Proiect SGBD Inchirieri Auto
Tudose Andrei-Daniel
Grupa 244

1. Prezentați pe scurt baza de date (utilitatea ei).

Baza de date este folosita de o firma de inchirieri auto pentru gestionarea activitatii sale, care se desfasoara astfel:

- o persoana (tabelul PROPRIETARI) isi lasa masina la firma de inchirieri auto

- firma stocheaza informatii despre masinile pe care le detine pentru inchiriere in tabelul MASINI

- un client poate inchiria acea masina prin intermediul unei cereri de inchiriere (many to many, deci tabelul asociativ CERERI_DE_INCHIRIERE), si o ridica de la o locatie unde se afla masina (tabelul LOCATII)

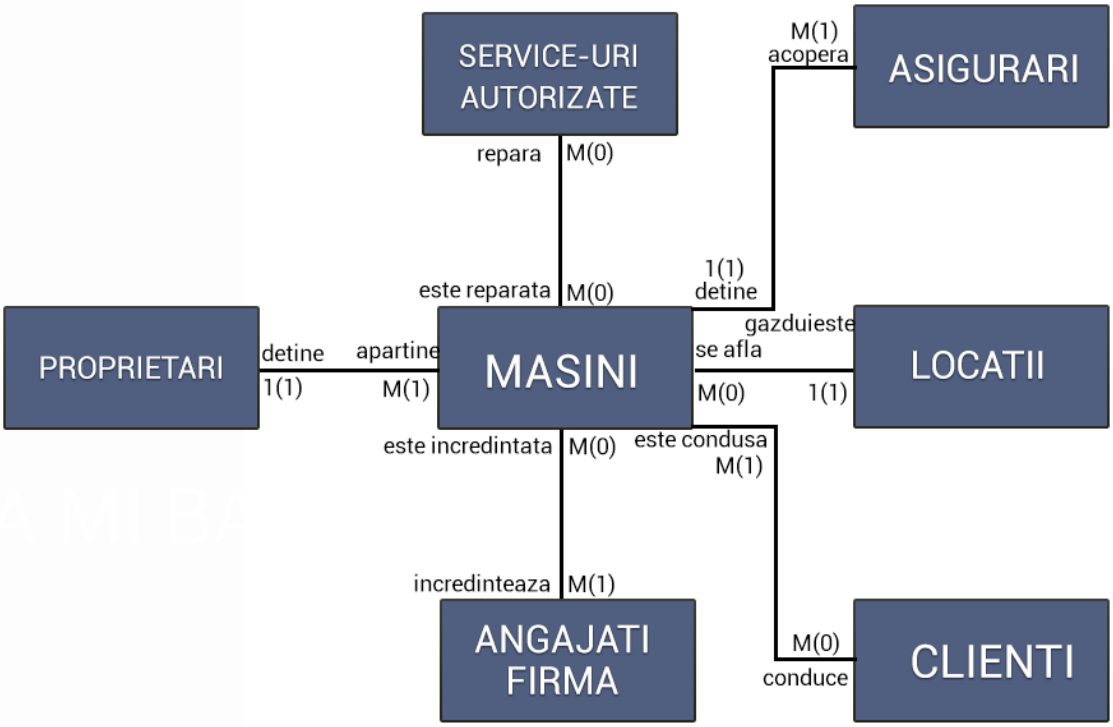
- de semnarea contractului de inchiriere se ocupa un angajat (tabelul ANGAJATI_FIRMA)

- masina detine un istoric cu accidentele pe care le-a avut in timpul in care a fost incredintata firmei de inchirieri (tabelul ACCIDENTE)

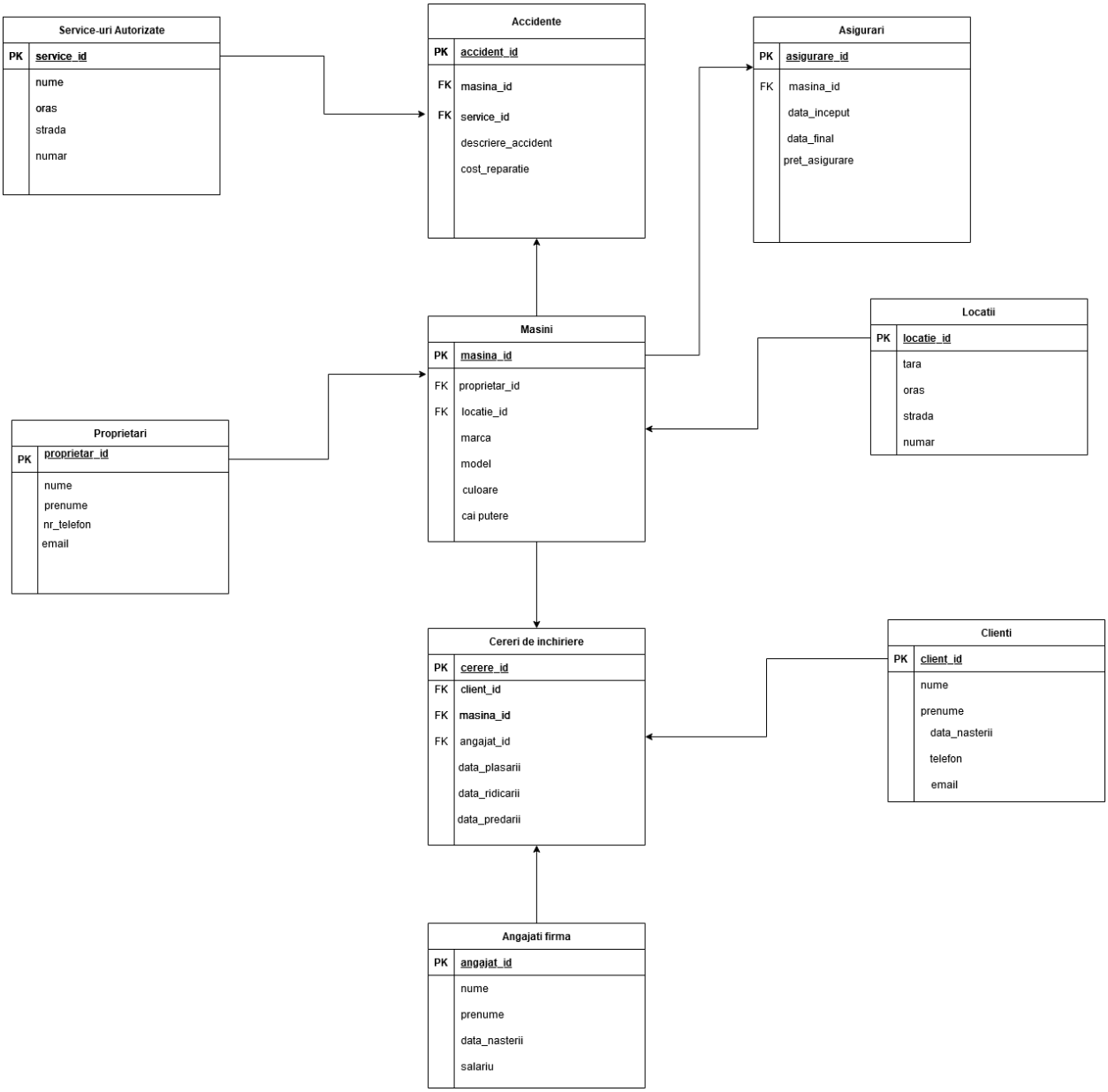
- fiecare avarie din urma unui accident este reparata obligatoriu intr-un service (SERVICEURI_AUTORIZATE)

- in tabelul ASIGURARI gasim istoricul de asigurari al unei masini, prin care ne dam seama daca o masina are asigurarea valabila

2. Realizați diagrama entitate-relație (ERD).

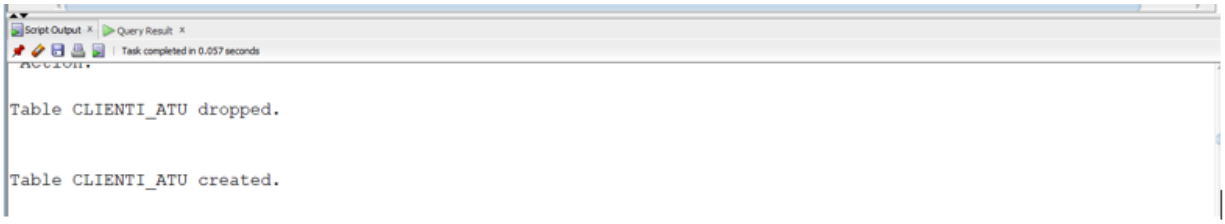


3. Pornind de la diagrama entitate-relație realizați diagrama conceptuală a modelului propus, integrand toate atributele necesare.



4. Implementați în Oracle diagrama conceptuală realizată: definiți toate tabelele, implementând toate constrângerile de integritate necesare (chei primare, cheile externe etc).

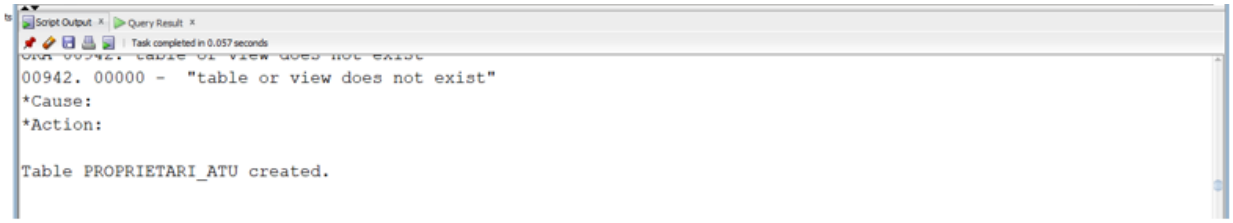
```
drop table clienti_atu;
CREATE TABLE clienti_atu (
  client_id number(10) NOT NULL,
  nume varchar(45) NOT NULL,
  prenume varchar(45) NOT NULL,
  data_nasterii date NOT NULL,
  telefon varchar(45) NOT NULL,
  email varchar(45) NOT NULL,
  CONSTRAINT clienti_atu_pk PRIMARY KEY (client_id)
);
```



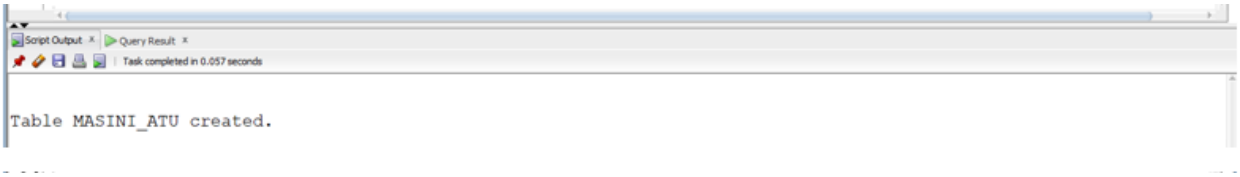

```
drop table serviceuri_autorizate_atu;
CREATE TABLE serviceuri_autorizate_atu (
  service_id number(20) NOT NULL,
  nume varchar(45) NOT NULL,
  oras varchar(45) NOT NULL,
  strada varchar(45) NOT NULL,
  numar number(20) NOT NULL,
  PRIMARY KEY (service_id)
);
```


```
drop table locatii_atu;
CREATE TABLE locatii_atu (
  locatie_id number(20) NOT NULL,
  tara varchar(45) NOT NULL,
  oras varchar(45) NOT NULL,
  strada varchar(45) NOT NULL,
  numar number(20) NOT NULL,
  CONSTRAINT locatii_atu_pk PRIMARY KEY (locatie_id)
);
```


```
drop table proprietari_atu;
CREATE TABLE proprietari_atu (
  proprietar_id number(20) NOT NULL,
  nume varchar(45) NOT NULL,
  prenume varchar(45) NOT NULL,
  nr_telefon varchar(45) NOT NULL,
  email varchar(45) NOT NULL,
  CONSTRAINT proprietari_atu_pk PRIMARY KEY (proprietar_id)
);
```

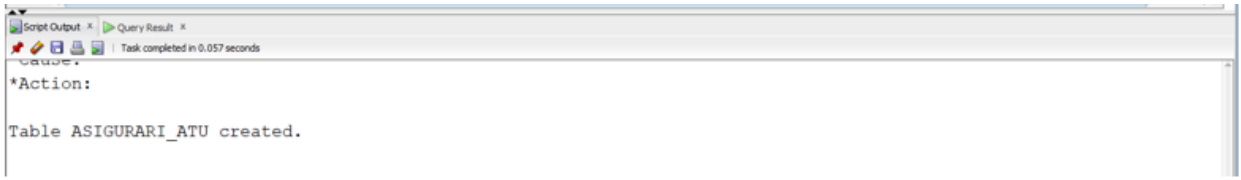



```
drop table masini_atu;

CREATE TABLE masini_atu (

  masina_id number(20) NOT NULL,

  marca varchar(45) NOT NULL,

  model varchar(45) NOT NULL,

  culoare varchar(45) NOT NULL,

  cai_putere number(20) NOT NULL,

  proprietar_id number(20) NOT NULL,

  locatie_id number(20) NOT NULL,

  CONSTRAINT masini_atu_pk PRIMARY KEY (masina_id),

  CONSTRAINT locatie_id FOREIGN KEY (locatie_id) REFERENCES
locatii_atu (locatie_id),

  CONSTRAINT proprietar_id FOREIGN KEY (proprietar_id) REFERENCES
proprietari_atu (proprietar_id) ON DELETE CASCADE

);
```

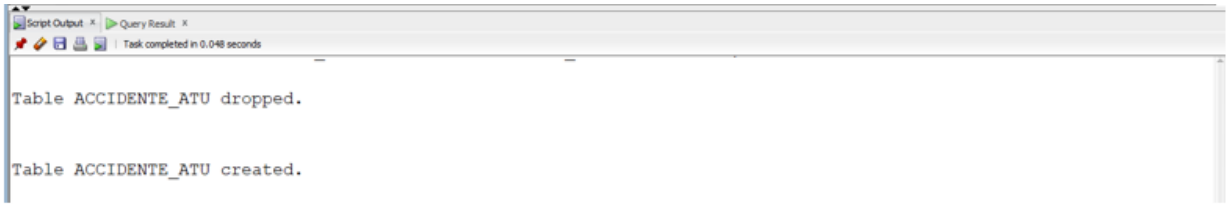


```
DROP TABLE angajati_firma_atu ;
CREATE TABLE angajati_firma_atu (
  angajat_id number(20) NOT NULL,
  nume varchar(45) NOT NULL,
  prenume varchar(45) NOT NULL,
  data_nasterii date NOT NULL,
  salariu number(30) NOT NULL,
  CONSTRAINT angajati_firma_atu_pk PRIMARY KEY (angajat_id)
);
```

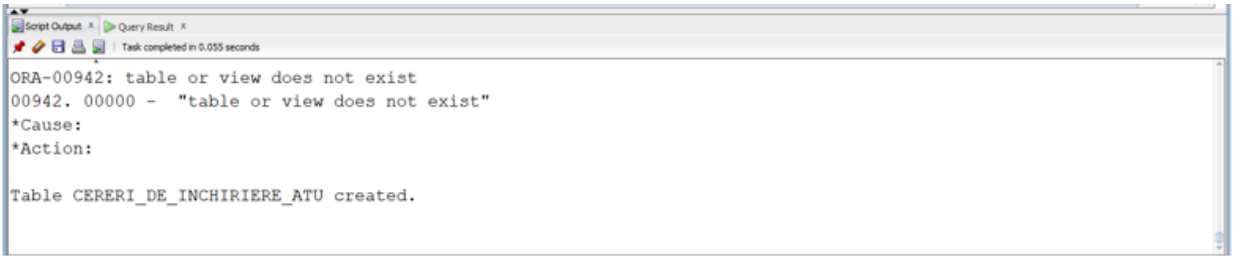

```
drop table asigurari_atu;
CREATE TABLE asigurari_atu (
  asigurare_id number(20) NOT NULL,
  masina_id number(20) NOT NULL,
  data_inceput date NOT NULL,
  data_final date NOT NULL,
  pret_asigurare number(20) NOT NULL,
  PRIMARY KEY (asigurare_id),
  CONSTRAINT masina_id_asigurari FOREIGN KEY (masina_id) REFERENCES
masini_atu (masina_id) ON DELETE CASCADE
);
```




```
drop table accidente_atu;

create TABLE accidente_atu

(

  accident_id number(20),

  masina_id number(20) NOT NULL REFERENCES masini_atu (masina_id) ON
DELETE CASCADE,

  service_id number(20) DEFAULT NULL REFERENCES
serviceuri_autorizate_atu (service_id) ON DELETE CASCADE,


  descriere_accident varchar(200) NOT NULL,

  cost_reparatie number(10) NOT NULL,

  CONSTRAINT accidente_atu_pk PRIMARY KEY(accident_id)

);
```



```
DROP TABLE cereri_de_inchiriere_atu;

CREATE TABLE cereri_de_inchiriere_atu (

  cerere_id number(20) NOT NULL,

  client_id number(20) NOT NULL,

  masina_id number(20) NOT NULL,

  angajat_id number(20) NOT NULL,

  data_plasarii date NOT NULL,

  data_ridicarii date NOT NULL,

  data_predarii date NOT NULL,

  CONSTRAINT cereri_de_inchiriere_atu_pk PRIMARY KEY (cerere_id),

  CONSTRAINT angajat_id FOREIGN KEY (angajat_id) REFERENCES
angajati_firma_atu (angajat_id) ON DELETE CASCADE,

  CONSTRAINT client_id FOREIGN KEY (client_id) REFERENCES clienti_atu
(client_id) ON DELETE CASCADE,

  CONSTRAINT masina_id FOREIGN KEY (masina_id) REFERENCES masini_atu
(masina_id) ON DELETE CASCADE

);
```



5. Adăugați informații coerente în tabelele create (minim 3-5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru tabela asociativă).

```
INSERT INTO locatii_atu
VALUES (1,'Romania','Bucuresti','Regina Elisabeta',483);

INSERT INTO locatii_atu
VALUES (2,'Romania','Bucuresti','Bld. Magheru',23);

INSERT INTO locatii_atu
VALUES (3,'Romania','Cluj','Sos. Oltenitei',82);

INSERT INTO locatii_atu
VALUES (4,'Romania','Timisoara','Mihai Eminescu',32);

INSERT INTO locatii_atu
VALUES (5,'Romania','Constanta','Bld. Tomis ',543);

INSERT INTO locatii_atu
VALUES (6,'Romania','Oradea','Sos. Oltenitei',93);

INSERT INTO locatii_atu
VALUES (7,'Romania','Bucuresti','Lanariei',78);

INSERT INTO locatii_atu
VALUES (8,'Romania','Bucuresti','Sos. Berceni',5);

INSERT INTO locatii_atu
VALUES (9,'Romania','Pitesti','George Enescu',3);
```

```
INSERT INTO locatii_atu
VALUES (10,'Romania','Braila','Vlad Tepes',87);
```

```
155 select * from locatii_atu;
156
```

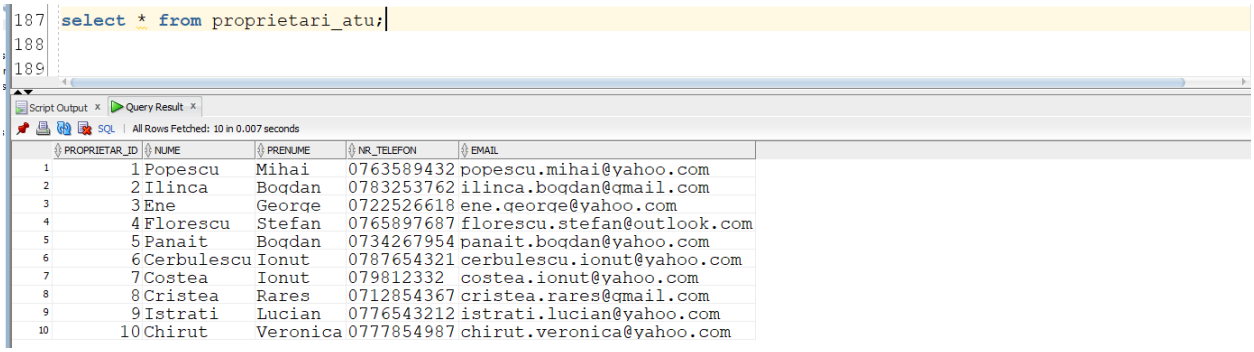
Script Output | Query Result

SQL | All Rows Fetched: 10 in 0.01 seconds

	LOCATIE_ID	TARA	ORAS	STRADA	NUMAR
1	1	Romania	Bucuresti	Regina Elisabeta	483
2	2	Romania	Bucuresti	Bld. Magheru	23
3	3	Romania	Cluj	Sos. Oltenitei	82
4	4	Romania	Timisoara	Mihai Eminescu	32
5	5	Romania	Constanta	Bld. Tomis	543
6	6	Romania	Oradea	Sos. Oltenitei	93
7	7	Romania	Bucuresti	Lanariei	78
8	8	Romania	Bucuresti	Sos. Berceni	5
9	9	Romania	Pitesti	George Enescu	3
10	10	Romania	Braila	Vlad Tepes	87

```
INSERT INTO proprietari_atu
VALUES (1,'Popescu','Mihai','0763589432','popescu.mihai@yahoo.com');

INSERT INTO proprietari_atu
VALUES(2,'Ilinca','Bogdan','0783253762','ilinca.bogdan@gmail.com');

INSERT INTO proprietari_atu
VALUES(3,'Ene','George','0722526618','ene.george@yahoo.com');

INSERT INTO proprietari_atu
VALUES(4,'Florescu','Stefan','0765897687','florescu.stefan@outlook.com');

INSERT INTO proprietari_atu
VALUES(5,'Panait','Bogdan','0734267954','panait.bogdan@yahoo.com');

INSERT INTO proprietari_atu
VALUES(6,'Cerbulescu','Ionut','0787654321','cerbulescu.ionut@yahoo.com');
```

```
INSERT INTO proprietari_atu
VALUES(7,'Costea','Ionut','079812332','costea.ionut@yahoo.com');

INSERT INTO proprietari_atu
VALUES(8,'Cristea','Rares','0712854367','cristea.rares@gmail.com');

INSERT INTO proprietari_atu
VALUES(9,'Istrati','Lucian','0776543212','istrati.lucian@yahoo.com');

INSERT INTO proprietari_atu
VALUES(10,'Chirut','Veronica','0777854987','chirut.veronica@yahoo.com');
```



```
187 select * from proprietari_atu;
188
189
```

Script Output | Query Result

All Rows Fetched: 10 in 0.007 seconds

	PROPRIETAR_ID	NUME	PRENUME	NR_TELEFON	EMAIL
1	1	Popescu	Mihai	0763589432	popescu.mihai@yahoo.com
2	2	Ilinca	Bogdan	0783253762	ilinca.bogdan@gmail.com
3	3	Ene	George	0722526618	ene.george@yahoo.com
4	4	Florescu	Stefan	0765897687	florescu.stefan@outlook.com
5	5	Panait	Bogdan	0734267954	panait.bogdan@yahoo.com
6	6	Cerbulescu	Ionut	0787654321	cerbulescu.ionut@yahoo.com
7	7	Costea	Ionut	079812332	costea.ionut@yahoo.com
8	8	Cristea	Rares	0712854367	cristea.rares@gmail.com
9	9	Istrati	Lucian	0776543212	istrati.lucian@yahoo.com
10	10	Chirut	Veronica	0777854987	chirut.veronica@yahoo.com

Service-uri Autorizate

```
INSERT INTO serviceuri_autorizate_atu
VALUES (1,'PlusAuto SRL','Bucuresti','Sos. Berceni',3);

INSERT INTO serviceuri_autorizate_atu
VALUES (2,'CarDoctor','Cluj','Bld. Ion Luca Caragiale',382);

INSERT INTO serviceuri_autorizate_atu
```

```
VALUES (3,'Bavaria','Bucuresti','Bld. I.C Bratianu',17);

INSERT INTO serviceuri_autorizate_atu
VALUES (4,'Custom Tuning','Bucuresti','Mosoaia',3);

INSERT INTO serviceuri_autorizate_atu
VALUES (5,'Dexcar','Timisoara','Bld. George Enescu',872);

INSERT INTO serviceuri_autorizate_atu
VALUES (6,'Auto Service International','Bucuresti','Bld.
Tineretului',7);

INSERT INTO serviceuri_autorizate_atu
VALUES (7,'TranzitAuto','Oradea','Bld. Ion Maiorescu',82);

INSERT INTO serviceuri_autorizate_atu
VALUES (8,'AutoTraian','Bucuresti','Viilor',3);

INSERT INTO serviceuri_autorizate_atu
VALUES (9,'CheckAuto','Constanta','Tomis Nord',97);

INSERT INTO serviceuri_autorizate_atu
VALUES (10,'Auto Repairs','Brasov','Bisericii',5);
```

```
220 select * from serviceuri_autorizate_atu;
```

Script Output | Query Result

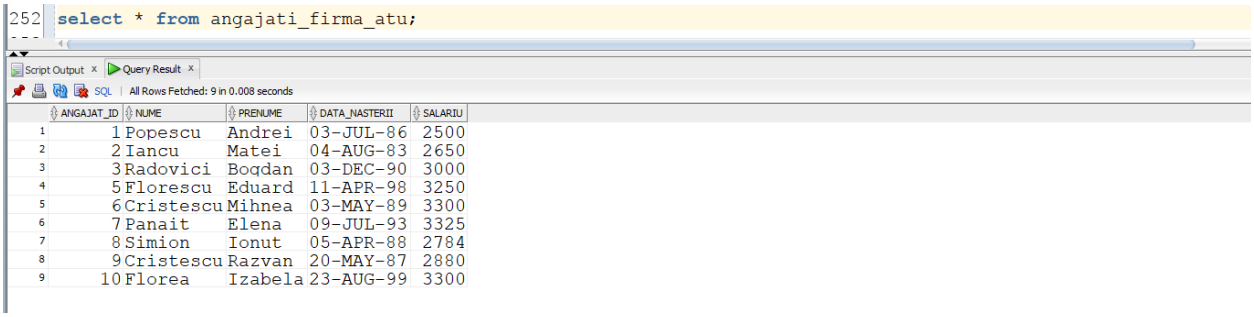
SQL | All Rows Fetched: 10 in 0.016 seconds

	SERVICE_ID	NUME	ORAS	STRADA	NUMAR
1	1	PlusAuto SRL	Bucuresti	Sos. Berceni	3
2	2	CarDoctor	Cluj	Bld. Ion Luca Caragiale	382
3	3	Bavaria	Bucuresti	Bld. I.C Bratianu	17
4	4	Custom Tuning	Bucuresti	Mosoaia	3
5	5	Dexcar	Timisoara	Bld. George Enescu	872
6	6	Auto Service International	Bucuresti	Bld. Tineretului	7
7	7	TranzitAuto	Oradea	Bld. Ion Maiorescu	82
8	8	AutoTraian	Bucuresti	Viilor	3
9	9	CheckAuto	Constanta	Tomis Nord	97
10	10	Auto Repairs	Brasov	Bisericii	5

Angajati firma

```
INSERT INTO angajati_firma_atu
VALUES (1,'Popescu','Andrei',to_date('1986-07-03', 'yy-mm-dd'),2500);

INSERT INTO angajati_firma_atu VALUES
(2,'Iancu','Matei',to_date('1983-08-04', 'yy-mm-dd'),2650);

INSERT INTO angajati_firma_atu VALUES
(3,'Radovici','Bogdan',to_date('1990-12-03', 'yy-mm-dd'),3000);

INSERT INTO angajati_firma_atu VALUES
(4,'Dilibau','Mihai',to_date('1992-10-04, 'yy-mm-dd')',3175);

INSERT INTO angajati_firma_atu VALUES
(5,'Florescu','Eduard',to_date('1998-04-11', 'yy-mm-dd'),3250);

INSERT INTO angajati_firma_atu VALUES
(6,'Cristescu','Mihnea',to_date('1989-05-03', 'yy-mm-dd'),3300);

INSERT INTO angajati_firma_atu VALUES
(7,'Panait','Elena',to_date('1993-07-09', 'yy-mm-dd'),3325);
```

```
INSERT INTO angajati_firma_atu VALUES
(8,'Simion','Ionut',to_date('1988-04-05', 'yy-mm-dd'),2784);

INSERT INTO angajati_firma_atu VALUES
(9,'Cristescu','Razvan',to_date('1987-05-20', 'yy-mm-dd'),2880);

INSERT INTO angajati_firma_atu VALUES
(10,'Florea','Izabela',to_date('1999-08-23', 'yy-mm-dd'),3300);
```



```
252 select * from angajati_firma_atu;
```

Script Output | Query Result | All Rows Fetched: 9 in 0.008 seconds

	ANGAJAT_ID	NUME	PRENUME	DATA_NASTERII	SALARIU
1	1	Popescu	Andrei	03-JUL-86	2500
2	2	Iancu	Matei	04-AUG-83	2650
3	3	Radovici	Bogdan	03-DEC-90	3000
4	5	Florescu	Eduard	11-APR-98	3250
5	6	Cristescu	Mihnea	03-MAY-89	3300
6	7	Panait	Elena	09-JUL-93	3325
7	8	Simion	Ionut	05-APR-88	2784
8	9	Cristescu	Razvan	20-MAY-87	2880
9	10	Florea	Izabela	23-AUG-99	3300

Masini

```
INSERT INTO masini_atu
VALUES (1,'BMW','Seria 3','Rosie',330,1,2);

INSERT INTO masini_atu
VALUES (2,'Mercedes','CLS','Negru',180,3,1);

INSERT INTO masini_atu
VALUES (3,'Land Rover','Range Rover','Alba',200,3,1);

INSERT INTO masini_atu
VALUES (4,'Renaul','Symbol','Verde',95,2,5);
```

```
INSERT INTO masini_atu
VALUES (5,'Mercedes','S Class','Negru',350,5,5);

INSERT INTO masini_atu
VALUES(6,'Skoda','Octavia','Albastra',120,7,3);

INSERT INTO masini_atu
VALUES(7,'Volswagen','Passat','Maro',170,8,9);

INSERT INTO masini_atu
VALUES(8,'Volswagen','Touran','Negru',120,6,9);

INSERT INTO masini_atu
VALUES(9,'Mercedes','GLE Coupe AMG','Alba',350,2,5);

INSERT INTO masini_atu
VALUES(10,'Lamborghini','Aventador','Negru',500,1,2);
```

```
285 select * from masini_atu;
```

Script Output | Query Result

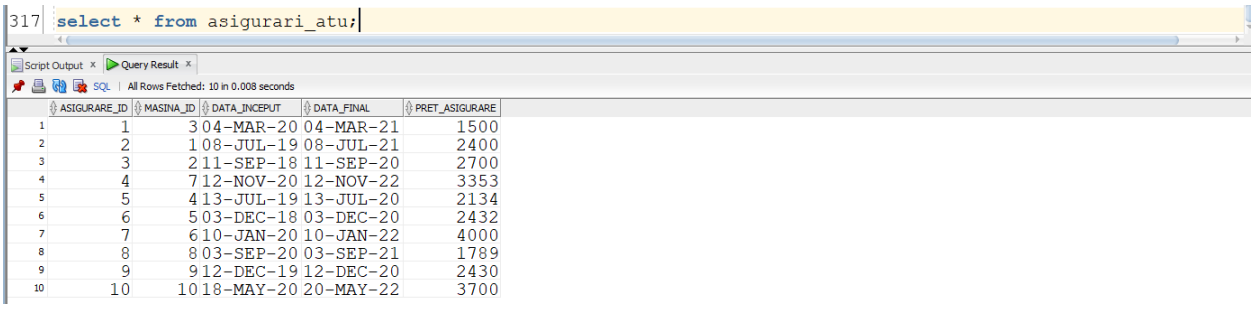
All Rows Fetched: 10 in 0.011 seconds

	MASINA_ID	MARCA	MODEL	CULOARE	CAI_PUTERE	PROPRIETAR_ID	LOCATIE_ID
1	1	BMW	Seria 3	Rosie	330	1	2
2	2	Mercedes	CLS	Negru	180	3	1
3	3	Land Rover	Range Rover	Alba	200	3	1
4	4	Renaul	Symbol	Verde	95	2	5
5	5	Mercedes	S Class	Negru	350	5	5
6	6	Skoda	Octavia	Albastra	120	7	3
7	7	Volswagen	Passat	Maro	170	8	9
8	8	Volswagen	Touran	Negru	120	6	9
9	9	Mercedes	GLE Coupe AMG	Alba	350	2	5
10	10	Lamborghini	Aventador	Negru	500	1	2

Asigurari

```
INSERT INTO asigurari_atu

VALUES (1,3,to_date('2020-03-04','yy-mm-dd'),to_date('2021-03-04','yy-mm-dd'),1500);


INSERT INTO asigurari_atu

VALUES (2,1,to_date('2019-07-08','yy-mm-dd'),to_date('2021-07-08','yy-mm-dd'),2400);


INSERT INTO asigurari_atu

VALUES (3,2,to_date('2018-09-11','yy-mm-dd'),to_date('2020-09-11','yy-mm-dd'),2700);


INSERT INTO asigurari_atu

VALUES (4,7,to_date('2020-11-12','yy-mm-dd'),to_date('2022-11-12','yy-mm-dd'),3353);


INSERT INTO asigurari_atu

VALUES (5,4,to_date('2019-07-13','yy-mm-dd'),to_date('2020-07-13','yy-mm-dd'),2134);


INSERT INTO asigurari_atu

VALUES (6,5,to_date('2018-12-03','yy-mm-dd'),to_date('2020-12-03','yy-mm-dd'),2432);


INSERT INTO asigurari_atu

VALUES (7,6,to_date('2020-01-10','yy-mm-dd'),to_date('2022-01-10','yy-mm-dd'),4000);
```

```
INSERT INTO asigurari_atu

VALUES (8,8,to_date('2020-09-03','yy-mm-dd'),to_date('2021-09-03','yy-mm-dd'),1789);


INSERT INTO asigurari_atu

VALUES (9,9,to_date('2019-12-12','yy-mm-dd'),to_date('2020-12-12','yy-mm-dd'),2430);


INSERT INTO asigurari_atu

VALUES (10,10,to_date('2020-05-18','yy-mm-dd'),to_date('2022-05-20','yy-mm-dd'),3700);
```



```
317 select * from asigurari_atu;
```

Script Output | Query Result | All Rows Fetched: 10 in 0.008 seconds

	ASIGURARE_ID	MASINA_ID	DATA_INCEPUT	DATA_FINAL	PRET_ASIGURARE
1	1	3	04-MAR-20	04-MAR-21	1500
2	2	1	08-JUL-19	08-JUL-21	2400
3	3	2	11-SEP-18	11-SEP-20	2700
4	4	7	12-NOV-20	12-NOV-22	3353
5	5	4	13-JUL-19	13-JUL-20	2134
6	6	5	03-DEC-18	03-DEC-20	2432
7	7	6	10-JAN-20	10-JAN-22	4000
8	8	8	03-SEP-20	03-SEP-21	1789
9	9	9	12-DEC-19	12-DEC-20	2430
10	10	10	18-MAY-20	20-MAY-22	3700

Accidente

```
INSERT INTO accidente_atu

VALUES (1,2,3,'Bara spate zgariata',1000);


INSERT INTO accidente_atu

VALUES (2,1,4,'Aripa fata indoita',1500);


INSERT INTO accidente_atu

VALUES (3,5,8,'Air Bag sarit in urma impactului frontal',3350);


INSERT INTO accidente_atu
```

```
VALUES (4,8,1,'Usa pasager indoita',3400);

INSERT INTO accidente_atu
VALUES (5,7,3,'Bara fata cazuta, far dreapta fata spart',5000);

INSERT INTO accidente_atu
VALUES (6,3,9,'Parbriz Spart, usa sofer indoita',5780);

INSERT INTO accidente_atu
VALUES (7,2,10,'Bloc motor distrus',8000);

INSERT INTO accidente_atu
VALUES (8,8,4,'Bara spate zgariata',2200);

INSERT INTO accidente_atu
VALUES (9,9,5,'Far stanga spate spart',1700);

INSERT INTO accidente_atu
VALUES (10,6,8,'Geam pasager si sofer sparte',2300);
```

```
349 select * from accidente_atu;
```

Script Output | Query Result

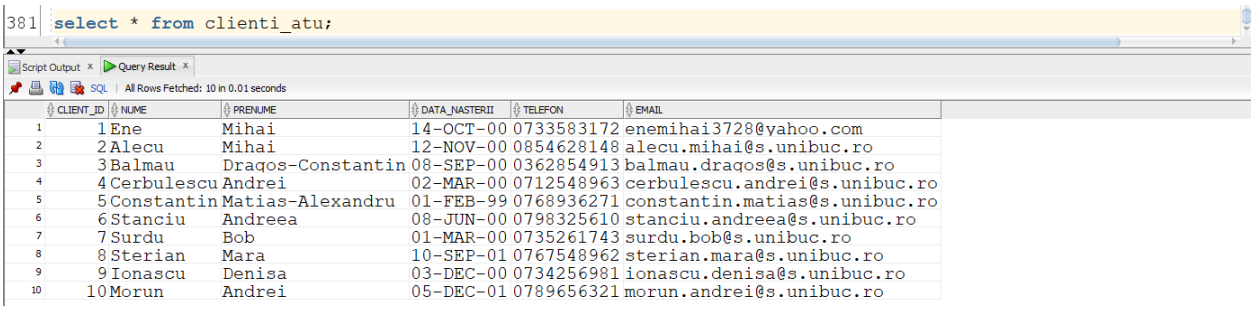
SQL | All Rows Fetched: 10 in 0.008 seconds

	ACCIDENT_ID	MASINA_ID	SERVICE_ID	DESCRIERE_ACCIDENT	COST_REPARATIE
1	1	2	3	Bara spate zgariata	1000
2	2	1	4	Aripa fata indoita	1500
3	3	5	8	Air Bag sarit in urma impactului frontal	3350
4	4	8	1	Usa pasager indoita	3400
5	5	7	3	Bara fata cazuta, far dreapta fata spart	5000
6	6	3	9	Parbriz Spart, usa sofer indoita	5780
7	7	2	10	Bloc motor distrus	8000
8	8	8	4	Bara spate zgariata	2200
9	9	9	5	Far stanga spate spart	1700
10	10	6	8	Geam pasager si sofer sparte	2300

Clienti

```
INSERT INTO clienti_atu

VALUES(1,'Ene','Mihai',to_date('2000-10-14', 'yy-mm-
dd'),'0733583172','enemihai3728@yahoo.com');


INSERT INTO clienti_atu

VALUES(2,'Alecu','Mihai',to_date('2000-11-12', 'yy-mm-
dd'),'0854628148','alecu.mihai@s.unibuc.ro');


INSERT INTO clienti_atu

VALUES(3,'Balmau','Dragos-Constantin',to_date('2000-09-08', 'yy-mm-
dd'),'0362854913','balmau.dragos@s.unibuc.ro');


INSERT INTO clienti_atu

VALUES(4,'Cerbulescu','Andrei',to_date('2000-03-02', 'yy-mm-
dd'),'0712548963','cerbulescu.andrei@s.unibuc.ro');


INSERT INTO clienti_atu

VALUES(5,'Constantin','Matias-Alexandru',to_date('1999-02-01', 'yy-mm-
dd'),'0768936271','constantin.matias@s.unibuc.ro');


INSERT INTO clienti_atu

VALUES(6,'Stanciu','Andreea',to_date('2000-06-08', 'yy-mm-
dd'),'0798325610','stanciu.andreea@s.unibuc.ro');


INSERT INTO clienti_atu

VALUES(7,'Surdu','Bob',to_date('2000-03-01', 'yy-mm-
dd'),'0735261743','surdu.bob@s.unibuc.ro');
```

```
INSERT INTO clienti_atu

VALUES(8,'Sterian','Mara',to_date('2001-09-10', 'yy-mm-
dd'),'0767548962','sterian.mara@s.unibuc.ro');


INSERT INTO clienti_atu

VALUES(9,'Ionascu','Denisa',to_date('2000-12-03', 'yy-mm-
dd'),'0734256981','ionascu.denisa@s.unibuc.ro');


INSERT INTO clienti_atu

VALUES(10,'Morun','Andrei',to_date('2001-12-05', 'yy-mm-
dd'),'0789656321','morun.andrei@s.unibuc.ro');
```



```
381 select * from clienti_atu;
```

Script Output | Query Result | SQL | All Rows Fetched: 10 in 0.01 seconds

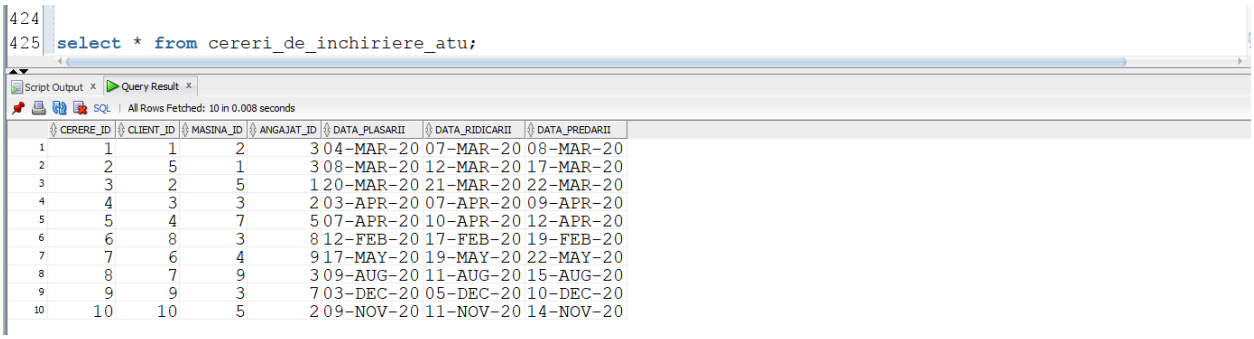
	CLIENT_ID	NUME	PRENUME	DATA_NASTERII	TELEFON	EMAIL
1	1	Ene	Mihai	14-OCT-00	0733583172	enemihai3728@yahoo.com
2	2	Alecu	Mihai	12-NOV-00	0854628148	alecu.mihai@s.unibuc.ro
3	3	Balmau	Dragos-Constantin	08-SEP-00	0362854913	balmau.dragos@s.unibuc.ro
4	4	Cerbulescu	Andrei	02-MAR-00	0712548963	cerbulescu.andrei@s.unibuc.ro
5	5	Constantin	Matias-Alexandru	01-FEB-99	0768936271	constantin.matias@s.unibuc.ro
6	6	Stanciu	Andreea	08-JUN-00	0798325610	stanciu.andreea@s.unibuc.ro
7	7	Surdu	Bob	01-MAR-00	0735261743	surdu.bob@s.unibuc.ro
8	8	Sterian	Mara	10-SEP-01	0767548962	sterian.mara@s.unibuc.ro
9	9	Ionascu	Denisa	03-DEC-00	0734256981	ionascu.denisa@s.unibuc.ro
10	10	Morun	Andrei	05-DEC-01	0789656321	morun.andrei@s.unibuc.ro

Cereri de inchiriere

```
INSERT INTO cereri_de_inchiriere_atu

VALUES(1,1,2,3,to_date('2020-03-04', 'yy-mm-dd'),to_date('2020-03-07',
'yy-mm-dd'),

to_date('2020-03-08', 'yy-mm-dd'));


INSERT INTO cereri_de_inchiriere_atu

VALUES(2,5,1,3,to_date('2020-03-08', 'yy-mm-dd'),to_date('2020-03-12',
'yy-mm-dd'),

to_date('2020-03-17', 'yy-mm-dd'));
```

```
INSERT INTO cereri_de_inchiriere_atu

VALUES(3,2,5,1,to_date('2020-03-20', 'yy-mm-dd'),to_date('2020-03-21',
'yy-mm-dd'),

to_date('2020-03-22', 'yy-mm-dd'));


INSERT INTO cereri_de_inchiriere_atu

VALUES(4,3,3,2,to_date('2020-04-03', 'yy-mm-dd'),to_date('2020-04-07',
'yy-mm-dd'),

to_date('2020-04-09', 'yy-mm-dd'));


INSERT INTO cereri_de_inchiriere_atu

VALUES(5,4,7,5,to_date('2020-04-07', 'yy-mm-dd'),to_date('2020-04-10',
'yy-mm-dd'),

to_date('2020-04-12', 'yy-mm-dd'));


INSERT INTO cereri_de_inchiriere_atu

VALUES(6,8,3,8,to_date('2020-02-12', 'yy-mm-dd'),to_date('2020-02-17',
'yy-mm-dd'),

to_date('2020-02-19', 'yy-mm-dd'));


INSERT INTO cereri_de_inchiriere_atu

VALUES(7,6,4,9,to_date('2020-05-17', 'yy-mm-dd'),to_date('2020-05-19',
'yy-mm-dd'),

to_date('2020-05-22', 'yy-mm-dd'));


INSERT INTO cereri_de_inchiriere_atu

VALUES(8,7,9,3,to_date('2020-08-09', 'yy-mm-dd'),to_date('2020-08-11',
'yy-mm-dd'),

to_date('2020-08-15', 'yy-mm-dd'));
```

```
INSERT INTO cereri_de_inchiriere_atu

VALUES(9,9,3,7,to_date('2020-12-03', 'yy-mm-dd'),to_date('2020-12-05',
'yy-mm-dd'),

to_date('2020-12-10', 'yy-mm-dd'));


INSERT INTO cereri_de_inchiriere_atu

VALUES(10,10,5,2,to_date('2020-11-09', 'yy-mm-dd'),to_date('2020-11-
11', 'yy-mm-dd'),

to_date('2020-11-14', 'yy-mm-dd'));
```



```
424
425 select * from cereri_de_inchiriere_atu;
```

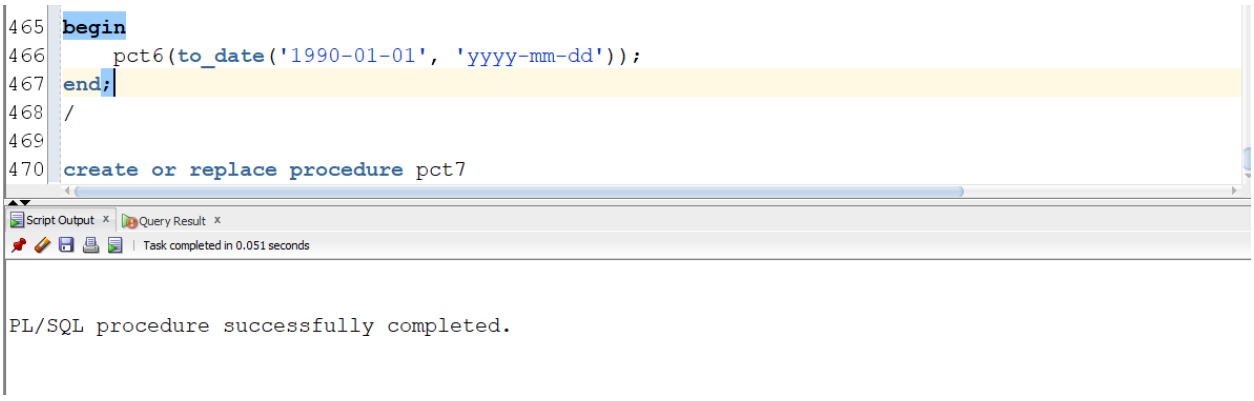
Script Output | Query Result | All Rows Fetched: 10 in 0.008 seconds

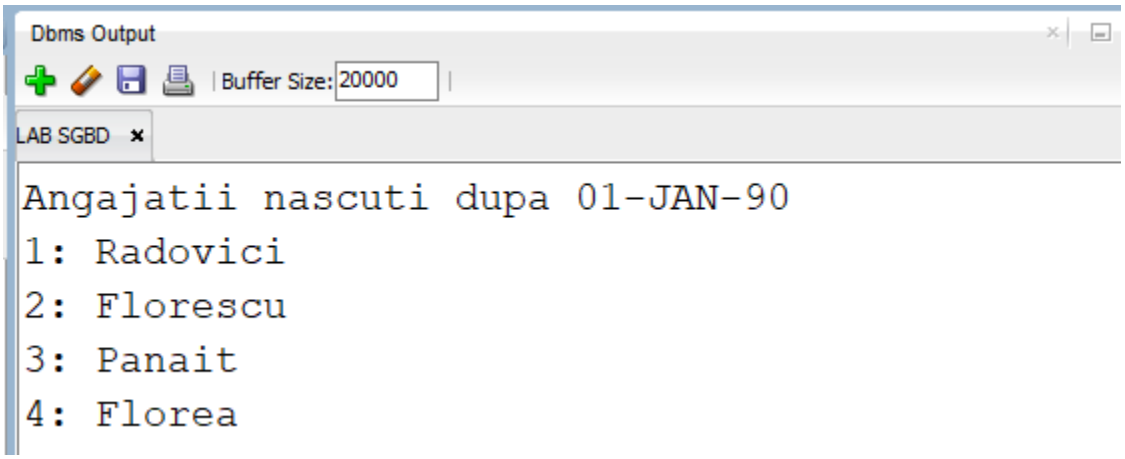
	CERERE_ID	CLIENT_ID	MASINA_ID	ANGAJAT_ID	DATA_PLASARII	DATA_RIDICARII	DATA_PREDARII
1	1	1	2	3	04-MAR-20	07-MAR-20	08-MAR-20
2	2	5	1	3	08-MAR-20	12-MAR-20	17-MAR-20
3	3	2	5	1	20-MAR-20	21-MAR-20	22-MAR-20
4	4	3	3	2	03-APR-20	07-APR-20	09-APR-20
5	5	4	7	5	07-APR-20	10-APR-20	12-APR-20
6	6	8	3	8	12-FEB-20	17-FEB-20	19-FEB-20
7	7	6	4	9	17-MAY-20	19-MAY-20	22-MAY-20
8	8	7	9	3	09-AUG-20	11-AUG-20	15-AUG-20
9	9	9	3	7	03-DEC-20	05-DEC-20	10-DEC-20
10	10	10	5	2	09-NOV-20	11-NOV-20	14-NOV-20

6. Definiți un subprogram stocat care să utilizeze un tip de colecție studiat. Apelați subprogramul.

```
--procedura afiseaza in dbms output toti angajatii nascuti dupa o data de nastere

--data ca parametru si se foloseste de un tablou imbircat si de un record

create or replace procedure pct6 (val_data date)

is

    TYPE record_6 IS RECORD

         (nume angajati_firma_atu.nume%TYPE,

         prenume angajati_firma_atu.prenume%TYPE,

         data_nasterii angajati_firma_atu.data_nasterii%TYPE);

    angajat record_6;



    TYPE nested_table is TABLE OF record_6;

    tablou_angajati nested_table := nested_table();


    cursor c is

        select nume, prenume, data_nasterii

        from angajati_firma_atu

        where data_nasterii > val_data;

BEGIN

    open c;

    loop

        fetch c into angajat.nume, angajat.prenume, angajat.data_nasterii;

        exit when c%notfound;

        tablou_angajati.extend;
```

```
        tablou_angajati(tablou_angajati.LAST) := angajat;
    end loop;
    close c;

    dbms_output.put_line('Angajatii nascuti dupa ' || val_data);
    for elem in 1 .. tablou_angajati.count loop
        dbms_output.put_line(elem || ': ' ||
tablou_angajati(elem).nume);
    end loop;
END;
/
select * from angajati_firma_atu;

begin
    pct6(to_date('1990-01-01', 'yyyy-mm-dd'));
end;
/
```

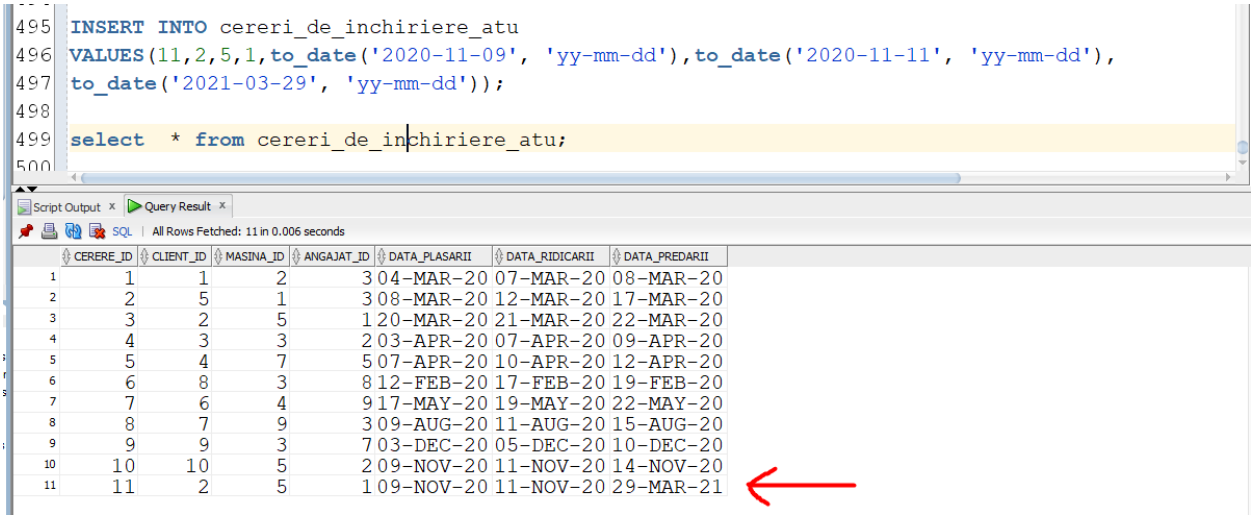



Dbms Output
Buffer Size: 20000
LAB SGBD
Angajatii nascuti dupa 01-JAN-90
1: Radovici
2: Florescu
3: Panait
4: Florea

7. Definiți un subprogram stocat care să utilizeze un tip de cursor studiat. Apelați subprogramul.

```
--procedura are efectul de a incheia toate inchirierile care sunt in
desfasurare,
--setand data_predarii ca fiind data curenta a utilizatorului
create or replace procedure pct7
is
    cursor c is
        select cerere_id, data_predarii
        from cereri_de_inchiriere_atu;
    temp_cod cereri_de_inchiriere_atu.cerere_id%type;
    temp_data cereri_de_inchiriere_atu.data_predarii%type;
begin
    open c;
    loop
        fetch c into temp_cod, temp_data;
        exit when c%notfound;
        if temp_data > sysdate then
            update cereri_de_inchiriere_atu
            set data_predarii = sysdate
            where cerere_id = temp_cod;
        end if;
    end loop;

    close c;
end;
/
```


```
INSERT INTO cereri_de_inchiriere_atu

VALUES(11,2,5,1,to_date('2020-11-09', 'yy-mm-dd'),to_date('2020-11-11', 'yy-mm-dd'),

to_date('2021-03-29', 'yy-mm-dd'));


select  * from cereri_de_inchiriere_atu;
```



```
begin
    pct7();
end;
/
```

```
501 begin
502     pct7();
503 end;
504 /
505
506 select  * from cereri_de_inchiriere_atu;
507
```

Script Output | Query Result

SQL | All Rows Fetched: 11 in 0.006 seconds

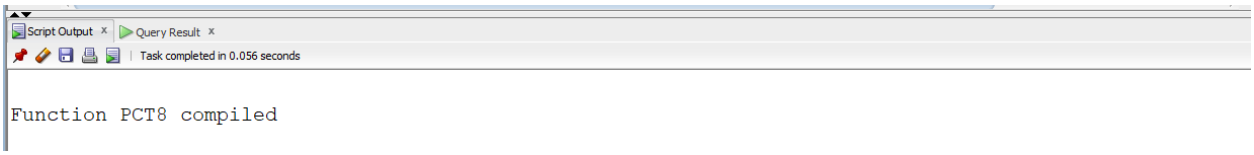
	CERERE_ID	CLIENT_ID	MASINA_ID	ANGAJAT_ID	DATA_PLASARII	DATA_RIDICARII	DATA_PREDARII
1	1	1	2	3	04-MAR-20	07-MAR-20	08-MAR-20
2	2	5	1	3	08-MAR-20	12-MAR-20	17-MAR-20
3	3	2	5	1	20-MAR-20	21-MAR-20	22-MAR-20
4	4	3	3	2	03-APR-20	07-APR-20	09-APR-20
5	5	4	7	5	07-APR-20	10-APR-20	12-APR-20
6	6	8	3	8	12-FEB-20	17-FEB-20	19-FEB-20
7	7	6	4	9	17-MAY-20	19-MAY-20	22-MAY-20
8	8	7	9	3	09-AUG-20	11-AUG-20	15-AUG-20
9	9	9	3	7	03-DEC-20	05-DEC-20	10-DEC-20
10	10	10	5	2	09-NOV-20	11-NOV-20	14-NOV-20
11	11	2	5	1	09-NOV-20	11-NOV-20	07-JAN-21

8. Definiți un subprogram stocat de tip funcție care să utilizeze 3 dintre tabelele definite. Tratați toate excepțiile care pot apărea. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

```
-- functia ia ca parametru codul unei cereri de inchiriere si cauta
--daca exista vreun angajat cu acelasi nume de familie precum al
--clientului care a semnat acea inchiriere, si in caz afirmativ
--ii afiseaza prenumele angajatului, in caz contrar afiseaza mesaje
create or replace function pct8 (cod_inchiriere
cereri_de_inchiriere_atu.cerere_id%type)
RETURN VARCHAR2
IS
    temp_client_id cereri_de_inchiriere_atu.client_id%type;
    temp_angajat_id cereri_de_inchiriere_atu.angajat_id%type;

    temp_client_nume clienti_atu.nume%type;

    cursor c_angajati (nume_client_var clienti_atu.nume%type)
    is
        select prenume
        from angajati_firma_atu
        where nume = nume_client_var;

    return_prenume angajati_firma_atu.prenume%type;

    cursor c_gasirecerere
    is
        select data_plasarii
        from cereri_de_inchiriere_atu
```

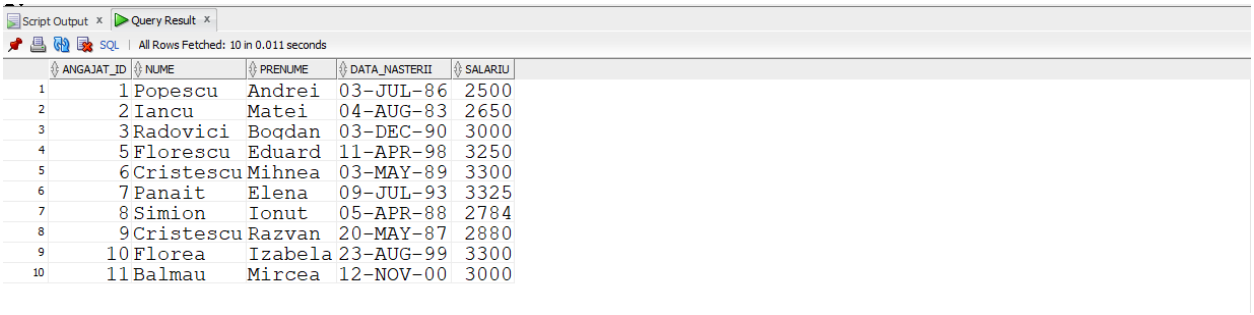
```
        where cerere_id = cod_inchiriere;

    gasirecerere cereri_de_inchiriere_atu.data_plasarii%type;

    nu_exista_cererea EXCEPTION;
    PRAGMA EXCEPTION_INIT (nu_exista_cererea, -20009);

    nu_exista_angajat EXCEPTION;
    PRAGMA EXCEPTION_INIT (nu_exista_angajat, -20010);
BEGIN
    open c_gasirecerere;
    loop
        fetch c_gasirecerere into gasirecerere;
        if gasirecerere is null then
            raise nu_exista_cererea;
        end if;
        exit when c_gasirecerere%notfound;
    end loop;
    close c_gasirecerere;

    --obtinem id_client din cererea aia
    SELECT client_id
    into temp_client_id
    from cereri_de_inchiriere_atu
    where cerere_id = cod_inchiriere;

    --obtinem numele clientului in functie de id
    select nume
```

```
    into temp_client_nume
    from clienti_atu
    where client_id = temp_client_id;

    --cautam angajatul cu acel nume si returnam prenumele sau
    open c_angajati(temp_client_nume);
    fetch c_angajati into return_prenume;
    if return_prenume is not null then
        return return_prenume;
    else
        raise nu_exista_angajat;
    end if;

    close c_angajati;
    return return_prenume;
EXCEPTION
    WHEN nu_exista_cererea then
        dbms_output.put_line('Cererea de inchiriere data ca parametru
nu exista');
        return '-';
     WHEN nu_exista_angajat then
        dbms_output.put_line('Numele clientului nu coincide cu niciun
angajat');
        return '-';
    WHEN OTHERS THEN
        dbms_output.put_line('alta eroare');
        return '-';
END;
/
```




```
--adaugam angajat care sa aiba prenumele Balmau, la fel ca clientul
--care a realizat inchirierea cu id-ul 4
insert into angajati_firma_atu
values(11, 'Balmau', 'Mircea', to_date('2000-11-12', 'yyyy-mm-dd'),
3000);
```

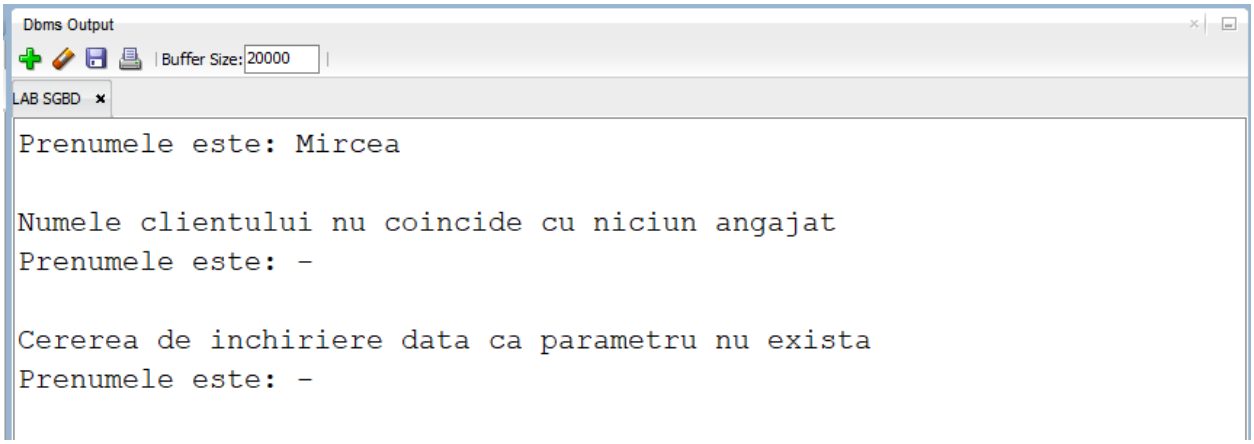

```
select * from angajati_firma_atu;
```



	ANGAJAT_ID	NUME	PRENUME	DATA_NASTERII	SALARIU
1	1	Popescu	Andrei	03-JUL-86	2500
2	2	Iancu	Matei	04-AUG-83	2650
3	3	Radovici	Bogdan	03-DEC-90	3000
4	5	Florescu	Eduard	11-APR-98	3250
5	6	Cristescu	Mihnea	03-MAY-89	3300
6	7	Panait	Elena	09-JUL-93	3325
7	8	Simion	Ionut	05-APR-88	2784
8	9	Cristescu	Razvan	20-MAY-87	2880
9	10	Florea	Izabela	23-AUG-99	3300
10	11	Balmau	Mircea	12-NOV-00	3000

```
declare
    rezultat VARCHAR(50);
begin
    rezultat := pct8(4);
    dbms_output.put_line('Prenumele este: ' || rezultat);
    dbms_output.put_line(' ');

    rezultat := pct8(2);
```

```
    dbms_output.put_line('Prenumele este: ' || rezultat);
    dbms_output.put_line(' ');

    rezultat := pct8(25);
    dbms_output.put_line('Prenumele este: ' || rezultat);

end;
/
```



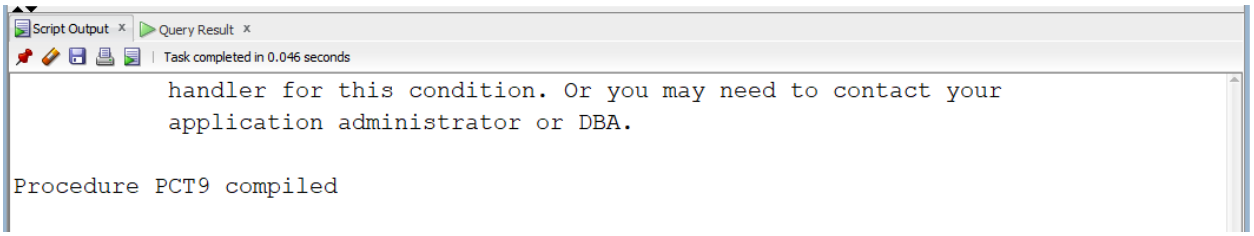
9. Definiți un subprogram stocat de tip procedură care să utilizeze 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

```
--procedura primeste marca si modelul si
--afiseaza accidentele in urma caruia masinile
--au fost reparate in orasul in care se afla iar costul reparatiei
--nu depaseste pretul asigurarii
create or replace procedure pct9(
    param_marca masini_atu.marca%type,
    param_model masini_atu.model%type
    )
is
    var_accident_id accidente_atu.accident_id%type;
    var_oras_service serviceuri_autorizate_atu.oras%type;
    var_oras_locatie locatii_atu.oras%type;
    var_cost_reparatie accidente_atu.cost_reparatie%type;
    var_pret_asigurare asigurari_atu.pret_asigurare%type;
    var_marca masini_atu.marca%type;
    var_model masini_atu.model%type;


    cursor c is
        select
            acc.accident_id "ID accident",
            ser.oras "Oras service",
            loc.oras "Oras locatie",
            acc.cost_reparatie "Cost reparatie",
            asi.pret_asigurare "Pret asigurare",
```

```
            mas.marca "Marca masinii",
            mas.model "Modelul masinii"
        from
            serviceuri_autorizate_atu ser
            join accidente_atu acc
                on acc.service_id = ser.service_id
            join masini_atu mas
                on mas.masina_id = acc.masina_id
            join asigurari_atu asi
                on asi.masina_id = mas.masina_id
            join locatii_atu loc
                on loc.locatie_id = mas.locatie_id;

    nu_sunt_masini EXCEPTION;
    nu_sunt_accidente EXCEPTION;
    nu_exista_masina EXCEPTION;
    i integer := 0;
    nr integer := 0;
begin
    select count(*)
    into nr
    from masini_atu
    where model = param_model and marca = param_marca;

    if nr = 0 then
        raise nu_exista_masina;
    end if;
    open c;
```

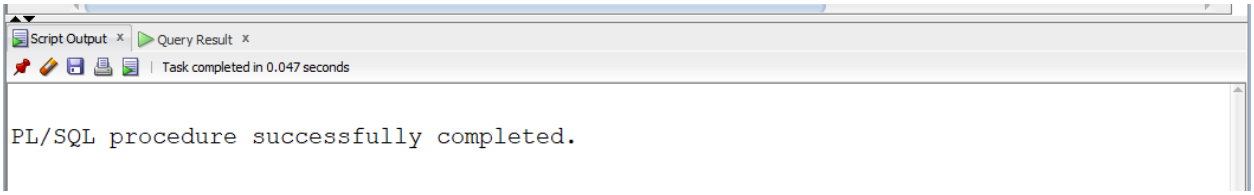
```
    loop
        fetch c into
            var_accident_id,
            var_oras_service,
            var_oras_locatie,
            var_cost_reparatie,
            var_pret_asigurare,
            var_marca,
            var_model;
        exit when c%notfound;
        if
            var_marca = param_marca
            and var_model = param_model
            and var_oras_service = var_oras_locatie
            and var_cost_reparatie <= var_pret_asigurare then
            dbms_output.put_line('Masina implicata in accident ' ||
var_marca || ' ' || var_model);
            dbms_output.put_line('  se afla in orasul ' ||
var_oras_locatie);
            dbms_output.put_line('  si a fost reparata tot in ' ||
var_oras_service);
            dbms_output.put_line('Costul reparatiei (' ||
var_cost_reparatie || ')' ||
            ' nu a depasit pretul asigurarii (' || var_pret_asigurare
|| ').');
            dbms_output.new_line();
            i := i+1;
        end if;
    end loop;
    if c%rowcount = 0 then
```

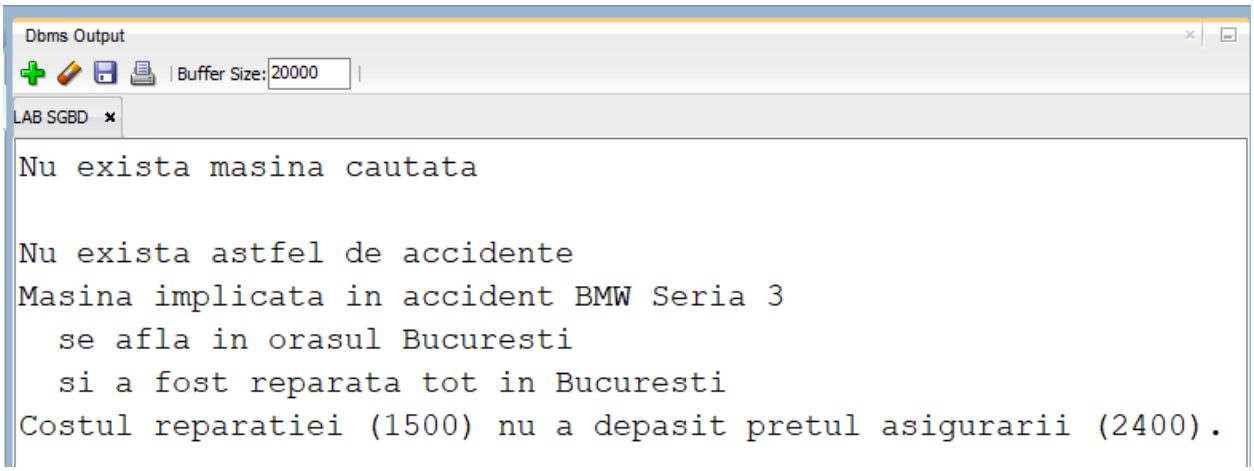
```
        raise nu_sunt_masini;
    elsif i = 0 then
        raise nu_sunt_accidente;
    end if;
    close c;
EXCEPTION
    when nu_sunt_masini then
        dbms_output.put_line('Nu exista masini in baza de date');
    when nu_sunt_accidente then
        dbms_output.put_line('Nu exista astfel de accidente');
    when nu_exista_masina then
        dbms_output.put_line('Nu exista masina cautata');
    when others then
        dbms_output.put_line('Ati apelat gresit cererea');
end;
/
```




```
begin
    pct9('Dacia', 'Logan'); --nu exista in baza de date

    pct9('Land Rover', 'Range Rover'); --exista dar nu sunt accidente

    pct9('BMW', 'Seria 3'); --aici afiseaza
```

```
EXCEPTION
    when OTHERS THEN
        dbms_output.put_line('Ati apelat gresit cererea');
end;
/
```

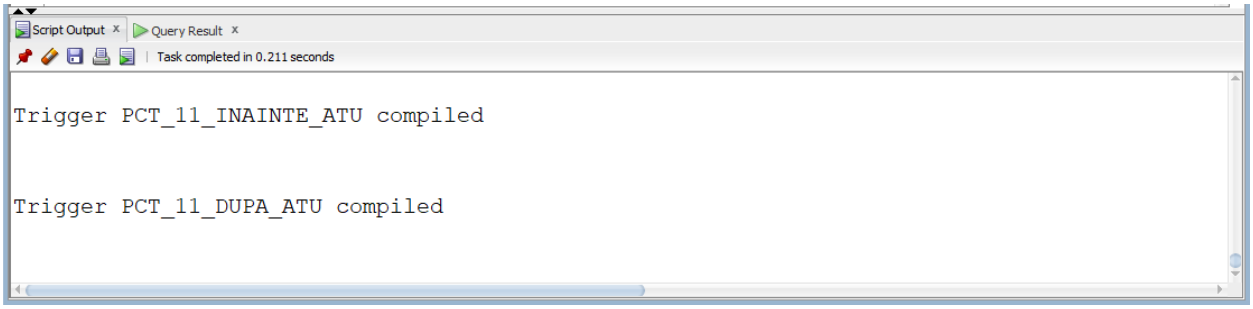




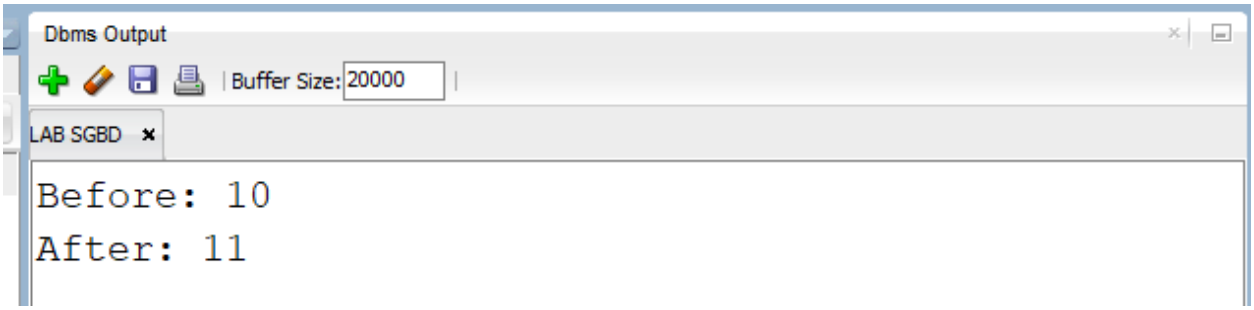
10. Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul.

```
--functie ajutatoare
create or replace function count_masini_atu
    return number
is
    numar integer;
begin
    select count(*)
    into numar
    from masini_atu;
    return numar;
end;
/

--doi triggeri, unul before si celalalt after, care afiseaza
--in dbms_output numarul de inregistrari din tabel inainte si dupa
--inserare sau stergere in tabelul masini_atu
CREATE OR REPLACE TRIGGER pct_11_inainte_atu
before insert or delete on masini_atu
begin
    dbms_output.put_line('Before: ' || count_masini_atu);
end;
/
```

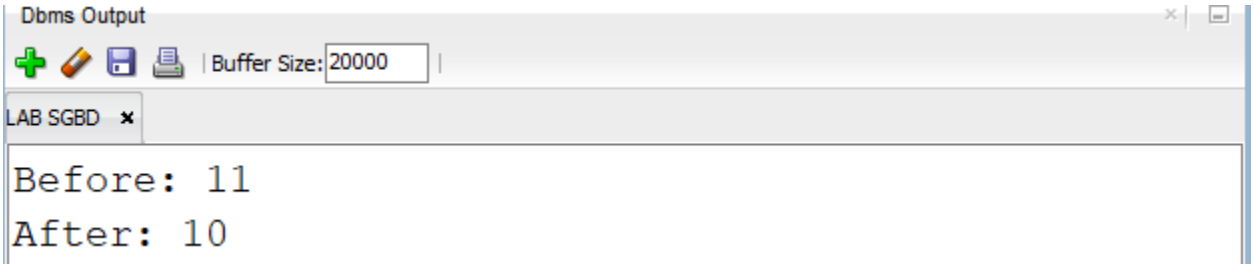
```
CREATE OR REPLACE TRIGGER pct_11_dupa_atu
after insert or delete on masini_atu
begin
    dbms_output.put_line('After: ' || count_masini_atu);
end;
/
```




```
insert into masini_atu values(11, 'Masina', 'Test', 'Alb', 100, 1, 1);
```

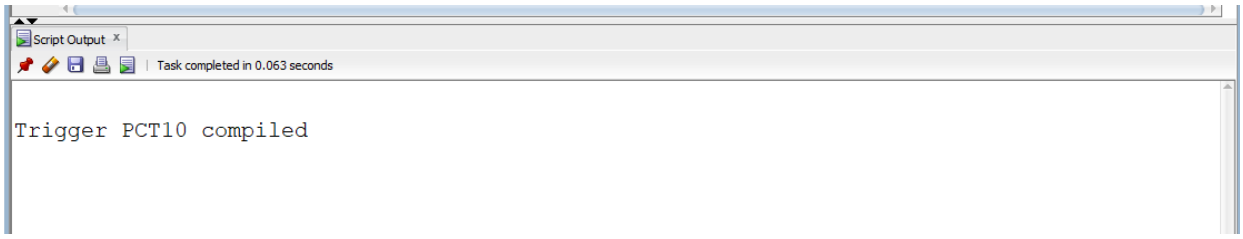



```
delete from masini_atu where masina_id = 11;
```

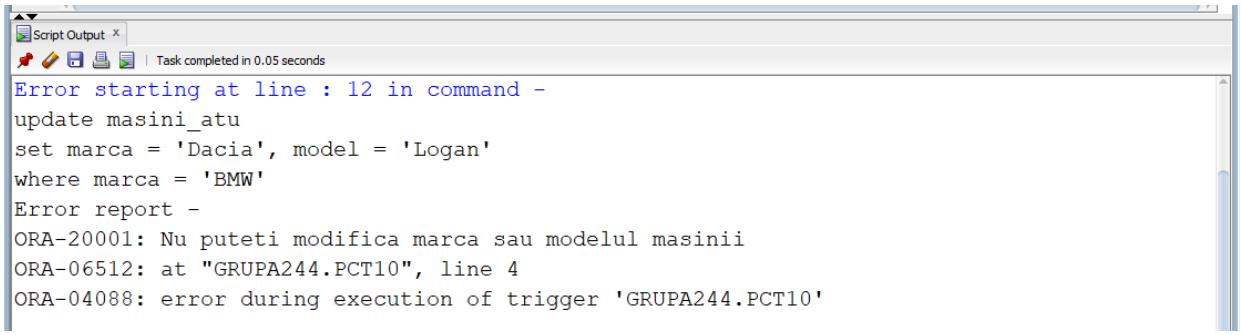


11. Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul.

```
--nu se poate modifica marca sau modelul masinii odata ce a fost
introdusa,
--deoarece marca si modelul unei masini nu pot fi
--schimbate legal in cartea de identitate a masinii
CREATE OR REPLACE TRIGGER pct10
    BEFORE UPDATE ON masini_atu
    FOR EACH ROW
BEGIN
    IF (:NEW.marca <> :OLD.marca) OR (:NEW.model <> :OLD.model)
    THEN
        RAISE_APPLICATION_ERROR(-20001,'Nu puteti modifica marca sau
modelul masinii');
    END IF;
END;
/
```

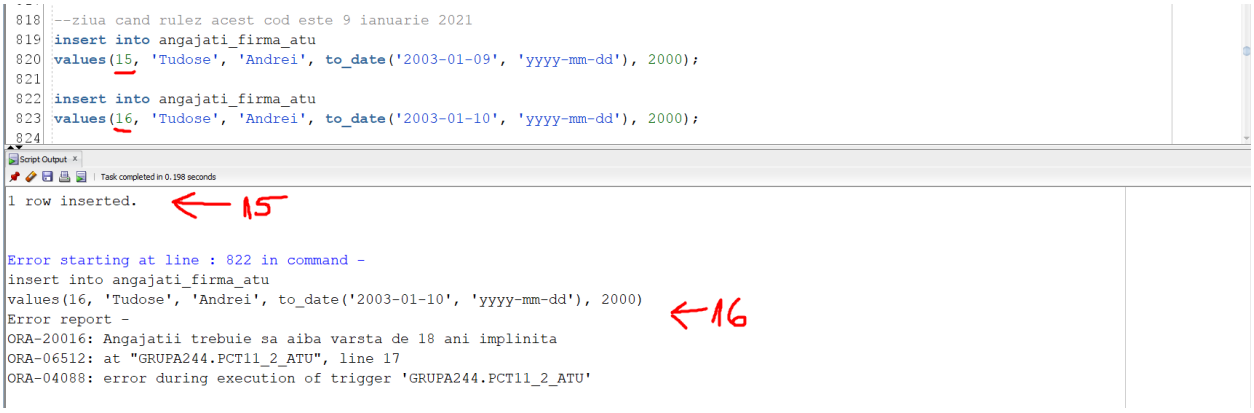



```
update masini_atu
set marca = 'Dacia'
where marca = 'BMW';
```




```
--un alt trigger care verifica ca angajatii introdusi

--sa aiba varsta de 18 ani implinita la data angajarii

CREATE OR REPLACE TRIGGER pct10_2

    BEFORE INSERT ON angajati_firma_atu

    for each row

declare

    ani_bisecti integer;

BEGIN

    ani_bisecti := 0;

    for i in extract(year from :new.data_nasterii)..extract(year from
sysdate) loop

        if MOD(i, 4) = 0 and MOD(i, 100) <> 0 then

            ani_bisecti := ani_bisecti + 1;

        elsif MOD(i, 400) = 0 then

            ani_bisecti := ani_bisecti + 1;

        end if;

    end loop;

    dbms_output.put_line(floor(sysdate - :new.data_nasterii -
ani_bisecti) / 365);

    if floor(sysdate - :new.data_nasterii - ani_bisecti) / 365 < 18

    THEN

        RAISE_APPLICATION_ERROR(-20016,'Angajatii trebuie sa aiba
varsta de 18 ani implinita');
```

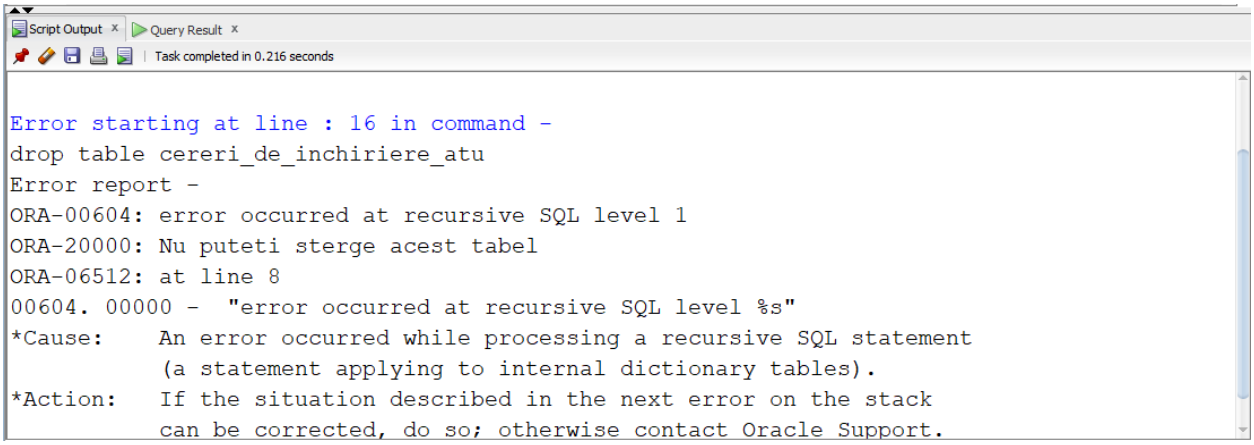
```
    END IF;
END;
/


--ziua cand rulez acest cod este 9 ianuarie 2021
insert into angajati_firma_atu
values(12, 'Tudose', 'Andrei', to_date('2003-01-09', 'yyyy-mm-dd'),
2000);


insert into angajati_firma_atu
values(12, 'Tudose', 'Andrei', to_date('2003-01-10', 'yyyy-mm-dd'),
2000);
```

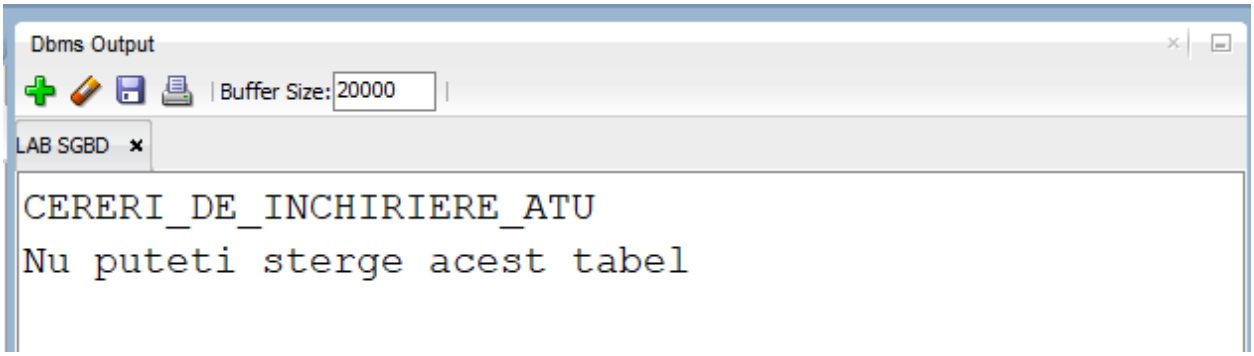


12. Definiți un trigger de tip LDD. Declanșați trigger-ul.

```
create or replace trigger pct12_atu
    AFTER DROP ON SCHEMA
begin
    dbms_output.put_line(ORA_DICT_OBJ_NAME);
    if UPPER(ORA_DICT_OBJ_NAME) in ('MASINI_ATU',
'ANGAJATI_FIRMA_ATU', 'LOCATII_ATU',
        'CLIENTI_ATU', 'ASIGURARI_ATU', 'PROPRIETARI_ATU',
'SERVICEURI_AUTORIZATE_ATU',
        'ACCIDENTE_ATU', 'CERERI_DE_INCHIRIERE_ATU')
    then
        dbms_output.put_line('Nu puteti sterge acest tabel');
        RAISE_APPLICATION_ERROR(-20000,'Nu puteti sterge acest
tabel');
    end if;
end;
/
```


```
drop table cereri_de_inchiriere_atu;
```


Script Output
Query Result
Task completed in 0.216 seconds
Error starting at line : 16 in command -
drop table cereri_de_inchiriere_atu
Error report -
ORA-00604: error occurred at recursive SQL level 1
ORA-20000: Nu puteti sterge acest tabel
ORA-06512: at line 8
00604. 00000 - "error occurred at recursive SQL level %s"
*Cause: An error occurred while processing a recursive SQL statement
(a statement applying to internal dictionary tables).
*Action: If the situation described in the next error on the stack
can be corrected, do so; otherwise contact Oracle Support.


Dbms Output
Buffer Size: 20000
LAB SGBD
CERERI_DE_INCHIRIERE_ATU
Nu puteti sterge acest tabel

13. Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului

```
CREATE OR REPLACE PACKAGE pachet_atu AS
    procedure pct6 (val_data date);
    procedure pct7;
    function pct8 (cod_inchiriere
cereri_de_inchiriere_atu.cerere_id%type)
           RETURN VARCHAR2;
     procedure pct9(param_marca masini_atu.marca%type, param_model
masini_atu.model%type);
     function count_masini_atu
           return number;
end pachet_atu;



CREATE OR REPLACE PACKAGE BODY pachet_atu AS
    procedure pct6 (val_data date)
    is
        TYPE record_6 IS RECORD
             (nume angajati_firma_atu.nume%TYPE,
             prenume angajati_firma_atu.prenume%TYPE,
             data_nasterii angajati_firma_atu.data_nasterii%TYPE);
        angajat record_6;



        TYPE nested_table is TABLE OF record_6;
        tablou_angajati nested_table := nested_table();


        cursor c is
```

```
            select nume, prenume, data_nasterii
            from angajati_firma_atu
            where data_nasterii > val_data;
    BEGIN
        open c;
        loop
            fetch c into angajat.nume, angajat.prenume,
angajat.data_nasterii;
            exit when c%notfound;
            tablou_angajati.extend;
            tablou_angajati(tablou_angajati.LAST) := angajat;
        end loop;
        close c;

        dbms_output.put_line('Angajatii nascuti dupa ' || val_data);
        for elem in 1 .. tablou_angajati.count loop
            dbms_output.put_line(elem || ': ' ||
tablou_angajati(elem).nume);
        end loop;

    END pct6;



    procedure pct7
    is
        cursor c is
            select cerere_id, data_predarii
            from cereri_de_inchiriere_atu;
        temp_cod cereri_de_inchiriere_atu.cerere_id%type;
```

```
        temp_data cereri_de_inchiriere_atu.data_predarii%type;
    begin
        open c;
        loop
            fetch c into temp_cod, temp_data;
            exit when c%notfound;
            if temp_data > sysdate then
                update cereri_de_inchiriere_atu
                set data_predarii = sysdate
                where cerere_id = temp_cod;
            end if;
        end loop;

        close c;
    end pct7;



    function pct8 (cod_inchiriere
cereri_de_inchiriere_atu.cerere_id%type)
    RETURN VARCHAR2
    IS
        temp_client_id cereri_de_inchiriere_atu.client_id%type;
        temp_angajat_id cereri_de_inchiriere_atu.angajat_id%type;

        temp_client_nume clienti_atu.nume%type;

        cursor c_angajati (nume_client_var clienti_atu.nume%type)
        is
```

```
        select prenume
        from angajati_firma_atu
        where nume = nume_client_var;

    return_prenume angajati_firma_atu.prenume%type;

    cursor c_gasirecerere
    is
        select data_plasarii
        from cereri_de_inchiriere_atu
        where cerere_id = cod_inchiriere;

    gasirecerere cereri_de_inchiriere_atu.data_plasarii%type;

    nu_exista_cererea EXCEPTION;
    PRAGMA EXCEPTION_INIT (nu_exista_cererea, -20009);

    nu_exista_angajat EXCEPTION;
    PRAGMA EXCEPTION_INIT (nu_exista_angajat, -20010);
BEGIN
    open c_gasirecerere;
    loop
        fetch c_gasirecerere into gasirecerere;
        if gasirecerere is null then
            raise nu_exista_cererea;
        end if;
        exit when c_gasirecerere%notfound;
    end loop;
```

```
close c_gasirecerere;


--obtinem id_client din cererea aia
SELECT client_id
into temp_client_id
from cereri_de_inchiriere_atu
where cerere_id = cod_inchiriere;


--obtinem numele clientului in functie de id
select nume
into temp_client_nume
from clienti_atu
where client_id = temp_client_id;

--cautam angajatul cu acel nume si returnam prenumele sau
open c_angajati(temp_client_nume);

fetch c_angajati into return_prenume;
if return_prenume is not null then
    return return_prenume;
else
    raise nu_exista_angajat;
end if;

close c_angajati;
return return_prenume;
```

```
    EXCEPTION
        WHEN nu_exista_cererea then
            dbms_output.put_line('Cererea de inchiriere data ca
parametru nu exista');
            return '-';
         WHEN nu_exista_angajat then
            dbms_output.put_line('Numele clientului nu coincide cu
niciun angajat');
            return '-';
        WHEN OTHERS THEN
            dbms_output.put_line('alta eroare');
            return '-';
    END pct8;



    procedure pct9(
        param_marca masini_atu.marca%type,
        param_model masini_atu.model%type
        )
    is
        var_accident_id accidente_atu.accident_id%type;
        var_oras_service serviceuri_autorizate_atu.oras%type;
        var_oras_locatie locatii_atu.oras%type;
        var_cost_reparatie accidente_atu.cost_reparatie%type;
        var_pret_asigurare asigurari_atu.pret_asigurare%type;
        var_marca masini_atu.marca%type;
        var_model masini_atu.model%type;
```

```
    cursor c is
        select
            acc.accident_id "ID accident",
            ser.oras "Oras service",
            loc.oras "Oras locatie",
            acc.cost_reparatie "Cost reparatie",
            asi.pret_asigurare "Pret asigurare",
            mas.marca "Marca masinii",
            mas.model "Modelul masinii"
        from
            serviceuri_autorizate_atu ser
            join accidente_atu acc
                on acc.service_id = ser.service_id
            join masini_atu mas
                on mas.masina_id = acc.masina_id
            join asigurari_atu asi
                on asi.masina_id = mas.masina_id
            join locatii_atu loc
                on loc.locatie_id = mas.locatie_id;

    nu_sunt_masini EXCEPTION;
    nu_sunt_accidente EXCEPTION;
    nu_exista_masina EXCEPTION;
    i integer := 0;
    nr integer := 0;
begin
    select count(*)
    into nr
```

```
        from masini_atu
        where model = param_model and marca = param_marca;

        if nr = 0 then
            raise nu_exista_masina;
        end if;
        open c;
        loop
            fetch c into
                var_accident_id,
                var_oras_service,
                var_oras_locatie,
                var_cost_reparatie,
                var_pret_asigurare,
                var_marca,
                var_model;
            exit when c%notfound;
            if
                var_marca = param_marca
                and var_model = param_model
                and var_oras_service = var_oras_locatie
                and var_cost_reparatie <= var_pret_asigurare then
                dbms_output.put_line('Masina implicata in accident '
|| var_marca || ' ' || var_model);
                dbms_output.put_line('  se afla in orasul ' ||
var_oras_locatie);
                dbms_output.put_line('  si a fost reparata tot in ' ||
var_oras_service);
```

```
                dbms_output.put_line('Costul reparatiei (' ||
var_cost_reparatie || ')' ||
                ' nu a depasit pretul asigurarii (' ||
var_pret_asigurare || ').');
                dbms_output.new_line();
                i := i+1;
            end if;
        end loop;
        if c%rowcount = 0 then
            raise nu_sunt_masini;
        elsif i = 0 then
            raise nu_sunt_accidente;
        end if;
        close c;
    EXCEPTION
        when nu_sunt_masini then
            dbms_output.put_line('Nu exista masini in baza de date');
        when nu_sunt_accidente then
            dbms_output.put_line('Nu exista astfel de accidente');
        when nu_exista_masina then
            dbms_output.put_line('Nu exista masina cautata');
            dbms_output.new_line();
        when others then
            dbms_output.put_line('Ati apelat gresit cererea');
            dbms_output.new_line();
    end pct9;



    function count_masini_atu
```

```
        return number
    is
        numar integer;
    begin
        select count(*)
        into numar
        from masini_atu;
        return numar;
    end count_masini_atu;

end pachet_atu;
/
```


```
declare
    rezultat VARCHAR(50);
begin
    pachet_atu.pct6(to_date('1990-01-01', 'yyyy-mm-dd'));

    pachet_atu.pct7();

    rezultat := pachet_atu.pct8(4);
    dbms_output.put_line('Prenumele este: ' || rezultat);
    dbms_output.put_line(' ');
    rezultat := pachet_atu.pct8(2);
    dbms_output.put_line('Prenumele este: ' || rezultat);
```

```
    dbms_output.put_line(' ');
    rezultat := pachet_atu.pct8(25);
    dbms_output.put_line('Prenumele este: ' || rezultat);

    pct9('Dacia', 'Logan'); --nu exista in baza de date
    pct9('Land Rover', 'Range Rover'); --exista dar nu sunt accidente
    pct9('BMW', 'Seria 3'); --aici afiseaza
EXCEPTION
    when OTHERS THEN
        dbms_output.put_line('Ati apelat gresit cererea');
end;
/
```