

Universitatea din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Tehnologia Informației

PROIECT – BAZE DE DATE

COORDONATOR ȘTIINȚIFIC :

Lector Univ. Dr. Vasile Silviu Laurențiu

STUDENT:

Nicoi Alexandru

BUCUREȘTI
2020

Universitatea din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Tehnologia Informației

MANAGEMENTUL UNUI DEPOU DE LOCOMOTIVE

COORDONATOR ȘTIINȚIFIC :

Lector Univ. Dr. Vasile Silviu Laurențiu

STUDENT:

Nicoi Alexandru

BUCUREȘTI
2020

Cuprins

1. Prezentare bază de date.....	4
1.1. Prezentare model din lumea reală.....	4
1.2. Reguli generale.....	4
2. Diagrama Entitate-Relație.....	5
2.1. Ilustrație diagramă.....	5
2.2. Descrierea componentelor diagramei.....	6
3. Diagrama conceptuală.....	8
3.1. Ilustrație diagramă.....	8
3.2. Descriere constrângeri de integritate.....	8
3.3. Scheme relaționale.....	11
4. Procesul de implementare.....	12
4.1. Crearea tabelor.....	12
4.2. Introducerea datelor.....	17

1. Prezentare bază de date

1.1. *Prezentare model din lumea reală*

Ramura feroviară a mijloacelor de transport reprezintă o componentă importantă în dezvoltarea lumii, transportul feroviar reprezentând o soluție optimă pentru transportul de persoane în siguranță, linia de cale ferată fiind constantă, fără multe viraje, dar și o soluție pentru transportul de mărfuri, deoarece prin intermediul căilor ferate pot fi transportate cantități de marfă de tonaj ridicat.

Motivația mea de a alege să realizez această temă este dată de pasiunea mea pentru lumea feroviară, de unde am acumulat multe informații interesante pe care le pot transpune într-o bază de date relațională.

În România, ansamblul feroviar este împărțit pe mai multe regionale ale Căilor Ferate Române. Unul dintre ele este „Regionala Muntenia Sud” care are mai multe depouri de locomotive, printre ele fiind regăsite „Depoul București Basarab”, „Depoul București Grivița”, sau „Depoul CFR Ploiești”.

Fiecare depou are o structură bine organizată, unde elementul principal este locomotiva, de unde se pot extrage diferite informații precum operatorul, mecanicii, detaliile tehnice și așa mai departe.

1.2. *Reguli generale*

Baza de date își propune stocarea datelor esențiale la realizarea și coordonarea depoului, bazată pe următoarele particularități:

- O locomotivă are întotdeauna un producător cunoscut.
- Orice locomotivă trebuie să aibă un rang setat pe sistemul său de siguranță numit INDUSI, sistem care comunică cu reductoarele de viteză amplasate pe linia de cale ferată, în apropierea semnalelor luminoase. Totodată prin rangul său cunoaștem și viteza care trebuie adaptată la semnale (V1 – reductorul de 1000 hz, V2 – reductorul de 500 hz).

2.2. Descrierea componentelor diagramei

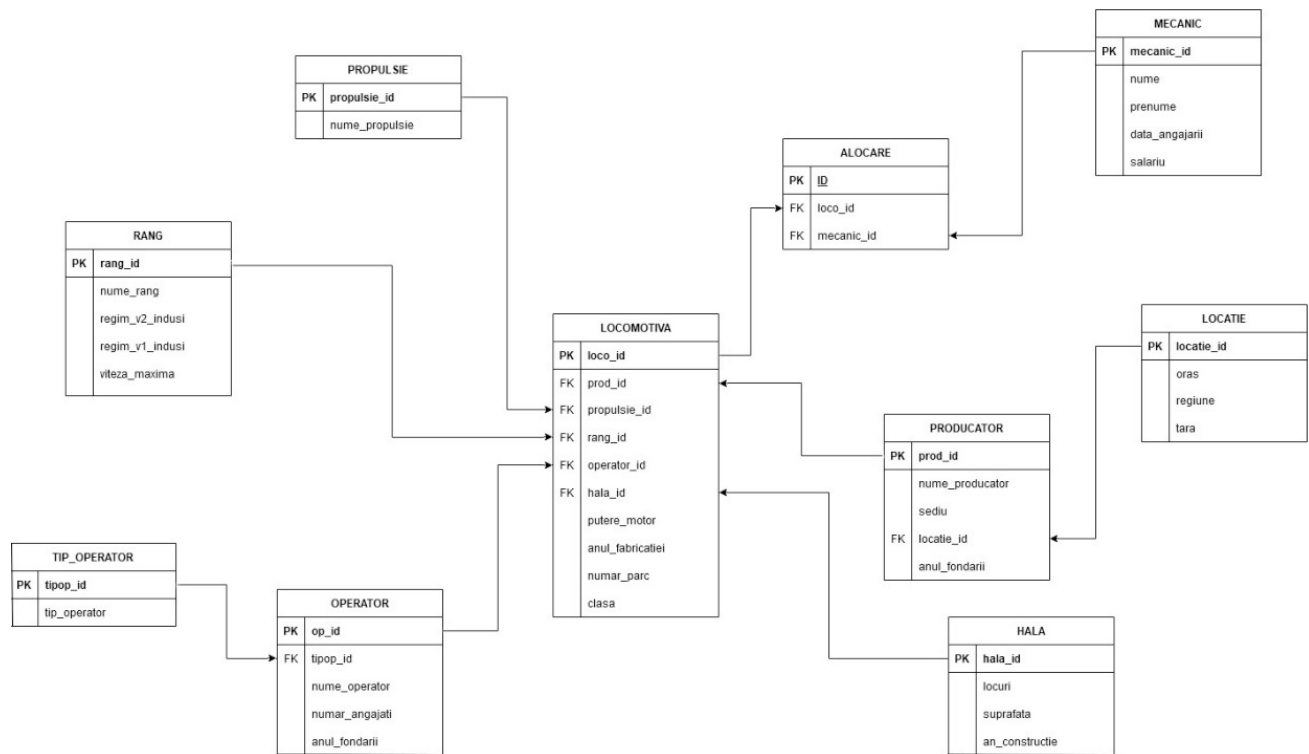
Pentru funcționarea acestei baze de date, am ales să compun 9 tabele care vor descrie într-un mod eficient structura unui depou de locomotive.

- Tabelul „Locomotivă” cuprinde particularitățile fiecărei locomotive în parte, caracteristici esențiale pentru găsirea informațiilor necesare în cadrul bazei de date. Pentru orice locomotivă trebuie să cunoaștem date despre producător, tipul de propulsie, tipul rangului, date despre operator, despre hală, dar și specificații tehnice precum puterea sa, anul fabricației, numărul de parc alocat și clasa de care aparține. În plus, locomotivele pot fi conduse de mai mulți mecanici, de aceea între tabelele „Locomotivă” și „Mecanic” există o relație de $M(1) - M(1)$, adică un mecanic poate opera pe mai multe locomotive, dar și locomotiva poate fi operată de mai mulți mecanici. Cheia primară de identificare a unei locomotive este unică, având denumirea „loco_id”.
- Tabelul „Mecanic” conține datele mecanicului de locomotivă, acestea fiind numele și prenumele său, data angajării, precum și salariul. Acest tabel se află în relație $M(1) - M(1)$ cu locomotiva datorită motivului enunțat mai sus. Cheia primară de identificare este „mecanic_id”.
- Tabelul „Operator” memorează datele operatorului de cale ferată. Este important să cunoaștem numele operatorului, numărul de angajați, anul fondării, dar și tipul de operator. Între „Tip operator” și „Operator” există o relație de $1(1) - M(1)$ deoarece un operator aparține unui anumit tip, iar un anumit tip poate fi atribuit mai multor operatori. Cheia primară de identificare a unui operator este „op_id”.
- Tabelul „Tip Operator” conține tipul de operator de cale ferată. Este important să cunoaștem acest tip pentru a ști natura finanțării operatorului de cale ferată. Cheia primară de identificare este „tipop_id”.
- Tabelul „Hala ” stochează date despre hala în care sunt alocate locomotivele, date precum numărul de locuri, suprafața și anul construcției. Acest tabel se află în relație de $1(1) - M(0)$ cu tabelul locomotivă deoarece o hală poate conține mai multe locomotive, iar o locomotivă poate aparține unei singure hale. Cheia primară de identificare a unei hale este „hala_id”.

- Tabelul „Propulsie” conține tipul de propulsie care este prezent pe locomotivă. Este important să cunoaștem acest tip pentru a diferenția locomotivele. Acest tabel se află în relație de $1(1) - M(0)$ cu tabelul „Locomotivă” deoarece un tip de propulsie poate fi prezent pe mai multe locomotive, iar orice locomotivă trebuie să aibă un anumit tip de propulsie. Cheia primară de identificare a unui tip de propulsie este „propulsie_id”.
- Tabelul „Rang” memorează date importante în ceea ce privește utilizarea locomotivei, date precum numele rangului, regimul de viteză v_2 indusi, regimul de viteză v_1 indusi, precum și viteza maximă. Aceste ultime trei valori sunt importante deoarece prin acestea cunoaștem vitezele cu care traversează o gară (în cazul valorilor de indusi), precum și viteza maximă cu care circulă pe linie. Viteza V_1 se îndeplinește la semnalul prevestitor semnalului de intrare în gară, iar viteza V_2 la semnalul de intrare în gară. Acest tabel se află în relație de $1(1) - M(0)$ cu tabelul locomotivă. Cheia primară de identificare a rangului de tren este „rang_id”.
- Tabelul „Producător” conține datele specifice prin care poate fi caracterizat, cum ar fi numele producătorului, sediul, locația producătorului, precum și anul fondării. Acest tabel se află în relație de $1(1) - M(0)$ cu tabelul „Locomotivă” deoarece un producător poate avea mai multe locomotive, iar o locomotivă aparține unui anumit producător. Cheia primară de identificare a producătorului este „prod_id”.
- Tabelul „Locație ” conține date despre orașul, regiunea și țara unei fabrici. Cum în anumite orașe există mai multe fabrici ale unor producători diferiți, relația dintre tabelele „Locație” și „Producător” este $1(1) - M(0)$. Cheia primară de identificare a locației unice este „locatie_id”.

3. Diagrama conceptuală

3.1. Ilustrație diagramă



3.2. Descriere constrângeri de integritate

Tabelul TIP_OPERATOR:

- NOT NULL:
 - tip_operator - varchar(255).
- PRIMARY KEY:
 - tipop_id - number(1).

Tabelul OPERATOR:

- NOT NULL:
 - tipop_id - number(1);
 - nume_operator - varchar(255);
 - numar_angajati - numeric(8);
 - anul_fondarii - numeric(4).
- PRIMARY KEY:

- op_id - number(3).
- FOREIGN KEY:
 - tipop_id – number(1).

Tabelul RANG:

- NOT NULL :
 - nume_rang – varchar(255).
- CHECK :
 - regim_v1_indusi – numeric(2) - >0 (viteza nu poate fi negativă);
 - regim_v2_indusi – numeric(2) - >0 (viteza nu poate fi negativă);
 - viteza_maxima – numeric(3) - >0 (viteza nu poate fi negativă).
- PRIMARY KEY:
 - rang_id – number(1).

Tabelul PROPULSIE:

- NOT NULL:
 - nume_propulsie – varchar(255).
- PRIMARY KEY:
 - propulsie_id – number(1).

Tabelul HALA:

- NOT NULL:
 - an_constructie – numeric(4).
- CHECK:
 - locuri – numeric(4) - >= 0 (numarul de locuri nu poate fi negativ);
 - suprafata – numeric(4) - >= 0 (suprafata nu poate fi negativa).
- PRIMARY KEY:
 - hala_id – number(3).

Tabelul LOCATIE:

- NOT NULL:
 - oras – varchar(255);
 - regiune – varchar(255);
 - tara – varchar(255).
- PRIMARY KEY:

- locatie_id – number(3).

Tabelul PRODUCATOR:

- NOT NULL:
 - nume_producator – varchar(255);
 - sediu – varchar(255);
 - locatie_id – numeric(3);
 - anul_fondarii – numeric(4).
- PRIMARY KEY:
 - prod_id – number(3).
- FOREIGN KEY:
 - locatie_id – numeric(3).

Tabelul MECANIC:

- NOT NULL:
 - nume – varchar(255);
 - prenume – varchar(255);
 - data_angajarii – date;
 - salariu – float(10).
- CHECK:
 - salariu - > 0.
- PRIMARY KEY:
 - mecanic_id – number(3).

Tabelul LOCOMOTIVA:

- NOT NULL:
 - prod_id – numeric(3);
 - propulsie_id – numeric(1);
 - rang_id – numeric(1);
 - op_id – numeric(3);
 - hala_id – numeric(3);
 - putere_motor – numeric(10);
 - anul_fabricatiei – numeric(4);
 - numar_parc – numeric(4);

- clasa – numeric(3);
- CHECK:
 - putere_motor - > 0
- PRIMARY KEY:
 - loco_id – number(5);
- FOREIGN KEY:
 - prod_id
 - propulsie_id
 - rang_id
 - op_id
 - hala_id

Tabelul ALOCARE:

- NOT NULL:
 - mecanic_id – number(3);
 - loco_id – number(5);
- PRIMARY KEY:
 - ID – number(3).
- FOREIGN KEY:
 - mecanic_id – number(3);
 - loco_id – number(5);

3.3. Scheme relaționale

În cadrul bazei de date, am utilizat opțiunea „on delete cascade”. Mai jos voi detalia ce se întâmplă în momentul ștergerii unei înregistrări care corespunde unei chei străine:

- Tabelul „OPERATOR” are cheia străină „tipop_id” care face legătura cu tabelul „TIP_OPERATOR”. În situația în care se va șterge o înregistrare din tabelul „TIP_OPERATOR”, se vor șterge toți operatorii care aparțineau acelei înregistrări deoarece nu se va îndeplini condiția obligatorie de existență (orice operator trebuie să aparțină unui anumit tip de operator)
- Tabelul „PRODUCATOR” are cheia străină „locatie_id” care face legătura cu tabelul „LOCATIE”. În momentul în care se șterge o înregistrare din „LOCATIE”, se va șterge și

înregistrarea și producătorul care avea acele date despre locație deoarece, nu există un producător care să nu aibă o locație.

- Tabelul „LOCOMOTIVA” are diverse chei străini de care depinde orice înregistrare din acest tabel. Cheia „prod_id” face legătură cu tabelul „PRODUCATOR”, cheia „propulsie_id” cu tabelul „PROPULSIE”, cheia „rang_id” cu tabelul „RANG”, cheia „operator_id” cu tabelul „OPERATOR”, iar cheia „hala_id” cu tabelul „HALA”. În momentul în care o înregistrare din tabelele conectate prin intermediul cheilor străine va fi ștearsă, implicit și locomotiva cu dependența spre acea înregistrare va fi ștearsă. Aceste date sunt extrem de prețioase și nu au cum să lipsească din datele unei locomotive, deoarece nu poate exista o locomotivă cu având aceste date (un producător, tip de propulsie, rang, operator sau hală) necunoscute.
- Tabelul „ALOCARE” este construit cu scopul de a îndeplini relația de MANY-TO-MANY între tabelul „LOCOMOTIVĂ” și „MECANIC”, prin intermediul cheilor străine care duc la cele două componente „loco_id” și „mecanic_id”. În momentul în care oricare din cele două înregistrări se șterg, automat și înregistrarea din alocare va fi ștearsă deoarece, nu are sens să menținem o alocare cu un element lipsă.

4. Procesul de implementare

4.1. Crearea tabelelor

```
CREATE TABLE propulsie
( propulsie_id number(1) CONSTRAINT pk_propulsie primary key,
  nume_propulsie varchar(255) not null
);
```

```
CREATE TABLE rang
( rang_id number(1) CONSTRAINT pk_rang primary key,
  nume_rang varchar(255) not null,
  regim_v1_indusi numeric(2) constraint check_regimv1 check
(regim_v1_indusi > 0),
  regim_v2_indusi numeric(2) constraint check_regimv2 check
(regim_v2_indusi > 0) ,
  viteza_maxima numeric(3) constraint check_vit_max check
(viteza_maxima > 0)
);
```

```
CREATE TABLE tip_operator
(
  tipop_id number(1) CONSTRAINT pk_tipop primary key,
  tip_operator varchar(255) CONSTRAINT vf_nume_rang not null
);
```

```
CREATE TABLE operator
(
  op_id number(3) constraint pk_op_id primary key,
  tipop_id numeric(1) not null,
  nume_operator varchar(255) not null,
```

```

    numar_angajati numeric(8) not null,
    anul_fondarii numeric(4) not null,
    constraint fk_op_tipop foreign key (tipop_id) references
tip_operator(tipop_id) on delete cascade
);

```

```

CREATE TABLE hala
(
    hala_id number(3) constraint pk_hala_id primary key,
    locuri numeric(4) constraint check_locuri check (locuri >= 0) ,
    suprafata numeric(4) constraint check_suprafata check (suprafata
>= 0) ,
    an_constructie numeric(4) not null
);

```

```

CREATE TABLE locatie
(
    locatie_id number(3) constraint pk_loc_id primary key,
    oras varchar(255) not null,
    regiune varchar(255) not null,
    tara varchar(255) not null
);

```

```

CREATE TABLE producator
(
    0 prod_id number(3) constraint pk_prod_id primary key,
    nume_producator varchar(255) not null,
    sediu varchar(255) not null,
    locatie_id numeric(3) not null,
    anul_fondarii numeric(4) not null,

```

```

    constraint fk_prod_loc foreign key (locatie_id) references
locatie(locatie_id) on delete cascade
);

```

```

CREATE TABLE mecanic

```

```

(
    mecanic_id number(3) constraint pk_mecanic_id primary key,
    nume varchar(255) not null,
    prenume varchar(255) not null,
    data_angajarii date not null,
    salariu float(10) not null,
    constraint check_salariu check (salariu>0)
);

```

```

CREATE TABLE locomotiva

```

```

(
    loco_id number(5) constraint pk_loco primary key,
    prod_id numeric(3) not null,
    propulsie_id numeric(1) not null,
    rang_id numeric(1) not null,
    op_id numeric(3) not null,
    hala_id numeric(3) not null,
    putere_motor numeric(10) not null,
    constraint check_power check (putere_motor > 0),
    anul_fabricatiei numeric(4) not null,
    numar_parc numeric(4) not null,
    clasa numeric(3) not null,
    constraint fk_locomotiva_prod
foreign key (prod_id) references producator(prod_id) on delete
cascade,
    constraint fk_locomotiva_propulsie

```

```

    foreign key (propulsie_id) references propulsie(propulsie_id) on
delete cascade,
    constraint fk_locomotiva_rang
    foreign key (rang_id) references rang(rang_id) on delete cascade,
    constraint fk_locomotiva_operator
    foreign key (op_id) references operator(op_id) on delete
cascade,
    constraint fk_locomotiva_hala
    foreign key (hala_id) references hala(hala_id) on delete cascade
);

```

```

CREATE TABLE alocare
(
    id number(3) constraint pk_aloc primary key,
    mecanic_id number(3) not null,
    loco_id number(5) not null,
    constraint fk_mecanic_aloc foreign key (mecanic_id) references
mecanic(mecanic_id) on delete cascade,
    constraint fk_loco_aloc foreign key (loco_id) references
locomotiva(loco_id) on delete cascade
);

```


4.2. *Introducerea datelor*

```
insert into propulsie values (1,'electric');
insert into propulsie values (2,'diesel-electric');
insert into propulsie values (3,'diesel-hidraulic');
insert into propulsie values (4,'aburi');

insert into rang values (1,'rapid',65,90,160);
insert into rang values (2,'personal',50,65,120);
insert into rang values (3,'marfa',40,50,100);

insert into tip_operator values (1,'de stat');
insert into tip_operator values (2,'privat');

insert into operator values (1,1,'CFR Calatori',12000,1998);
insert into operator values (2,1,'CFR Marfa',16220,1998);
insert into operator values (3,2,'Transferoviar
Calatori',400,2011);
insert into operator values (4,2,'Regio Calatori',700,2004);
insert into operator values (5,2,'Grup Feroviar Roman',1500,2001);
insert into operator values (6,2,'Servtrans',650,2002);

insert into hala values (1,12,1400,1995);
insert into hala values (2,25,2925,1980);
insert into hala values (3,30,3510,1970);

insert into locatie values (1,'Craiova','Dolj','Romania');
insert into locatie values (2,'Bucuresti','Bucuresti-
Ilfov','Romania');
insert into locatie values (3,'Pascani','Iasi','Romania');
insert into locatie values (4,'Munchen','Bavaria','Germania');
```

```

insert into locatie values (5,'Saint-Ouen','Paris','Franta');

insert into producator values (1,'Electroputere','Calea Bucuresti
80',1,1949);
insert into producator values (2,'Softronic','Calea Severinului
40',1,1999);
insert into producator values (3,'Electroputere VFU','Garii
18',3,1869);
insert into producator values (4,'Faur','Basarabia 256',2,1921);
insert into producator values (5,'Siemens','Wittelsbacherpl.
1',4,1989);
insert into producator values (6,'Alstom','Rue Albert Dhalenne
48',5,1989);

insert into mecanic values (1,'Popescu','Ionel',TO_DATE('27-01-
1971','dd-mm-yyyy'),3774);
insert into mecanic values (2,'Kogalniceanu','Carol',TO_DATE('29-
02-1972','dd-mm-yyyy'),3435);
insert into mecanic values (3,'Rebreanu','Marin',TO_DATE('25-08-
1973','dd-mm-yyyy'),4093);
insert into mecanic values (4,'Zamfir','Nicolas',TO_DATE('10-04-
1974','dd-mm-yyyy'),4152);
insert into mecanic values (5,'Hasdeu','Flaviu',TO_DATE('21-12-
1974','dd-mm-yyyy'),3892);
insert into mecanic values (6,'Cantacuzino','Bogdan',TO_DATE('28-
01-1975','dd-mm-yyyy'),3184);
insert into mecanic values (7,'Parasca','Mario',TO_DATE('14-11-
1976','dd-mm-yyyy'),4612);
insert into mecanic values (8,'Ciora','Nandru',TO_DATE('20-12-
1976','dd-mm-yyyy'),3382);
insert into mecanic values (9,'Vlaicu','Shaithis',TO_DATE('06-01-
1979','dd-mm-yyyy'),4182);

```

```

insert into mecanic values (10,'Raducan','Adam',TO_DATE('23-03-
1980','dd-mm-yyyy'),4947);
insert            into            mecanic            values
(11,'Marandici','Catarino',TO_DATE('15-01-1981','dd-mm-
yyyy'),4145);
insert into mecanic values (12,'Filipescu','Dorin',TO_DATE('11-
09-1981','dd-mm-yyyy'),4118);
insert into mecanic values (13,'Saguna','Flaviu',TO_DATE('29-06-
1982','dd-mm-yyyy'),3744);
insert into mecanic values (14,'Stoica','Cezar',TO_DATE('07-07-
1982','dd-mm-yyyy'),4818);
insert into mecanic values (15,'Ianculescu','Mazon',TO_DATE('22-
03-1984','dd-mm-yyyy'),4167);
insert into mecanic values (16,'Celibidache','Ion',TO_DATE('15-
07-1984','dd-mm-yyyy'),4763);
insert into mecanic values (17,'Enache','Viorel',TO_DATE('03-05-
1985','dd-mm-yyyy'),4495);
insert into mecanic values (18,'Giurgiu','Costica',TO_DATE('05-
06-1985','dd-mm-yyyy'),4790);
insert            into            mecanic            values
(19,'Caragiale','Valentin',TO_DATE('03-12-1986','dd-mm-
yyyy'),3071);
insert into mecanic values (20,'Cazacu','Sorin',TO_DATE('03-01-
1987','dd-mm-yyyy'),3239);
insert into mecanic values (21,'Moculescu','Liviu',TO_DATE('31-
08-1987','dd-mm-yyyy'),4604);
insert into mecanic values (22,'Arcos','Neculai',TO_DATE('16-02-
1988','dd-mm-yyyy'),4280);
insert into mecanic values (23,'Dinescu','Ionel',TO_DATE('22-01-
1991','dd-mm-yyyy'),4215);

```

```

insert          into          mecanic          values
(24,'Raducanu','Cristofor',TO_DATE('14-08-1991','dd-mm-
yyyy'),3779);
insert          into          mecanic          values
(25,'Hutopila','Laurentiu',TO_DATE('11-05-1992','dd-mm-
yyyy'),4885);
insert          into          mecanic          values
(26,'Dinescu','Haralambie',TO_DATE('04-05-1981','dd-mm-
yyyy'),3734);
insert into mecanic values (27,'Raducan','Teo',TO_DATE('07-08-
1988','dd-mm-yyyy'),3636);
insert into mecanic values (28,'Stolojan','Vasile',TO_DATE('24-
06-1975','dd-mm-yyyy'),3088);
insert into mecanic values (29,'Tavitian','Victor',TO_DATE('01-
05-1992','dd-mm-yyyy'),3642);
insert into mecanic values (30,'Gilca','Luca',TO_DATE('30-08-
1974','dd-mm-yyyy'),3004);

insert into locomotiva values (1,1,1,1,1,1,5100,1968,4,41);
insert into locomotiva values (2,1,1,2,1,1,5100,1971,106,41);
insert into locomotiva values (3,1,1,1,1,1,5100,1973,387,41);
insert into locomotiva values (4,2,1,1,1,1,5400,2010,538,477);
insert into locomotiva values (5,2,1,1,1,2,5400,2009,613,477);
insert into locomotiva values (6,2,1,1,1,2,5400,2011,871,477);
insert into locomotiva values (7,1,1,3,2,2,5100,1979,603,40);
insert into locomotiva values (8,1,1,3,2,2,5100,1980,342,40);
insert into locomotiva values (9,1,1,3,2,2,5100,1982,170,41);
insert into locomotiva values (10,1,1,3,5,3,5100,1984,54,40);
insert into locomotiva values (11,1,1,3,5,3,5100,1980,52,40);
insert into locomotiva values (12,1,1,3,5,3,5100,1978,604,41);
insert into locomotiva values (13,1,1,3,5,3,5100,1974,2,40);
insert into locomotiva values (14,1,1,3,5,3,5100,1968,176,41);

```

```

insert into locomotiva values (15,2,1,1,1,3,5100,2009,498,477);
insert into locomotiva values (16,1,2,2,2,1,2100,1976,423,62);
insert into locomotiva values (17,1,2,2,6,1,2100,1980,741,62);
insert into locomotiva values (18,1,2,2,6,1,2100,1977,768,62);
insert into locomotiva values (19,4,2,2,1,1,444,1946,905,77);
insert into locomotiva values (20,4,2,2,1,2,444,1950,917,77);
insert into locomotiva values (21,4,2,2,1,2,444,1952,781,78);
insert into locomotiva values (22,5,2,1,1,2,1800,1991,2001,96);
insert into locomotiva values (23,5,2,1,1,2,1800,2000,2010,96);
insert into locomotiva values (24,5,2,1,1,2,1800,2001,2018,96);
insert into locomotiva values (25,6,2,2,4,3,2000,1975,337,97);
insert into locomotiva values (26,6,2,1,4,3,2000,1968,502,97);
insert into locomotiva values (27,6,2,1,4,3,2000,1969,507,97);
insert into locomotiva values (28,1,3,3,1,3,1100,1965,136,82);
insert into locomotiva values (29,1,3,3,1,3,1100,1970,260,82);
insert into locomotiva values (30,1,3,3,1,3,1100,1968,338,82);

```

```

insert into alocare values (1,30,1);
insert into alocare values (2,29,2);
insert into alocare values (3,28,3);
insert into alocare values (4,27,4);
insert into alocare values (5,26,5);
insert into alocare values (6,25,6);
insert into alocare values (7,24,7);
insert into alocare values (8,23,8);
insert into alocare values (9,22,9);
insert into alocare values (10,21,10);
insert into alocare values (11,20,11);
insert into alocare values (12,19,12);
insert into alocare values (13,18,13);
insert into alocare values (14,17,14);
insert into alocare values (15,16,15);

```

```
insert into alocare values (16,15,16);
insert into alocare values (17,14,17);
insert into alocare values (18,13,18);
insert into alocare values (19,12,19);
insert into alocare values (20,11,20);
insert into alocare values (21,10,21);
insert into alocare values (22,9,22);
insert into alocare values (23,8,23);
insert into alocare values (24,7,24);
insert into alocare values (25,6,25);
insert into alocare values (26,5,26);
insert into alocare values (27,4,27);
insert into alocare values (28,3,28);
insert into alocare values (29,2,29);
insert into alocare values (30,1,30);
```

```
commit
```