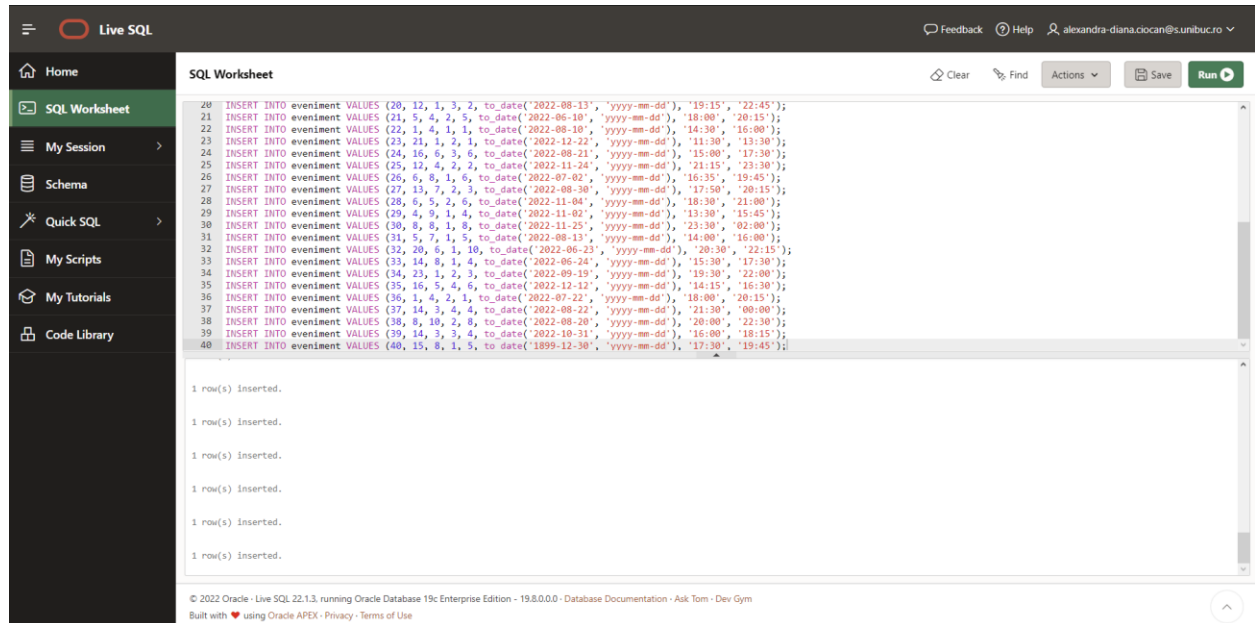
INSERT INTO eveniment VALUES (13, 15, 6, 2, 5, to_date('2022-10-16', 'yyyy-mm-dd'), '17:00', '20:00');

```
INSERT INTO eveniment VALUES (14, 1, 8, 1, 1, to_date('2022-09-21', 'yyyy-mm-dd'), '13:45', '16:00');
INSERT INTO eveniment VALUES (15, 19, 9, 1, 9, to_date('2022-10-24', 'yyyy-mm-dd'), '16:30', '18:30');
INSERT INTO eveniment VALUES (16, 2, 7, 2, 2, to_date('2022-12-04', 'yyyy-mm-dd'), '18:30', '21:00');
INSERT INTO eveniment VALUES (17, 20, 1, 1, 10, to_date('2022-06-19', 'yyyy-mm-dd'), '19:00', '21:30');
INSERT INTO eveniment VALUES (18, 15, 2, 2, 5, to_date('2022-09-15', 'yyyy-mm-dd'), '15:30', '17:00');
INSERT INTO eveniment VALUES (19, 22, 7, 1, 2, to_date('2022-09-27', 'yyyy-mm-dd'), '21:15', '23:30');
INSERT INTO eveniment VALUES (20, 12, 1, 3, 2, to_date('2022-08-13', 'yyyy-mm-dd'), '19:15', '22:45');
INSERT INTO eveniment VALUES (21, 5, 4, 2, 5, to_date('2022-06-10', 'yyyy-mm-dd'), '18:00', '20:15');
INSERT INTO eveniment VALUES (22, 1, 4, 1, 1, to_date('2022-08-10', 'yyyy-mm-dd'), '14:30', '16:00');
INSERT INTO eveniment VALUES (23, 21, 1, 2, 1, to_date('2022-12-22', 'yyyy-mm-dd'), '11:30', '13:30');
INSERT INTO eveniment VALUES (24, 16, 6, 3, 6, to_date('2022-08-21', 'yyyy-mm-dd'), '15:00', '17:30');
INSERT INTO eveniment VALUES (25, 12, 4, 2, 2, to_date('2022-11-24', 'yyyy-mm-dd'), '21:15', '23:30');
INSERT INTO eveniment VALUES (26, 6, 8, 1, 6, to_date('2022-07-02', 'yyyy-mm-dd'), '16:35', '19:45');
INSERT INTO eveniment VALUES (27, 13, 7, 2, 3, to_date('2022-08-30', 'yyyy-mm-dd'), '17:50', '20:15');
INSERT INTO eveniment VALUES (28, 6, 5, 2, 6, to_date('2022-11-04', 'yyyy-mm-dd'), '18:30', '21:00');
INSERT INTO eveniment VALUES (29, 4, 9, 1, 4, to_date('2022-11-02', 'yyyy-mm-dd'), '13:30', '15:45');
INSERT INTO eveniment VALUES (30, 8, 8, 1, 8, to_date('2022-11-25', 'yyyy-mm-dd'), '23:30', '02:00');
INSERT INTO eveniment VALUES (31, 5, 7, 1, 5, to_date('2022-08-13', 'yyyy-mm-dd'), '14:00', '16:00');
INSERT INTO eveniment VALUES (32, 20, 6, 1, 10, to_date('2022-06-23', 'yyyy-mm-dd'), '20:30', '22:15');
INSERT INTO eveniment VALUES (33, 14, 8, 1, 4, to_date('2022-06-24', 'yyyy-mm-dd'), '15:30', '17:30');
INSERT INTO eveniment VALUES (34, 23, 1, 2, 3, to_date('2022-09-19', 'yyyy-mm-dd'), '19:30', '22:00');
INSERT INTO eveniment VALUES (35, 16, 5, 4, 6, to_date('2022-12-12', 'yyyy-mm-dd'), '14:15', '16:30');
INSERT INTO eveniment VALUES (36, 1, 4, 2, 1, to_date('2022-07-22', 'yyyy-mm-dd'), '18:00', '20:15');
INSERT INTO eveniment VALUES (37, 14, 3, 4, 4, to_date('2022-08-22', 'yyyy-mm-dd'), '21:30', '00:00');
INSERT INTO eveniment VALUES (38, 8, 10, 2, 8, to_date('2022-08-20', 'yyyy-mm-dd'), '20:00', '22:30');
```


Ciocan Alexandra-Diana, grupa 132

INSERT INTO eveniment VALUES (39, 14, 3, 3, 4, to_date('2022-10-31', 'yyyy-mm-dd'), '16:00', '18:15');

INSERT INTO eveniment VALUES (40, 15, 8, 1, 5, to_date('1899-12-30', 'yyyy-mm-dd'), '17:30', '19:45');



INSERT INTO utilizator VALUES (secv_utilizator.nextval, 'Lungu David', 'sdfsreds12313', 'lungudavid@gmail.com', '0753296131');

INSERT INTO utilizator VALUES (secv_utilizator.nextval, 'Birsan Constantin', 'wdsagreds3', 'birsanconstantin@hotmail.com', '0793884781');

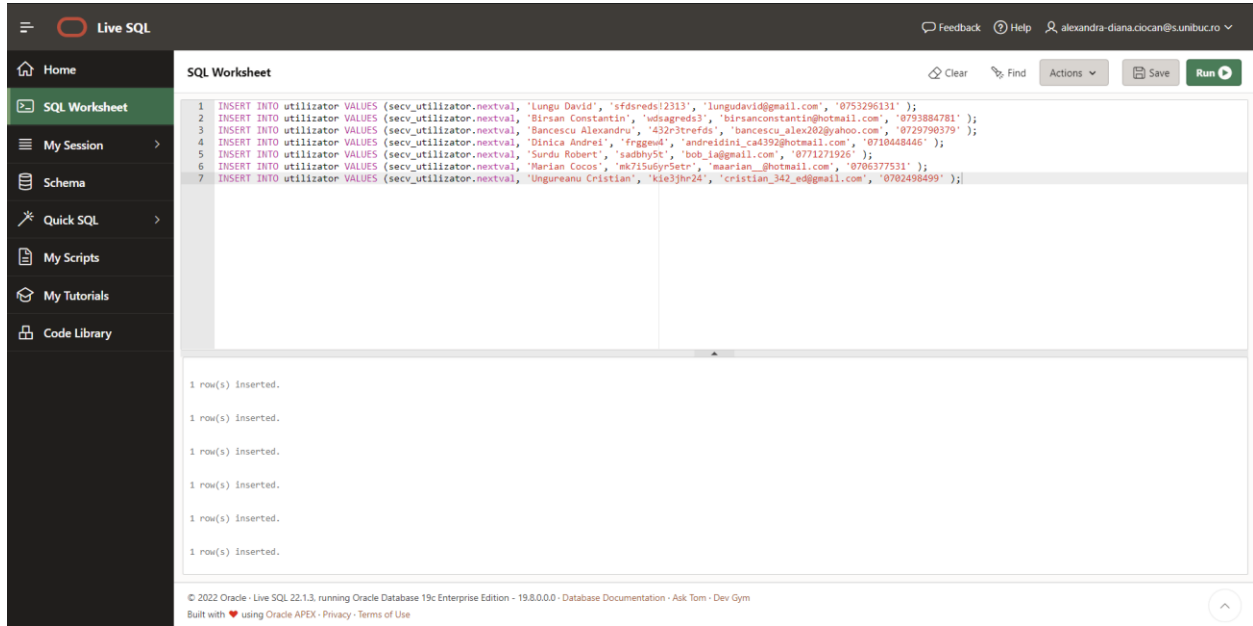
INSERT INTO utilizator VALUES (secv_utilizator.nextval, 'Bancescu Alexandru', '432r3trefds', 'bancescu_alex202@yahoo.com', '0729790379');

INSERT INTO utilizator VALUES (secv_utilizator.nextval, 'Dinica Andrei', 'frggew4', 'andreidini_ca4392@hotmail.com', '0710448446');

INSERT INTO utilizator VALUES (secv_utilizator.nextval, 'Surdu Robert', 'sadbhy5t', 'bob_ia@gmail.com', '0771271926');

INSERT INTO utilizator VALUES (secv_utilizator.nextval, 'Marian Cocos', 'mk7i5u6yr5etr', 'maarian__@hotmail.com', '0706377531');

INSERT INTO utilizator VALUES (secv_utilizator.nextval, 'Ungureanu Cristian', 'kie3jhr24', 'cristian_342_ed@gmail.com', '0702498499');



The screenshot shows the Live SQL interface. On the left is a sidebar with navigation options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains 7 lines of SQL code, all 'INSERT INTO UTILIZATOR' statements. Below the code, the execution results are shown as a list of 7 '1 row(s) inserted.' messages. At the bottom, a footer indicates the version is '© 2022 Oracle · Live SQL 22.1.3, running Oracle Database 19c Enterprise Edition - 19.0.0.0.0 - Database Documentation · Ask Tom · Dev Gym'.

```
1 INSERT INTO UTILIZATOR VALUES (sevc_utilizator.nextval, 'Lungu David', 'sdfsreds12313', 'lungudevid@gmail.com', '0753296131' );
2 INSERT INTO UTILIZATOR VALUES (sevc_utilizator.nextval, 'Birsan Constantin', 'udsagreds3', 'birsanconstantin@hotmail.com', '0793884781' );
3 INSERT INTO UTILIZATOR VALUES (sevc_utilizator.nextval, 'Bancescu Alexandru', '432r3trefds', 'bancescu_alex20@yahoo.com', '0729790379' );
4 INSERT INTO UTILIZATOR VALUES (sevc_utilizator.nextval, 'Dinica Andrei', 'frggewd', 'andreidini_ca4392@hotmail.com', '0710448446' );
5 INSERT INTO UTILIZATOR VALUES (sevc_utilizator.nextval, 'Sardu Robert', 'sadbhy5t', 'bob_la@gmail.com', '0771271926' );
6 INSERT INTO UTILIZATOR VALUES (sevc_utilizator.nextval, 'Marian Cocos', 'mk715u6yr5etr', 'maarian_@hotmail.com', '0706377531' );
7 INSERT INTO UTILIZATOR VALUES (sevc_utilizator.nextval, 'Ungureanu Cristian', 'kie3jhr24', 'cristian_342_ed@gmail.com', '0702498499' );
```

1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.

© 2022 Oracle · Live SQL 22.1.3, running Oracle Database 19c Enterprise Edition - 19.0.0.0.0 - Database Documentation · Ask Tom · Dev Gym
Built with ♥ using Oracle APEX · Privacy · Terms of Use

INSERT INTO rezervare VALUES (1, 4, to_date('2022-07-01', 'yyyy-mm-dd'), 2, 1);

INSERT INTO rezervare VALUES (2, 6, to_date('2022-07-03', 'yyyy-mm-dd'), 5, 1);

INSERT INTO rezervare VALUES (3, 3, to_date('2022-08-06', 'yyyy-mm-dd'), 3, 1);

INSERT INTO rezervare VALUES (4, 2, to_date('2022-08-24', 'yyyy-mm-dd'), 1, 0);

INSERT INTO rezervare VALUES (5, 6, to_date('2022-10-25', 'yyyy-mm-dd'), 2, 0);

INSERT INTO rezervare VALUES (6, 6, to_date('2022-09-21', 'yyyy-mm-dd'), 3, 1);

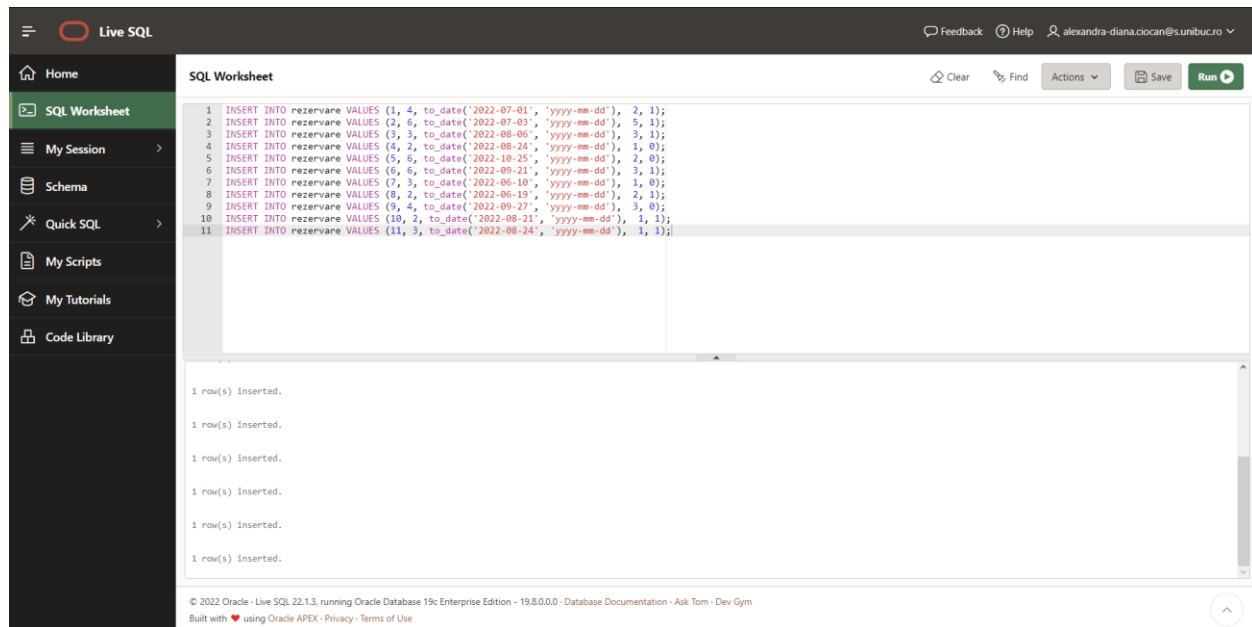
INSERT INTO rezervare VALUES (7, 3, to_date('2022-06-10', 'yyyy-mm-dd'), 1, 0);

INSERT INTO rezervare VALUES (8, 2, to_date('2022-06-19', 'yyyy-mm-dd'), 2, 1);

INSERT INTO rezervare VALUES (9, 4, to_date('2022-09-27', 'yyyy-mm-dd'), 3, 0);

INSERT INTO rezervare VALUES (10, 2, to_date('2022-08-21', 'yyyy-mm-dd'), 1, 1);

INSERT INTO rezervare VALUES (11, 3, to_date('2022-08-24', 'yyyy-mm-dd'), 1, 1);



The screenshot shows the Live SQL interface with the following SQL Worksheet content:

```
1 INSERT INTO rezervare VALUES (1, 4, to_date('2022-07-01', 'yyyy-mm-dd'), 2, 1);
2 INSERT INTO rezervare VALUES (2, 6, to_date('2022-07-03', 'yyyy-mm-dd'), 5, 1);
3 INSERT INTO rezervare VALUES (3, 3, to_date('2022-08-06', 'yyyy-mm-dd'), 3, 1);
4 INSERT INTO rezervare VALUES (4, 2, to_date('2022-08-24', 'yyyy-mm-dd'), 1, 0);
5 INSERT INTO rezervare VALUES (5, 6, to_date('2022-10-25', 'yyyy-mm-dd'), 2, 0);
6 INSERT INTO rezervare VALUES (6, 6, to_date('2022-09-21', 'yyyy-mm-dd'), 5, 1);
7 INSERT INTO rezervare VALUES (7, 3, to_date('2022-06-10', 'yyyy-mm-dd'), 1, 0);
8 INSERT INTO rezervare VALUES (8, 2, to_date('2022-06-19', 'yyyy-mm-dd'), 2, 1);
9 INSERT INTO rezervare VALUES (9, 4, to_date('2022-09-27', 'yyyy-mm-dd'), 3, 0);
10 INSERT INTO rezervare VALUES (10, 2, to_date('2022-08-21', 'yyyy-mm-dd'), 1, 1);
11 INSERT INTO rezervare VALUES (11, 3, to_date('2022-08-24', 'yyyy-mm-dd'), 1, 1);
```

The output shows 11 rows inserted successfully:

```
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
```

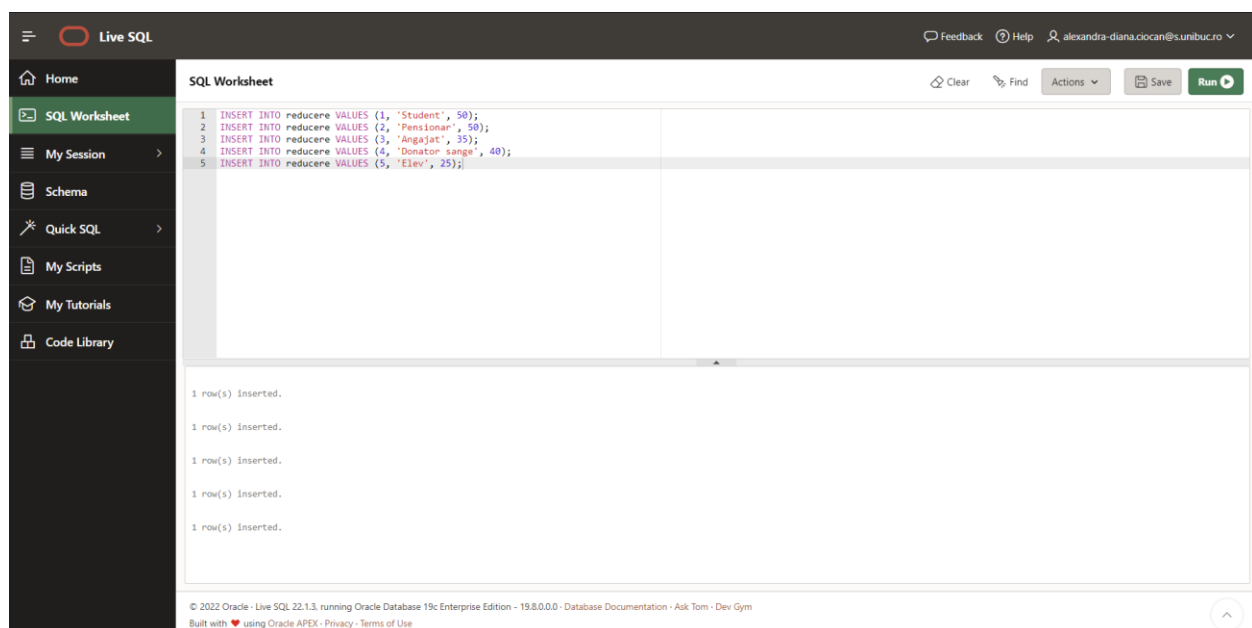
INSERT INTO reducere VALUES (1, 'Student', 50);

INSERT INTO reducere VALUES (2, 'Pensionar', 50);

INSERT INTO reducere VALUES (3, 'Angajat', 35);

INSERT INTO reducere VALUES (4, 'Donator sange', 40);

INSERT INTO reducere VALUES (5, 'Elev', 25);



The screenshot shows the Live SQL interface with the following SQL Worksheet content:

```
1 INSERT INTO reducere VALUES (1, 'Student', 50);
2 INSERT INTO reducere VALUES (2, 'Pensionar', 50);
3 INSERT INTO reducere VALUES (3, 'Angajat', 35);
4 INSERT INTO reducere VALUES (4, 'Donator sange', 40);
5 INSERT INTO reducere VALUES (5, 'Elev', 25);
```

The output shows 5 rows inserted successfully:

```
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
```

INSERT INTO plata VALUES (1, 90, to_date('2022-07-01', 'yyyy-mm-dd'), 1);

```
INSERT INTO plata VALUES (2, 150, to_date('2022-07-03', 'yyyy-mm-dd'), 2);
```

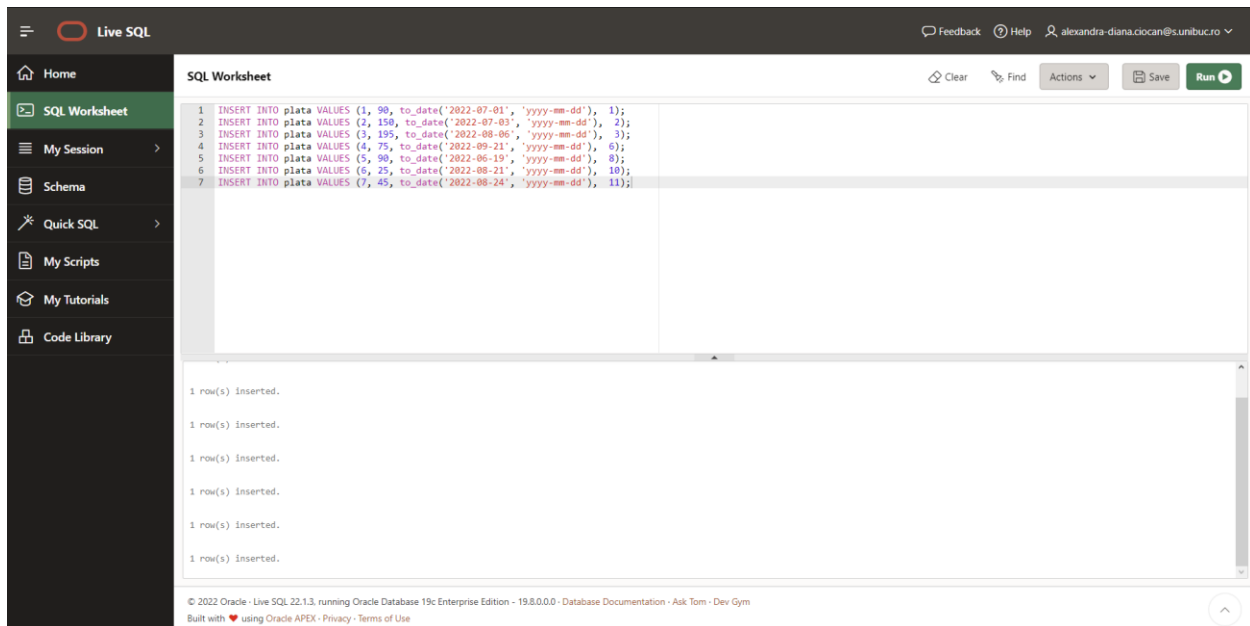
```
INSERT INTO plata VALUES (3, 195, to_date('2022-08-06', 'yyyy-mm-dd'), 3);
```

```
INSERT INTO plata VALUES (4, 75, to_date('2022-09-21', 'yyyy-mm-dd'), 6);
```

```
INSERT INTO plata VALUES (5, 90, to_date('2022-06-19', 'yyyy-mm-dd'), 8);
```

```
INSERT INTO plata VALUES (6, 25, to_date('2022-08-21', 'yyyy-mm-dd'), 10);
```

```
INSERT INTO plata VALUES (7, 45, to_date('2022-08-24', 'yyyy-mm-dd'), 11);
```



```
INSERT INTO implica VALUES (1, 1);
```

```
INSERT INTO implica VALUES (1, 2);
```

```
INSERT INTO implica VALUES (2, 1);
```

```
INSERT INTO implica VALUES (2, 2);
```

```
INSERT INTO implica VALUES (2, 3);
```

```
INSERT INTO implica VALUES (2, 4);
```

```
INSERT INTO implica VALUES (3, 4);
```

```
INSERT INTO implica VALUES (6, 1);
```

```
INSERT INTO implica VALUES (6, 5);
```

```
INSERT INTO implica VALUES (6, 3);
```

INSERT INTO implica VALUES (7, 1);

INSERT INTO implica VALUES (7, 3);

The screenshot shows the Oracle Live SQL interface. On the left is a navigation menu with options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains 12 lines of SQL code: 1 INSERT INTO implica VALUES (1, 1);, 2 INSERT INTO implica VALUES (1, 2);, 3 INSERT INTO implica VALUES (2, 1);, 4 INSERT INTO implica VALUES (2, 2);, 5 INSERT INTO implica VALUES (2, 3);, 6 INSERT INTO implica VALUES (2, 4);, 7 INSERT INTO implica VALUES (3, 4);, 8 INSERT INTO implica VALUES (6, 1);, 9 INSERT INTO implica VALUES (6, 5);, 10 INSERT INTO implica VALUES (6, 3);, 11 INSERT INTO implica VALUES (7, 1);, and 12 INSERT INTO implica VALUES (7, 3);. Below the code, the execution results are shown: '1 row(s) inserted.' for the first three statements, and '1 row(s) inserted.' for the last three statements. The interface also includes a top bar with 'Live SQL' and a user profile, and a bottom bar with copyright information.

INSERT INTO билет VALUES (1, 1, 1, 1, 7, NULL, 35, 0);

INSERT INTO билет VALUES (1, 2, 1, 1, 8, NULL, 30, 0);

INSERT INTO билет VALUES (2, 2, 2, 1, 8, NULL, 30, 0);

INSERT INTO билет VALUES (3, 2, 3, 1, 8, NULL, 30, 0);

INSERT INTO билет VALUES (4, 2, 4, 1, 8, NULL, 30, 0);

INSERT INTO билет VALUES (5, 2, 5, 1, 8, NULL, 30, 0);

INSERT INTO билет VALUES (1, 3, 1, 2, 4, 1, 45, 1);

INSERT INTO билет VALUES (2, 3, 2, 2, 4, 1, 45, 1);

INSERT INTO билет VALUES (3, 3, 3, 2, 4, NULL, 45, 0);

INSERT INTO билет VALUES (1, 4, 1, 1, 9, 3, 65, 1);

INSERT INTO билет VALUES (2, 4, 2, 1, 9, 3, 65, 1);

INSERT INTO билет VALUES (3, 4, 3, 1, 9, 3, 65, 1);

INSERT INTO билет VALUES (4, 4, 4, 1, 9, NULL, 65, 0);

```
INSERT INTO bilet VALUES (5, 4, 5, 1, 9, NULL, 65, 0);
INSERT INTO bilet VALUES (1, 5, 1, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (2, 5, 2, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (3, 5, 3, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (4, 5, 4, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (5, 5, 5, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (1, 6, 1, 2, 2, NULL, 35, 0);
INSERT INTO bilet VALUES (2, 6, 2, 2, 2, NULL, 35, 0);
INSERT INTO bilet VALUES (3, 6, 3, 2, 2, NULL, 35, 0);
INSERT INTO bilet VALUES (4, 6, 4, 2, 2, NULL, 35, 0);
INSERT INTO bilet VALUES (5, 6, 5, 2, 2, NULL, 35, 0);
INSERT INTO bilet VALUES (1, 7, 1, 1, 9, 2, 30, 1);
INSERT INTO bilet VALUES (2, 7, 2, 1, 9, 2, 30, 1);
INSERT INTO bilet VALUES (3, 7, 3, 1, 9, 2, 30, 1);
INSERT INTO bilet VALUES (4, 7, 4, 1, 9, 2, 30, 1);
INSERT INTO bilet VALUES (5, 7, 5, 1, 9, 2, 30, 1);
INSERT INTO bilet VALUES (1, 8, 1, 1, 5, 4, 45, 1);
INSERT INTO bilet VALUES (2, 8, 2, 1, 5, NULL, 45, 0);
INSERT INTO bilet VALUES (3, 8, 3, 1, 5, 11, 45, 1);
INSERT INTO bilet VALUES (4, 8, 4, 1, 5, NULL, 45, 0);
INSERT INTO bilet VALUES (1, 9, 1, 2, 7, NULL, 65, 0);
INSERT INTO bilet VALUES (2, 9, 2, 2, 7, NULL, 65, 0);
INSERT INTO bilet VALUES (3, 9, 3, 2, 7, NULL, 65, 0);
INSERT INTO bilet VALUES (4, 9, 4, 2, 7, NULL, 65, 0);
INSERT INTO bilet VALUES (5, 9, 5, 2, 7, NULL, 65, 0);
```

```
INSERT INTO bilet VALUES (1, 10, 1, 2, 4, NULL, 25, 0);
INSERT INTO bilet VALUES (2, 10, 2, 2, 4, NULL, 25, 0);
INSERT INTO bilet VALUES (3, 10, 3, 2, 4, NULL, 25, 0);
INSERT INTO bilet VALUES (1, 11, 1, 3, 1, NULL, 35, 0);
INSERT INTO bilet VALUES (2, 11, 2, 3, 1, NULL, 35, 0);
INSERT INTO bilet VALUES (3, 11, 3, 3, 1, NULL, 35, 0);
INSERT INTO bilet VALUES (4, 11, 4, 3, 1, NULL, 35, 0);
INSERT INTO bilet VALUES (5, 11, 5, 3, 1, NULL, 35, 0);
INSERT INTO bilet VALUES (1, 12, 1, 2, 7, NULL, 30, 0);
INSERT INTO bilet VALUES (2, 12, 2, 2, 7, NULL, 30, 0);
INSERT INTO bilet VALUES (3, 12, 3, 2, 7, NULL, 30, 0);
INSERT INTO bilet VALUES (4, 12, 4, 2, 7, NULL, 30, 0);
INSERT INTO bilet VALUES (5, 12, 5, 2, 7, NULL, 30, 0);
INSERT INTO bilet VALUES (1, 13, 1, 2, 6, 5, 45, 1);
INSERT INTO bilet VALUES (2, 13, 2, 2, 6, 5, 45, 1);
INSERT INTO bilet VALUES (3, 13, 3, 2, 6, NULL, 45, 0);
INSERT INTO bilet VALUES (4, 13, 4, 2, 6, NULL, 45, 0);
INSERT INTO bilet VALUES (5, 13, 5, 2, 6, NULL, 45, 0);
INSERT INTO bilet VALUES (1, 14, 1, 1, 8, NULL, 65, 0);
INSERT INTO bilet VALUES (2, 14, 2, 1, 8, NULL, 65, 0);
INSERT INTO bilet VALUES (3, 14, 3, 1, 8, NULL, 65, 0);
INSERT INTO bilet VALUES (4, 14, 4, 1, 8, NULL, 65, 0);
INSERT INTO bilet VALUES (5, 14, 5, 1, 8, NULL, 65, 0);
INSERT INTO bilet VALUES (1, 15, 1, 1, 9, 6, 25, 1);
INSERT INTO bilet VALUES (2, 15, 2, 1, 9, 6, 25, 1);
```

```
INSERT INTO bilet VALUES (3, 15, 3, 1, 9, 6, 25, 1);
INSERT INTO bilet VALUES (4, 15, 4, 1, 9, NULL, 25, 0);
INSERT INTO bilet VALUES (5, 15, 5, 1, 9, NULL, 25, 0);
INSERT INTO bilet VALUES (1, 16, 1, 2, 7, NULL, 35, 0);
INSERT INTO bilet VALUES (2, 16, 2, 2, 7, NULL, 35, 0);
INSERT INTO bilet VALUES (3, 16, 3, 2, 7, NULL, 35, 0);
INSERT INTO bilet VALUES (4, 16, 4, 2, 7, NULL, 35, 0);
INSERT INTO bilet VALUES (5, 16, 5, 2, 7, NULL, 35, 0);
INSERT INTO bilet VALUES (1, 17, 1, 1, 1, NULL, 30, 0);
INSERT INTO bilet VALUES (2, 17, 2, 1, 1, NULL, 30, 0);
INSERT INTO bilet VALUES (3, 17, 3, 1, 1, NULL, 30, 0);
INSERT INTO bilet VALUES (4, 17, 4, 1, 1, NULL, 30, 0);
INSERT INTO bilet VALUES (5, 17, 5, 1, 1, NULL, 30, 0);
INSERT INTO bilet VALUES (1, 18, 1, 2, 2, NULL, 45, 0);
INSERT INTO bilet VALUES (2, 18, 2, 2, 2, 8, 45, 1);
INSERT INTO bilet VALUES (3, 18, 3, 2, 2, 8, 45, 1);
INSERT INTO bilet VALUES (4, 18, 4, 2, 2, NULL, 45, 0);
INSERT INTO bilet VALUES (5, 18, 5, 2, 2, NULL, 45, 0);
INSERT INTO bilet VALUES (1, 19, 1, 1, 7, NULL, 65, 0);
INSERT INTO bilet VALUES (1, 20, 1, 3, 1, NULL, 25, 0);
INSERT INTO bilet VALUES (2, 20, 2, 3, 1, NULL, 25, 0);
INSERT INTO bilet VALUES (3, 20, 3, 3, 1, 9, 25, 1);
INSERT INTO bilet VALUES (4, 20, 4, 3, 1, 9, 25, 1);
INSERT INTO bilet VALUES (5, 20, 5, 3, 1, 9, 25, 1);
INSERT INTO bilet VALUES (1, 21, 1, 2, 4, NULL, 35, 0);
```



```
INSERT INTO bilet VALUES (2, 21, 2, 2, 4, NULL, 35, 0);
INSERT INTO bilet VALUES (3, 21, 3, 2, 4, NULL, 35, 0);
INSERT INTO bilet VALUES (1, 22, 1, 1, 4, 7, 30, 1);
INSERT INTO bilet VALUES (2, 22, 2, 1, 4, NULL, 30, 0);
INSERT INTO bilet VALUES (3, 22, 3, 1, 4, NULL, 30, 0);
INSERT INTO bilet VALUES (4, 22, 4, 1, 4, NULL, 30, 0);
INSERT INTO bilet VALUES (5, 22, 5, 1, 4, NULL, 30, 0);
INSERT INTO bilet VALUES (1, 23, 1, 2, 1, NULL, 45, 0);
INSERT INTO bilet VALUES (2, 23, 2, 2, 1, NULL, 45, 0);
INSERT INTO bilet VALUES (3, 23, 3, 2, 1, NULL, 45, 0);
INSERT INTO bilet VALUES (4, 23, 4, 2, 1, NULL, 45, 0);
INSERT INTO bilet VALUES (5, 23, 5, 2, 1, NULL, 45, 0);
INSERT INTO bilet VALUES (1, 24, 1, 3, 6, NULL, 65, 0);
INSERT INTO bilet VALUES (2, 24, 2, 3, 6, NULL, 65, 0);
INSERT INTO bilet VALUES (3, 24, 3, 3, 6, NULL, 65, 0);
INSERT INTO bilet VALUES (1, 25, 1, 2, 4, 10, 25, 1);
INSERT INTO bilet VALUES (2, 25, 2, 2, 4, NULL, 25, 0);
INSERT INTO bilet VALUES (3, 25, 3, 2, 4, NULL, 25, 0);
INSERT INTO bilet VALUES (1, 26, 1, 1, 8, NULL, 35, 0);
INSERT INTO bilet VALUES (2, 26, 2, 1, 8, NULL, 35, 0);
INSERT INTO bilet VALUES (3, 26, 3, 1, 8, NULL, 35, 0);
INSERT INTO bilet VALUES (4, 26, 4, 1, 8, NULL, 35, 0);
INSERT INTO bilet VALUES (5, 26, 5, 1, 8, NULL, 35, 0);
INSERT INTO bilet VALUES (1, 27, 1, 2, 7, NULL, 30, 0);
INSERT INTO bilet VALUES (2, 27, 2, 2, 7, NULL, 30, 0);
```

```
INSERT INTO bilet VALUES (3, 27, 3, 2, 7, NULL, 30, 0);
INSERT INTO bilet VALUES (4, 27, 4, 2, 7, NULL, 30, 0);
INSERT INTO bilet VALUES (5, 27, 5, 2, 7, NULL, 30, 0);
INSERT INTO bilet VALUES (1, 28, 1, 2, 5, NULL, 45, 0);
INSERT INTO bilet VALUES (2, 28, 2, 2, 5, NULL, 45, 0);
INSERT INTO bilet VALUES (3, 28, 3, 2, 5, NULL, 45, 0);
INSERT INTO bilet VALUES (1, 29, 1, 1, 9, NULL, 65, 0);
INSERT INTO bilet VALUES (2, 29, 2, 1, 9, NULL, 65, 0);
INSERT INTO bilet VALUES (3, 29, 3, 1, 9, NULL, 65, 0);
INSERT INTO bilet VALUES (4, 29, 4, 1, 9, NULL, 65, 0);
INSERT INTO bilet VALUES (5, 29, 5, 1, 9, NULL, 65, 0);
INSERT INTO bilet VALUES (1, 30, 1, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (2, 30, 2, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (3, 30, 3, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (4, 30, 4, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (5, 30, 5, 1, 8, NULL, 25, 0);
INSERT INTO bilet VALUES (1, 31, 1, 1, 7, NULL, 35, 0);
INSERT INTO bilet VALUES (1, 32, 1, 1, 6, NULL, 30, 0);
INSERT INTO bilet VALUES (2, 32, 2, 1, 6, NULL, 30, 0);
INSERT INTO bilet VALUES (3, 32, 3, 1, 6, NULL, 30, 0);
INSERT INTO bilet VALUES (4, 32, 4, 1, 6, NULL, 30, 0);
INSERT INTO bilet VALUES (5, 32, 5, 1, 6, NULL, 30, 0);
INSERT INTO bilet VALUES (1, 33, 1, 1, 8, NULL, 45, 0);
INSERT INTO bilet VALUES (2, 33, 2, 1, 8, NULL, 45, 0);
INSERT INTO bilet VALUES (3, 33, 3, 1, 8, NULL, 45, 0);
```

```
INSERT INTO bilet VALUES (4, 33, 4, 1, 8, NULL, 45, 0);
INSERT INTO bilet VALUES (5, 33, 5, 1, 8, NULL, 45, 0);
INSERT INTO bilet VALUES (1, 34, 1, 2, 1, NULL, 65, 0);
INSERT INTO bilet VALUES (2, 34, 2, 2, 1, NULL, 65, 0);
INSERT INTO bilet VALUES (3, 34, 3, 2, 1, NULL, 65, 0);
INSERT INTO bilet VALUES (4, 34, 4, 2, 1, NULL, 65, 0);
INSERT INTO bilet VALUES (5, 34, 5, 2, 1, NULL, 65, 0);
INSERT INTO bilet VALUES (1, 35, 1, 4, 5, NULL, 25, 0);
INSERT INTO bilet VALUES (2, 35, 2, 4, 5, NULL, 25, 0);
INSERT INTO bilet VALUES (3, 35, 3, 4, 5, NULL, 25, 0);
INSERT INTO bilet VALUES (4, 35, 4, 4, 5, NULL, 25, 0);
INSERT INTO bilet VALUES (5, 35, 5, 4, 5, NULL, 25, 0);
INSERT INTO bilet VALUES (1, 36, 1, 2, 4, NULL, 35, 0);
INSERT INTO bilet VALUES (2, 36, 2, 2, 4, NULL, 35, 0);
INSERT INTO bilet VALUES (3, 36, 3, 2, 4, NULL, 35, 0);
INSERT INTO bilet VALUES (1, 37, 1, 4, 3, NULL, 30, 0);
INSERT INTO bilet VALUES (2, 37, 2, 4, 3, NULL, 30, 0);
INSERT INTO bilet VALUES (3, 37, 3, 4, 3, NULL, 30, 0);
INSERT INTO bilet VALUES (4, 37, 4, 4, 3, NULL, 30, 0);
INSERT INTO bilet VALUES (5, 37, 5, 4, 3, NULL, 30, 0);
INSERT INTO bilet VALUES (1, 39, 1, 3, 3, NULL, 65, 0);
INSERT INTO bilet VALUES (2, 39, 2, 3, 3, NULL, 65, 0);
INSERT INTO bilet VALUES (3, 39, 3, 3, 3, NULL, 65, 0);
INSERT INTO bilet VALUES (4, 39, 4, 3, 3, NULL, 65, 0);
INSERT INTO bilet VALUES (5, 39, 5, 3, 3, NULL, 65, 0);
```

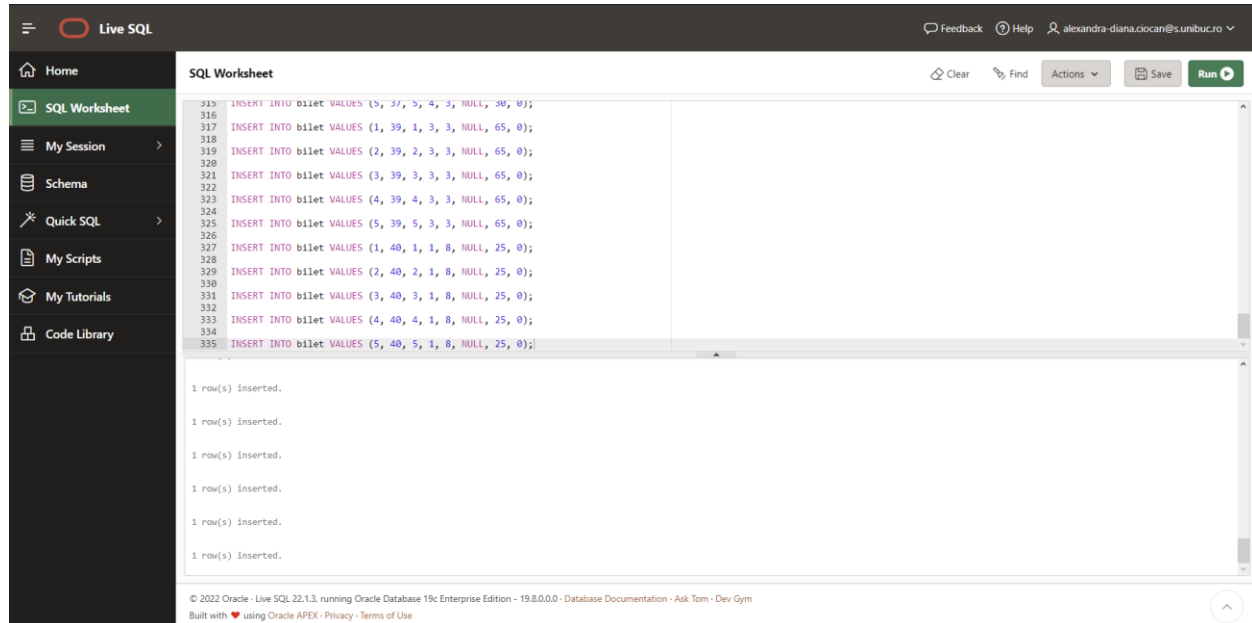
INSERT INTO bilet VALUES (1, 40, 1, 1, 8, NULL, 25, 0);

INSERT INTO bilet VALUES (2, 40, 2, 1, 8, NULL, 25, 0);

INSERT INTO bilet VALUES (3, 40, 3, 1, 8, NULL, 25, 0);

INSERT INTO bilet VALUES (4, 40, 4, 1, 8, NULL, 25, 0);

INSERT INTO bilet VALUES (5, 40, 5, 1, 8, NULL, 25, 0);



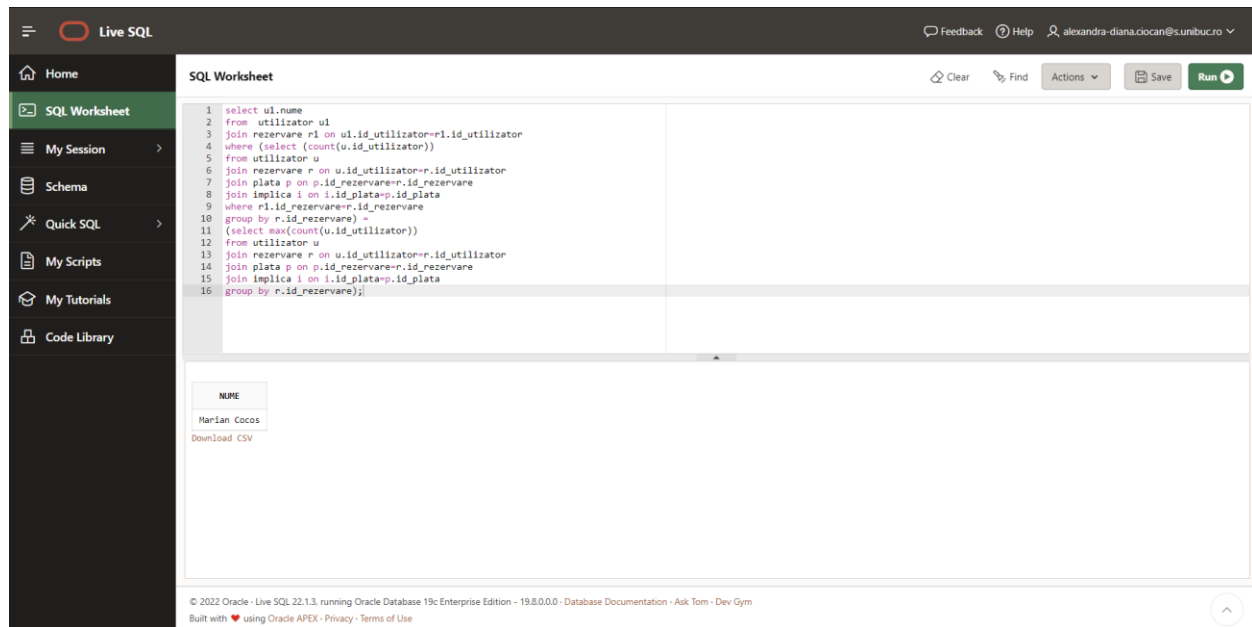
12. Formulați în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

- *operație join pe cel puțin 4 tabele*
- *filtrare la nivel de linii*
- *subcereri sincronizate în care intervin cel puțin 3 tabele*
- *subcereri nesincronizate în care intervin cel puțin 3 tabele*
- *grupări de date, funcții grup, filtrare la nivel de grupuri*
- *ordonări*
- *utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a funcțiilor NVL și DECODE, a cel puțin unei expresii CASE*
- *utilizarea a cel puțin 1 bloc de cerere (clauza WITH)*

Observație: Într-o cerere se vor regăsi mai multe elemente dintre cele enumerate mai sus, astfel încât cele 5 cereri să le cuprindă pe toate.

1. Să se afișeze numele utilizatorului care a plasat comanda cu cel mai mare număr de reduceri.
(operație join pe cel puțin 4 tabele, filtrare la nivel de linii, grupări de date, funcții grup, filtrare la nivel de grupuri, subcereri sincronizate în care intervin cel puțin 3 tabele)

```
select u1.nume
from utilizator u1
join rezervare r1 on u1.id_utilizator=r1.id_utilizator
where (select (count(u.id_utilizator))
from utilizator u
join rezervare r on u.id_utilizator=r.id_utilizator
join plata p on p.id_rezervare=r.id_rezervare
join implica i on i.id_plata=p.id_plata
where r1.id_rezervare=r.id_rezervare
group by r.id_rezervare) =
(select max(count(u.id_utilizator))
from utilizator u
join rezervare r on u.id_utilizator=r.id_utilizator
join plata p on p.id_rezervare=r.id_rezervare
join implica i on i.id_plata=p.id_plata
group by r.id_rezervare);
```



2. Afișați id-ul, statusul(rezervat/disponibil) și data spectacolului tuturor biletelor puse la dispoziție pentru spectacole care conțin litera "o" și au loc în Baia Mare, în ordine cronologică a evenimentelor. (ordonări, 2 funcții pe șiruri de caractere, DECODE, subcereri nesincronizate în care intervin cel puțin 3 tabele)

```
select id_bilet, decode(b.stare, 0, 'rezervat', 1, 'disponibil') "STATUS", e.data_eveniment
```

```
from bilet b
```

```
join eveniment e on b.id_eveniment=e.id_eveniment
```

```
join spectacol s on e.id_spectacol=s.id_spectacol
```

```
where lower(s.ume) like '%o%' and e.id_eveniment in (
```

```
select e1.id_eveniment
```

```
from eveniment e1
```

```
join locatie l on e1.id_locatie=l.id_locatie
```

```
join oras o on o.id_oras=l.id_locatie
```

```
where upper(o.ume)='BAIA MARE'
```

```
)
```

```
order by e.data_eveniment;
```

The screenshot shows the Live SQL interface. On the left is a sidebar with navigation options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains an SQL query. Below the query, the results are displayed in a table.

```

1 select id_bilet, decode(b.stare, 0, 'rezervat', 1, 'disponibil') "STATUS", e.data_eveniment
2 from bilet b
3 join eveniment e on b.id_eveniment=e.id_eveniment
4 join spectacol s on e.id_spectacol=s.id_spectacol
5 where lower(s.name) like '%ol%' and e.id_eveniment in (
6 select e1.id_eveniment
7 from eveniment e1
8 join locatie l on e1.id_locatie=l.id_locatie
9 join oras o on e1.id_oras=o.id_oras
10 where upper(o.name)='BAIA MARE'
11 )
12 order by e.data_eveniment;
  
```

ID_BILET	STATUS	DATA_EVENIMENT
1	rezervat	30-DEC-99
2	rezervat	30-DEC-99
3	rezervat	30-DEC-99
4	rezervat	30-DEC-99
5	rezervat	30-DEC-99
5	rezervat	01-JUL-22
4	rezervat	01-JUL-22
3	rezervat	01-JUL-22

© 2022 Oracle - Live SQL 22.1.3, running Oracle Database 19c Enterprise Edition - 19.0.0.0 - Database Documentation - Ask Tom - Dev Gym
Built with ♥ using Oracle APEX - Privacy - Terms of Use

3. Afisati în ordine crescătoare a sumelor, numele fiecărui utilizator și suma cheltuită pe bilete la spectacole, împreună cu mesajul "Utilizator fidel" în cazul în care a efectuat cel puțin o plată, sau "Utilizator obișnuit" în cazul în care nu a efectuat nicio plată.

```
select u.ume, nvl(sum(p.suma),'0'), case when nvl(sum(p.suma),'0')=0 then 'Utilizator normal' else 'Utilizator fidel' end "TIP UTILIZATOR"
```

```
from utilizator u
```

```
left join rezervare r on r.id_utilizator=u.id_utilizator
```

```
left join plata p on p.id_rezervare=r.id_rezervare
```

```
group by u.ume;
```

The screenshot shows the Live SQL interface. On the left is a sidebar with navigation options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains a query editor with the following SQL code:

```
1 select u.ume, nvl(sum(p.suma),'0'), case when nvl(sum(p.suma),'0')=0 then 'Utilizator normal' else 'Utilizator fidel' end "TIP UTILIZATOR"
2 from utilizator u
3 left join rezervare r on r.id_utilizator=u.id_utilizator
4 left join plata p on p.id_rezervare=r.id_rezervare
5 group by u.ume;
```

Below the editor, the results are displayed in a table:

NUME	NVL(SUM(P.SUMA),'0')	TIP UTILIZATOR
Marian Cocos	225	Utilizator fidel
Birsan Constantin	115	Utilizator fidel
Ungureanu Cristian	0	Utilizator normal
Dinica Andrei	90	Utilizator fidel
Lungu David	0	Utilizator normal
Sundu Robert	0	Utilizator normal
Bancescu Alexandru	240	Utilizator fidel

Below the table, it says 'Download CSV' and '7 rows collected'. At the bottom, there is a footer with copyright information: '© 2022 Oracle - Live SQL 22.1.3, running Oracle Database 19c Enterprise Edition - 19.0.0.0 - Database Documentation - Ask Tom - Dev Gym' and 'Built with ❤️ using Oracle APEX - Privacy - Terms of Use'.

4. Afişaţi reducerile care au fost folosite vreodată, de la cele mai recent folosite la cele mai departe în timp folosite.

```
select r.id_reducere, max(p.data_plata) "ULTIMA UTILIZARE"
```

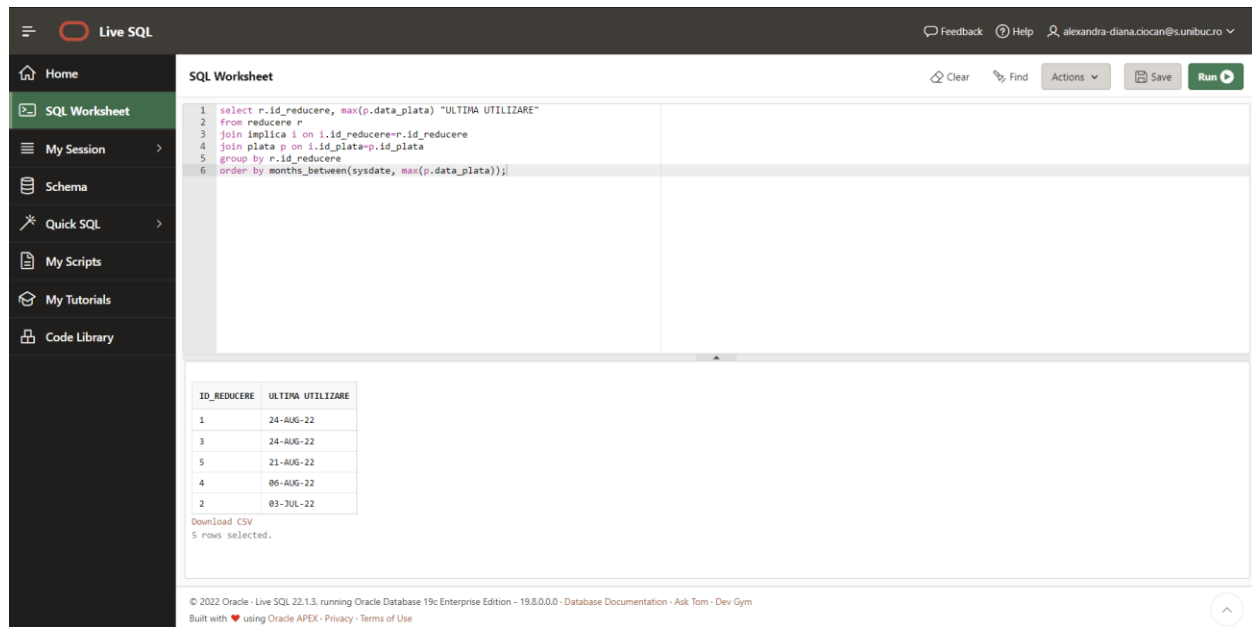
```
from reducere r
```

```
join implica i on i.id_reducere=r.id_reducere
```

```
join plata p on i.id_plata=p.id_plata
```

```
group by r.id_reducere
```

```
order by months_between(sysdate, max(p.data_plata));
```

The screenshot shows the Live SQL interface. On the left is a sidebar with navigation links: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains a SQL query:

```
1 select r.id_reducere, max(p.data_plata) "ULTIMA UTILIZARE"
2 from reducere r
3 join implica i on i.id_reducere=r.id_reducere
4 join plata p on i.id_plata=p.id_plata
5 group by r.id_reducere
6 order by months_between(sysdate, max(p.data_plata));
```

Below the query, the results are displayed in a table:

ID_REUCERE	ULTIMA UTILIZARE
1	24-AUG-22
3	24-AUG-22
5	21-AUG-22
4	06-AUG-22
2	03-JUL-22

Below the table, it says 'Download CSV' and '5 rows selected.' At the bottom, there is a footer with copyright information: '© 2022 Oracle - Live SQL 22.1.3, running Oracle Database 19c Enterprise Edition - 19.0.0.0 - Database Documentation - Ask Tom - Dev Gym' and 'Built with ♥ using Oracle APEX - Privacy - Terms of Use'.

5. Afișați suma cheltuită în medie de un utilizator pe platforma.

with suma_plata

as

(

select nvl(sum(p.suma),'0') suma_totala

from utilizator u

left join rezervare r on r.id_utilizator=u.id_utilizator

left join plata p on p.id_rezervare=r.id_rezervare

group by u.num

)

select

round(avg(suma_totala),2) "MEDIE SUMA CHELTUITA"

from suma_plata;

The screenshot shows the Live SQL interface with a sidebar on the left containing links to Home, SQL Worksheet, My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains a SQL query. Below the query, the result is displayed as a table with one row and one column.

```
1 with suma_plata
2 as
3 (
4   select nvl(sum(p.suma),'0') suma_totala
5   from utilizator u
6   left join rezervare r on r.id_utilizator=u.id_utilizator
7   left join plata p on p.id_rezervare=r.id_rezervare
8   group by u.ume
9 )
10 select
11 round(avg(suma_totala),2) "MEDIE SUMA CHELTUITA"
12 from suma_plata;
```

MEDIE SUMA CHELTUITA
95.71

Download CSV

© 2022 Oracle - Live SQL 22.1.3, running Oracle Database 19c Enterprise Edition - 19.0.0.0 - Database Documentation - Ask Tom - Dev Gym
Built with ♥ using Oracle APEX - Privacy - Terms of Use

13. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri.

Toate biletele de la evenimente ce au loc dupa data de 20 august 2022 s-au vândut:

update bilet

set stare=1

where id_eveniment in (select id_eveniment from eveniment where data_eveniment>to_date('2022-08-20', 'yyyy-mm-dd'));

The screenshot shows the Live SQL interface with the same sidebar as the previous image. The main area is titled 'SQL Worksheet' and contains an SQL update query. Below the query, the result is displayed as a message indicating the number of rows updated.

```
1 update bilet
2 set stare=1
3 where id_eveniment in (select id_eveniment from eveniment where data_eveniment>to_date('2022-08-20', 'yyyy-mm-dd'));
```

101 row(s) updated.

© 2022 Oracle - Live SQL 22.1.3, running Oracle Database 19c Enterprise Edition - 19.0.0.0 - Database Documentation - Ask Tom - Dev Gym
Built with ♥ using Oracle APEX - Privacy - Terms of Use

Utilizatorii care nu au plasat nicio rezervare sunt eliminati din baza de date:

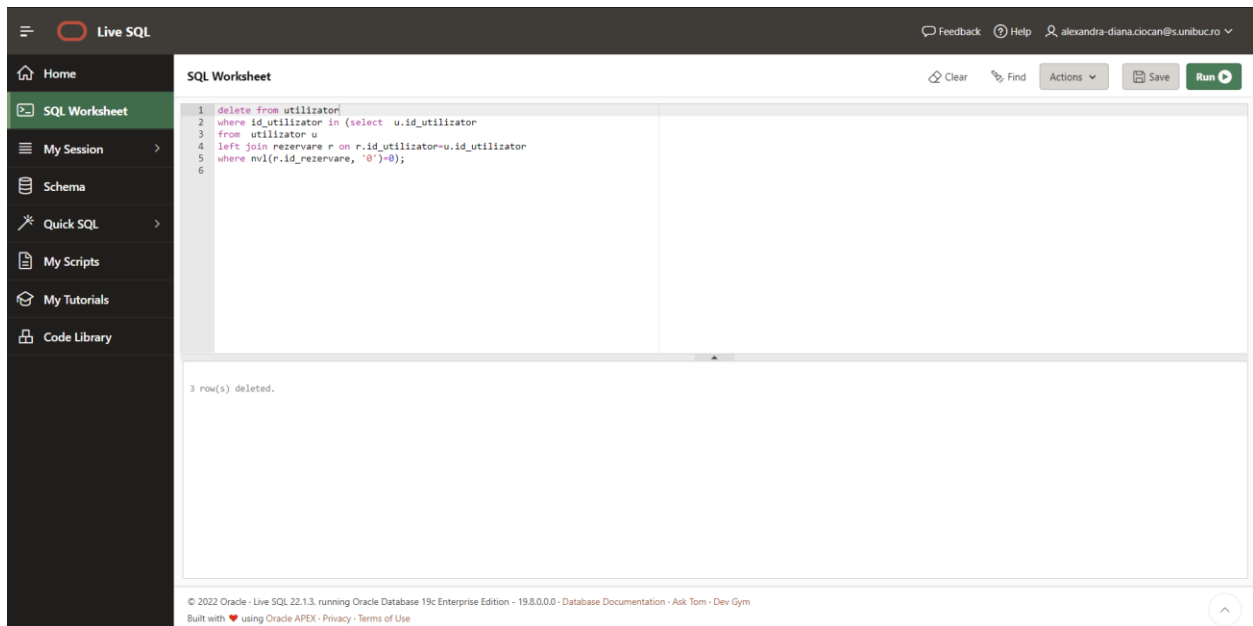
delete from utilizator

where id_utilizator in (select u.id_utilizator

from utilizator u

left join rezervare r on r.id_utilizator=u.id_utilizator

where nvl(r.id_rezervare, '0')=0);

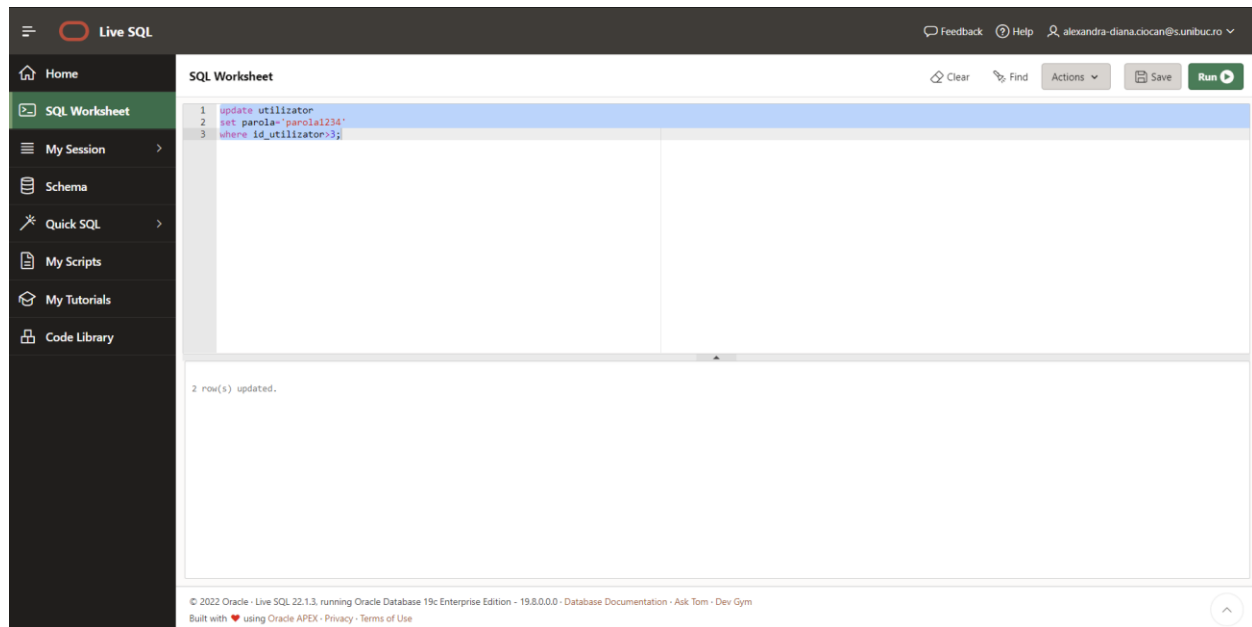


Toți utilizatorii înregistrați după al 3-lea utilizator și-au uitat parola, vor să o reseteze:

update utilizator

set parola='parola1234'

where id_utilizator>3;



18. a. Realizarea normalizării BCNF, FN4, FN5.

Modelul de bază de date implementat de mine este:

SPECTACOL(id_spectacol#, tip, nume)

FILM(id_spectacol#, durata, limba, data_lansarii, tara, gen)

TEATRU(id_spectacol#, autor, regizor, scenograf)

CONCERT(id_spectacol#, artist, invitati, gen)

ORGANIZATOR(id_organizator#, nume, email, nr_telefon)

EVENIMENT(id_eveniment#, id_spectacol, id_locatie, id_sala, id_organizator, data_eveniment, ora_start, ora_stop)

SALA(id_sala#, id_locatie#, nume, nr_locuri)

LOCATIE(id_locatie#, id_oras, nume, nr_sali, adresa)

ORAS(id_oras#, nume, localitate)

REZERVARE(id_rezervare#, id_utilizator, data_rezervare, nr_bilete, stare)

BILET(id_bilet#, id_eveniment#, id_loc, id_sala, id_locatie, id_rezervare, pret, stare)

LOC(id_loc#, id_sala#, id_locatie#, tip)

UTILIZATOR(id_utilizator#, nume, parola, email, nr_telefon)

PLATA(id_plata#, suma, data_plata, id_rezervare)

IMPLICA(id_plata#, id_reducere#)

REDUCERE(id_reducere#, nume, valoare)

Acest model nu se află în BCNF deoarece attributele id_sala și id_locatie ale entității BILET, chei externe ce fac legătura cu tabelul LOC creează redundanță, noi știind din tabelul EVENIMENT ce id_sala și id_locatie sunt corespondente cheii primare id_eveniment. Pentru a duce tabelul în BCNF putem renunța la id_sala și id_locatie ca attribute ale entității BILET:

SPECTACOL(id_spectacol#, tip, nume)

FILM(id_spectacol#, durata, limba, data_lansarii, tara, gen)

TEATRU(id_spectacol#, autor, regizor, scenograf)

CONCERT(id_spectacol#, artist, invitati, gen)

ORGANIZATOR(id_organizator#, nume, email, nr_telefon)

EVENIMENT(id_eveniment#, id_spectacol, id_locatie, id_sala, id_organizator, data_eveniment, ora_start, ora_stop)

SALA(id_sala#, id_locatie#, nume, nr_locuri)

LOCATIE(id_locatie#, id_oras, nume, nr_sali, adresa)

ORAS(id_oras#, nume, localitate)

REZERVARE(id_rezervare#, id_utilizator, data_rezervare, nr_bilete, stare)

BILET(id_bilet#, id_eveniment#, id_rezervare, pret, stare)

LOC(id_loc#, id_sala#, id_locatie#, tip)

UTILIZATOR(id_utilizator#, nume, parola, email, nr_telefon)

PLATA(id_plata#, suma, data_plata, id_rezervare)

IMPLICA(id_plata#, id_reducere#)

REDUCERE(id_reducere#, nume, valoare)