PROIECT – BAZE DE DATE

COORDONATOR ȘTIINȚIFIC :

Prof. Silviu-Laurentiu Vasile

STUDENT:

Dura Alexandru-Bogdan

GRUPA: 263

GESTIONAREA UNUI MAGAZIN DE PIESE AUTO

COORDONATOR ȘTIINȚIFIC :

Prof. Silviu-Laurentiu Vasile

STUDENT:

Dura Alexandru-Bogdan

GRUPA: 263

**Cuprins**

[1. Prezentarea modelului și a regulilor acestuia 4](#_Toc103800469)

[1.1. Regulile modelului: 4](#_Toc103800470)

[2. Diagrama Entitate-Relație 5](#_Toc103800471)

[2.1. Descrierea componentelor Diagramei Entitate-Relatie 5](#_Toc103800472)

[3. Diagrama Conceptuala 7](#_Toc103800473)

[3.1. Descrierea entităților si atributelor acestora: 8](#_Toc103800474)

[4. Comenzile pentru crearea tabelelor și pentru popularea acestora 13](#_Toc103800475)

[4.1. Comenzile pentru crearea tabelelor și a schemei 13](#_Toc103800476)

[4.2. Exemple de comenzi pentru popularea tabelelor. 19](#_Toc103800477)

[5. Interacțiunea cu baza de date 29](#_Toc103800478)

[5.1. Introducere 29](#_Toc103800479)

[5.2. Listarea conținutului 32](#_Toc103800480)

[5.3. Modificarea Informațiilor 35](#_Toc103800481)

[5.4. Afișarea rezultatului unei cereri care extrage informații din cel puțin 3 tabele 37](#_Toc103800482)

[5.5 Rezultatul unei cereri ce folosește o clauză having 38](#_Toc103800483)

[5.6 Exemplificarea unei constrângeri de tip ON DELETE CASCADE 39](#_Toc103800484)

[6. Interacțiunea cu baza de date (din modul de admin al site-ului) 42](#_Toc103800485)

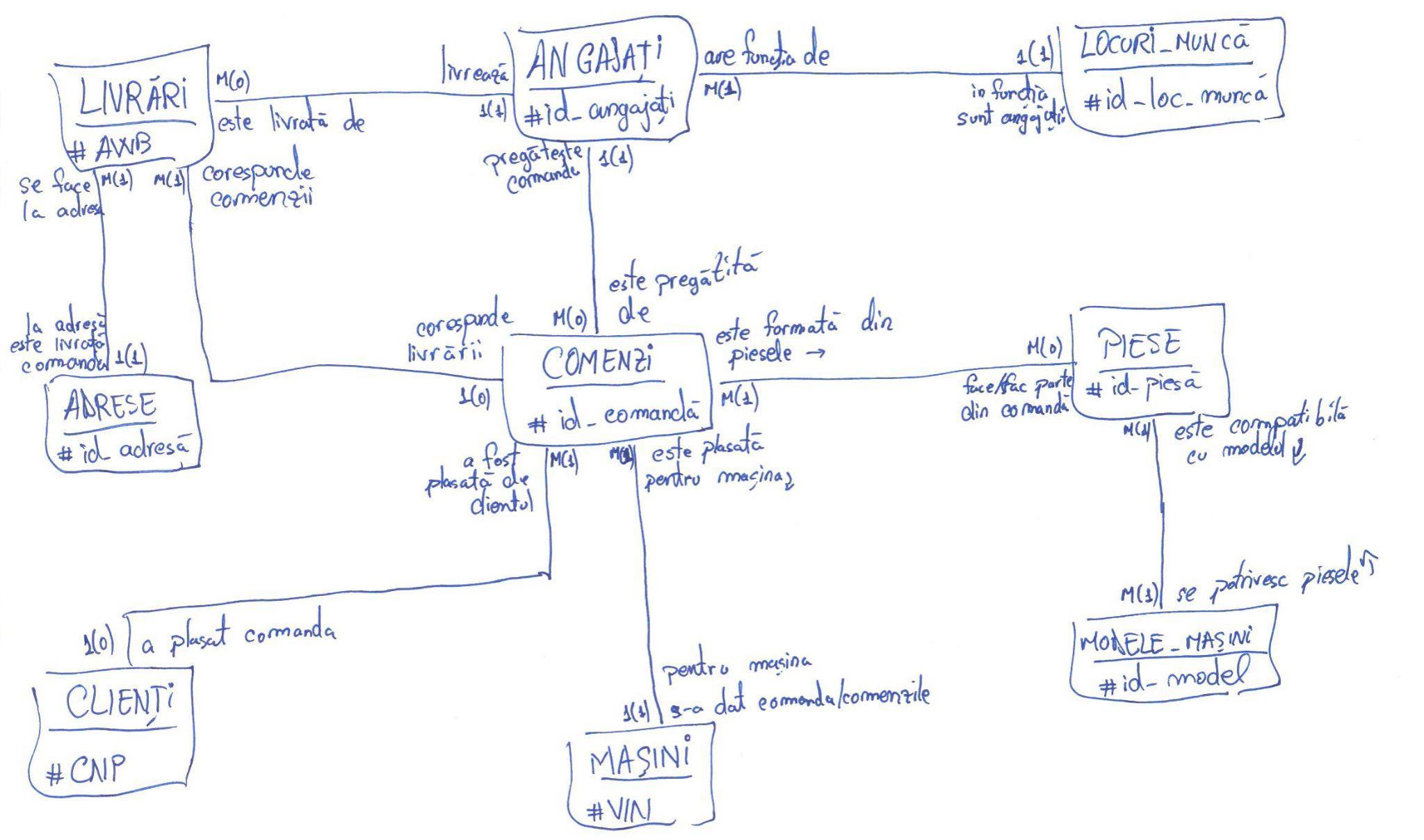
# Prezentarea modelului și a regulilor acestuia

Tema aleasă pentru proiectul la materia Baze de Date este ”Gestionarea unui magazin de piese auto”. Rolul acestei baze de date este de a stoca informații despre piesele si modelele de mașini pentru care respectivul magazin vinde piese, despre comenzile și datele de livrare plasate și, nu în ultimul rând, despre clienții și angajații magazinului. Pentru a ține mai bine evidența acestor date, am făcut următoarea bază de date, pentru a cărei proiectare am folosit MySQL.

## Regulile modelului:

* Toți angajații trebuie să aibă o funcție (un loc de munca) în cadrul magazinului.
* Orice comandă trebuie să fie pregătită de un angajat (unul singur).
* Nu toți angajații trebuie să pregătească sau să livreze comenzi. (angajații pot avea diferite funcții: manager de magazin, it-ist, supervizor, curier etc.).
* Nu este obligatoriu ca angajatul care se ocupă de pregătirea comenzii să o și livreze, dar este posibil.
* Nu toate comenzile trebuie livrate la adresă, pot exista și comenzi cu ridicare personală din magazin.
* Orice comandă care este cu livrare la adresă, este livrată de un singur angajat (de preferat să fie curier).
* O comandă poate fi formată din mai multe piese, dar nu mai puțin de una.
* Orice mașină trebuie să aibă un VIN (Vehicle Identification Number / Serie de șasiu), cu ajutorul căruia este posibilă identificarea mașinii.
* În tabelul mașini, atributul „kilometraj” reprezintă numărul de kilometri pe care îi are mașina la momentul comenzii.
* Pentru o persoană (client sau angajat), în câmpul „data\_naștere” voi completa cu data nașterii, nu cu vârsta acestuia.
* În cadrul unei comenzi pot adăuga piese pentru mai multe mașini (seria de șasiu din cadrul tabelului comenzi nu restricționează în vreun fel ce piese pot fi adăugate la comanda)
* Pot exista piese care sunt compatibile cu mai multe modele de mașini.
* Pentru a plasa o comandă, clientul nu trebuie să fie neapărat proprietarul mașinii (motiv pentru care nu există o legătură directă între mașină și client).
* În baza de date pot exista clienți care încă nu au plasat vreo comandă. (clienți care s-au înregistrat dar, din diferite motive, nu au apucat să plaseze vreo comandă).

# Diagrama Entitate-Relație



## Descrierea componentelor Diagramei Entitate-Relatie

***TABELUL „LOCURI\_MUNCA”***

Acest tabel cuprinde informațiile necesare despre locurile de munca, un id (pentru identificare), denumirea locului de munca, salariul minim și salariul maxim pentru respectivul loc de muncă. Cheia primară aferentă acestei entități se numește „id\_loc\_munca”.

***TABELUL „ANGAJATI”***

Acest tabel cuprinde informațiile necesare despre angajați: un id (pentru identificare), numele de familie, prenumele, salariul, adresa de email, data angajării, data nașterii și id-ul locului său de munca. Cheia primară aferentă acestei entități se numește „id\_angajat”.

***TABELUL „CLIENTI”***

Acest tabel cuprinde informațiile necesare despre clienți: cnp-ul (pentru identificare), numele de familie, prenumele, adresa de mail și numărul de telefon. Cheia primară aferentă acestei entități se numește „cnp”.

***TABELUL „ADRESE”***

Acest tabel cuprinde informațiile necesare despre fiecare adresă: un id (pentru identificare), țara, județul, localitatea, strada, numărul străzii, blocul, scara, etajul și apartamentul. Cheia primară aferentă acestei entități se numește „id\_adresă”.

***TABELUL „MAȘINI”***

Acest tabel cuprinde informațiile necesare despre fiecare mașină: un VIN (pentru identificare), numărul de înmatriculare, anul de fabricație și numărul de kilometri pe care îi are mașina. Cheia primară aferentă acestei entități se numește „vin” (Vehicle Identification Number). Nu am putut folosi numărul de înmatriculare ca și cheie primară deoarece o mașină îsi poate schimba numărul de înmatriculare.

***TABELUL „MODELE MAȘINI”***

Acest tabel cuprinde informațiile necesare despre fiecare model de mașină: un id (pentru identificare), marca, modelul, anul în care a început producția, anul în care s-a finalizat producția, motorizarea (exprimată în centimetri cubi), puterea (exprimată în cai putere), combustibilul, codul motorului și tipul de tracțiune. Cheia primară aferentă acestei entități se numește „id\_model”.

***TABELUL „PIESE”***

Acest tabel cuprinde informațiile necesare despre fiecare piesă: un id (pentru identificare), denumirea, producătorul, intervalul (de km sau ani), când este recomandat să schimbi piesa și prețul acesteia. Cheia primară aferentă acestei entități se numește „id\_piesa”.

***TABELUL „COMENZI”***

Acest tabel cuprinde informațiile necesare despre fiecare comandă: un id (pentru identificare), data comenzii, VIN-ul mașinii pentru care este plasată comanda, id-ul clientului care a plasat comanda, id-ul angajatului care se ocupă de pregătirea comenzii, tipul de livrare si statusul în care se află comanda.

***TABELUL „LIVRARI”***

Acest tabel cuprinde informațiile necesare despre fiecare livrare: un AWB (pentru identificare), id-ul angajatului care se ocupă de livrare, id-ul comenzii pentru care se face livrarea, adresa de livrare, statusul livrării și data la care este programată livrarea.

# Diagrama Conceptuala

## Descrierea entităților si atributelor acestora:

***Tabelul locuri\_munca :*** Această entitate conține informații despre locurile de munca din cadrul magazinului de piese auto.

Cheile acestui tabel sunt:

* id\_loc\_munca (PK) – este identificatorul unic pentru fiecare loc de munca. Acesta se generează automat. Tip de date : INT.

Atributele acestui tabel sunt:

* denumire – reține numele locului de munca. Exemple de locuri de munca in cadrul magazinului sunt: curier, manager de magazin, it-ist, manipulator (persoana care pregateste comenzi, etc). Tip de date: VARCHAR(45).
* salariu\_minim – reține salariul minim pentru respectiva funcție. Tip de date: INT.
* salariu\_maxim– reține salariul maxim pentru respectiva funcție. Tip de date: INT.

***Tabelul angajati :*** Această entitate conține informații despre angajații magazinului. Un loc de munca poate să fie șters doar în cazul în care acest loc de muncă nu este ocupat de niciun angajat, pentru a nu exista posibilitatea de a șterge angajați din greșeală. Pentru acest lucru voi folosi „ON DELETE RESTRICT”.

Cheile acestui tabel sunt:

* id\_angajat (PK) - este identificatorul unic pentru fiecare angajat, acesta se genereaza automat. Tip de date : INT.
* loc\_munca (FK) – este identificatorul care stabilește relația dintre tabelele angajați si locuri de muncă (face referire la cheia primară id\_loc\_munca din cadrul tabelului locuri\_munca). Tip de date : INT.

Atributele acestui tabel sunt:

* nume\_familie – reține numele de familie al angajatului. Tip de date: VARCHAR(45).
* prenume – reține prenumele angajatului. Tip de date: VARCHAR(45).
* salariu - reține salariul angajatului, exprimat in ron. Tip de date: INT.
* email – reține email-ul angajatului. Tip de date: VARCHAR(45).
* data\_angajare – reține data la care a fost semnat contractul de angajare. Tip de date: DATE.
* data\_nastere – reține data nașterii a angajatului. Tip de date: DATE.

***Tabelul modele\_masini:*** Această entitate conține informații despre diferite modele de masini.

Cheile acestui tabel sunt:

* id\_model (PK) - este identificatorul unic pentru fiecare model de mașina, acesta se genereaza automat. Tip de date : INT.

Atributele acestui tabel sunt:

* marca – retine numele producatorului modelului. Tip de date: VARCHAR(45).
* model – retine denumirea modelului. Tip de date: VARCHAR(45).
* an\_incepere\_fabricatie – reține anul în care a început sa se produca respectivul model. Tip de date: YEAR.
* an\_finalizare\_fabricatie – reține anul în care s-a oprit producția respectivului model. Tip de date: YEAR.
* motorizare\_cc - reține capacitatea motorului exprimată în centimetri cubi. Tip de date: INT.
* putere\_cp - reține puterea motorului exprimat în cai putere. Tip de date: INT.
* combustibil - reține tipul de combustibil pe care îl folosește masina . Tip de date: ENUM('benzina', 'diesel', 'electric', 'hybrid-benzina', 'hybrid-diesel', 'hidrogen').
* cod\_motor - retine codul motorului. Tip de date: VARCHAR(10)
* tip\_tractiune - reține ce fel de tracțiune are respectivul model de mașină. Tip de date: ENUM('fata', 'spate', '4x4').

***Tabelul piese:*** Această entitate conține informații despre diferite piese.

Cheile acestui tabel sunt:

* id\_piesa (PK) - este identificatorul unic pentru fiecare piesa, acesta se genereaza automat. Tip de date : INT.

Atributele acestui tabel sunt:

* denumire\_piesa – reține denumirea respectivei piese. Tip de date: VARCHAR(45).
* producator – reține numele producatorului piesei. Tip de date: VARCHAR(45).
* interval\_recomandat\_schimb - reține intervalul de km sau ani după care este recomandată schimbarea respectivei piese. Tip de date: INT.
* pret\_ron - reține prețul care corespunde piesei, este exprimat în ron. Tip de date: INT

***Tabelul piesa\_model:*** Această entitate face legatura dintre tabelele modele\_masini si piese. Dacă voi șterge o piesă sau un model de mașină atunci trebuie șterse și legăturile din care acestea făceau parte. Pentru acest lucru voi folosi „ON DELETE CASCADE”.

Cheile acestui tabel sunt:

* id\_piesa (PK, FK) - este identificatorul pentru fiecare piesa. Face referire la cheia primara id\_piesa din tabelul piese. Tip de date : INT.
* id\_model (PK, FK) - este identificatorul pentru fiecare model de mașină. Face referire la cheia primara id\_model din tabelul model. Tip de date : INT.

Cheia primară a acestui tabel este compusă din cele doua chei.

Atributele acestui tabel sunt:

* originala – reține dacă piesa este originală pentru respectivul model de mașină. Tip de date: ENUM('originala', 'compatibila').

***Tabelul masini:*** Această entitate conține informații despre diferite masini.

Cheile acestui tabel sunt:

* vin (PK) - este identificatorul unic pentru fiecare masina, VIN (Vehicle Identification Number). Tip de date : VARCHAR(17).

Atributele acestui tabel sunt:

* nr\_inmatriculare - retine numărul de înmatriculare al acestei mașini (acesta nu putea fi PK deoarece pe parcursul vieții unei mașini, masina poate avea mai multe numere de inmatriculare.). Tip de date : VARCHAR(10).
* an\_fabricatie - reține anul cand a fost fabricată masina. Tip de date : DATE.
* kilometraj - reține numărul de kilometri pe care îi are mașina la momentul plasării comenzii. Tip de date : VARCHAR(10).

***Tabelul clienti:*** Această entitate conține informații despre clienții magazinului.

Cheile acestui tabel sunt:

* cnp (PK)- cod numeric personal, este identificatorul unic pentru fiecare client. Tip de date : VARCHAR(13).

Atributele acestui tabel sunt:

* nume\_familie – reține numele de familie al clientului. Tip de date: VARCHAR(45).
* prenume – reține prenumele clientului. Tip de date: VARCHAR(45).
* nr\_telefon - retine numarul de telefon al clientului. VARCHAR(12).
* email – reține email-ul angajatului. Tip de date: VARCHAR(45).

***Tabel adresa:*** Această entitate conține informații despre adresele clienților.

Cheile acestui tabel sunt:

* id\_adresa (PK) - este identificatorul unic pentru fiecare adresa, acesta se genereaza automat. Tip de date : INT.

Atributele acestui tabel sunt:

* tara - reține numele țării în care se face livrarea. Tip de date: VARCHAR(45).
* judet - - reține numele județului în care se face livrarea. Tip de date: VARCHAR(45).
* localitate - reține numele localității unde se face livrarea. Tip de date: VARCHAR(45).
* strada - reține numele străzii unde se face livrarea. Tip de date: VARCHAR(45).
* nr\_strada - reține numărul străzii unde se face livrarea. Tip de date: INT.
* bloc - reține numărul blocului unde se face livrarea. Dacă respectiva adresa este la casa si nu la bloc, acest camp poate ramane nul. Tip de date: INT.
* scara - reține numărul scării unde se face livrarea. Dacă respectiva adresa este la casa si nu la bloc, acest camp poate ramane nul. Tip de date: VARCHAR(10).
* etaj - reține numărul etajului unde se face livrarea. Dacă respectiva adresa este la casa si nu la bloc, acest camp poate ramane nul. Tip de date: INT.
* apartament - reține numărul apartamentului unde se face livrarea. Dacă respectiva adresa este la casa și nu la bloc, acest camp poate ramane nul. Tip de date: INT.

***Tabel comenzi:*** Această entitate conține informații despre comenzile plasate. Se vor putea șterge o mașină, un client sau un angajat doar daca nu au comenzi asociate acestora. Pentru acest lucru voi folosi „ON DELETE RESTRICT”.

Cheile acestui tabel sunt:

* id\_comanda (PK) - este identificatorul unic pentru fiecare adresa, acesta se genereaza automat. Tip de date : INT.
* masina (FK) - este identificatorul pentru masini, astfel putem vedea pentru ce masina a fost plasată comanda. El stabilește relația dintre tabelele comenzi și masini (face referire la cheia primara vin din cadrul tabelului masini). Tip de date :VARCHAR(17).
* client (FK) - este identificatorul pentru clienți, astfel putem vedea ce client a plasat comanda. El stabilește relația dintre tabelele comenzi și client (face referire la cheia primară cnp din cadrul tabelului client). Tip de date :VARCHAR(13).
* pregatita\_de\_angajat (FK) - este identificatorul pentru angajați, cu ajutorul lui putem vedea care este angajatul care trebuie sa se ocupe de pregătirea respectivei comenzi. El stabilește relația dintre tabelele clienți și angajați (face referire la cheia primara id\_angajat din cadrul tabelului angajati.). Tip de date : INT.

Atributele acestui tabel sunt:

* data\_comanda - reține data la care a fost plasată comanda. Tip de date : DATE.
* status\_comanda - reține etapa în care se afla comanda. Tip de date : ENUM('plasata', 'pregatita pentru ridicare', 'în curs de livrare', 'ridicata', 'livrata').
* tip\_livrare - reține tipul de livrare pentru care a optat clientul (ridicare personală sau livrare la adresa). Tip de date : ENUM('ridicare personala', 'livrare').

***Tabelul livrari:*** Această entitate conține informații despre datele de livrare. O adresă sau un angajat vor putea fi șterși din baza de date doar dacă aceștia nu au livrări asociate. Pentru acest lucru voi folosi „ON DELETE RESTRICT”. Dacă voi șterge o comandă care este asociată unei livrări, atunci și livrarea asociată ei trebuie ștearsă. Pentru acest lucru voi folosi „ON DELETE CASCADE”.

Cheile acestui tabel sunt:

* AWB (PK) - este numărul de urmarire al comenzii, acesta este identificatorul unic pentru livrare. Tip de date : VARCHAR(20).
* livrata\_de\_angajat (FK) - este identificatorul pentru angajați, cu ajutorul lui putem vedea care este angajatul care trebuie sa se ocupe de livrarea respectivei comenzi. El stabilește relația dintre tabelele livrare și angajați (face referire la cheia primara id\_angajat din cadrul tabelului angajati.). Tip de date : INT.
* adresa (FK) - este identificatorul unic pentru adrese, cu ajutorul lui putem vedea adresa la care trebuie făcută livrarea. El stabilește relația dintre tabelele livrare și adresa (face referire la cheia primara id\_adresa din cadrul tabelului adresa). Tip de date : INT.

Atributele acestui tabel sunt:

* status\_livrare - retine etapa în care se afla livrarea, dacă acest status nu se cunoaște, acest camp poate ramane null. Tip de date: ENUM('primita', 'in curs de livrare', 'livrata').
* data\_programata\_livrare - reține data pentru cand a fost programată livrarea, dacă nu se cunoaste inca, acest camp poate ramane null. Tip de date : DATE.

***Tabelul comanda\_piesa:*** Această entitate face legatura dintre tabelele modele\_masini si piese. Dacă voi șterge o piesă sau o comandă atunci trebuie șterse și legăturile din care acestea făceau parte. Pentru acest lucru voi folosi „ON DELETE CASCADE”.

Cheile acestui tabel sunt:

* id\_comanda (PK, FK) - este identificatorul pentru fiecare comanda. Face referire la cheia primara id\_piesa din tabelul piesa. Tip de date : INT.
* id\_piesa (PK, FK) - este identificatorul pentru fiecare piesa. Face referire la cheia primara id\_piesa din tabelul piese. Tip de date : INT.

Cheia primară a acestui tabel este compusă din cele doua chei.

Atributele acestui tabel sunt:

* nr\_piese – reține cantitatea de piese care este adăugată la comandă. Tip de date: INT.

# 4. Comenzile pentru crearea tabelelor și pentru popularea acestora

## 4.1. Comenzile pentru crearea tabelelor și a schemei

***Crearea schemei mag\_piese.bd***

CREATE SCHEMA `mag\_piese.bd` ;

***Crearea tabelului locuri\_munca***

CREATE TABLE `mag\_piese.bd`.`locuri\_munca` (

`id\_loc\_munca` INT NOT NULL AUTO\_INCREMENT,

`denumire` VARCHAR(45) NOT NULL,

`salariu\_min` INT UNSIGNED NOT NULL,

`salariu\_max` INT UNSIGNED NOT NULL,

PRIMARY KEY (`id\_loc\_munca`));

***Crearea tabelului piese***

CREATE TABLE `mag\_piese.bd`.`piese` (

`id\_piesa` INT NOT NULL AUTO\_INCREMENT,

`denumire\_piesa` VARCHAR(45) NOT NULL,

`producator` VARCHAR(45) NOT NULL,

`interval\_recomandat\_schimb` VARCHAR(45) NOT NULL DEFAULT 'nu exista',

`pret\_ron` INT UNSIGNED NOT NULL,

PRIMARY KEY (`id\_piesa`));

***Crearea tabelului modele\_masini***

CREATE TABLE `mag\_piese.bd`.`modele\_masini` (

`id\_model` INT NOT NULL AUTO\_INCREMENT,

`marca` VARCHAR(45) NOT NULL,

`model` VARCHAR(45) NOT NULL,

`an\_incepere\_fabricatie` YEAR NOT NULL,

`an\_finalizare\_fabricatie` YEAR NOT NULL,

`motorizare\_cc` INT UNSIGNED NULL,

`putere\_cp` INT UNSIGNED NOT NULL,

`combustibil` ENUM('benzina', 'diesel', 'hybrid-benzina', 'hybrid-diesel', 'electric', 'hidrogen') NOT NULL,

`cod\_motor` VARCHAR(10) NOT NULL DEFAULT 'nu exista',

`tip\_tractiune` ENUM('fata', 'spare', '4x4') NOT NULL,

PRIMARY KEY (`id\_model`));

***Crearea tabelului piesa\_model***

CREATE TABLE `mag\_piese.bd`.`piesa\_model` (

`id\_piesa` INT NOT NULL,

`id\_model` INT NOT NULL,

`originala` ENUM('originala', 'compatibila') NOT NULL DEFAULT 'originala',

PRIMARY KEY (`id\_piesa`, `id\_model`),

INDEX `id\_model\_idx` (`id\_model` ASC) VISIBLE,

CONSTRAINT `id\_piesa`

FOREIGN KEY (`id\_piesa`)

REFERENCES `mag\_piese.bd`.`piese` (`id\_piesa`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `id\_model`

FOREIGN KEY (`id\_model`)

REFERENCES `mag\_piese.bd`.`modele\_masini` (`id\_model`)

ON DELETE CASCADE

ON UPDATE CASCADE);

***Crearea tabelului masini***

CREATE TABLE `mag\_piese.bd`.`masini` (

`vin` VARCHAR(17) NOT NULL,

`nr\_inmatriculare` VARCHAR(10) NOT NULL,

`an\_fabricatie` YEAR NOT NULL,

`kilometraj` VARCHAR(10) NOT NULL,

PRIMARY KEY (`vin`));

***Crearea tabelului angajati***

CREATE TABLE `mag\_piese.bd`.`angajati` (

`id\_angajat` INT NOT NULL AUTO\_INCREMENT,

`nume\_familie` VARCHAR(45) NOT NULL,

`prenume` VARCHAR(45) NOT NULL,

`salariu` INT UNSIGNED NOT NULL,

`email` VARCHAR(45) NOT NULL,

`data\_angajare` DATE NOT NULL,

`data\_nastere` DATE NOT NULL,

`loc\_munca` INT NOT NULL,

PRIMARY KEY (`id\_angajat`),

INDEX `loc\_munca\_idx` (`loc\_munca` ASC) VISIBLE,

CONSTRAINT `loc\_munca`

FOREIGN KEY (`loc\_munca`)

REFERENCES `mag\_piese.bd`.`locuri\_munca` (`id\_loc\_munca`)

ON DELETE RESTRICT

ON UPDATE RESTRICT);

***Crearea tabelului clienti***

CREATE TABLE `mag\_piese.bd`.`clienti` (

`cnp` VARCHAR(13) NOT NULL,

`nume\_familie` VARCHAR(45) NOT NULL,

`prenume` VARCHAR(45) NOT NULL,

`nr\_telefon` VARCHAR(12) NOT NULL,

`email` VARCHAR(45) NOT NULL,

PRIMARY KEY (`cnp`));

***Crearea tabelului adrese***

CREATE TABLE `mag\_piese.bd`.`adrese` (

`id\_adresa` INT NOT NULL AUTO\_INCREMENT,

`tara` VARCHAR(45) NOT NULL,

`judet` VARCHAR(45) NOT NULL,

`localitate` VARCHAR(45) NOT NULL,

`strada` VARCHAR(45) NOT NULL,

`nr\_strada` INT UNSIGNED NOT NULL,

`bloc` INT UNSIGNED NULL,

`scara` VARCHAR(10) NULL,

`etaj` INT UNSIGNED NULL,

`apartament` INT UNSIGNED NULL,

PRIMARY KEY (`id\_adresa`));

***Crearea tabelului comenzi***

CREATE TABLE `mag\_piese.bd`.`comenzi` (

`id\_comanda` INT NOT NULL AUTO\_INCREMENT,

`data\_comanda` DATE NOT NULL,

`masina` VARCHAR(17) NOT NULL,

`client` VARCHAR(13) NOT NULL,

`pregatita\_de\_angajat` INT NOT NULL,

`tip\_livrare` ENUM('ridicare personala', 'livrare') NOT NULL DEFAULT 'ridicare personala',

`status\_comanda` ENUM('plasata', 'pregatita pentru ridicare', 'in curs de livrare', 'ridicata', 'livrata') NOT NULL DEFAULT 'plasata',

PRIMARY KEY (`id\_comanda`),

INDEX `masina\_idx` (`masina` ASC) VISIBLE,

INDEX `client\_idx` (`client` ASC) VISIBLE,

INDEX `pregatita\_de\_angajat\_idx` (`pregatita\_de\_angajat` ASC) VISIBLE,

CONSTRAINT `masina`

FOREIGN KEY (`masina`)

REFERENCES `mag\_piese.bd`.`masini` (`vin`)

ON DELETE RESTRICT

ON UPDATE RESTRICT,

CONSTRAINT `client`

FOREIGN KEY (`client`)

REFERENCES `mag\_piese.bd`.`clienti` (`cnp`)

ON DELETE RESTRICT

ON UPDATE RESTRICT,

CONSTRAINT `pregatita\_de\_angajat`

FOREIGN KEY (`pregatita\_de\_angajat`)

REFERENCES `mag\_piese.bd`.`angajati` (`id\_angajat`)

ON DELETE RESTRICT

ON UPDATE RESTRICT);

***Crearea tabelului comanda\_piesa***

CREATE TABLE `mag\_piese.bd`.`comanda\_piesa` (

`id-comanda` INT NOT NULL,

`id-piesa` INT NOT NULL,

`nr\_piese` INT UNSIGNED NOT NULL,

PRIMARY KEY (`id-comanda`, `id-piesa`),

INDEX `id\_piesa\_idx` (`id-piesa` ASC) VISIBLE,

CONSTRAINT `id-comanda`

FOREIGN KEY (`id-comanda`)

REFERENCES `mag\_piese.bd`.`comenzi` (`id\_comanda`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `id-piesa`

FOREIGN KEY (`id-piesa`)

REFERENCES `mag\_piese.bd`.`piese` (`id\_piesa`)

ON DELETE CASCADE

ON UPDATE CASCADE);

***Crearea tabelului livrari***

CREATE TABLE `mag\_piese.bd`.`livrari` (

`AWB` VARCHAR(20) NOT NULL,

`livrata\_de\_angajat` INT NOT NULL,

`comanda` INT NOT NULL,

`adresa` INT NOT NULL,

`status\_livrare` ENUM('primita', 'in curs de livrare', 'livrata') NOT NULL DEFAULT 'primita',

`data\_programata\_livrare` DATE NOT NULL,

PRIMARY KEY (`AWB`),

INDEX `livrata\_de\_angajat\_idx` (`livrata\_de\_angajat` ASC) VISIBLE,

INDEX `comanda\_idx` (`comanda` ASC) VISIBLE,

INDEX `adresa\_idx` (`adresa` ASC) VISIBLE,

CONSTRAINT `livrata\_de\_angajat`

FOREIGN KEY (`livrata\_de\_angajat`)

REFERENCES `mag\_piese.bd`.`angajati` (`id\_angajat`)

ON DELETE RESTRICT

ON UPDATE RESTRICT,

CONSTRAINT `comanda`

FOREIGN KEY (`comanda`)

REFERENCES `mag\_piese.bd`.`comenzi` (`id\_comanda`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `adresa`

FOREIGN KEY (`adresa`)

REFERENCES `mag\_piese.bd`.`adrese` (`id\_adresa`)

ON DELETE RESTRICT

ON UPDATE RESTRICT);

## 4.2. Exemple de comenzi pentru popularea tabelelor.

***Popularea tabelului clienti***

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('5010502430038', 'Ionescu', 'Bogdan', '0748053805', 'ionescu.bogdan@gmail.com');

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('5010231412428', 'Barbu', 'Andrada', '0748053235', 'barbu.andrada@gmail.com');

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('5321321320038', 'Popa', 'Dragos', '0728053235', 'popa.dragos@gmail.com');

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('5015235234333', 'Neamtu', 'Liviu', '0734342405', 'neamtu.liviu@gmail.com');

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('5015232523532', 'Dumitrescu', 'Florin', '0708053995', 'dumitrescu.florin@gmail.com');

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('5014324324338', 'Dura', 'Bogdan', '0731248443', 'bogdan2435@gmail.com');

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('5014324234238', 'Stefanescu', 'Viorel', '0734353443', 'stefanescu.viorel@gmail.com');

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('5010702550014', 'Popescu', 'Stefan', '0770434654', 'popescu@stefan.ro');

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('1991225330023', 'Ionescu', 'Vlad', '0744553344', 'ionescu@vlad.ro');

INSERT INTO `mag\_piese.bd`.`clienti` (`cnp`, `nume\_familie`, `prenume`, `nr\_telefon`, `email`) VALUES ('2961111660088', 'Popescu', 'Cristina', '0755234654', 'popescu@cristina.ro');

***Popularea tabelului adrese***

INSERT INTO `mag\_piese.bd`.`adrese` (`tara`, `judet` ,`localitate`, `strada`, `nr\_strada`, `bloc`, `scara`, `etaj`, `apartament`) VALUES ('Romania', 'Bucuresti', 'Sector 4', 'Viorele', '51', '30', '3', '7', '55');

INSERT INTO `mag\_piese.bd`.`adrese` (`tara`, `judet` ,`localitate`, `strada`, `nr\_strada`) VALUES ('Romania', 'Prahova', 'Ploiesti', 'Sarii', '30');

INSERT INTO `mag\_piese.bd`.`adrese` (`tara`, `judet` ,`localitate`, `strada`, `nr\_strada`, `bloc`, `scara`, `etaj`, `apartament`) VALUES ('Romania', 'Bucuresti', 'Sector 4', 'Cercetatorilor', '61', '37', '2', '4', '75');

INSERT INTO `mag\_piese.bd`.`adrese` (`tara`, `judet` ,`localitate`, `strada`, `nr\_strada`, `bloc`, `scara`, `etaj`, `apartament`) VALUES ('Romania', 'Bucuresti', 'Sector 1', 'Aripilor', '3', '22', 'A', '0', '33');

INSERT INTO `mag\_piese.bd`.`adrese` (`tara`, `judet` ,`localitate`, `strada`, `nr\_strada`, `bloc`, `scara`, `etaj`, `apartament`) VALUES ('Romania', 'Ploiesti', 'Ploiesti', 'Vornicei', '59', '13', 'B', '2', '15');

INSERT INTO `mag\_piese.bd`.`adrese` (`tara`, `judet` ,`localitate`, `strada`, `nr\_strada`) VALUES ('Romania', 'Ilfov', 'Snagov', 'Florilor', '87');

***Popularea tabelului locuri\_munca***

INSERT INTO `mag\_piese.bd`.`locuri\_munca` (`denumire`, `salariu\_min`, `salariu\_max`) VALUES ('programator', '3000', '6000');

INSERT INTO `mag\_piese.bd`.`locuri\_munca` (`denumire`, `salariu\_min`, `salariu\_max`) VALUES ('programator-ajutor', '1000', '2000');

INSERT INTO `mag\_piese.bd`.`locuri\_munca` (`denumire`, `salariu\_min`, `salariu\_max`) VALUES ('curier', '2312', '3132');

INSERT INTO `mag\_piese.bd`.`locuri\_munca` (`denumire`, `salariu\_min`, `salariu\_max`) VALUES ('staff', '1500', '3000');

INSERT INTO `mag\_piese.bd`.`locuri\_munca` (`denumire`, `salariu\_min`, `salariu\_max`) VALUES ('manipulant', '1800', '3300');

INSERT INTO `mag\_piese.bd`.`locuri\_munca` (`denumire`, `salariu\_min`, `salariu\_max`) VALUES ('manager de magazin', '3500', '5500');

INSERT INTO `mag\_piese.bd`.`locuri\_munca` (`denumire`, `salariu\_min`, `salariu\_max`) VALUES ('it-ist', '2700', '4800');

***Popularea tabelului angajati***

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Costea', 'Marian', '2400', 'costea.marian@yahoo.com', '2019-03-01', '1987-07-08', '3');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Popescu', 'Ion', '2700', 'popescu.ion@yahoo.com', '2019-05-23', '1987-09-23', '4');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Cristescu', 'Maria', '3030', 'cristescu.maria@yahoo.com', '2020-11-20', '1992-04-02', '1');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Macovei', 'Tudor', '4322', 'macovei.tudor@yahoo.com', '2022-12-23', '2000-02-05', '1');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Munteanu', 'Florin', '1700', 'munteanu.florin@gmail.com', '2022-11-28', '2001-06-05', '4');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Mihai', 'Bogdan', '3000', 'mihai.bogdan@gmail.com', '2021-06-01', '2001-02-05', '3');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Stefansecu', 'Mihai', '2765', 'stefanescu7@yahoo.com', '2019-08-23', '1987-02-05', '5');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Tudor', 'Alexandru', '2234', 'alexandru.tudor@gmail.com', '2021-10-11', '1989-09-08', '5');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Andreescu', 'Mihaela', '3045', 'andreescu.mihaela22@icloud.com', '2020-05-22', '1990-06-19', '3');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Baciu', 'Andreea', '3400', 'baciu@andreea.ro', '2018-03-16', '1985-12-15', '3');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Badea', 'Achim', '1600', 'badea@achim.ro', '2022-01-02', '1990-01-12', '2');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Begu', 'Codrin', '3150', 'begu@codrin.ro', '2020-04-27', '1984-07-13', '1');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Bonciu', 'Carol', '2730', 'bonciu@carol.ro', '2018-01-19', '1981-05-29', '4');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Nita', 'Alexandra', '1800', 'nita@alexandra.ro', '2021-12-18', '1996-06-06', '2');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Stanciu', 'George', '2370', 'stanciu@george.ro', '2019-10-21', '1995-11-25', '3');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Diaconu', 'Anca', '3220', 'diaconu@anca.ro', '2020-02-03', '1988-06-27', '3');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Mocanu', 'Florin', '5430', 'mocanu@florin.ro', '2017-09-07', '1980-10-09', '6');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Toma', 'Dani', '4800', 'toma@dani.ro', '2018-12-09', '1982-07-11', '6');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Rusu', 'Bianca', '4750', 'rusu@bianca.ro', '2020-10-11', '1997-03-20', '6');

INSERT INTO `mag\_piese.bd`.`angajati` (`nume\_familie`, `prenume`, `salariu`, `email`, `data\_angajare`, `data\_nastere`, `loc\_munca`) VALUES ('Lupu', 'Bogdan', '4020', 'lupu@bogdan.ro', '2021-08-14', '1993-02-18', '6');

***Popularea tabelului*** ***modele\_masini***

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Skoda', 'Octavia II', 2007, 2010, '1968', '140', 'diesel', 'BKD', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Volvo', 'S80 II', 2006, 2016, '1984', '163', 'diesel', 'D5204T3', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('BMW', 'M2', 2016, 2020, '2999', '450', 'benzina', 'fsda22', '4x4');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Dacia', 'Logan', 2004, 2009, '1399', '100', 'benzina', 'ffd', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Opel', 'Corsa', 2016, 2020, '1399', '90', 'benzina', '55', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Volvo', 'S80 II', 2006, 2016, '1984', '136', 'diesel', 'D3136T2', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Volkswagen', 'Golf V', 2005, 2009, '1968', '140', 'diesel', 'BKD', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Seat', 'Leon 1P', 2005, 2009, '1968', '140', 'diesel', 'BKD', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Volvo', 'S80 II', 2006, 2016, '4414', '315', 'benzina', 'B8444S', '4x4');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Skoda', 'Octavia I', 1996, 2010, '1896', '90', 'diesel', 'ALH', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Skoda', 'Octavia II', 2004, 2013, '1968', '170', 'diesel', 'CEGA', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Volkswagen', 'Golf IV', 1999, 2004, '1896', '110', 'diesel', 'AXR', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Volkswage', 'Golf VII', 2012, 2019, '1984', '300', 'benzina', 'CJXC', '4x4');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `putere\_cp`, `combustibil`, `tip\_tractiune`) VALUES ('Volkswagen', 'e-Golf', 2017, 2021, '136', 'electric', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Seat', 'Leon 5F', 2012, 2020, '1984', '280', 'benzina', 'CJXA', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Seat', 'Leon 5F', 2012, 2020, '1968', '150', 'diesel', 'CKFC', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Seat', 'Leon 1P', 2005, 2012, '1798', '160', 'benzina', 'BZB', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Seat', 'Leon KL1', 2020, 2022, '1395', '204', 'hybrid-benzina', 'DGEA', 'fata');

INSERT INTO `mag\_piese.bd`.`modele\_masini` (`marca`, `model`, `an\_incepere\_fabricatie`, `an\_finalizare\_fabricatie`, `motorizare\_cc`, `putere\_cp`, `combustibil`, `cod\_motor`, `tip\_tractiune`) VALUES ('Peugeot', '508', 2011, 2014, '1997', '200', 'hybrid-diesel', 'RXH', '4x4');

***Popularea tabelului piese***

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('filtru aer vag', 'mann', '15000 km', '56');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('stergator luneta opel corsa', 'bosch', '2 ani', '99');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('filtru motorina vag', 'mann', '15000 km', '44');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `pret\_ron`) VALUES ('senzor parcare volvo', 'valeo', '54');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('placute frana dacia', 'trw', '40000 km', '164');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('placute frana bmw', 'brembo', '50000 km', '254');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('discuri frana dacia', 'trw', '60000 km', '204');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('filtru aer volvo', 'mann', '15000 km', '54');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('distributie volvo', 'continental', '180000 km', '896');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('distributie vag', 'luk', '210000 km', '1236');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `interval\_recomandat\_schimb`, `pret\_ron`) VALUES ('filtru ulei vag', 'mann', '15000 km', '40');

INSERT INTO `mag\_piese.bd`.`piese` (`denumire\_piesa`, `producator`, `pret\_ron`) VALUES ('valva control turbo volvo', 'continental', '568');

***Popularea tabelului masini***

INSERT INTO `mag\_piese.bd`.`masini` (`vin`, `nr\_inmatriculare`, `an\_fabricatie`, `kilometraj`) VALUES ('OP1WS530XY1123456', 'IF65BAM', 2017, '91000');

INSERT INTO `mag\_piese.bd`.`masini` (`vin`, `nr\_inmatriculare`, `an\_fabricatie`, `kilometraj`) VALUES ('TMBBE1234567890NJ', 'B67NJK', 2008, '189918');

INSERT INTO `mag\_piese.bd`.`masini` (`vin`, `nr\_inmatriculare`, `an\_fabricatie`, `kilometraj`) VALUES ('FSSDE1234567890NJ', 'MM31HDD', 2008, '140000');

INSERT INTO `mag\_piese.bd`.`masini` (`vin`, `nr\_inmatriculare`, `an\_fabricatie`, `kilometraj`) VALUES ('GREWE1234567890NJ', 'PH31ADE', 2005, '343232');

INSERT INTO `mag\_piese.bd`.`masini` (`vin`, `nr\_inmatriculare`, `an\_fabricatie`, `kilometraj`) VALUES ('YV1WS530XY1123456', 'B555BXD', 2012, '184236');

INSERT INTO `mag\_piese.bd`.`masini` (`vin`, `nr\_inmatriculare`, `an\_fabricatie`, `kilometraj`) VALUES ('YV1WS530XY1323456', 'B777BXD', 2013, '84236');

INSERT INTO `mag\_piese.bd`.`masini` (`vin`, `nr\_inmatriculare`, `an\_fabricatie`, `kilometraj`) VALUES ('YV1MS98D123456789', 'B502BGD', 2012, '168116');

***Popularea tabelului piesa\_model***

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('1', '1');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('1', '7');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('1', '8');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('1', '10');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('1', '11');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('1', '12');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('1', '16');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('2', '5');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('3', '1');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('3', '7');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('3', '8');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('3', '10');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('3', '11');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('3', '12');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('3', '16');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('4', '2');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('4', '6');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('4', '9');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`, `originala`) VALUES ('5', '4', 'compatibila');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`, `originala`) VALUES ('7', '4', 'compatibila');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('6', '3');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('8', '2');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('8', '6');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('8', '9');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('9', '2');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('10', '1');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('10', '7');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('10', '8');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('10', '10');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('10', '11');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('10', '12');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('10', '16');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('11', '1');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('11', '7');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('11', '8');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('11', '10');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('11', '11');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('11', '12');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('11', '16');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('12', '2');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('12', '6');

INSERT INTO `mag\_piese.bd`.`piesa\_model` (`id\_piesa`, `id\_model`) VALUES ('12', '9');

***Popularea tabelului comenzi***

INSERT INTO `mag\_piese.bd`.`comenzi` (`data\_comanda`, `masina`, `client`, `pregatita\_de\_angajat`, `tip\_livrare`, `status\_comanda`) VALUES ('2022-01-12', 'YV1WS530XY1123456', '5014324324338', '1', '1', '1');

INSERT INTO `mag\_piese.bd`.`comenzi` (`data\_comanda`, `masina`, `client`, `pregatita\_de\_angajat`, `tip\_livrare`, `status\_comanda`) VALUES ('2022-04-28', 'TMBBE1234567890NJ', '5014324324338', '5', '1', '2');

INSERT INTO `mag\_piese.bd`.`comenzi` (`data\_comanda`, `masina`, `client`, `pregatita\_de\_angajat`, `tip\_livrare`, `status\_comanda`) VALUES ('2022-04-28', 'GREWE1234567890NJ', '5015232523532', '5', '1', '2');

INSERT INTO `mag\_piese.bd`.`comenzi` (`data\_comanda`, `masina`, `client`, `pregatita\_de\_angajat`, `tip\_livrare`, `status\_comanda`) VALUES ('2022-05-14', 'FSSDE1234567890NJ', '5014324234238', '9', '1', '1');

INSERT INTO `mag\_piese.bd`.`comenzi` (`data\_comanda`, `masina`, `client`, `pregatita\_de\_angajat`, `tip\_livrare`, `status\_comanda`) VALUES ('2022-05-14', 'TMBBE1234567890NJ', '5014324324338', '5', '2', '1');

INSERT INTO `mag\_piese.bd`.`comenzi` (`data\_comanda`, `masina`, `client`, `pregatita\_de\_angajat`, `tip\_livrare`, `status\_comanda`) VALUES ('2022-05-14', 'YV1WS530XY1123456', '5014324324338', '5', '1', '1');

INSERT INTO `mag\_piese.bd`.`comenzi` (`data\_comanda`, `masina`, `client`, `pregatita\_de\_angajat`, `tip\_livrare`, `status\_comanda`) VALUES ('2022-04-28', 'GREWE1234567890NJ', '5010502430038', '5', '2', '1');

***Popularea tabelului comanda\_piesa***

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('1', '4', '1');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('1', '12', '1');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('2', '1', '1');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('2', '3', '2');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('2', '11', '1');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('3', '5', '2');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('3', '7', '2');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('4', '3', '3');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('4', '1', '1');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('5', '3', '1');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('5', '1', '1');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('6', '4', '3');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('7', '1', '1');

INSERT INTO `mag\_piese.bd`.`comanda\_piesa` (`id-comanda`, `id-piesa`, `nr\_piese`) VALUES ('7', '3', '1');

***Popularea tabelului livrari***

INSERT INTO `mag\_piese.bd`.`livrari` (`awb`, `livrata\_de\_angajat`, `comanda`, `adresa`, `status\_livrare`, `data\_programata\_livrare`) VALUES ('DFASF8DSADAF8AS8FAS8', '6', '5', '1', '1', '2022-05-15');

INSERT INTO `mag\_piese.bd`.`livrari` (`awb`, `livrata\_de\_angajat`, `comanda`, `adresa`, `status\_livrare`, `data\_programata\_livrare`) VALUES ('DJJN2IK31I2314913942', '6', '7', '2', '1', '2022-05-15');

# Interacțiunea cu baza de date

## Introducere

Pentru Interacțiunea cu baza de date am ales să creez o interfață web cu ajutorul modulului Django din limbajul programare Python. Am încărcat proiectul pe platforma Github unde am pus și un fișier cu instrucțiunile necesare pentru a porni interfața web.

Link-ul către github este urmatorul: <https://github.com/Bogdan2435/magazin_piese_auto> .

La prima accesare a interfeței vom fi redirecționați spre pagina de home, unde vom avea mai multe opțiuni de interacțiune cu baza de date.

Figura 1 – Pagina de Home.

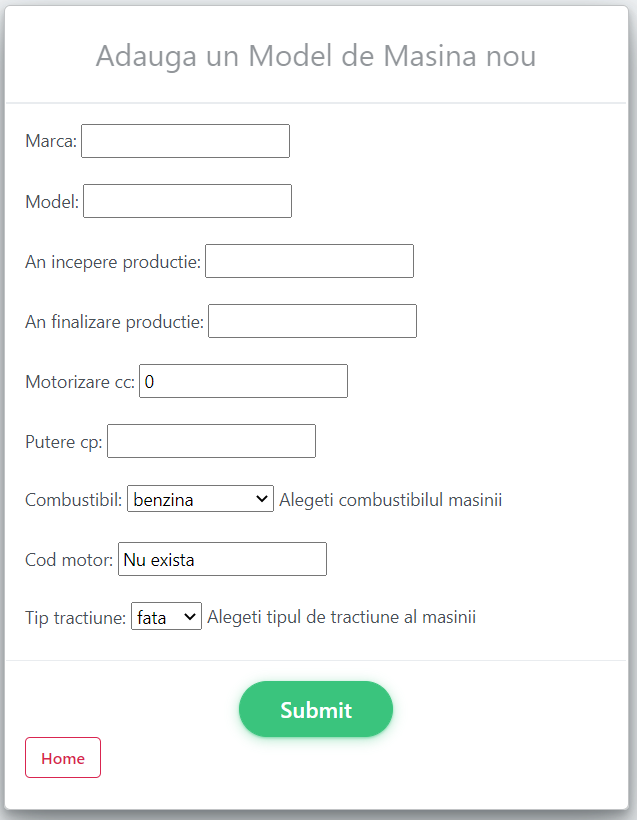
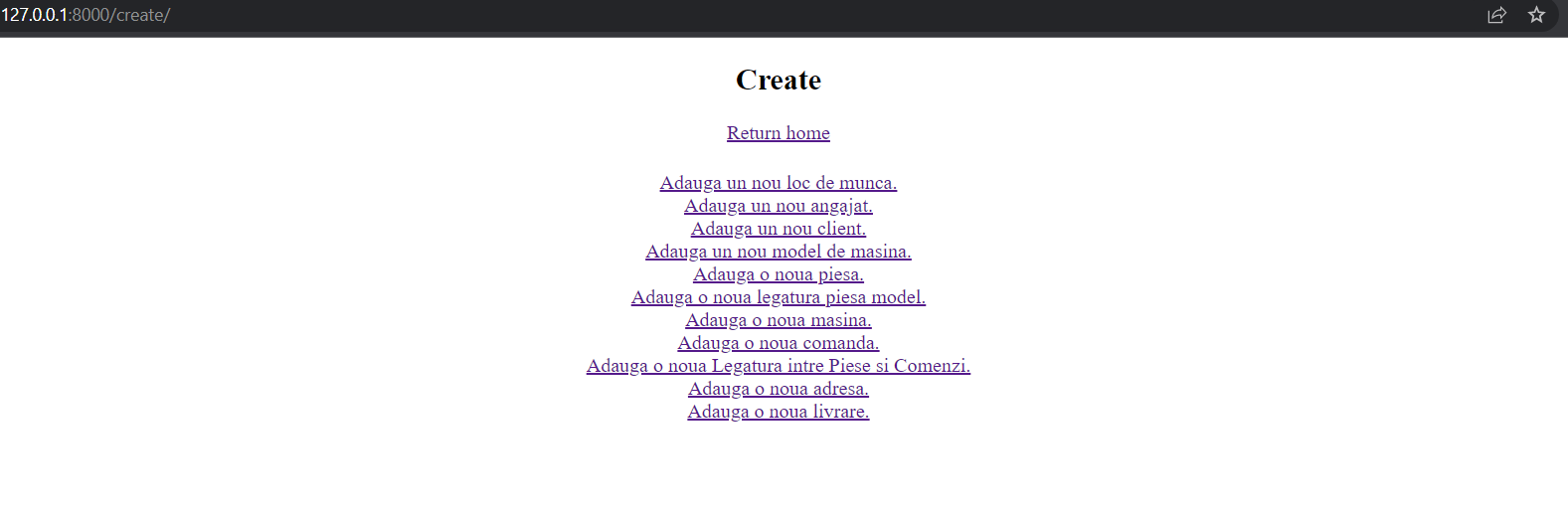
Figura 2 – Pagina Create.

Figura 3 – Adăugarea unui model nou de mașină.

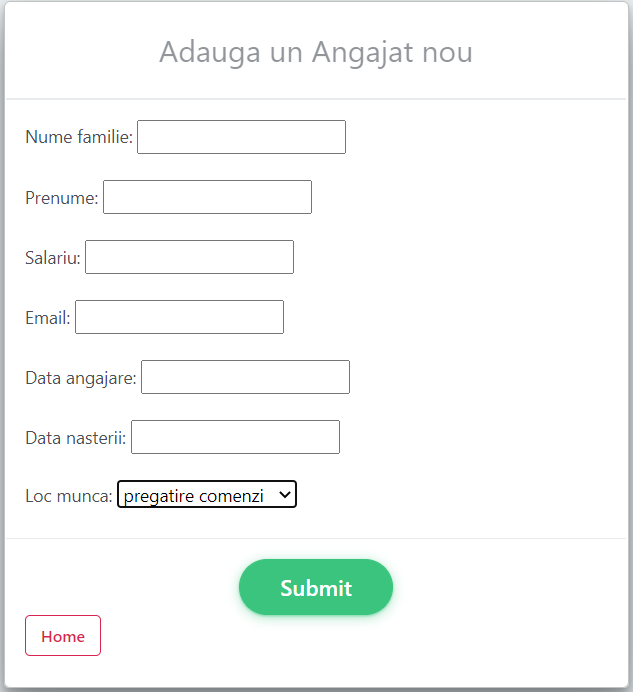


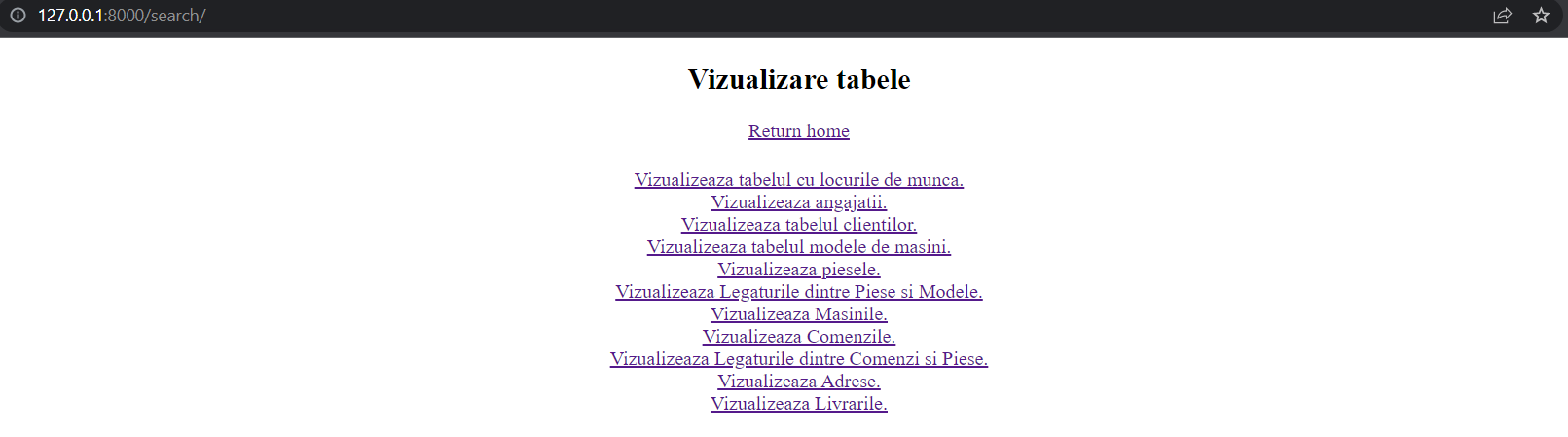
Figura 4 – Adăugarea unui angajat in baza de date.

Figura 5 – Pagina de unde pot selecta tabelul pe care doresc să îl vizualizez.

## Listarea conținutului

Pentru fiecare tabel din baza de date am creat câte o pagină pentru a putea vizualiza informațiile. Informațiile din fiecare tabel le pot sorta după fiecare coloană, în mod crescător sau descrescător, dar am implementat și funcția de căutare.

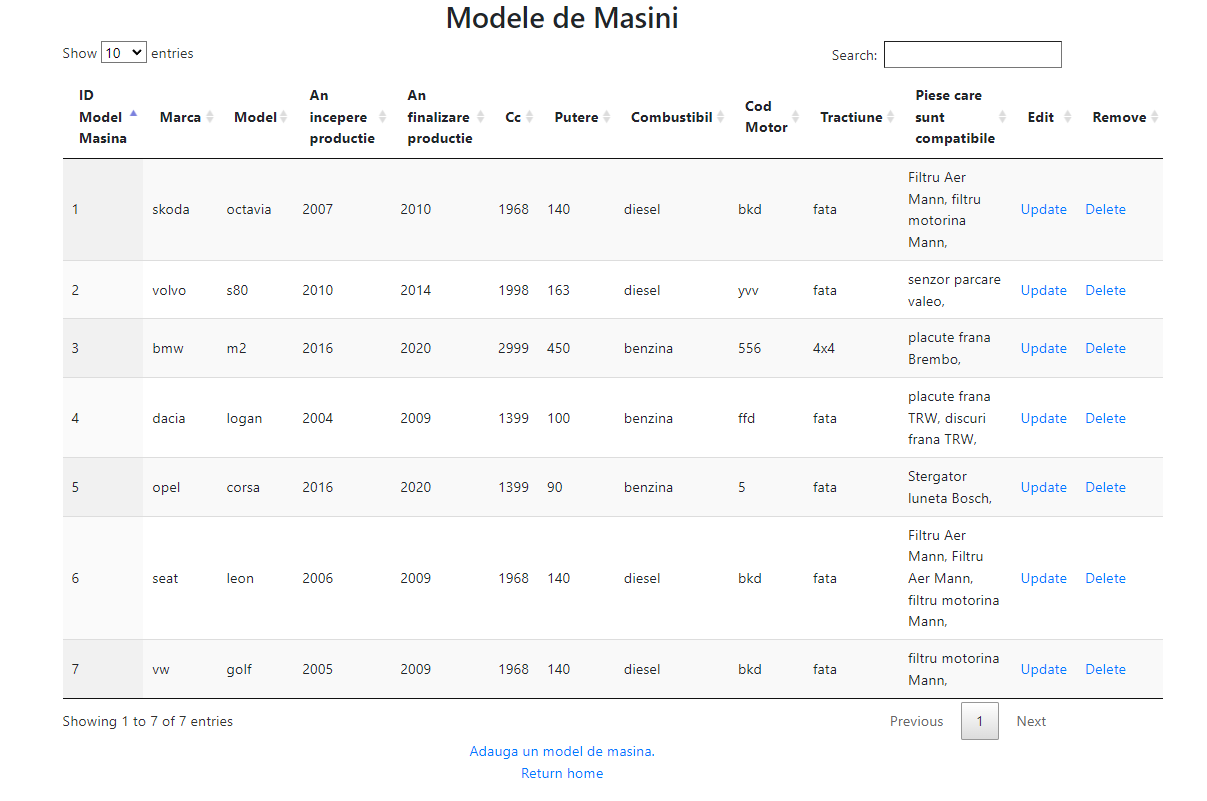


Figura 6 – Vizualizarea modelelor de mașini ordonat după id.

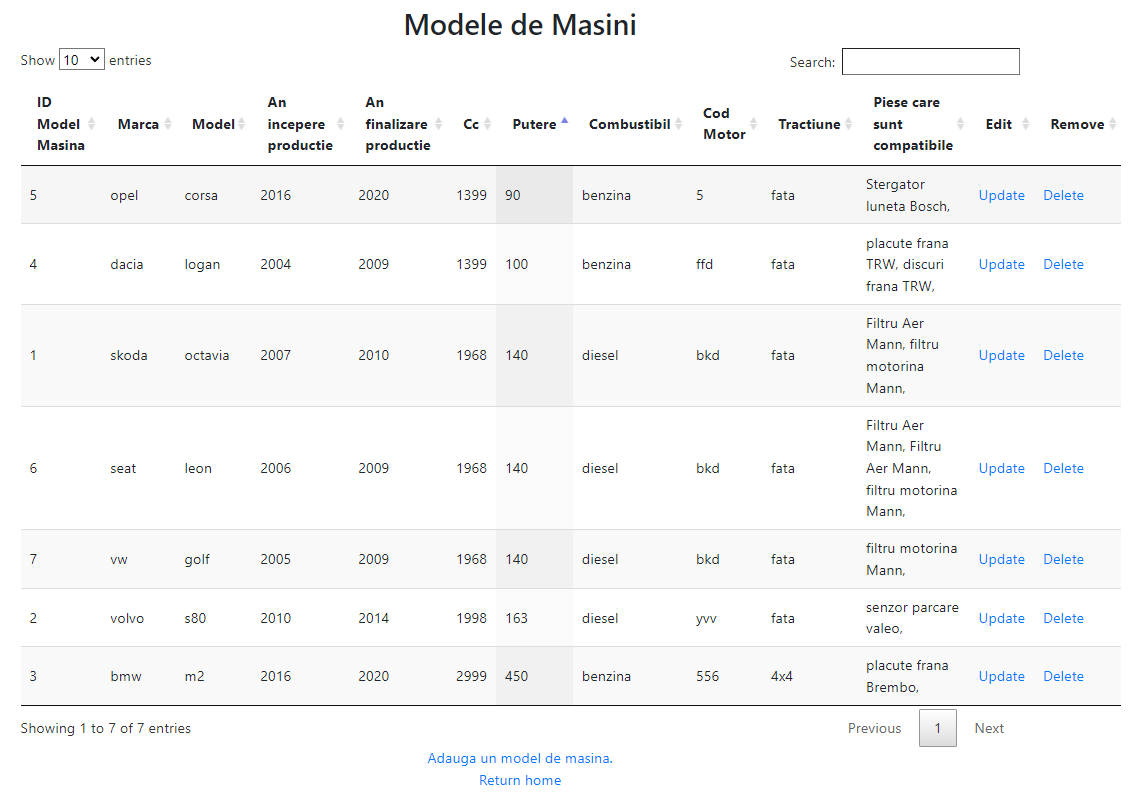


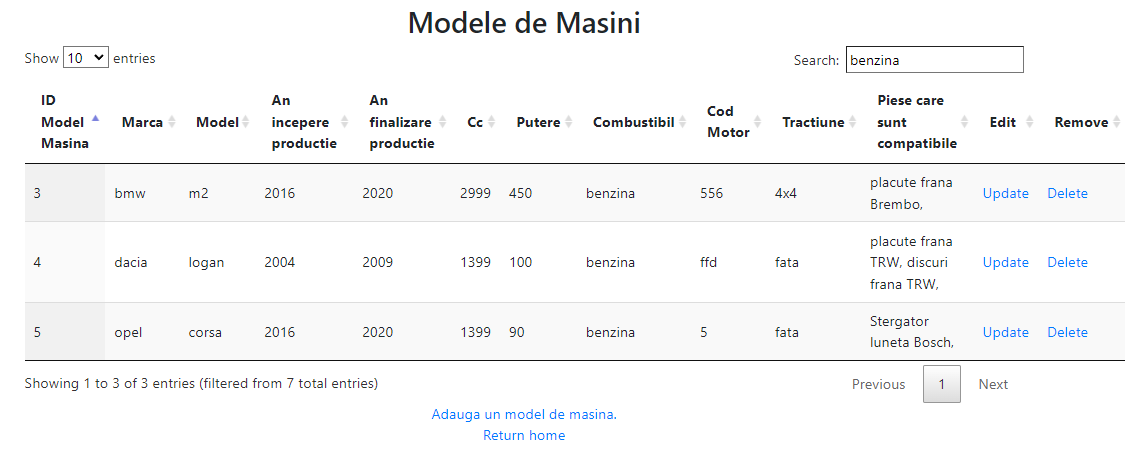
Figura 7 – Vizualizarea tabelului ordonat crescător după puterea mașinilor.

Figura 8 – Vizualizarea mașinilor ce folosesc benzină.

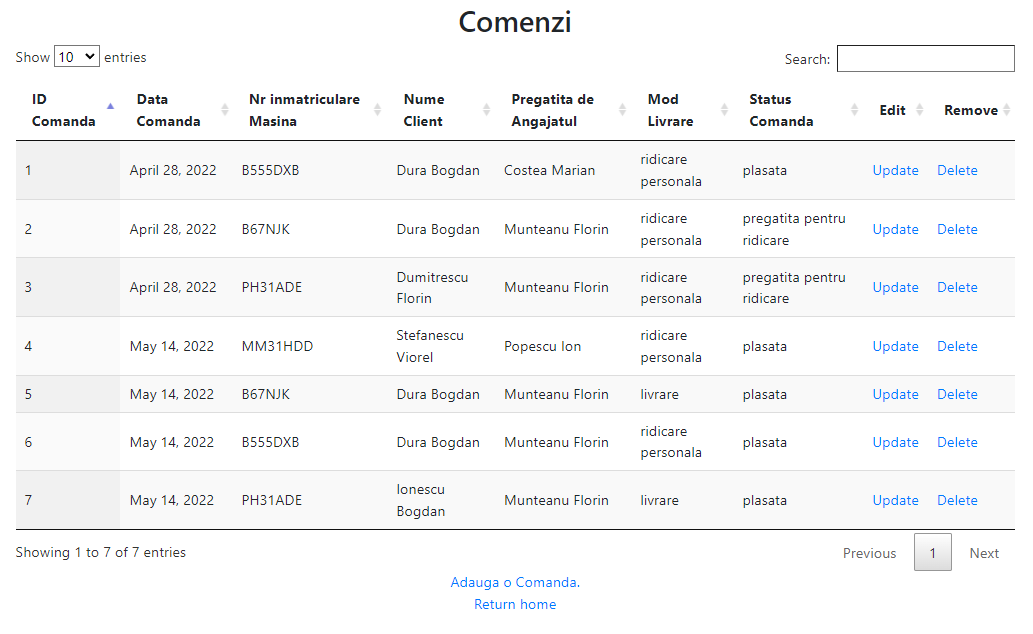


Figura 9 – Tabelul Comenzilor

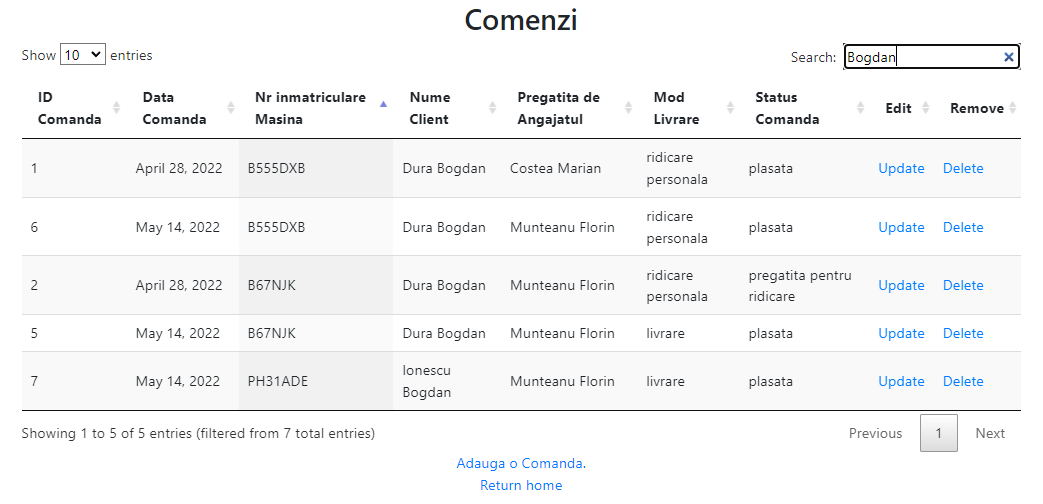
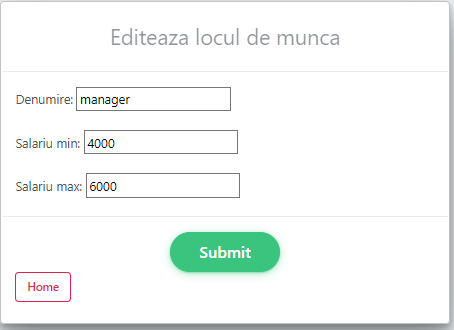


Figura 10 – Tabelul Comenzilor, căutare Bogdan

## Modificarea Informațiilor

Pentru a modifica sau a șterge informațiile, pentru fiecare intrare din fiecare tabel, avem doua butoane, update și search ce redirecționează spre paginile destinate funcțiilor de editare și ștergere.

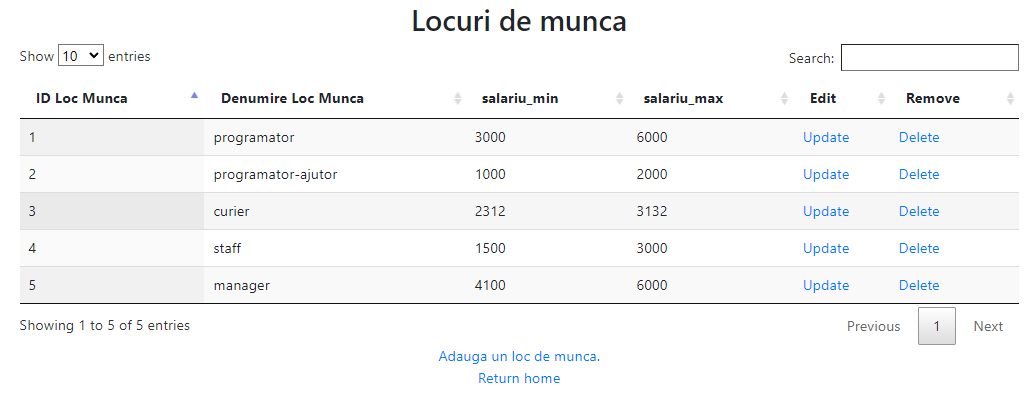
Fingura 11 – Editarea locului de muncă ‘’manager’’.

Figura 12 – Putem observa că modificarea locului de muncă a avut efect.

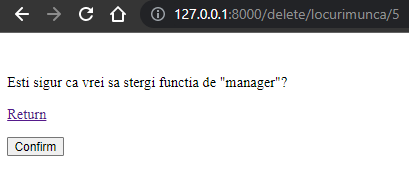
Figura 13 – Pagina de stergere a unei intrări.



Figura 14 – Putem observa că ștergerea locului de muncă a avut efect.

## Afișarea rezultatului unei cereri care extrage informații din cel puțin 3 tabele

Pentru a vedea rezultatul unei cereri ce extrage informații din cel puțin 3 tabele și le flirtează cu ajutorul a cel puțin două condiții vom intra selecta din „home”, rezolvarea cerinței c).

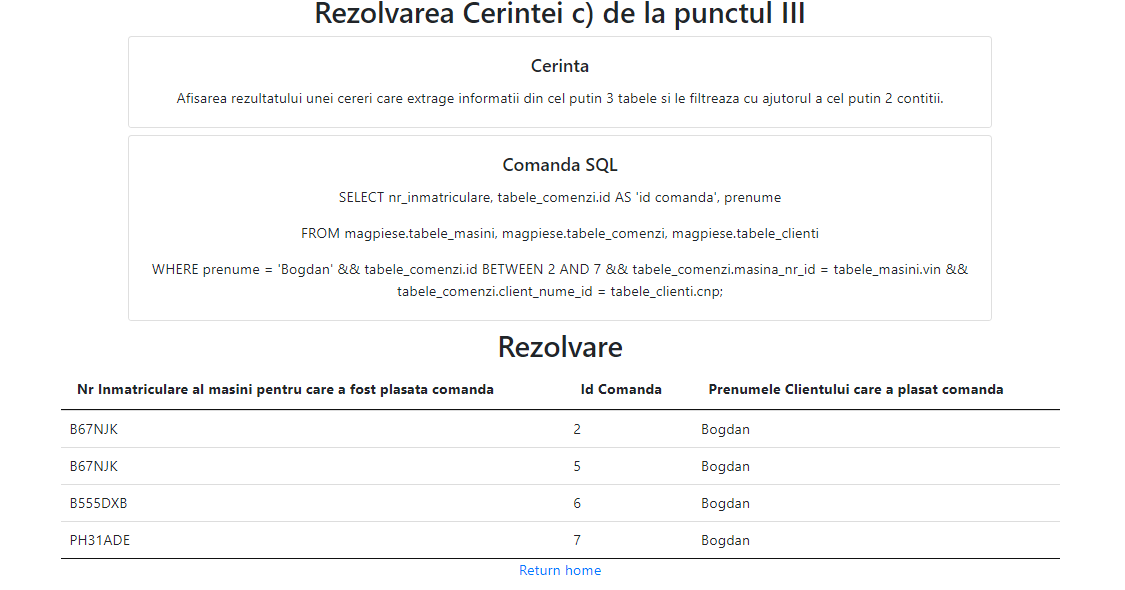


Figura 15 – Rezolvarea cerinței c)

## Rezultatul unei cereri ce folosește o clauză having

La fel ca în cazul cerinței c), va trebui să selectăm rezolvarea cerinței d) din pagina „home”. Dorim să selectăm pentru fiecare loc de muncă codul și salariul celui mai prost plătit angajat. Vom exclude locurile de muncă unde salariul minim este de peste 3000 de ron.

Putem observa că pentru locul de muncă cu id-ul 1, toți angajații au salariul de peste 3000 ron iar pentru locurile de muncă programator ajutor, curier și staff, salarile minime sunt 2132, 2400 și respectiv 1700.

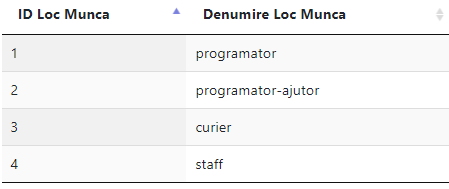


Figura 16 – Id-ul si denumirile locurilor de muncă

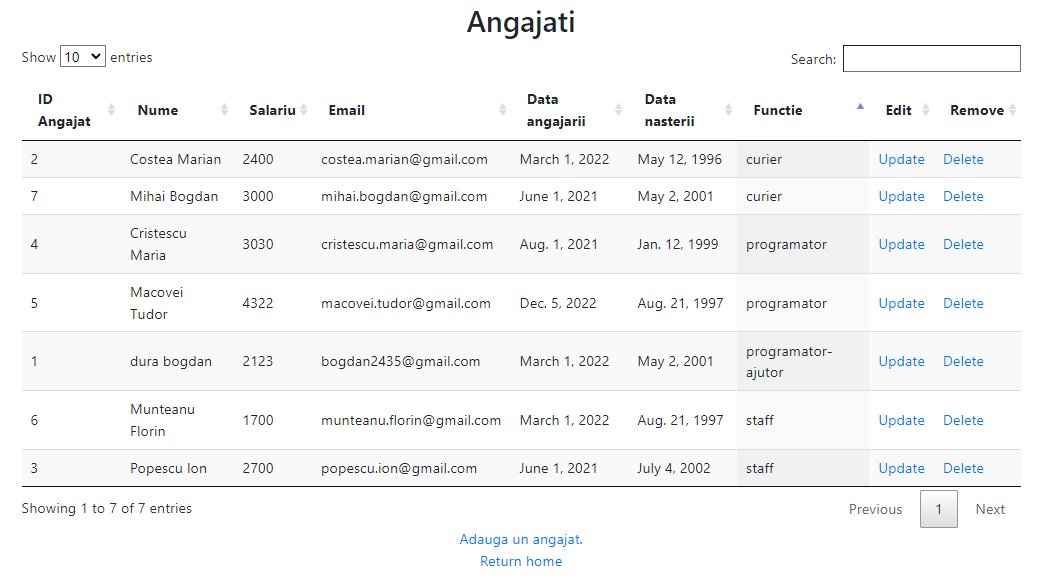
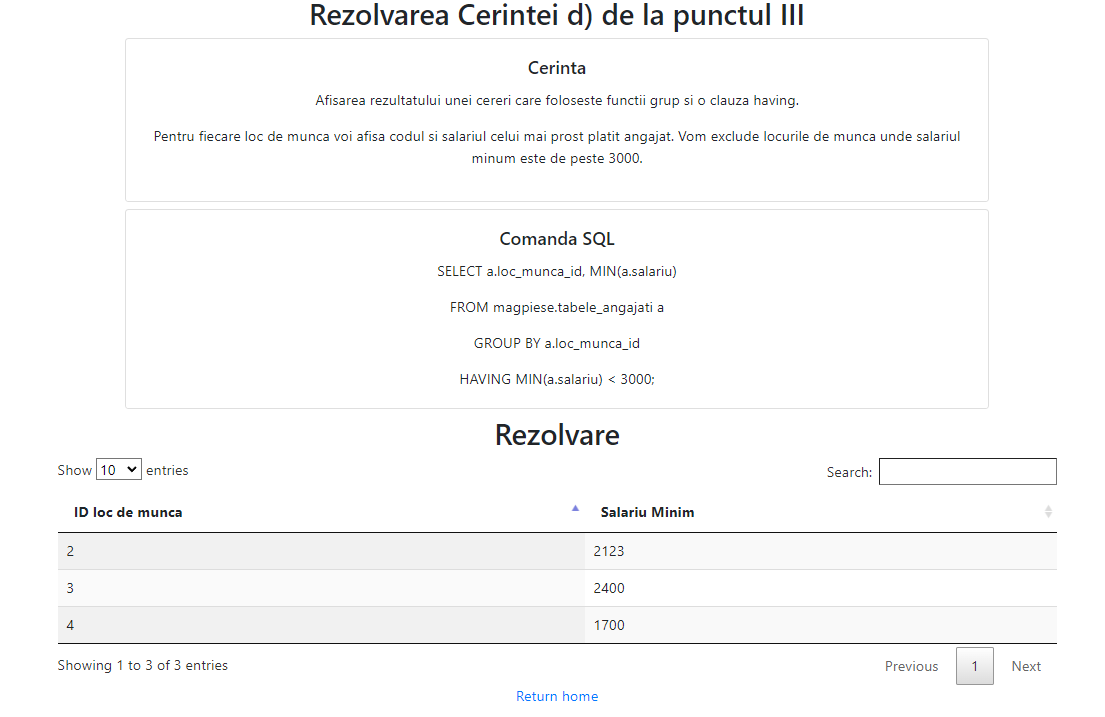


Figura 17 – Datele din tabelul Angajaților

Figura 18 – Rezolvarea cerinței d)

## Exemplificarea unei constrângeri de tip ON DELETE CASCADE

Voi exemplifica efectele constrângerii de tip „ON DELETE CASCADE” dintre tabelul modelelor de mașini și tabelul legăturilor dintre piese și modele. Atunci când sterg un model de mașină, trebuie să fie șterse automat și legăturile din care acest model de mașină face parte.

Pentru simplitate, am adăugat un model de mașină și o piesă, ambele denumite „test”.

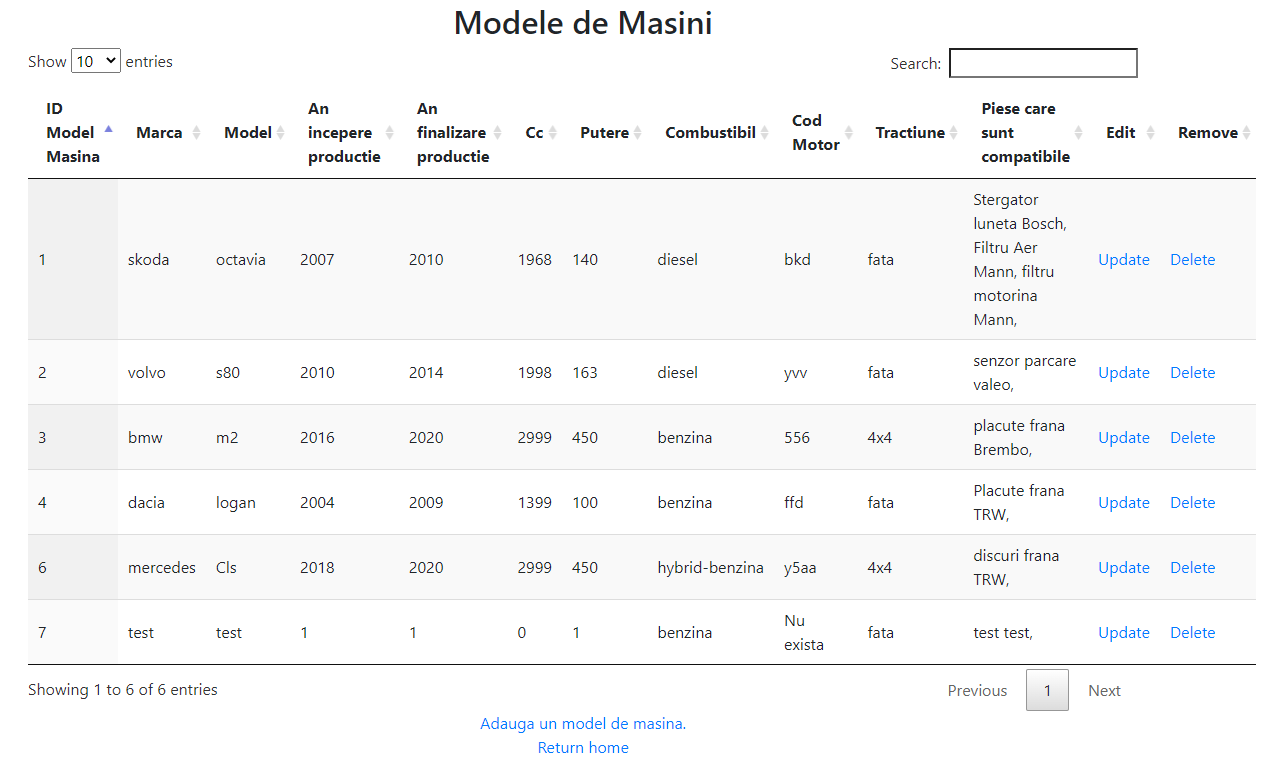
Figura 19 – Tabelul modelelor de mașini

Figura 20 – Tabelul pieselor.

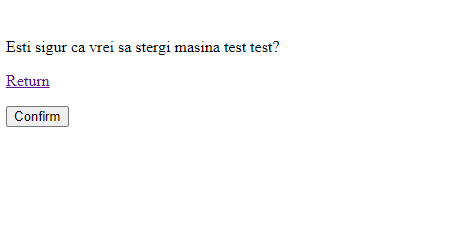
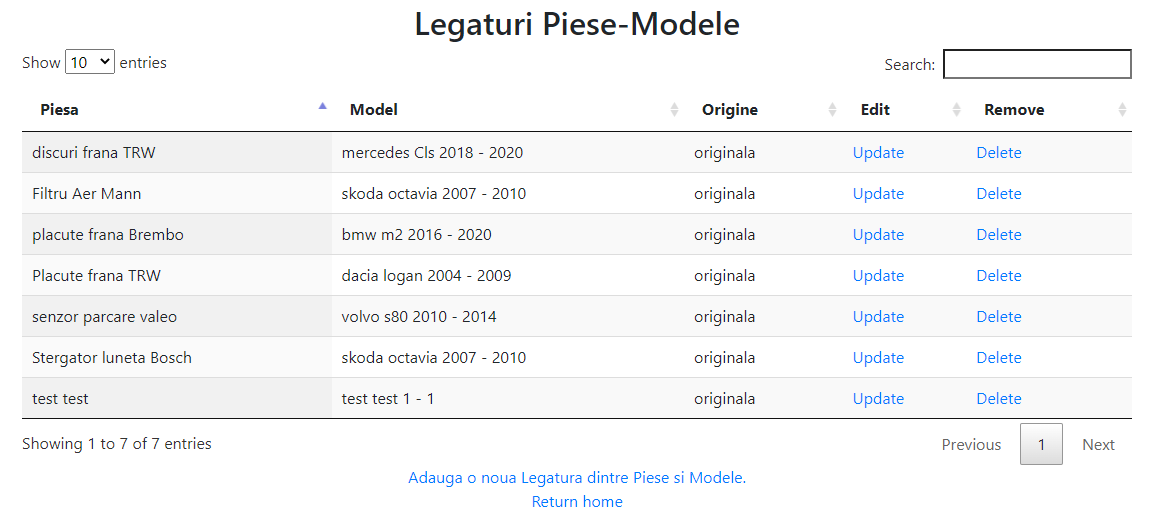
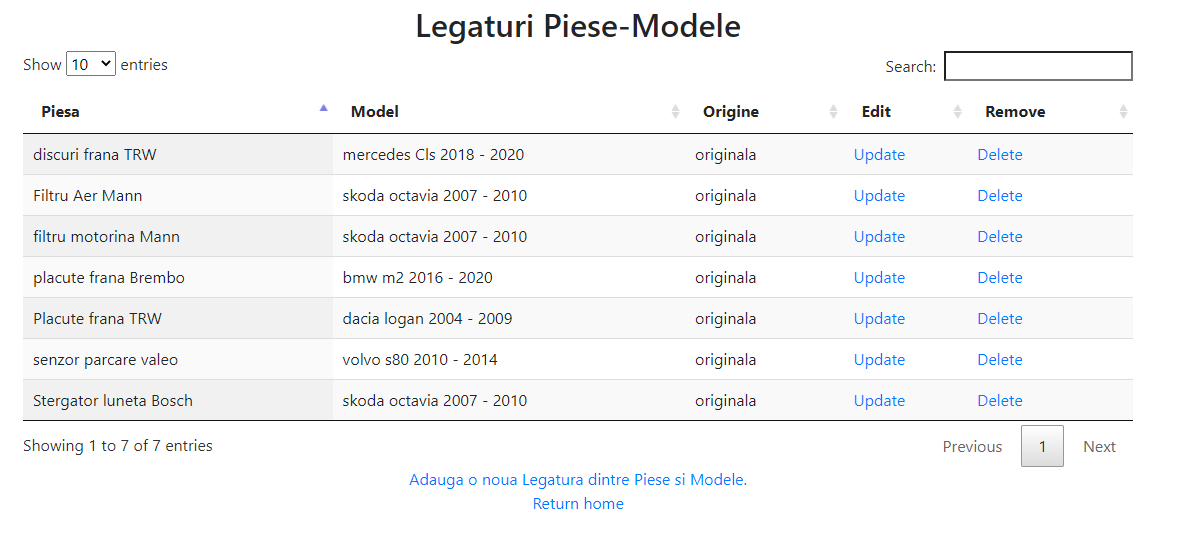
Figura 21 – Tabelul Legăturilor dintre piese și modele.

Figura 22 – Ștergerea Mașinii de test.

Figura 23 – Tabelul de legături după ștergerea mașinii de test.

Putem observa în pozele de mai sus că odată ce am șters un model de mașină din baza de date, automat se vor șterge din tabelul de legătură, toate intrările asociate cu respectivul model de mașină. Acest lucru poate fi exemplificat foarte ușor și în cazul tabelelor comenzi și livrări, unde dacă ștergem o comandă, atunci și livrarea asociată ei, va fi ștearsă.

# Interacțiunea cu baza de date (din modul de admin al site-ului)

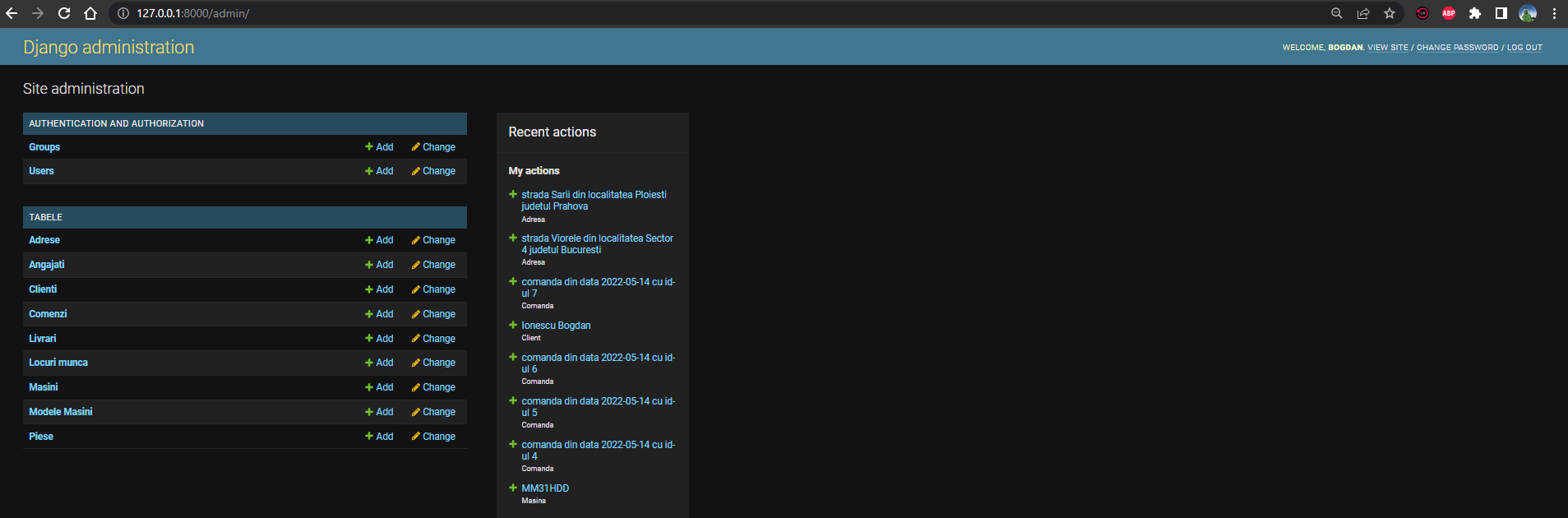
Un alt mod de interacționare cu baza de date poate fi direct din partea de Admin a site-ului, pe care o putem accesa la adresa 127.0.0.1:8000/admin.

Figura 24 – Admin Django.

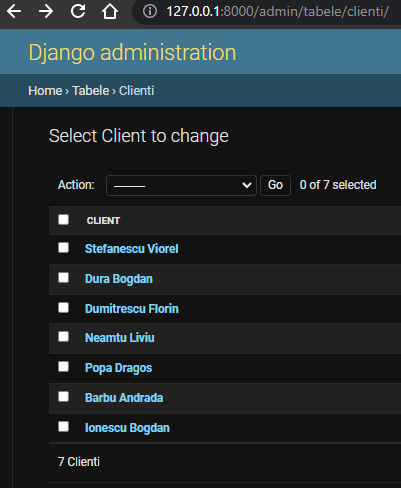


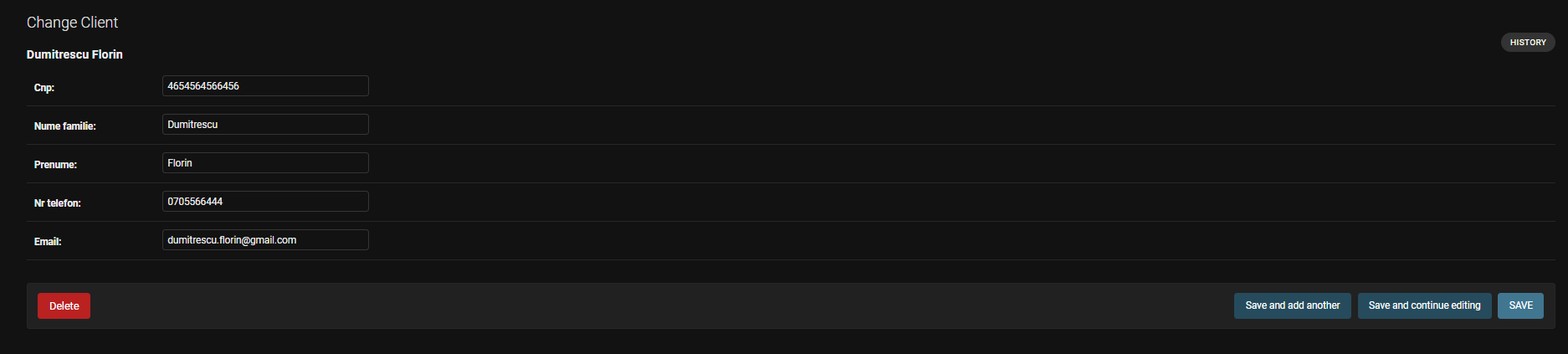
Figura 25 – Vizualizarea Clienților din baza de date.

Figura 26 – Modificarea informațiilor din baza de date.

Figura 27 – Modificarea informațiilor din baza de date.