**Documentatie proiect Baze de Date**

Am ales ca tema Vanzarea/Gestiunea medicamentelor intr-o farmacie. Aplicatia are urmatoarele functionalitati: adaugare de noi inregistrare, modificare inregistrare curenta, stergere inregistrare curenta, refresh-ul tabelelor dupa fiecare operatiune facuta( se face automat in interfata), calcularea unor campuri in interfata si inserarea lor in baza de date.

**Baza de date cuprinde urmatoarele tabele:**

-farmacie cu cheia primara FarmacieID (am presupus ca pot exista mai multe farmacii cu diferite adrese)

-angajati cu cheia primara AngajatID ( autoincrementare), camp unic CNP si cheia externa farmacieID care arata la care farmacie este angajat (prima relatie 1:N)

-salarii cu cheia primara SalariuID ( autoincrementare) si cheie externa AngajatID( care este de tip unic in tabelul salarii pentru ca fiecare angajat trebuie sa aiba un singur salariu)

-medici cu cheia primara MedicID ( autoincrementare)

--produse cu cheia primara ProdusID ( autoincrementare)

-pacienti cu cheia primara PacientID( autoincrementare), cheia externa MedicID( a 2 a relatie 1:N) si campul CNP unic

-retete –tabel de legatura-cu cheia primara RetetaID(autoincrementare), chei externe:ProdusID(produsul care o sa fie cumparat) si PacientID(pacientul care are nevoie de medicamentul respectiv)

-bonuri –tabel de legatura-cu cheia primara BonID(autoincrementare), chei externe:AngajatID(angajatul care ofera bonul) si PacientID(pacientului pentru care se elibereaza)

-date tabel in care se stocheaza datele de login din interfata

**Relatii intre tabele:**

Angajati-Salarii relatie 1:1

Farmacie-Angajati relatie 1:N deoarece o farmacie poate avea mai multi angajati

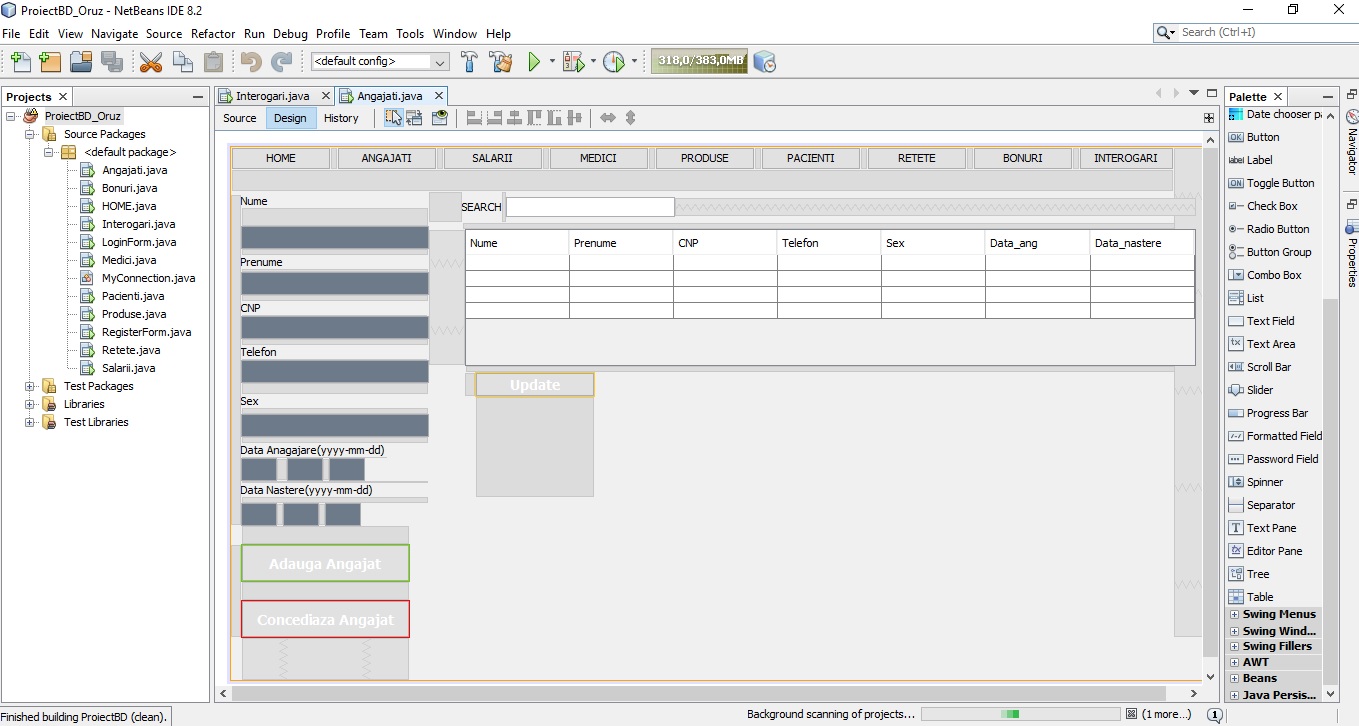
Medic-Pacienti relatie 1:N deoarece un medic poate avea mai multi pacienti

Pacienti-Produse relatie N:N cu ajutorul tabelului de legatura retete

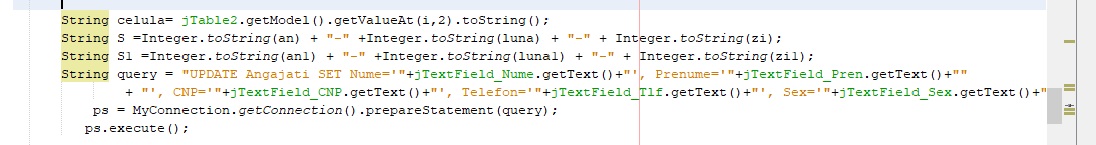
Pacienti-Angajati relatie N:N cu ajutorul tabelului de legatura bonuri

**Interfata:**

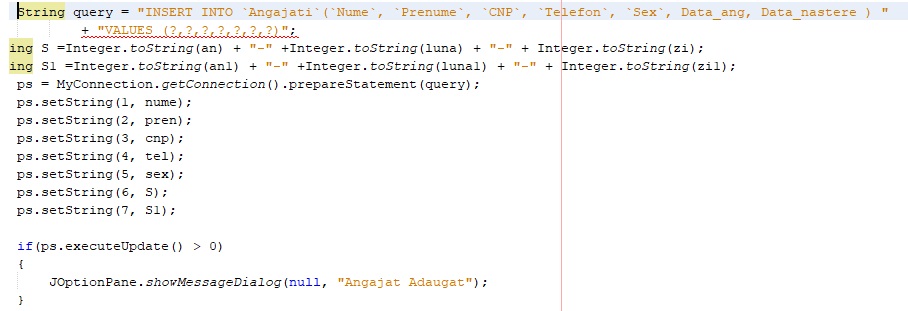
Interfata a fost facuta in java cu ajutorul programului NetBeans.



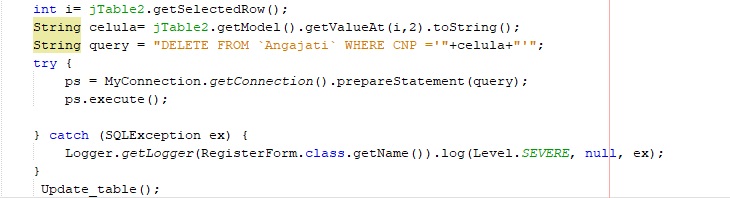
Exemplu de update(update ul se face dupa cnp-ul angajatului care a fost selectat din tabel(deoarece este un camp unic) :



Exemplu de insert(am inserat in campurile tabelului din baza de date valrorile prelucrate si citite din interfata):



Exemplu de delete ( am folosit campurile unice la stergere, daca acestea nu existau cautam dupa mai multe campuri ex:denumire+categorie+pret)



**Interogari simple:**

**1."Select `nouu`.`Angajati`.`Nume`,`nouu`.`Angajati`.`Prenume`,Salariu, ore\_saptamana, Bonus**

**from Salarii**

**INNER JOIN `nouu`.`Angajati` on `nouu`.`Angajati`.`AngajatID`=`nouu`.`Salarii`.`AngajatID`"**

Selecteaza Numele si Prenumele angajatului si detaliile despre salariile acestuia folosindu se de cheia externa.

**2.** **"Select `nouu`.`Angajati`.`Nume`,`nouu`.`Angajati`.`Prenume`,Salariu, ore\_saptamana, Bonus**

**from Salarii**

**INNER JOIN `nouu`.`Angajati` on `nouu`.`Angajati`.`AngajatID`=`nouu`.`Salarii`.`AngajatID`**

**WHERE `nouu`.`Angajati`.`Nume` =? OR `nouu`.`Angajati`.`Prenume` =? OR `Salariu` =?"**

Selecteaza Numele si Prenumele angajatului si detaliile despre salariile acestuia dupa numele, prenumele sau salariul introdus in campul de cautare.

**3.** **"Select `nouu`.`Medici`.`Nume`,`nouu`.`Medici`.`Prenume`,`NumeP`, `PrenumeP`, `CNP`, `Telefon`, `Sex` from Pacienti**

**INNER JOIN `nouu`.`Medici` on `nouu`.`Medici`.`MedicID`=`nouu`.`Pacienti`.`MedicID`"**

Selecteaza detaliile pacientului + numele si prenumele Medicului la care acesta a fost.

**4.** **"Select `nouu`.`Medici`.`Nume`,`nouu`.`Medici`.`Prenume`,`NumeP`, `PrenumeP`, `CNP`, `Telefon`, `Sex` from Pacienti**

**INNER JOIN `nouu`.`Medici` on `nouu`.`Medici`.`MedicID`=`nouu`.`Pacienti`.`MedicID`**

**WHERE `nouu`.`Medici`.`Nume` =? OR `nouu`.`Medici`.`Prenume` =? OR `NumeP` =? OR `PrenumeP`=?"**

Selecteaza detaliile pacientului + numele si prenumele Medicului la care acesta a fost dupa numele, prenumele sau dupa numele/ prenumele medicului in campul de cautare.

**5."Select `nouu`.`Pacienti`.`NumeP`,`nouu`.`Pacienti`.`PrenumeP`,`nouu`.`Produse`.`Denumire`, Diagnostic`, `Durata\_tratament(zile)`, `Compensata` from Retete**

**INNER JOIN `nouu`.`Pacienti` on `nouu`.`Pacienti`.`PacientID`=`nouu`.`Retete`.`PacientID`**

**INNER JOIN `nouu`.`Produse` on `nouu`.`Produse`.`ProdusID`=`nouu`.`Retete`.`ProdusID`"**

Selecteaza detaliile retetei + numele si prenumele Pacientului + Denumirea produsului de pe reteta.

**6.** **"Select `nouu`.`Pacienti`.`NumeP`,`nouu`.`Pacienti`.`PrenumeP`,**

**SUM(`nouu`.`Produse`.`Pret`) As Total,**

**`TVA` from Bonuri**

**INNER JOIN `nouu`.`Pacienti` on `nouu`.`Pacienti`.`PacientID`=`nouu`.`Bonuri`.`PacientID`**

**INNER JOIN `nouu`.`Retete` on `nouu`.`Retete`.`PacientID`=`nouu`.`Bonuri`.`PacientID`**

**INNER JOIN `nouu`.`Produse` on `nouu`.`Produse`.`ProdusID`=`nouu`.`Retete`.`ProdusID`"**

**+ " Group By `nouu`.`Bonuri`.`PacientID`"**

Selecteaza detaliile bonului + numele si prenumele Pacientului + Totalul pe care fiecare pacient il are de platit.

**Interogari complexe:**

**1.** **"Select `Nume`, `Prenume`, Data\_ang "**

**+ "from Angajati "**

**+ "Where Data\_ang <= ( "**

**+ "SELECT Data\_ang from Angajati "**

**+ "WHERE AngajatID ='"+2+"')"**

Selecteaza numele, prenumele si data angajarii pentru toti angajatii care au data de angajare mai mica decat angajatul cu ID-ul 2.

**2.** **"Select `nouu`.`Angajati`.`Nume`,`nouu`.`Angajati`.`Prenume`,Salariu"**

**+ " from Salarii"**

**+ " INNER JOIN `nouu`.`Angajati` on `nouu`.`Angajati`.`AngajatID`=`nouu`.`Salarii`.`AngajatID`"**

**+ " Where Salariu <= ( SELECT AVG(Salariu) from Salarii)"**

Selecteaza numele, prenumele si salariul angajatilor care au salariul mai mic sau egal cu media tuturor salariilor.

**3.** **"Select `nouu`.`Angajati`.`Nume`,`nouu`.`Angajati`.`Prenume`, ore\_saptamana"**

**+ " from Salarii"**

**+ " INNER JOIN `nouu`.`Angajati` on `nouu`.`Angajati`.`AngajatID`=`nouu`.`Salarii`.`AngajatID`"**

**+ " Where Exists ( "**

**+ "Select `AngajatID` FROM `nouu`.`Angajati` Where ore\_saptamana < 50)"**

Selecteaza numele, prenumele si numarul de ore lucrate ale angajatilor care lucreaza mai putin de 50 de ore pe saptamana.

**4.** **"Select year(`Data\_ang`) As AN, count(`AngajatID`) As Numar\_ang"**

**+ " from Angajati "**

**+ "Group by year(`Data\_ang`) "**

**+ "Having count(`AngajatID`)=( "**

**+ "Select max(ss.nrrang) "**

**+ "From (select ( count(`AngajatID`)) AS nrrang "**

**+ "From Angajati "**

**+ "Group by year(`Data\_ang`)) As ss)"**

Verifica anul in care s-au angajat cei mai multi angajati. Afiseaza anul si numarul angajatilor.