**ACADEMIA DE STUDII ECONOMICE BUCUREŞTI**

**FACULTATEA DE CIBERNETICĂ, STATISTICĂ ȘI INFORMATICĂ ECONOMICĂ**

**PROIECT**

**BAZE DE DATE**

**GESTIONAREA BAZEI DE DATE A UNUI HOTEL SI A ANGAJATILOR SAI**

**STUDENT: CIOCAN ANDREI**

**GRUPA 1053, SERIA C**

Tema proiectului

Tema proiectului este “Proiectarea si implementarea unei baze de date pentru gestionarea unui hotel si a angajatilor sai”. Am ales aceasta tema deoarece mi se pare un subiect ce necesita atentie in asocierea clientilor cu camera/camerele rezervate, a duratei sejurului fiecarui client, viitoarele rezervari, a serviciilor oferite de hotel, luarea in evidenta a eventualelor reclamatii, generarea facturilor, tinerea in evidenta a tuturor clientilor cat si a angajatilor si a departamentelor din care fac parte.

Descrierea proiectului

Interactiunea clientilor cu hotelul este reprezentata de sejurul pe care acestia il au in respectivul hotel. Angajatii detin anumite functii specifice domeniului de activitate in care se incadreaza si sunt direct responsabili de clienti prin gestionarea facturilor, rezervarilor, eventualelor reclamatii, asignarea clientilor catre camerele pentru care acestia au facut rezervare cat si prin serviciile prestate. Clientii sunt beneficiarii acestor multitudini de servicii oferite de hotel si au posibilitatea de a alege dintre diferite camere, cu diferite dotari pentru nevoile si preferintele fiecaruia dar si servicii extra: spa, tranfer aeroport, inchiriere sala de conferinte etc. De asemenea, pentru imbunatatirea activitatii si a performantei angajatilor, este important sa tinem o evidenta si a reclamatiilor facute de clienti.

Baza de date ce contine toate aceste informatii asigura o buna desfasurare a obiectivului de activitate a hotelului. O buna proiectare a acestei baze de date va determina un numar mai mic de erori la locul de munca, ducand la un nivel de satisfactie al clientilor mai mare si un renume mai bun al hotelului.

Structura tabelelor

**TABELA CAMERA**

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_camera | Primary Key, Not Null | NUMBER(3) |
| etaj | Check (>=0) | NUMBER(2) |
| tip | Check Not Null | VARCHAR2(15) |
| pret | Check Not Null | NUMBER(4) |
| dotari | Check Not Null | VARCHAR2(100) |

create table CAMERA(

id\_camera number(3) constraint CAMERA\_PK primary key,

etaj number(2) check(etaj>=0),

tip varchar2(15) not null,

pret number(4) not null,

dotari varchar2(100) not null);

alter table CAMERA

add constraint ID\_CAMERA\_NN

check (id\_camera IS NOT NULL);

**TABELA CLIENTI**

create table CLIENTI(

id\_client number(3) constraint CLIENTI\_PK primary key,

nume varchar2(50) not null,

prenume varchar2(50) not null,

cnp varchar2(13) not null,

adresa varchar2(100) not null,

telefon varchar2(10) unique);

alter table CLIENTI

add constraint ID\_CLIENT\_NN

check (id\_client IS NOT NULL);

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_client | Primary Key, Not Null | NUMBER(3) |
| nume | Check Not Null | VARCHAR2(50) |
| prenume | Check Not Null | VARCHAR2(50) |
| cnp | Check Not Null | VARCHAR2(13) |
| adresa | Check Not Null | VARCHAR2(100) |
| telefon | Unique | VARCHAR2(100) |

**TABELA SEJUR**

Ilustreaza o relatie MANY-TO-MANY intre tabela CAMERA si tabela CLIENTI: o camera poate fi rezervata de mai multi clienti si un client poate sta in mai multe camere. Astfel, cheile primare din cele 2 tabele mentionate anterior vor deveni **chei straine** in tabela SEJUR.

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_sejur | Primary Key, Not Null | NUMBER(3) |
| data\_cazare | Check Not Null | DATE |
| nr\_nopti | Check Not Null | NUMBER(3) |
| id\_client | Foreign Key | NUMBER(3) |
| id\_camera | Foreign Key | NUMBER(3) |

create table SEJUR(

id\_sejur number(3) constraint SEJUR\_PK primary key,

data\_cazare date not null,

nr\_nopti number(2) not null,

id\_client number(3) not null,

constraint CLIENTI\_SEJUR\_FK foreign key (id\_client) references CLIENTI(id\_client),

id\_camera number(3) not null,

constraint CAMERA\_FK foreign key(id\_camera) references CAMERA(id\_camera));

alter table SEJUR

add constraint ID\_SEJUR\_NN

check (id\_sejur IS NOT NULL);

**TABELA DEPARTAMENTE**

create table DEPARTAMENTE(

id\_departament number(3) constraint DEP\_PK primary key,

denumire varchar2(20));

alter table DEPARTAMENTE

add constraint ID\_DEPARTAMENT\_NN

check (id\_departament IS NOT NULL);

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_departament | Primary Key, Not Null | NUMBER(3) |
| denumire |  | VARCHAR2(20) |

**TABELA ANGAJATI**

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_angajat | Primary Key, Check Not Null | NUMBER(3) |
| nume | Check Not Null | VARCHAR2(50) |
| prenume | Check Not Null | VARCHAR2(50) |
| cnp | Check Not Null | VARCHAR2(13) |
| telefon | Unique | VARCHAR2(10) |
| salariu | Check Not Null | NUMBER(4) |
| data\_ang | Check Not Null | DATE |
| functie | Check Not Null | VARCHAR2(50) |
| bonuri | Check In (0,1) | NUMBER(1) |
| id\_departament | Foreign Key | NUMBER(3) |
| id\_manager |  | NUMBER(3) |

Atributul bonuri marcheaza optiunea angajatului cu privirea la integrarea bonurilor de masa in salariu.

create table ANGAJATI(

id\_angajat number(3) constraint ANG\_PK primary key,

nume varchar2(50) not null,

prenume varchar2(50) not null,

cnp varchar2(13) not null,

telefon varchar2(10) unique,

salariu number(4) not null,

data\_ang date not null,

functie varchar2(50),

bonuri number(1) CHECK (bonuri IN(0,1)),

id\_departament number(3) not null,

constraint DEP\_ANG\_FK foreign key(id\_departament) references DEPARTAMENTE(id\_departament),

id\_manager number(3));

alter table ANGAJATI

add constraint ID\_ANGAJAT\_NN

check (id\_angajat IS NOT NULL);

**TABELA FACTURA**

Are rol de tabela de jonctiune pentru a ilustra o relatie MANY-TO-MANY intre CLIENTI si ANGAJATI: un angajat poate gestiona mai multe facturi iar un client poate avea mai multe facturi. Astfel, cheile primare din cele 2 tabele mentionate anterior vor deveni **chei straine** in tabela FACTURA.

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_factura | Primary Key, Not Null | NUMBER(3) |
| valoare | Check Not Null | NUMBER(7) |
| numar | Check Not Null | NUMBER(5) |
| id\_client | Foreign Key | NUMBER(3) |
| id\_angajat | Foreign Key | NUMBER(3) |
| data\_factura | Check Not Null | DATE |

create table FACTURA(

id\_factura number(3) constraint FACT\_PK primary key,

valoare number(7) not null,

numar number(5) not null,

id\_client number(3) not null,

constraint CLIENTI\_FACT\_FK foreign key(id\_client) references CLIENTI(id\_client),

id\_angajat number(3) not null,

constraint ANG\_FACT\_FK foreign key(id\_angajat) references ANGAJATI(id\_angajat),

data\_factura date not null

);

alter table FACTURA

add constraint ID\_FACTURA\_NN

check (id\_factura IS NOT NULL);

**TABELA REZERVARI**

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_rezervare | Primary Key, Check Not Null | NUMBER(3) |
| data\_inc | Check Not Null | DATE |
| nr\_nopti |  | NUMBER(3) |
| id\_camera | Foreign Key | NUMBER(3) |
| id\_client | Foreign Key | NUMBER(3) |

CREATE TABLE REZERVARI(

id\_rezervare NUMBER(3) CONSTRAINT REZ\_PK PRIMARY KEY,

data\_inc date NOT NULL,

nr\_nopti NUMBER(3),

id\_camera NUMBER(3) NOT NULL,

constraint CAMERA\_REZ\_FK FOREIGN KEY(id\_camera) REFERENCES CAMERA(id\_camera),

id\_client NUMBER(3) NOT NULL,

constraint CLIENTI\_REZ\_FK FOREIGN KEY(id\_client) REFERENCES CLIENTI(id\_client)

);

alter TABLE REZERVARI

add constraint ID\_REZ\_NN

CHECK(id\_rezervare IS NOT NULL);

**TABELA SERVICII**

create TABLE SERVICII(

id\_serviciu number(3) constraint SERV\_PK primary key,

denumire\_serviciu varchar2(50),

pret\_serviciu number(3)

);

alter TABLE SERVICII

add constraint ID\_SERV\_NN

CHECK(id\_serviciu IS NOT NULL);

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_serviciu | Primary Key, Check Not Null | NUMBER(3) |
| denumire\_serviciu |  | VARCHAR2(50) |
| pret\_serviciu |  | NUMBER(3) |

**TABELA RAND\_SERVICII**

Are rol de tabela de jonctiune pentru a ilustra o relatie MANY-TO-MANY intre SERVICII si CLIENTI: un client poate cere mai multe servicii, iar un serviciu poate fi cerut de mai multi clienti.

De asemenea, ilustreaza o relatie MANY-TO-MANY intre tabela FACTURA si SERVICII. Aceeasi situatie este si in cazul tabelelor ANGAJATI si SERVICII.

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_serviciu | Foreign Key | NUMBER(3) |
| id\_client | Foreign Key | NUMBER(3) |
| id\_facura | Foreign Key | NUMBER(3) |
| id\_angajat | Foreign Key | NUMBER(3) |
| pret | Check Not Null | NUMBER(4) |
| cantitate | Check Not Null | NUMBER(3) |

create table RAND\_SERVICII(

id\_serviciu NUMBER(3),

constraint SERV\_RSERV\_FK FOREIGN KEY(id\_serviciu) references SERVICII(id\_serviciu),

id\_client NUMBER(3) not null,

constraint CLIENTI\_RSERV\_FK FOREIGN KEY(id\_client) references CLIENTI(id\_client),

id\_factura number(3) not null,

constraint FACT\_RSERV\_FK foreign key(id\_factura) references FACTURA(id\_factura),

id\_angajat number(3) not null,

constraint ANG\_RSERV\_FK foreign key(id\_angajat) references ANGAJATI(id\_angajat),

pret NUMBER(4) NOT NULL,

cantitate NUMBER(3) NOT NULL);

**TABELA RECLAMATII**

|  |  |  |
| --- | --- | --- |
| Nume coloana | Restrictie | Tipul de date |
| id\_reclamatie | Primary Key, Check Not Null | NUMBER(3) |
| mesaj\_reclamatie |  | VARCHAR2(100) |
| id\_client | Foreign Key | NUMBER(3) |
| id\_angajat | Foreign Key | NUMBER(3) |

create table RECLAMATII(

id\_reclamatie NUMBER(3) CONSTRAINT REC\_PK primary key,

mesaj\_reclamatie VARCHAR2(100),

id\_client NUMBER(3) not null,

constraint CLIENTI\_REC\_FK foreign key(id\_client) references CLIENTI(id\_client),

id\_angajat number(3) not null,

constraint ANG\_REC\_FK foreign key(id\_angajat) references ANGAJATI(id\_angajat)

);

alter TABLE RECLAMATII

add constraint ID\_REC\_NN

CHECK(id\_reclamatie IS NOT NULL);

Schema conceptuala a bazei de date

A screenshot of a computer

Description automatically generated

Popularea tabelelor

**Tabela CAMERA**

INSERT INTO CAMERA(Id\_camera,Etaj,Tip,Pret,Dotari)

VALUES('001', '1','Single','195','Mini-bar, acces internet, telefon, tv diagonala 51 cm,dus');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('002', '2','Single','225','Mini-bar, acces internet, telefon, tv diagonala 60 cm, cada');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('003', '1','Double','275','Mini-bar, acces internet, telefon, tv diagonala 60 cm, cada,balcon');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('004', '2','Double','290','Mini-bar, acces internet, telefon, tv diagonala 51 cm, dus, feon');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('005', '1','Apartment','350','Mini-bar, acces internet, telefon, tv diagonala 80 cm, cada, dus, balcon');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('006', '1','Double','275','Mini-bar, acces internet, telefon, tv diagonala 60 cm, cada,balcon');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('007', '0','Single','150','Mini-bar, acces internet, telefon, cada');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('008', '3','Apartment','450','Mini-bar, acces internet, telefon, tv diagonala 80 cm, cada,dus,balcon cu vedere');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('009', '0','Single','150','Mini-bar, acces internet, telefon, cada');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('010', '2','Single','225','Mini-bar, acces internet, telefon, tv diagonala 60 cm, dus');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('011', '2','Single','150','Mini-bar, acces internet, telefon, cada');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('012', '1','Single','150','Mini-bar, acces internet, telefon, dus');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('013', '4','Sala conferinte','800','Televizor, videoproiector,calculator, masa 12 persoane');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('014', '4','Sala conferinte','800','Televizor, videoproiector,calculator, masa 12 persoane');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('015', '3','Apartment','450','Mini-bar, acces internet, telefon, tv diagonala 80 cm, cada,dus,balcon cu vedere');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('016', '3','Apartment','400','Mini-bar, acces internet, telefon, tv diagonala 80 cm, cada,dus,balcon');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('017', '0','Single','150','Mini-bar, acces internet, telefon, cada');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip ,Pret , Dotari)

VALUES('018', '1','Single','180','Mini-bar, acces internet, telefon, dus');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('019', '1','Double','275','Mini-bar, acces internet, telefon, tv diagonala 60 cm, cada,balcon');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('020', '2','Double','290','Mini-bar, acces internet, telefon, tv diagonala 80 cm, dus,balcon');

INSERT INTO CAMERA(Id\_camera, Etaj , Tip , Pret , Dotari)

VALUES('021', '3','Apartment','425','Mini-bar, acces internet, telefon, tv diagonala 80 cm, cada,dus,balcon cu vedere partiala');

**Tabela CLIENTI**

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('01', 'Palade','Ioan', '1781216562089','Str. Oltului 53, bl P5, Galati', '0745349015');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('02', 'Burlacu','Matei', '1781216071789','Str. Constructorilor 27, bl M1, Slobozia', '0725346725');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('03', 'Condrea','Alina', '2871216026490','Str. Mihai Bravu, bl A1, Bucuresti', '0745342435');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('04', 'Cristea','Anca', '2781216070890','Calea Dorobantilor 114, bl A2, Bucuresti', '0765327455');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('05', 'Comanescu','Alina', '2781216070119','Str. Padesu 14, bl B3, Bucuresti', '0725322455');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('06', 'Moraru','Mihai', '1781215091389','Str. Lalelelor 27, bl G9, Iasi', '0725676554');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('07', 'Marin','Alexandra', '2741015071790','Str. Domneasca, bl BR5, Timisoara', '0769872200')

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('08', 'Anghel','Daria', '2781216450119','Str. Mihai Eminescu 13, bl F1, Galati', '0745667789');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('09', 'Balauta','Rodica', '2871216776528','Str. Pacii, bl I6, Slobozia', '0759687445');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('10', 'Serban','Marius', '1781285496836','Str. Marii Uniri 29, bl M10, Slobozia', '0729667823');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('11', 'Agapie','Florin', '1781208988765','Str. 1 Decembrie 1918, bl P10, Galati', '0755432789');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('12', 'Nita','Stefan', '1881105767552','Str.Francezilor, bl A12, Brasov', '0725677125');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('13', 'Dumitrescu','Paula', '2870911877266','Str. Cristofor Columb, bl B2, Sibiu', '0744897632');

INSERT INTO CLIENTI(id\_client, nume, prenume, cnp, adresa, telefon)

VALUES('14', 'Dobre','Ioana', '2850211155676','Str. Mihai Eminescu, bl Y7, Ploiesti', '0758963324');

**Tabela SEJUR**

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('001',to\_date('24-12-2019', 'dd-mm-yyyy'),'5', '01', '01');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('002',to\_date('25-12-2017', 'dd-mm-yyyy'),'3','02','02');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('003', to\_date('29-12-2015', 'dd-mm-yyyy'), '4','03', '01');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('004', to\_date('01-01-2015', 'dd-mm-yyyy'),'5','04','04');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('005', to\_date('02-01-2010', 'dd-mm-yyyy'),'7','05','06');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('006', to\_date('13-06-2020', 'dd-mm-yyyy'),'5','03','06');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('007', to\_date('15-07-2017', 'dd-mm-yyyy'),'7','05','01');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('008', to\_date('13-06-2018', 'dd-mm-yyyy'),'5','04','02');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('009', to\_date('12-05-2019', 'dd-mm-yyyy'),'4','03','06');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('010', to\_date('17-07-2015', 'dd-mm-yyyy'),'7','06','01');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('011', to\_date('17-07-2020', 'dd-mm-yyyy'),'3','09','02');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('012', to\_date('10-09-2020', 'dd-mm-yyyy'),'2','08','03');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('013', to\_date('01-12-2019', 'dd-mm-yyyy'),'1','10','07');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('014', to\_date('15-08-2021', 'dd-mm-yyyy'),'4','10','20');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('015', to\_date('16-02-2021', 'dd-mm-yyyy'),'2','11','16');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('016', to\_date('02-02-2018', 'dd-mm-yyyy'),'2','11','18');

INSERT INTO sejur(id\_sejur,data\_cazare,nr\_nopti, id\_client, id\_camera)

VALUES('017', to\_date('12-02-2018', 'dd-mm-yyyy'),'2','10','17');

**Tabela DEPARTAMENTE**

INSERT INTO departamente(id\_departament,denumire)

VALUES ('001', 'Receptie');

INSERT INTO departamente(id\_departament,denumire)

VALUES('002', 'Administratie');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('003', 'Restaurant');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('004', 'Curatenie');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('005', 'Securitate');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('006', 'Contabilitate');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('007', 'Tehnic');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('008', 'Resurse umane');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('009', 'Marketing');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('010', 'Aprovizionare');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('011', 'Servicii haine');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('012', 'Transfer aeroport');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('013', 'Room service');

INSERT INTO departamente(id\_departament,denumire)

VALUES ('014', 'Relaxare');

**Tabela ANGAJATI**

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('001','Hincu','Adelina','2750212096381','0725673112','1500',to\_date('21-09-1989', 'dd-mm-yyyy'),'Operator receptie', '001','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('002','Popescu','George','1760304571965','0724219898','2700',to\_date('13-07-2000', 'dd-mm-yyyy'),'Director', '002','002','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('003','Popa','Mariana','2730395121859','0764269763','1700',to\_date('05-06-1998', 'dd-mm-yyyy'), 'Contabila', '006','002','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('004','Toma','Alin','1870413112945','0755312269','1200',to\_date('10-03-2012', 'dd-mm-yyyy'),'Ospatar', '003','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('005','Panait','Lucas','185090212347','0726368569','1000',to\_date('06-10-2002', 'dd-mm-yyyy'),'Mecanic', '007','007',null);

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('006','Stancescu','Liana','2870302757294','0743942532','1000',to\_date('01-07-2011', 'dd-mm-yyyy'), 'Operator receptie', '001','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('007','Ionescu','Roxana','286080753194','0723265969','2000',to\_date('10-06-2020', 'dd-mm-yyyy'),'Director secundar', '002','002','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('008','Iliescu','Cosmin','2890502167239','0729876876','1800',to\_date('09-06-2010', 'dd-mm-yyyy'),'Barman', '003','007', null);

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('009','Matache','Ioana','287050719395','0776335659','1000',to\_date('01-07-2006', 'dd-mm-yyyy'), 'Camerista','004','007', null);

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('010','Macovei','Diana','2890578957239','0745703116','1370',to\_date('09-06-2012', 'dd-mm-yyyy'),'Agent securitate', '005','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('011','Popescu','Alina','2860807562194','0774156774','1400',to\_date('10-06-2016', 'dd-mm-yyyy'),'Operator masina spalat', '011','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('012','Filip','Teodora','2789315187859','0764269763','1700',to\_date('05-06-1998', 'dd-mm-yyyy'), 'Operator statie de calcat haine', '011','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('013','Georgescu','Andreea','2720015071790','0554493337','1400',to\_date('15-09-2013', 'dd-mm-yyyy'),'Operator masina spalat rufe', '011','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('014','Diaconescu','Mioara','2809321602649','0764269763','1700',to\_date('17-12-2008', 'dd-mm-yyyy'), 'Operator statie de calcat haine', '011','007', null);

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('015','Palade','Mircea','1781216562089','0727980769','2100',to\_date('21-10-2014', 'dd-mm-yyyy'),'Sofer', '012','007', null);

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('016','Irimia','Dan','178027980769','0745677912','2300',to\_date('21-10-2001', 'dd-mm-yyyy'),'Sofer', '012','007', null);

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('017','Mitrea','Carmen','2860804982694','0774156774','1400',to\_date('10-06-2016', 'dd-mm-yyyy'),'Responsabil Room-Service', '013','007', null);

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('018','Iosifescu','Cosmina','289070914395','0725665441','2500',to\_date('01-07-2006', 'dd-mm-yyyy'), 'Maseuza','014','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('019','Ionita','Daniela','2920217086381','0725963112','1500',to\_date('21-09-2015', 'dd-mm-yyyy'),'Operator receptie', '001','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('020','Stoica','Laura','2960903975563','0751448326','2700',to\_date('13-12-2019', 'dd-mm-yyyy'), 'Cosmeticiana', '014','007', null);

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('021','Mocanu','Robert','1780217225784','0745796688','1400',to\_date('09-11-2014', 'dd-mm-yyyy'),'Responsabil Room-Service', '013','007','1');

INSERT INTO ANGAJATI(id\_angajat,nume,prenume,cnp,telefon,salariu,data\_ang,functie,id\_departament,id\_manager,bonuri)

VALUES ('022','Ciocan','Andrei','5030904181972','0772167742','4400',to\_date('21-07-2018', 'dd-mm-yyyy'),'', '003','007',null);

**A screenshot of a computer

Description automatically generated**

**Tabela FACTURA**

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura)

VALUES('001','1175','1001','01','003',to\_date('24-12-2019', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('002','675','1002','02','003', to\_date('25-12-2017', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('003','780','1003','03','001', to\_date('29-12-2015', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('004','1450','1004','04','006', to\_date('01-01-2015', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('005','1650','1005','05','003', to\_date('02-01-2010', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('006','1650','1006','03','003', to\_date('13-06-2020', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('007','1515','1008','09','001', to\_date('15-07-2017', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('008','1125','1008','10','001', to\_date('13-06-2018', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('009','1100','1009','03','001', to\_date('12-05-2019', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('010','1180','1010','07','006', to\_date('17-05-2015', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('011','515','1011','11','019', to\_date('17-07-2020', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('012','1265','1012','12','019', to\_date('10-09-2020', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('013','340','1013','13','001', to\_date('01-12-2019', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('014','1190','1014','14','006', to\_date('15-08-2021', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('015','800','1015','05','019', to\_date('16-02-2021', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('016','360','1016','10','019', to\_date('02-02-2018', 'dd-mm-yyyy'));

INSERT INTO FACTURA(id\_factura, valoare,numar, id\_client,id\_angajat,data\_factura )

VALUES('017','300','1017','11','019', to\_date('12-02-2018', 'dd-mm-yyyy'));

**Tabela SERVICII**

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('001','Spalare rufe colorate','35');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('002','Spalare rufe albe','25');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('003','Calcare bluze','20');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('004','Calcare pantaloni','30');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('005','Transfer aeroport in masina de lux','200');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('006','Transfer aeroport in masina normala','100');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('007','Room Service','15');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('008','Masaj relaxant','150');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('009','Masaj cu pietre vulcanice','250');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('010','Masaj cu bete din bambus','200');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('011','Tratament facial anti-age','150');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('012','Tratament facial cu oxigen hiperbaric','350');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('013','Masaj facial','50');

INSERT INTO SERVICII(id\_serviciu,denumire\_serviciu,pret\_serviciu)

VALUES('014','Inchiriere sala conferinte','800');

**Tabela RAND\_SERVICII**

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('001','09','009','001','35','1');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('002','10','008','013','25','1');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('003','07','010','012','20','2');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('004','07','010','014','30','2');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('004','14','014','014','30','1');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('005','01','001','015','200','1');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('006','05','016','016','100','1');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('007','11','017','017','15','2');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('007','13','013','021','15','2');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('007','12','012','017','15','1');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('008','09','007','018','15','1');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('010','13','013','018','200','1');

INSERT INTO RAND\_SERVICII(id\_serviciu,id\_client,id\_factura,id\_angajat,pret,cantitate)

VALUES('014','13','013','007','800','1');

**Tabela REZERVARI**

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('001','014',to\_date('02-02-2018', 'dd-mm-yyyy'),'4','020');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('002','012',to\_date('02-02-2016', 'dd-mm-yyyy'),'3','018');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('003','014',to\_date('01-12-2020', 'dd-mm-yyyy'),'8','001');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('004','011',to\_date('30-12-2021', 'dd-mm-yyyy'),'4','002');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('005','009',to\_date('09-11-2021', 'dd-mm-yyyy'),'1','008');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('006','001',to\_date('17-02-2022', 'dd-mm-yyyy'),'2','004');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('007','008',to\_date('19-10-2022', 'dd-mm-yyyy'),'7','002');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('008','012',to\_date('01-05-2023', 'dd-mm-yyyy'),'2','012');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('009','011',to\_date('19-10-2022', 'dd-mm-yyyy'),'3','017');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('010','005',to\_date('19-10-2022', 'dd-mm-yyyy'),'4','019');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('011','003',to\_date('19-10-2022', 'dd-mm-yyyy'),'1','010');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('012','013',to\_date('05-07-2023', 'dd-mm-yyyy'),'2','003');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('013','006',to\_date('29-12-2022', 'dd-mm-yyyy'),'1','009');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('014','014',to\_date('12-09-2022', 'dd-mm-yyyy'),'5','001');

INSERT INTO REZERVARI(id\_rezervare,id\_client,data\_inc,nr\_nopti,id\_camera)

VALUES('015','004',to\_date('21-07-2022', 'dd-mm-yyyy'),'3','008');

**Tabela RECLAMATII**

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('001','Probleme apa calda','03','001');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('002','Probleme aer conditionat','14','019');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('003','Personal spa irascibil','13','006');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('004','Transport aeroport intarziat','01','006');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('005','Transport aeroport intarziat','05','001');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('006','Mancarea a venit rece prin room-service','12','001');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('007','Mancarea a venit rece prin room-service','11','019');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('008','Mancarea a venit tarziu prin room-service','11','019');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('009','Camasa nu a fost calcata bine','12','019');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

VALUES('010','Pantalonii nu au fost calcati bine','07','006');

INSERT INTO RECLAMATII(id\_reclamatie,mesaj\_reclamatie,id\_client,id\_angajat)

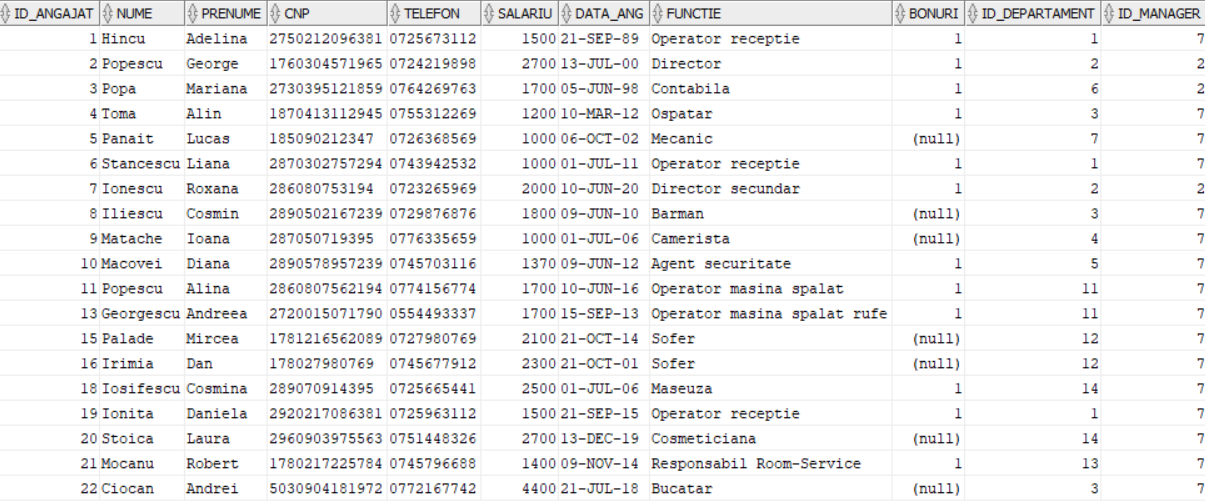
VALUES('011','Nu a fost scoasa pata de pe bluza','10','006');

**Exercitii cu SELECT. Jonctiuni**

**Exemple cu SELECT**

1. Sa se selecteze toti angajatii din tabela ANGAJATI

SELECT \* FROM ANGAJATI;



2.Sa se selecteze coloanele id\_angajat,nume,prenume si salariu din tabela ANGAJATI

SELECT id\_angajat,nume,prenume,salariu

FROM ANGAJATI

A table with numbers and letters

Description automatically generated

3.Sa se selecteze angajatii care fac parte din categoria Operator

SELECT \*FROM ANGAJATI

A picture containing text

Description automatically generatedWHERE UPPER(functie) LIKE 'OPERATOR%';

4.Sa se selecteze toate facturile incheiate de angajatul cu id\_angajat=19;

SELECT \* FROM FACTURA

WHERE id\_angajat=19;

Graphical user interface, table

Description automatically generated

5.Sa se selecteze toate facturile incheiate dupa 1 ianuarie 2018

SELECT \* FROM FACTURA

WHERE data\_factura>TO\_DATE('01.01.2018','DD.MM.YYYY');

Table

Description automatically generated

**Operatorul ANY si ALL**

6.Sa se afiseze id\_angajat,nume,prenume si salariul pentru angajatii care nu lucreaza in departamentul Receptie si al caror salariu este mai mare decat fiecare dintre salariile angajatilor care lucreaza in departamentul Receptie

Graphical user interface, text, table, Excel

Description automatically generatedSELECT id\_angajat,nume,prenume,salariu

FROM ANGAJATI

WHERE salariu>ALL(

SELECT salariu FROM ANGAJATI

WHERE id\_departament='1')

AND id\_departament <>'1'

ORDER BY salariu DESC;

7. Sa se afiseze id\_angajat, prenume, id\_departament si salariu pentru angajatii care nu lucreaza in departamentul cu id-ul 01 (Receptie) si al caror salariu este mai mic decat oricare dintre salariile angajatilor care lucreaza in departamentul cu id-ul 01

SELECT a.id\_angajat, a.prenume, a.id\_departament, a.salariu

O imagine care conține masă

Descriere generată automatFROM ANGAJATI a

WHERE a.salariu < ANY

(SELECT salariu FROM ANGAJATI

WHERE id\_departament = '01')

AND id\_departament <> '01'

ORDER BY salariu DESC;

**Jonctiunea de egalitate**

8.Sa se afiseze angajatii care nu au incheiat facturi in Februarie 2018

SELECT a.nume,a.prenume,f.data\_factura

FROM ANGAJATI a, FACTURA f

WHERE a.id\_angajat=f.id\_angajat AND EXTRACT (YEAR FROM f.data\_factura)<>'2018'

Graphical user interface, table, Excel

Description automatically generatedAND EXTRACT (MONTH FROM f.data\_factura)<>'02'

ORDER BY a.nume;

9.Sa se selecteze facturile gestionate de toti angajatii

SELECT a.\*, f.\*

FROM ANGAJATI a, FACTURA f

O imagine care conține text, interior, decorat, captură de ecran

Descriere generată automatWHERE a.id\_angajat=f.id\_angajat;

**Jonctiunea externa**

10.Sa se afiseze id\_serviciu, denumirea serviciului si cantitatea chiar daca nu au fost cerute de niciun client

SELECT s.id\_serviciu, s.denumire\_serviciu, rs.cantitate,s.pret\_serviciu

FROM SERVICII s, RAND\_SERVICII rs

WHERE s.id\_serviciu=rs.id\_serviciu(+);

O imagine care conține text

Descriere generată automat

**Jonctiunea dintre o tabela cu aceeasi tabela**

11. Sa se afiseze numele fiecarui angajat si numele managerului(directorului) caruia ii este subordonat.

SELECT a.nume, a.prenume ||' este subordonat managerului: ' || m.nume

FROM ANGAJATI a, ANGAJATI m

WHERE a.id\_manager=m.id\_angajat;

A screenshot of a computer

Description automatically generated

**Functii SINGLE-ROW.Functii de grup**

**Functii SINGLE-ROW**

**Functia LOWER() , UPPER(), INITCAP()**

1.Sa se afiseze toti angajatii cu numele Popescu utilizand functiile INITCAP, UPPER, LOWER.

SELECT id\_angajat, nume

FROM ANGAJATI

WHERE INITCAP(nume)='Popescu';

SELECT id\_angajat, nume

FROM ANGAJATI

WHERE UPPER(nume)='POPESCU';

SELECT id\_angajat, nume

FROM ANGAJATI

WHERE lower(nume)='popescu';

**Operatorul de concatenare** (||)

2. Sa se afiseze denumirea serviciilor si pretul lor.

SELECT 'Serviciul: ' || INITCAP(denumire\_serviciu) || ' are pretul: ' || pret\_serviciu

O imagine care conține text

Descriere generată automatFROM SERVICII;

**Funcţia CONCAT() , funcţia LENGTH() , funcţia SUBSTR()**

3. Sa se afiseze id\_angajat, nume angajat concatenat cu salariul si lungimea prenumelui pentru angajatii cu primele 2 litere Po.

SELECT id\_angajat, CONCAT(nume,salariu), LENGTH(prenume), data\_ang

FROM ANGAJATI

WHERE SUBSTR(nume,1,2)='Po';

O imagine care conține masă

Descriere generată automat

O imagine care conține text

Descriere generată automat**Funcţia SYSDATE**4. Sa se afiseze perioada de timp coresp(in saptamani) intre data incheierii facturii si data curenta.

SELECT data\_factura, (SYSDATE-data\_factura)/7 saptamani

FROM FACTURA;

**Funcţiile MONTH\_BETWEEN() , ADD\_MONTHS() , NEXT\_DAY() , LAST\_DAY()**

5. Sa se afiseze facturile, data inchierii facturii, numarul de luni intre data curenta si data incheierii, urmatoarea zi de vineri dupa data incheierii, ultima zi din luna din care face parte data incheierii precum si data corespunzatoare dupa 2 luni de la data incheierii facturii.

SELECT numar, data\_factura, round(MONTHS\_BETWEEN(sysdate, data\_factura)) luni,

NEXT\_DAY(data\_factura, 'FRIDAY'),

LAST\_DAY(data\_factura),

ADD\_MONTHS(data\_factura,2)

FROM FACTURA;

O imagine care conține masă

Descriere generată automat

**Funcţia ROUND()**

6. Sa se afiseze facturile incheiate in 2020. Se va rotunji data incheierii la prima zi din luna corespunzatoare daca data incheierii este in prima jumatate a lunii sau la prima zi din luna urmatoare.

SELECT numar, data\_factura, ROUND(data\_factura,'MONTH') FROM FACTURA

WHERE data\_factura LIKE '%-20%';

O imagine care conține masă

Descriere generată automat

**Funcţia TO\_CHAR**

7. Sa se afiseze facturile si data incheierii in formatul initial si in formatul "MM/YY".

SELECT numar, data\_factura, TO\_CHAR(data\_factura, 'MM/YY') data\_incheierii\_facturii FROM FACTURA;

O imagine care conține masă

Descriere generată automat

**Funcţia TO\_DATE**

8. Sa se afiseze facturile incheiate intre 15 ianuarie si 15 decembrie 2018.

SELECT numar, data\_factura

FROM FACTURA

O imagine care conține text

Descriere generată automatWHERE data\_factura BETWEEN TO\_DATE( 'January 15, 2018', 'Month dd,YYYY') AND TO\_DATE( 'December 15, 2018', 'Month dd,YYYY');

**Funcţia EXTRACT()**

O imagine care conține text

Descriere generată automat9. Sa se afiseze facturile incheiate in anii 2015 si 2019.

SELECT numar, data\_factura FROM FACTURA

WHERE EXTRACT(YEAR FROM data\_factura) IN (2015, 2019);

**Functia NULLIF**

O imagine care conține masă

Descriere generată automat10.Sa se afiseze lungimea numelui, lungimea prenumelui, daca acestea sunt egale sa se returneze nul ca rezultat, iar daca nu sunt egale se va returna lungimea numelui.

SELECT nume, length(nume), prenume, length(prenume), NULLIF(length(nume), length(prenume)) rezultat

FROM CLIENTI;

**FUNCTII DE GRUP**

1. Sa se afiseze valoarea maxima, valoarea medie, valoarea minima si valoarea totala a serviciilor cerute de clienti.

SELECT AVG(rs.cantitate\*rs.pret) AS VALOARE\_MEDIE,

MAX(rs.cantitate\*rs.pret) AS VALOARE\_MAXIMA,

MIN(rs.cantitate\*rs.pret) AS VALOARE\_MINIMA,

SUM(rs.cantitate\*pret) AS VALOARE\_TOTALA

FROM RAND\_SERVICII rs;



2.Să se afişeze data primei rezervari încheiate şi data celei mai vechi rezervari încheiate.

SELECT MIN(data\_inc) AS PRIMA\_REZERVARE, MAX(data\_inc) AS CEA\_MAI\_RECENTA\_REZERVARE

FROM REZERVARI;

3.Să se afişeze numărul de servicii al căror pret\_serviciu>100.

SELECT COUNT(\*) nr\_servicii

FROM SERVICII

WHERE pret\_serviciu>100;

4.Sa se afiseze numarul de salarii (distincte) din tabela angajati.



SELECT COUNT (salariu) FROM ANGAJATI;



SELECT COUNT (DISTINCT salariu) FROM ANGAJATI;

**Expresii folosind DECODE si CASE**

1.Sa se calculeze comision angajatilor in functie de pozitia ocupata:

-0.1% din salariu daca functia este Operator receptie

-0.2% din salariu daca functia este Sofer

-0.3% din salariu daca functia este Director

Pentru celelalte comisionul va fi 0.

SELECT nume, functie,

CASE WHEN UPPER(functie)='DIRECTOR' THEN 0.3

WHEN UPPER(functie)='SOFER' THEN 0.2

WHEN UPPER(functie)='OPERATOR RECEPTIE' THEN 0.1

ELSE 0 END comision

FROM ANGAJATI;

A screenshot of a computer

Description automatically generated

2. Rezolvarea cu functia DECODE

SELECT nume,functie,

DECODE(UPPER(functie),'DIRECTOR',0.3,'SOFER',0.2,'OPERATOR RECEPTIE',0.1,0) comision

FROM ANGAJATI;

**Operatorii Algebrici Relationali: UNION, INTERSECT, MINUS**

**Operatorul MINUS**

1. Sa se afiseze angajatii care au salariul intre 1000 si 2500 fara cei care au salariul 1700 si 2000.

SELECT \* FROM ANGAJATI WHERE salariu BETWEEN 1000 AND 2500

MINUS

A screen shot of a computer

Description automatically generatedSELECT \* FROM ANGAJATI WHERE salariu IN (1700, 2000);

**Operatorul UNION**

2. Sa se calculeze distinct comisionul pentru angajati folosind operatorul UNION:

* daca au efectuat 1 serviciu comisionul va fi de 10% din valoare totala a serviciilor;
* daca au efectuat 2 servicii comisionul va fi de 20% din valoare totala a serviciilor;
* daca au efectuat 2 servicii comisionul va fi de 30% din valoare totala a serviciilor.

SELECT a.nume, COUNT(s.id\_serviciu) numar\_servicii,

0.1\* SUM(rs.cantitate\*rs.pret) valoare\_servicii

FROM ANGAJATI a,SERVICII s, RAND\_SERVICII rs

WHERE a.id\_angajat=rs.id\_angajat

AND s.id\_serviciu=rs.id\_serviciu

GROUP BY a.nume

HAVING COUNT(s.id\_serviciu)=1

UNION

SELECT a.nume, COUNT(s.id\_serviciu) numar\_servicii,

0.2\* SUM(rs.cantitate\*rs.pret) valoare\_servicii

FROM ANGAJATI a,SERVICII s, RAND\_SERVICII rs

WHERE a.id\_angajat=rs.id\_angajat

AND s.id\_serviciu=rs.id\_serviciu

GROUP BY a.nume

HAVING COUNT(s.id\_serviciu)=2

UNION

SELECT a.nume, COUNT(s.id\_serviciu) numar\_servicii,

0.3\* SUM(rs.cantitate\*rs.pret) valoare\_servicii

FROM ANGAJATI a,SERVICII s, RAND\_SERVICII rs

WHERE a.id\_angajat=rs.id\_angajat

AND s.id\_serviciu=rs.id\_serviciu

GROUP BY a.nume

HAVING COUNT(s.id\_serviciu)>=3;

O imagine care conține masă

Descriere generată automat

**Operatorul INTERSECT**

3. Sa se selecteze denumirea serviciilor, valoare totala ceruta de clienti si id\_serviciu pentru produsele comandate de cel putin 2 ori si care au valoarea totala diferita de 100, 35.

SELECT s.denumire\_serviciu, SUM(rs.cantitate\*rs.pret) valoare, COUNT(rs.id\_serviciu) numar\_servicii

FROM SERVICII s, RAND\_SERVICII rs

WHERE rs.id\_serviciu=s.id\_serviciu

GROUP BY s.denumire\_serviciu

HAVING COUNT(rs.id\_serviciu)<=2

INTERSECT

SELECT s.denumire\_serviciu, SUM(rs.cantitate\*rs.pret) valoare, COUNT(rs.id\_serviciu) numar\_servicii

FROM SERVICII s, RAND\_SERVICII rs

WHERE rs.id\_serviciu=s.id\_serviciu

GROUP BY s.denumire\_serviciu

HAVING SUM(rs.cantitate\*rs.pret) NOT IN (100, 35);

O imagine care conține text

Descriere generată automat

**Prelucrarea Cererilor Ierarhice**

A table of numbers with black text

Description automatically generated with medium confidence

1.Sa se afiseze angajatii si nivelul ierarhic al acestora pornind de la angajatul cu id-ul 7 (sa se ordoneze in functie de nivelul ierahic).

SELECT id\_angajat, nume, id\_manager,

LEVEL FROM angajati

CONNECT BY PRIOR id\_angajat= id\_manager

START WITH id\_angajat = 7

ORDER BY LEVEL;

A table of names on a white background

Description automatically generated2. Sa se afiseze angajatii companiei subordonati inregistrarii radacina sub forma de organigrama.

SELECT LEVEL, LPAD(' ', LEVEL)|| nume FROM angajati

CONNECT BY PRIOR id\_angajat = id\_manager

START WITH id\_angajat= 7

ORDER BY LEVEL;

3. Afisati angajatii companiei subordonati inregistrarii radacina specificand numarul de superiori si toti superiorii sai, id-urile (se utilizeaza clauzele: SYS\_CONNECT\_BY\_PATH, LEVEL-1):

SELECT id\_angajat, nume,

LEVEL-1 Numar\_Superiori, SYS\_CONNECT\_BY\_PATH(id\_angajat, '/') ID\_Superiori

FROM angajati

START WITH id\_angajat=7

CONNECT BY PRIOR id\_angajat = id\_manager;

O imagine care conține masă

Descriere generată automat

4. Sa se selecteze angajati si gradul de subordonare numai pentru cei din departamentele 1 si 11:

SELECT id\_angajat, nume, id\_manager, level FROM angajati

WHERE id\_departament IN (1, 11)

CONNECT BY PRIOR id\_angajat = id\_manager

START WITH id\_angajat = 7;

O imagine care conține masă

Descriere generată automat

5. Sa se afiseze superiorii angajatilor aflati pe ultimul nivel de subordonare:

SELECT nume, LEVEL-1 Numar\_Superiori, SYS\_CONNECT\_BY\_PATH(nume, '/') Nume\_Superiori

FROM angajati

START WITH id\_angajat = 7

CONNECT BY PRIOR id\_angajat = id\_manager

ORDER BY LEVEL desc;

O imagine care conține masă

Descriere generată automat

6. Sa se afiseze toti superiorii lui ‘Stoica’:

O imagine care conține masă

Descriere generată automatSELECT id\_angajat, nume, id\_manager, LEVEL FROM angajati

CONNECT BY id\_angajat = PRIOR id\_manager

START WITH nume= 'Stoica';

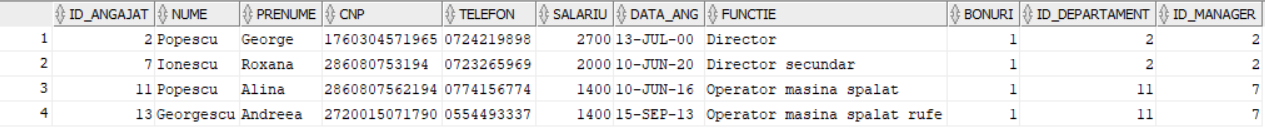
**Subcereri**

1.Sa se selecteze angajatii care sunt in acelasi departament cu angajatul Popescu

SELECT \* FROM ANGAJATI

WHERE id\_departament IN

(SELECT id\_departament FROM ANGAJATI WHERE upper(nume)='POPESCU');



2. Sa se afiseze serviciile care au pretul unitar cel mai mare

SELECT s.denumire\_serviciu,rs.pret,s.id\_serviciu

FROM SERVICII s, RAND\_SERVICII rs

WHERE s.id\_serviciu=rs.id\_serviciu

AND s.pret\_serviciu=(SELECT MAX(RAND\_SERVICII.pret) FROM RAND\_SERVICII);



**Tabele virtuale. Indecsi. Sinonime. Secvente**

**Tabele virtuale**

1. Sa se realizeze o tabela virtuala cu toti angajatii din departamentul 11. Actualizam salariul.

CREATE OR REPLACE VIEW v\_angajati\_11

AS SELECT \* FROM ANGAJATI

WHERE id\_departament=11;

O imagine care conține text

Descriere generată automat

UPDATE v\_angajati\_11

SET salariu=salariu+300;

Dupa actualizarea salariului:

O imagine care conține text

Descriere generată automat

**Indecsi**

1. Sa se creeze un index pe tabela angajati pe coloana nume

SELECT \* FROM ANGAJATI WHERE nume='Hincu';

CREATE INDEX idx\_nume ON ANGAJATI(nume);

SELECT \* FROM ANGAJATI WHERE nume='Hincu';

SELECT \* FROM ANGAJATI WHERE UPPER(nume)='HINCU';

CREATE INDEX idx\_upper\_nume ON ANGAJATI(UPPER(nume));

O imagine care conține text

Descriere generată automat

2. Vizualiarea indecsilor unui anumit utilizator

SELECT \* FROM USER\_INDEXES;

A screenshot of a computer

Description automatically generated

3. Sa se stearga indexul creat anterior

DROP INDEX idx\_nume;



**Sinonime**

1. .Sa se creeze un sinonim pentru tabela rand\_servicii;

CREATE SYNONYM rs FOR RAND\_SERVICII;

2. Vizualizarea sinonimelor

SELECT \* FROM USER\_SYNONYMS; 

3. Sa se stearga sinonimul

DROP SYNONYM rs;

**Secvente**

1. Sa se creeze o secventa pt asigurarea unicitatii cheii primare din tabela Servicii.

CREATE SEQUENCE seq\_id\_serviciu

START WITH 100 INCREMENT BY 10

MAXVALUE 1000 NOCYCLE;



INSERT INTO SERVICII

VALUES(seq\_id\_serviciu.NEXTVAL,'Spalare masina',80);

SELECT \* FROM SERVICII;

O imagine care conține text

Descriere generată automat

2. Sa se afiseze valoarea curenta a secventei

SELECT seq\_id\_serviciu.CURRVAL FROM DUAL;



3. Sa se modifice pasul de incrementare si valoarea maxima pt secventa seq\_id\_serviciu;

ALTER SEQUENCE seq\_id\_serviciu INCREMENT BY 100;

ALTER SEQUENCE seq\_id\_serviciu MAXVALUE 2000;

INSERT INTO SERVICII

VALUES(seq\_id\_serviciu.NEXTVAL,'Parcare',20);

O imagine care conține text

Descriere generată automat

4. Sa se vizualizeze informatiile despre secventele utilizatorului

SELECT \* FROM USER\_SEQUENCES;



5. Sa se stearga secventa seq\_id\_serviciu

DROP SEQUENCE seq\_id\_serviciu;