**Практическа работа БД**

На Стефан Христов Димитров

Трети Курс, ФИТА

КСТ 1Б

19621416



**Задание:**

Да се проектриа и реализира база от данни за **КОЛИ ПОД НАЕМ**, която да съхранява следната информация:

* Автомобил - вид, марка, модел, година, цвят, изминати километри,цена за ден
* Клиент - име, адрес - град, улица, телефон
* Служител - име, позиция, телефон
* Заемане под наем - клиент, автомобил, служител, дата на заемане, брой дни

Правила:

* Всеки клиент не може да наема повече от една кола в едно и също време
* Всеки автомобил може да бъде само от една марка и един модел

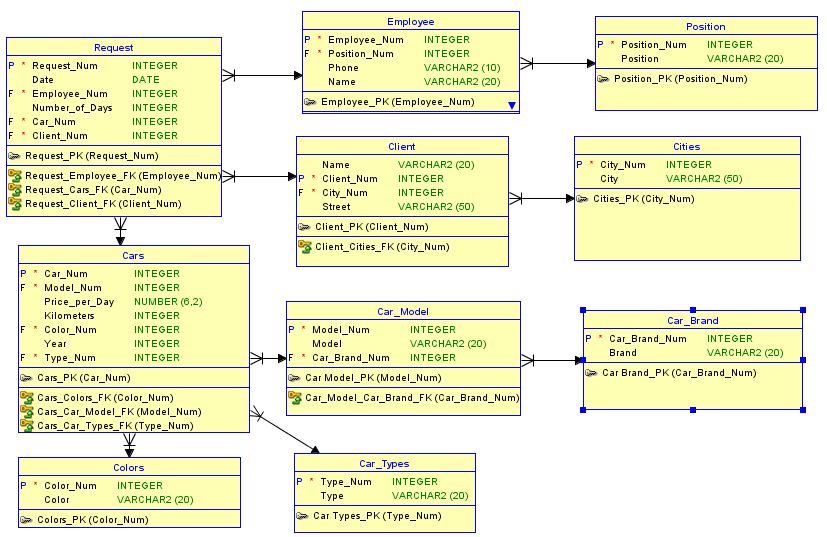
Базата данни трябва да е **нормализирана** и да позволява:

1. Въвеждане и корекция на данни
2. Търсене на автомобил по вид, марка, модел, цена за ден
3. Справки

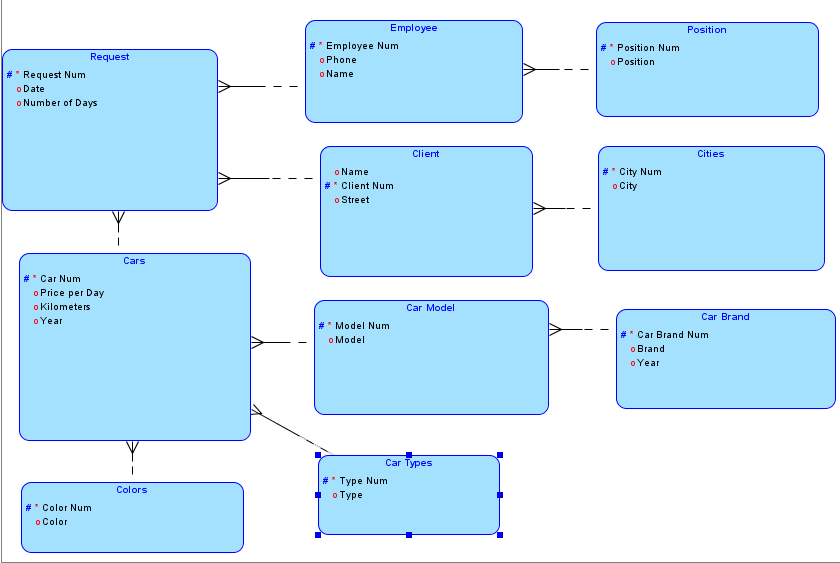
* Отдадени автомобили под наем от служител, подредени по вид и дата
* Последните 10 заемания, подредени по дни на заемане
* Наети автомобили от клиент, подредени по дата
* Отдаване на автомобили за период, подредени по клиенти, които заемат

**Модели**

Релационен:



Логически:



**Примерни данни**

Таблица **Request**

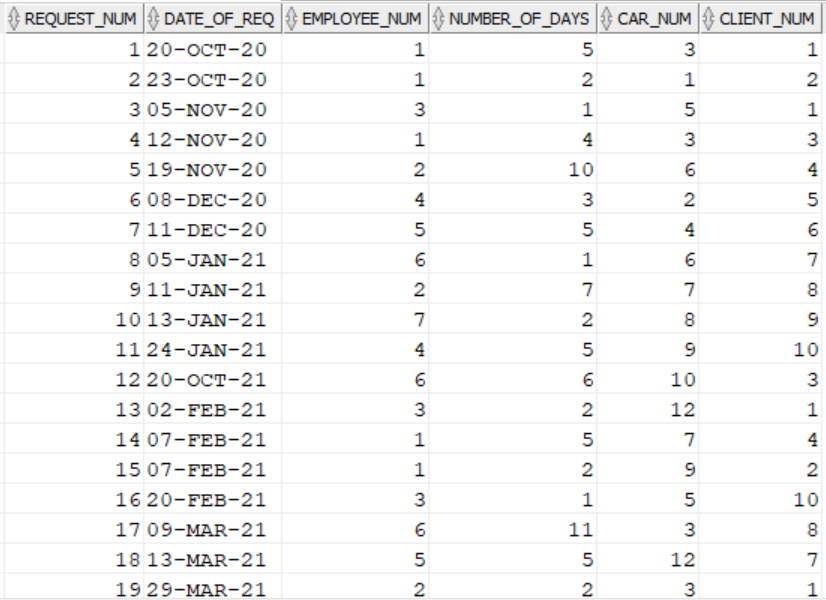


Таблица **Positions**

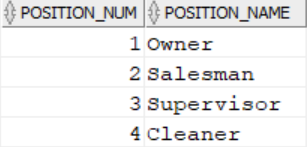


Таблица **Employee**

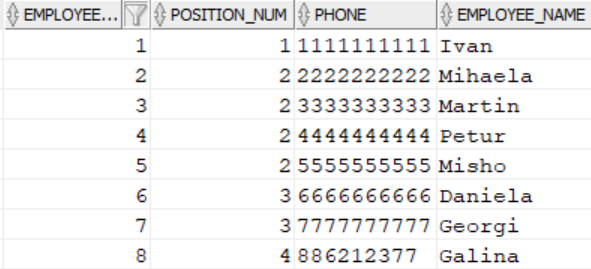


Таблица **Colors**

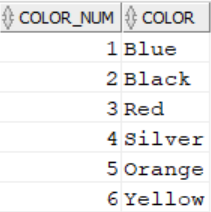


Таблица **Clients**



Таблица **Cities**

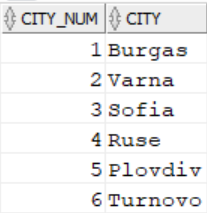


Таблица **Car\_Brand**

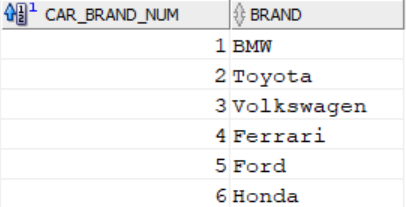


Таблица **Car\_Model**

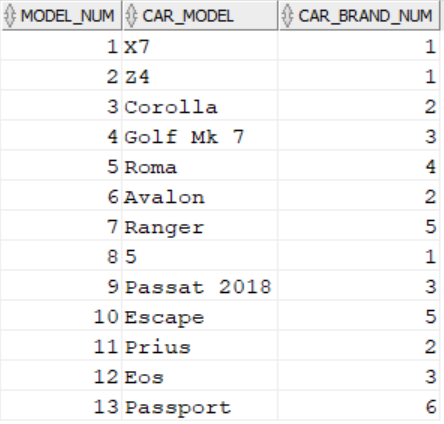


Таблица **Car\_Types**

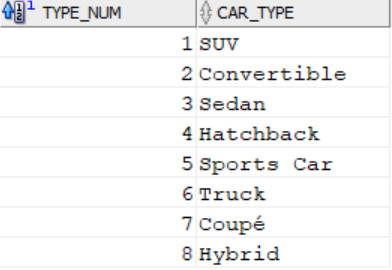
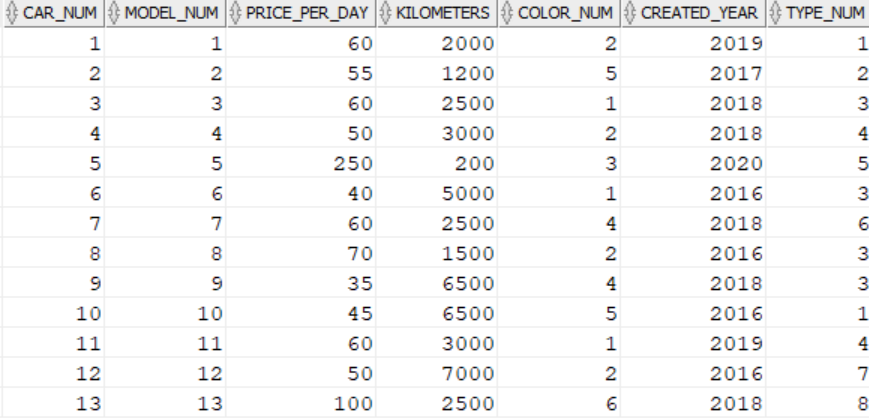


Таблица **Cars**



**SQL Команди:**

**DDL**

drop table request;

drop table employee;

drop table positions;

drop table cars;

drop table car\_model;

drop table car\_brand;

drop table colors;

drop table car\_types;

drop table clients;

drop table cities;

truncate table colors;

create table positions

(

Position\_Num int,

Position\_Name varchar2(20),

constraint Positions\_PK primary key(Position\_Num)

);

create table employee

(

Employee\_Num int,

Position\_Num int,

Phone varchar2(10),

Employee\_Name varchar2(20),

constraint Employee\_PK primary key(Employee\_Num),

constraint Positions\_Employee\_FK foreign key(Position\_Num) references positions(Position\_Num)

);

create table cities

(

City\_Num int,

City varchar2(20),

constraint Cities\_PK primary key(City\_Num)

);

create table clients

(

Client\_Num int,

Client\_Name varchar2(20),

City\_Num int,

Street varchar2(50),

constraint Clients\_PK primary key(Client\_Num),

constraint Clients\_Cities\_FK foreign key(City\_Num) references cities(City\_Num)

);

create table car\_brand

(

Car\_Brand\_Num int,

Brand varchar2(20) UNIQUE,

constraint Car\_Brand\_PK primary key(Car\_Brand\_Num)

);

create table car\_model

(

Model\_Num int,

Car\_Model varchar2(20) UNIQUE,

Car\_Brand\_Num int UNIQUE,

constraint Model\_PK primary key(Model\_Num),

constraint Model\_Brand\_FK foreign key(Car\_Brand\_Num) references car\_brand(Car\_Brand\_Num)

);

create table car\_types

(

Type\_Num int,

Car\_Type varchar2(20),

constraint Types\_PK primary key(Type\_Num)

);

create table colors

(

Color\_Num int,

Color varchar2(20),

constraint Colors\_PK primary key(Color\_Num)

);

create table cars

(

Car\_Num int,

Model\_Num int UNIQUE,

Price\_per\_Day number(6,2),

Kilometers int,

Color\_Num int,

Created\_Year int,

Type\_Num int,

constraint Cars\_PK primary key(Car\_Num),

constraint Cars\_Colors\_FK foreign key(Color\_Num) references colors(Color\_Num),

constraint Cars\_Model\_FK foreign key(Model\_Num) references car\_model(Model\_Num),

constraint Cars\_Types\_FK foreign key(Type\_Num) references car\_types(Type\_Num)

);

-- table for the Rent a car requests

create table request

(

Request\_Num int,

Date\_of\_Req date,

Employee\_Num int,

Number\_of\_Days int,

Car\_Num int,

Client\_Num int,

constraint Request\_PK primary key(Request\_Num),

constraint Request\_Employee\_FK foreign key(Employee\_Num) references employee(Employee\_Num),

constraint Request\_Client\_FK foreign key(Client\_Num) references clients(Client\_Num),

constraint Request\_Cars\_FK foreign key(Car\_Num) references cars(Car\_Num)

);

alter table colors

drop column Color;

alter table colors

add (Color varchar2(10));

alter table request

modify Date\_of\_Req date;

alter table request

rename to Rent\_A\_Car;

drop table Rent\_A\_Car

**DML**

insert into positions values (1,'Owner');

insert into positions values (2,'Salesman');

insert into positions values (3,'Supervisor');

insert into employee values (1,1,'1111111111','Ivan');

insert into employee values (2,2,'2222222222','Mihaela');

insert into employee values (3,2,'3333333333','Martin');

insert into employee values (4,2,'4444444444','Petur');

insert into employee values (5,2,'5555555555','Misho');

insert into employee values (6,3,'6666666666','Daniela');

insert into employee values (7,3,'7777777777','Georgi');

insert into cities values(1,'Burgas');

insert into cities values(2,'Varna');

insert into cities values(3,'Sofia');

insert into cities values(4,'Ruse');

insert into cities values(5,'Plovdiv');

insert into clients values(1,'Ivanka',1,'Vardar 21');

insert into clients values(2,'Mihail',3,'Kostenec 3');

insert into clients values(3,'Martina',4,'Medkovec 5');

insert into clients values(4,'Petka',1,'Stara Palnina 19');

insert into clients values(5,'Daniel',2,'Knijovnik 11');

insert into clients values(6,'Gergana',5,'Tumba 1');

insert into clients values(7,'Valq',5,'Tumba 13');

insert into clients values(8,'Mihail',2,'Pliska 27');

insert into clients values(9,'Petur',4,'Debur 14');

insert into clients values(10,'Todor',5,'Brail 24');

insert into car\_brand values(1,'BMW');

insert into car\_brand values(2,'Toyota');

insert into car\_brand values(3,'Volkswagen');

insert into car\_brand values(4,'Ferrari');

insert into car\_brand values(5,'Ford');

insert into car\_model values(1,'X7',1);

insert into car\_model values(2,'Z4',1);

insert into car\_model values(3,'Corolla',2);

insert into car\_model values(4,'Golf Mk 7',3);

insert into car\_model values(5,'Roma',4);

insert into car\_model values(6,'Avalon',2);

insert into car\_model values(7,'Ranger',5);

insert into car\_model values(8,'5',1);

insert into car\_model values(9,'Passat 2018',3);

insert into car\_model values(10,'Escape',5);

insert into car\_model values(11,'Prius',2);

insert into car\_model values(12,'Eos',3);

insert into car\_types values(1,'SUV');

insert into car\_types values(2,'Convertible');

insert into car\_types values(3,'Sedan');

insert into car\_types values(4,'Hatchback');

insert into car\_types values(5,'Sports Car');

insert into car\_types values(6,'Truck');

insert into car\_types values(7,'Coupé');

insert into colors values(1,'Blue');

insert into colors values(2,'Black');

insert into colors values(3,'Red');

insert into colors values(4,'Silver');

insert into colors values(5,'Orange');

insert into cars values(1,1,60,2000,2,2019,1);

insert into cars values(2,2,55,1200,5,2017,2);

insert into cars values(3,3,60,2500,1,2018,3);

insert into cars values(4,4,50,3000,2,2018,4);

insert into cars values(5,5,250,200,3,2020,5);

insert into cars values(6,6,40,5000,1,2016,3);

insert into cars values(7,7,60,2500,4,2018,6);

insert into cars values(8,8,70,1500,2,2016,3);

insert into cars values(9,9,55,3500,4,2018,3);

insert into cars values(10,10,45,6500,5,2016,1);

insert into cars values(11,11,60,3000,1,2019,4);

insert into cars values(12,12,50,7000,2,2016,7);

insert into request values(1,to\_date('20/10/2020','DD/MM/YYYY'),1,5,3,1);

insert into request values(2,to\_date('23/10/2020','DD/MM/YYYY'),1,2,1,2);

insert into request values(3,to\_date('5/11/2020','DD/MM/YYYY'),3,1,5,1);

insert into request values(4,to\_date('12/11/2020','DD/MM/YYYY'),1,4,3,3);

insert into request values(5,to\_date('19/11/2020','DD/MM/YYYY'),2,10,6,4);

insert into request values(6,to\_date('8/12/2020','DD/MM/YYYY'),4,3,2,5);

insert into request values(7,to\_date('11/12/2020','DD/MM/YYYY'),5,5,4,6);

insert into request values(8,to\_date('5/1/2021','DD/MM/YYYY'),6,1,6,7);

insert into request values(9,to\_date('11/01/2021','DD/MM/YYYY'),2,7,7,8);

insert into request values(10,to\_date('13/1/2021','DD/MM/YYYY'),7,2,8,9);

insert into request values(11,to\_date('24/1/2021','DD/MM/YYYY'),4,5,9,10);

insert into request values(12,to\_date('20/10/2021','DD/MM/YYYY'),6,6,10,3);

insert into request values(13,to\_date('2/2/2021','DD/MM/YYYY'),3,2,12,1);

insert into request values(14,to\_date('7/2/2021','DD/MM/YYYY'),1,5,7,4);

insert into request values(15,to\_date('7/2/2021','DD/MM/YYYY'),1,2,9,2);

insert into request values(16,to\_date('20/2/2021','DD/MM/YYYY'),3,1,5,10);

insert into request values(17,to\_date('9/3/2021','DD/MM/YYYY'),6,11,3,8);

insert into request values(18,to\_date('13/3/2021','DD/MM/YYYY'),5,5,12,7);

insert into request values(19,to\_date('29/3/2021','DD/MM/YYYY'),2,2,3,1);

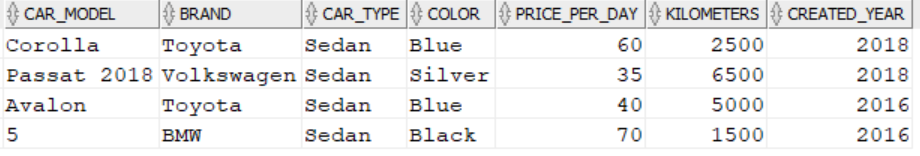
insert into request values(20,to\_date('1/4/2021','DD/MM/YYYY'),1,5,8,9);

update clients set Client\_Name = 'Patrik' where Client\_Num = 2;

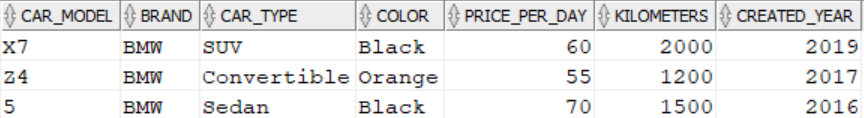
delete from employee where Employee\_Name = 'Galina';

**Резултати от изпълнението на заявките**

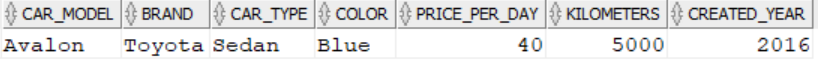
**Търсене на автомобил по тип (Sedan)**



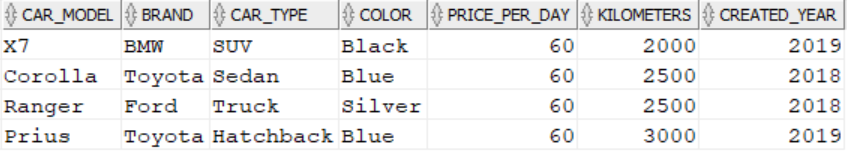
**Търсене на автомобил по марка(BMW)**



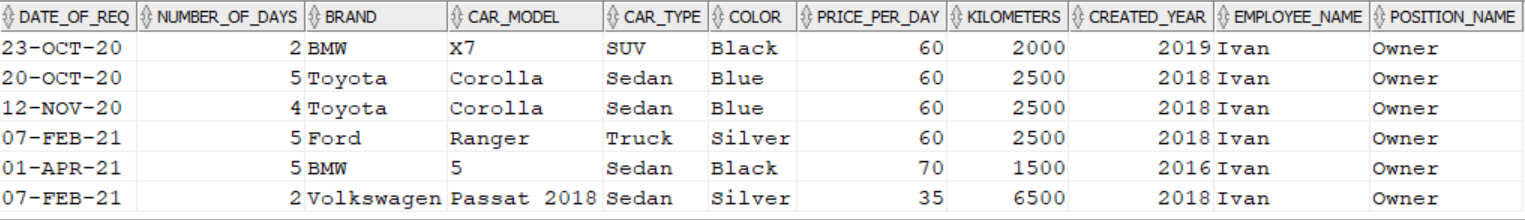
**Търсене на автомобил по модел(Avalon)**



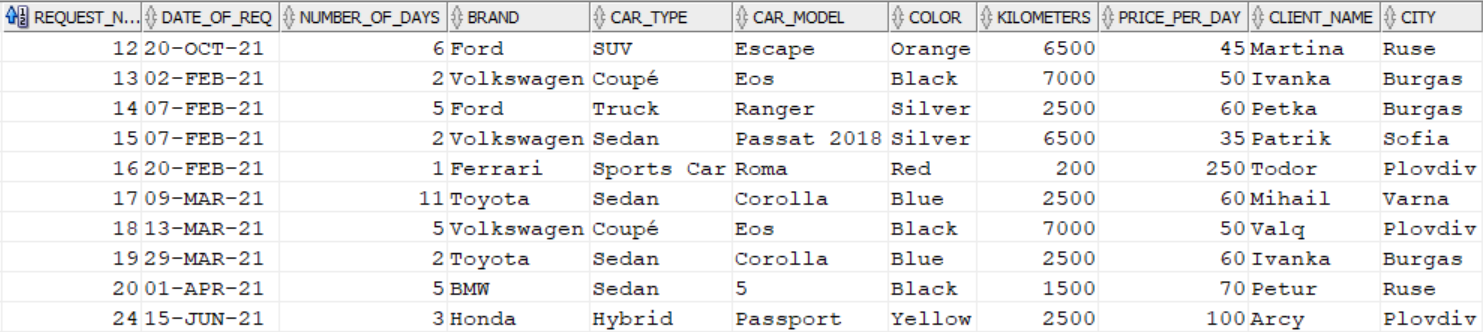
**Tърсене на автомобил по цена за ден(60)**



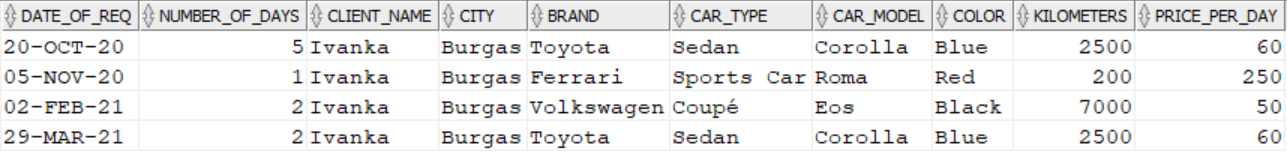
**Справка за отдадени автомобили под наем от служител(1)**



**Справка за последните 10 заемания**

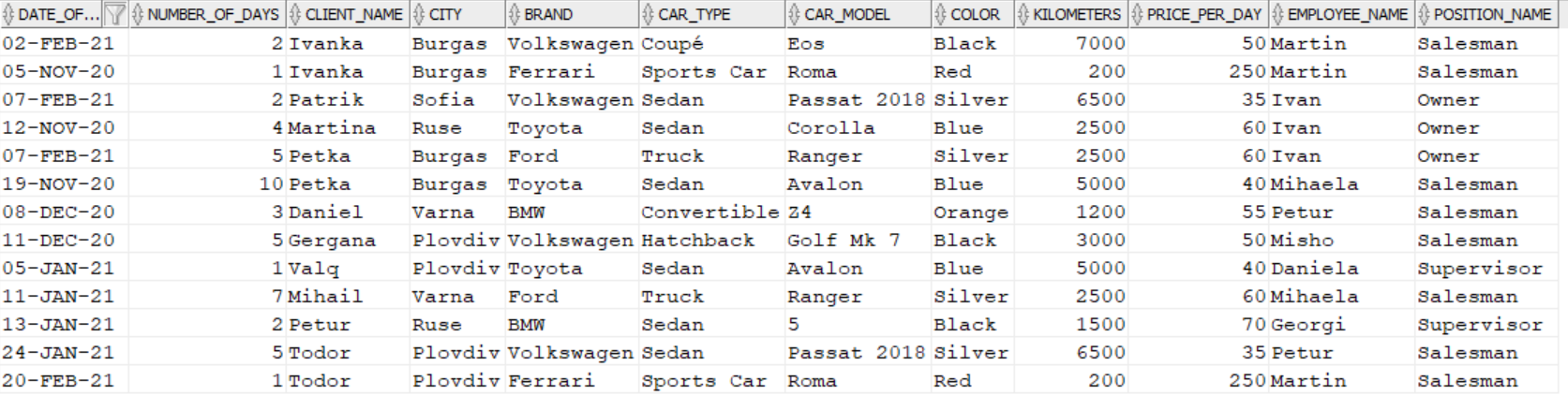


**Справка за наети автомобили от клиент(1)**



**Спрвака за отдаване на автомобили за период**

**(5-NOV-2020 – 04-MAR-2021)**



**PL/SQL**

**Процедури**

**brand\_inp**

create or replace procedure brand\_inp(v\_brand Car\_Brand.Brand%TYPE)

AS

BEGIN

insert into Car\_Brand(Brand)

Values(v\_brand);

END;

**brand\_upd**

create or replace procedure brand\_upd(v\_brand\_num Car\_Brand.Car\_Brand\_Num%TYPE,v\_brand Car\_Brand.Brand%TYPE)

AS

Begin

Update Car\_Brand set Brand = v\_brand

where Car\_Brand\_Num = v\_brand\_num;

END;

**brand\_del**

create or replace procedure brand\_del (v\_brand\_num Car\_Brand.Car\_Brand\_Num%TYPE)

as

BEGIN

delete from Car\_Brand where Car\_Brand\_Num = v\_brand\_num;

end;

**car\_inp**

create or replace procedure car\_inp(v\_model\_num Cars.Model\_Num%TYPE, v\_price Cars.Price\_per\_Day%TYPE,

v\_kilometers Cars.Kilometers%TYPE, v\_color\_num Cars.Color\_Num%TYPE, v\_year Cars.Created\_Year%TYPE,

v\_type\_num Cars.Type\_Num%TYPE)

AS

BEGIN

insert into Cars(Model\_Num, Price\_per\_Day, Kilometers, Color\_Num, Created\_Year, Type\_Num)

Values(v\_model\_num,v\_price,v\_kilometers,v\_color\_num,v\_year,v\_type\_num);

END;

**car\_upd**

create or replace procedure car\_upd(v\_car\_num Cars.Car\_Num%TYPE,v\_model\_num Cars.Model\_Num%TYPE, v\_price Cars.Price\_per\_Day%TYPE,

v\_kilometers Cars.Kilometers%TYPE, v\_color\_num Cars.Color\_Num%TYPE, v\_year Cars.Created\_Year%TYPE, v\_type\_num Cars.Type\_Num%TYPE)

AS

Begin

Update Cars set Model\_Num = v\_model\_num,Price\_per\_Day = v\_price, Kilometers = v\_kilometers, Color\_Num = v\_color\_num,

Created\_Year = v\_year, Type\_Num = v\_type\_num

where Car\_Num = v\_car\_num;

END;

**car\_del**

create or replace procedure car\_del (v\_car\_num Cars.Car\_Num%TYPE)

as

BEGIN

delete from Cars where Car\_Num = v\_car\_num;

end;

**city\_inp**

create or replace procedure city\_inp(v\_city\_name cities.City%TYPE)

AS

BEGIN

insert into cities(City)

Values(v\_city\_name);

END;

**city\_upd**

create or replace procedure city\_upd(v\_city\_num cities.City\_Num%TYPE,v\_city\_name cities.City%TYPE)

AS

Begin

Update cities set City = v\_city\_name

where City\_Num = v\_city\_num;

END;

**city\_del**

create or replace procedure city\_del (v\_city\_num Cities.City\_Num%TYPE)

as

BEGIN

delete from Cities where City\_Num = v\_city\_num;

end;

**client\_inp**

create or replace procedure client\_inp(v\_client\_name Clients.Client\_Name%TYPE,

v\_city\_num Clients.City\_Num%TYPE, v\_street Clients.Street%TYPE)

AS

BEGIN

insert into Clients(Client\_Name,City\_Num,Street)

Values(v\_client\_name,v\_city\_num,v\_street);

END;

**client\_upd**

create or replace procedure client\_upd(v\_client\_num Clients.Client\_Num%TYPE,v\_client\_name Clients.Client\_Name%TYPE,

v\_city\_num Clients.City\_Num%TYPE, v\_street Clients.Street%TYPE)

AS

Begin

Update Clients set Client\_Name = v\_client\_name,

City\_Num = v\_city\_num,

Street = v\_street

where Client\_Num = v\_client\_num;

END;

**client\_del**

create or replace procedure client\_del (v\_client\_num Clients.Client\_Num%TYPE)

as

BEGIN

delete from Clients where Client\_Num = v\_client\_num;

end;

**color\_inp**

create or replace procedure color\_inp(v\_color Colors.Color%TYPE)

AS

BEGIN

insert into Colors(Color)

Values(v\_color);

END;

**color\_upd**

create or replace procedure color\_upd(v\_color\_num Colors.Color\_Num%TYPE,v\_color Colors.Color%TYPE)

AS

Begin

Update Colors set Color = v\_color

where Color\_Num = v\_color\_num;

END;

**color\_del**

create or replace procedure color\_del (v\_color\_num Colors.Color\_Num%TYPE)

as

BEGIN

delete from Colors where Color\_Num = v\_color\_num;

end;

**employee\_inp**

create or replace procedure employee\_inp(v\_position Employee.Position\_Num%TYPE,

v\_phone Employee.Phone%TYPE, v\_name Employee.Employee\_Name%TYPE)

AS

BEGIN

insert into Employee(Position\_Num, Phone, Employee\_Name)

Values(v\_position,v\_phone,v\_name);

END;

**employee\_upd**

create or replace procedure employee\_upd(v\_emplyoee\_num Employee.Employee\_Num%TYPE,v\_position Employee.Position\_Num%TYPE,

v\_phone Employee.Phone%TYPE, v\_name Employee.Employee\_Name%TYPE)

AS

Begin

Update Employee set Position\_Num = v\_position,

Phone = v\_phone, Employee\_Name = v\_name

where Employee\_Num = v\_emplyoee\_num;

END;

**employee\_del**

create or replace procedure employee\_del (v\_employee\_num Employee.Employee\_Num%TYPE)

as

BEGIN

delete from Employee where Employee\_Num = v\_employee\_num;

end;

**model\_inp**

create or replace procedure model\_inp(v\_model Car\_Model.Car\_Model%TYPE,

v\_brand Car\_Model.Car\_Brand\_Num%TYPE)

AS

BEGIN

insert into Car\_Model(Car\_Model, Car\_Brand\_Num)

Values(v\_model,v\_brand);

END;

**model\_upd**

create or replace procedure model\_upd(v\_model\_num Car\_Model.Model\_Num%TYPE,v\_model Car\_Model.Car\_Model%TYPE,

v\_brand Car\_Model.Car\_Brand\_Num%TYPE)

AS

Begin

Update Car\_Model set Car\_Model = v\_model,

Car\_Brand\_Num = v\_brand

where Model\_Num = v\_model\_num;

END;

**model\_del**

create or replace procedure model\_del (v\_model\_num Car\_Model.Model\_Num%TYPE)

as

BEGIN

delete from Car\_Model where Model\_Num = v\_model\_num;

end;

**position\_inp**

create or replace procedure position\_inp(v\_position Positions.Position\_Name%TYPE)

AS

BEGIN

insert into Positions(Position\_Name)

Values(v\_position);

END;

**position\_upd**

create or replace procedure position\_upd(v\_position\_num Positions.Position\_Num%TYPE,v\_position Positions.Position\_Name%TYPE)

AS

Begin

Update Positions set Position\_Name = v\_position

where Position\_Num = v\_position\_num;

END;

**position\_del**

create or replace procedure position\_del (v\_position\_num Positions.Position\_Num%TYPE)

as

BEGIN

delete from Positions where Position\_Num = v\_position\_num;

end;

**request\_inp**

create or replace procedure request\_inp(v\_date Request.Date\_of\_Req%TYPE,

v\_employee Request.Employee\_Num%TYPE,v\_days Request.Number\_of\_Days%TYPE, v\_car Request.Car\_Num%TYPE,

v\_client Request.Client\_Num%TYPE)

AS

BEGIN

insert into Request(Date\_of\_Req, Employee\_Num, Number\_of\_Days, Car\_Num, Client\_Num)

Values(v\_date,v\_employee,v\_days,v\_car,v\_client);

END;

**request\_upd**

create or replace procedure request\_upd(v\_request\_num Request.Request\_Num%TYPE,v\_date Request.Date\_of\_Req%TYPE,

v\_employee Request.Employee\_Num%TYPE,v\_days Request.Number\_of\_Days%TYPE, v\_car Request.Car\_Num%TYPE,

v\_client Request.Client\_Num%TYPE)

AS

Begin

Update Request set Date\_of\_Req = v\_date,Employee\_Num = v\_employee, Number\_of\_Days = v\_days,

Car\_Num = v\_car, Client\_Num = v\_client

where Request\_Num = v\_request\_num;

END;

**request\_del**

create or replace procedure request\_del (v\_request\_num Request.Request\_Num%TYPE)

as

BEGIN

delete from Request where Request\_Num = v\_request\_num;

end;

**type\_inp**

create or replace procedure type\_inp(v\_type Car\_Types.Car\_Type%TYPE)

AS

BEGIN

insert into Car\_Types(Car\_Type)

Values(v\_type);

END;

**type\_upd**

create or replace procedure type\_upd(v\_type\_num Car\_Types.Type\_Num%TYPE,v\_type Car\_Types.Car\_Type%TYPE)

AS

Begin

Update Car\_Types set Car\_Type = v\_type

where Type\_Num = v\_type\_num;

END;

**type\_del**

create or replace procedure type\_del (v\_type\_num Car\_Types.Type\_Num%TYPE)

as

BEGIN

delete from Car\_Types where Type\_Num = v\_type\_num;

end;

**Тригери**

**brand\_tr**

create or replace TRIGGER brand\_tr

BEFORE INSERT ON Car\_Brand

FOR EACH ROW

WHEN (NEW.Car\_Brand\_Num IS NULL)

BEGIN

:NEW.Car\_Brand\_Num := Brand\_Seq.NEXTVAL;

END;

**car\_tr**

create or replace TRIGGER car\_tr

BEFORE INSERT ON Cars

FOR EACH ROW

WHEN (NEW.Car\_Num IS NULL)

BEGIN

:NEW.Car\_Num := Car\_Seq.NEXTVAL;

END;

**city\_tr**

create or replace TRIGGER city\_tr

BEFORE INSERT ON Cities

FOR EACH ROW

WHEN (NEW.City\_Num IS NULL)

BEGIN

:NEW.City\_Num := City\_Seq.NEXTVAL;

END;

**client\_tr**

create or replace TRIGGER client\_tr

BEFORE INSERT ON Clients

FOR EACH ROW

WHEN (NEW.Client\_Num IS NULL)

BEGIN

:NEW.Client\_Num := Client\_Seq.NEXTVAL;

END;

**color\_tr**

create or replace TRIGGER color\_tr

BEFORE INSERT ON Colors

FOR EACH ROW

WHEN (NEW.Color\_Num IS NULL)

BEGIN

:NEW.Color\_Num := Color\_Seq.NEXTVAL;

END;

**employee\_tr**

create or replace TRIGGER employee\_tr

BEFORE INSERT ON Employee

FOR EACH ROW

WHEN (NEW.Employee\_Num IS NULL)

BEGIN

:NEW.Employee\_Num := Employee\_Seq.NEXTVAL;

END;

**model\_tr**

create or replace TRIGGER model\_tr

BEFORE INSERT ON Car\_Model

FOR EACH ROW

WHEN (NEW.Model\_Num IS NULL)

BEGIN

:NEW.Model\_Num := Model\_Seq.NEXTVAL;

END;

**position\_tr**

create or replace TRIGGER position\_tr

BEFORE INSERT ON Positions

FOR EACH ROW

WHEN (NEW.Position\_Num IS NULL)

BEGIN

:NEW.Position\_Num := Position\_Seq.NEXTVAL;

END;

**request\_tr**

create or replace TRIGGER request\_tr

BEFORE INSERT ON Request

FOR EACH ROW

WHEN (NEW.Request\_Num IS NULL)

BEGIN

:NEW.Request\_Num := Req\_Seq.NEXTVAL;

END;

**type\_tr**

create or replace TRIGGER type\_tr

BEFORE INSERT ON Car\_Types

FOR EACH ROW

WHEN (NEW.Type\_Num IS NULL)

BEGIN

:NEW.Type\_Num := Type\_Seq.NEXTVAL;

END;

**Курсори**

**search\_car\_by\_type**

create or replace Procedure search\_car\_by\_type(v\_type Car\_Types.Car\_Type%TYPE)

IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Car Type:');

DECLARE

CURSOR Car\_TYP IS

select m.Car\_Model,b.Brand,t.Car\_Type,color.Color,c.Price\_per\_Day,c.Kilometers,

c.Created\_Year

from Cars c

join Car\_Types t

on c.Type\_Num = t.Type\_Num

join Car\_Model m

on c.Model\_Num = m.Model\_Num

join Car\_Brand b

on m.Car\_Brand\_Num = b.Car\_Brand\_Num

join Colors color

on c.Color\_Num = color.Color\_Num

join Car\_Types t

on c.Type\_Num = t.Type\_Num

where t.Car\_Type = v\_type;

BEGIN

FOR Car\_Types in Car\_TYP

LOOP

DBMS\_OUTPUT.PUT\_LINE(Car\_Types.Car\_Model || ' ' || Car\_Types.Brand || ' ' ||Car\_Types.Car\_Type

|| ' ' ||Car\_Types.Color || ' ' || Car\_Types.Price\_per\_Day || ' ' || Car\_Types.Kilometers|| ' ' ||

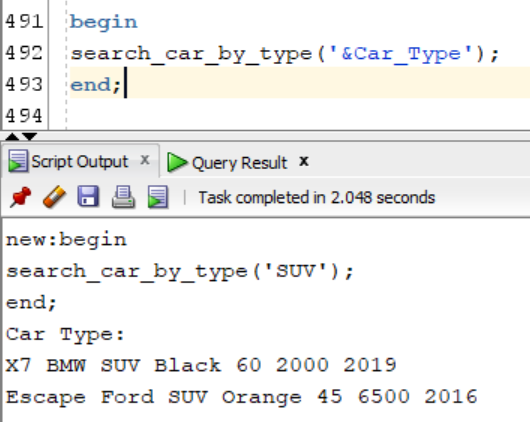
Car\_Types.Created\_Year);

end loop;

end;

end;

**При въвеждане на SUV**



**Search\_car\_by\_brand**

create or replace Procedure search\_car\_by\_brand(v\_brand Car\_Brand.Brand%TYPE)

IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Car Brand:');

DECLARE

CURSOR Car\_BRND IS

select m.Car\_Model,b.Brand,t.Car\_Type,color.Color,c.Price\_per\_Day,c.Kilometers,

c.Created\_Year

from Cars c

join Car\_Types t

on c.Type\_Num = t.Type\_Num

join Car\_Model m

on c.Model\_Num = m.Model\_Num

join Car\_Brand b

on m.Car\_Brand\_Num = b.Car\_Brand\_Num

join Colors color

on c.Color\_Num = color.Color\_Num

join Car\_Types t

on c.Type\_Num = t.Type\_Num

where b.Brand = v\_brand;

BEGIN

FOR Car\_Brand in Car\_BRND

LOOP

DBMS\_OUTPUT.PUT\_LINE(Car\_Brand.Car\_Model || ' ' || Car\_Brand.Brand || ' ' ||Car\_Brand.Car\_Type

|| ' ' ||Car\_Brand.Color || ' ' || Car\_Brand.Price\_per\_Day || ' ' || Car\_Brand.Kilometers|| ' ' ||

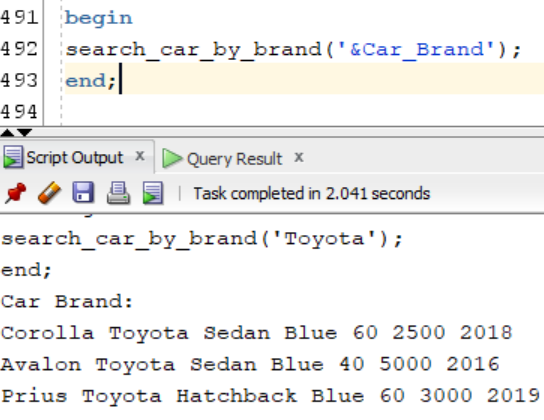
Car\_Brand.Created\_Year);

end loop;

end;

end;

**При въвеждане на Toyota**



**search\_car\_by\_model**

create or replace Procedure search\_car\_by\_model(v\_model Car\_Model.Car\_Model%TYPE)

IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Car Model:');

DECLARE

CURSOR Car\_MDL IS

select m.Car\_Model,b.Brand,t.Car\_Type,color.Color,c.Price\_per\_Day,c.Kilometers,

c.Created\_Year

from Cars c

join Car\_Types t

on c.Type\_Num = t.Type\_Num

join Car\_Model m

on c.Model\_Num = m.Model\_Num

join Car\_Brand b

on m.Car\_Brand\_Num = b.Car\_Brand\_Num

join Colors color

on c.Color\_Num = color.Color\_Num

join Car\_Types t

on c.Type\_Num = t.Type\_Num

where m.Car\_Model = v\_model;

BEGIN

FOR Car\_Model in Car\_MDL

LOOP

DBMS\_OUTPUT.PUT\_LINE(Car\_Model.Car\_Model || ' ' || Car\_Model.Brand || ' ' ||Car\_Model.Car\_Type

|| ' ' ||Car\_Model.Color || ' ' || Car\_Model.Price\_per\_Day || ' ' || Car\_Model.Kilometers|| ' ' ||

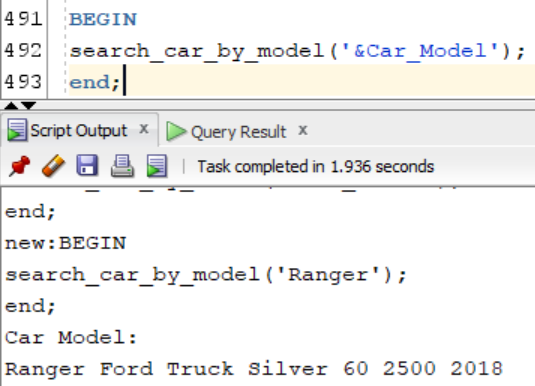
Car\_Model.Created\_Year);

end loop;

end;

end;

**При въвеждане на Ranger**



**search\_car\_by\_price**

create or replace Procedure search\_car\_by\_price(v\_price Cars.Price\_per\_Day%TYPE)

IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Car Price:');

DECLARE

CURSOR Car\_PRC IS

select m.Car\_Model,b.Brand,t.Car\_Type,color.Color,c.Price\_per\_Day,c.Kilometers,

c.Created\_Year

from Cars c

join Car\_Types t

on c.Type\_Num = t.Type\_Num

join Car\_Model m

on c.Model\_Num = m.Model\_Num

join Car\_Brand b

on m.Car\_Brand\_Num = b.Car\_Brand\_Num

join Colors color

on c.Color\_Num = color.Color\_Num

join Car\_Types t

on c.Type\_Num = t.Type\_Num

where c.Price\_per\_Day = v\_price;

BEGIN

FOR Price\_Per\_Day in Car\_PRC

LOOP

DBMS\_OUTPUT.PUT\_LINE(Price\_Per\_Day.Car\_Model || ' ' || Price\_Per\_Day.Brand || ' ' ||

Price\_Per\_Day.Car\_Type|| ' ' ||Price\_Per\_Day.Color || ' ' || Price\_Per\_Day.Price\_per\_Day || ' ' ||

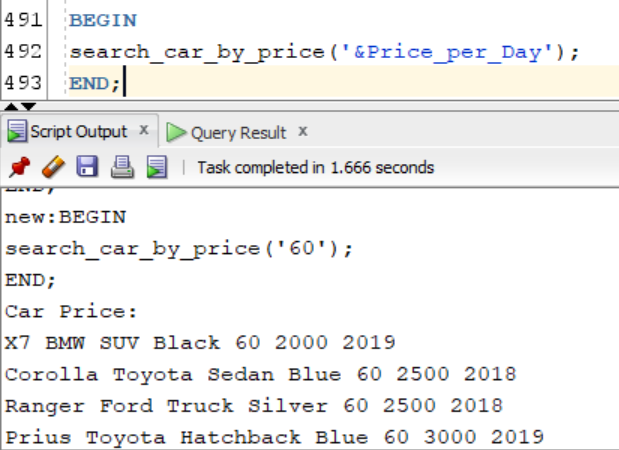
Price\_Per\_Day.Kilometers|| ' ' || Price\_Per\_Day.Created\_Year);

end loop;

end;

end;

**При въвеждане на 60**



**cars\_from\_employee**

create or replace PROCEDURE cars\_from\_employee(Employee\_Name varchar2)

IS

Employee\_Name\_new varchar2(20);

BEGIN

Employee\_Name\_new :='%'||Employee\_Name||'%';

DECLARE

CURSOR Car\_Emp IS

select req.Date\_of\_Req,req.Number\_of\_Days,b.Brand, m.Car\_Model,t.Car\_Type,color.Color, c.Price\_per\_Day,

c.Kilometers,c.Created\_Year,e.Employee\_Name,p.Position\_Name

from request req

join cars c

on req.Car\_Num = c.Car\_Num

join car\_model m

on c.Model\_Num = m.Model\_Num

join Employee e

on req.Employee\_Num = e.Employee\_Num

join Positions p

on e.Position\_Num = p.Position\_Num

join Colors color

on c.Color\_Num = color.Color\_Num

join Car\_Brand b

on m.Car\_Brand\_Num = b.Car\_Brand\_Num

join Car\_Types t

on c.Type\_Num = t.Type\_Num

where e.Employee\_Name like Employee\_Name\_new

order by m.Model\_Num, req.Date\_of\_Req;

BEGIN

FOR Cars in Car\_Emp

LOOP

DBMS\_OUTPUT.PUT\_LINE(Cars.Date\_of\_Req || ' ' || Cars.Number\_of\_Days || ' ' || Cars.Brand || ' ' ||

Cars.Car\_Model || ' ' || Cars.Car\_Type || ' ' || Cars.Color || ' ' || Cars.Price\_per\_Day || ' ' ||

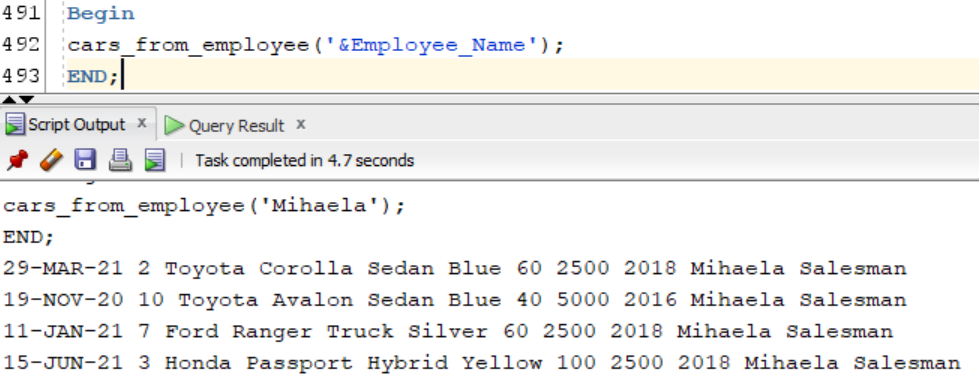
Cars.Kilometers || ' ' || Cars.Created\_Year || ' ' || Cars.Employee\_Name || ' ' || Cars.Position\_Name);

END LOOP;

END;

END;

**При въвеждане на Mihaela**



**last\_ten\_req**

create or replace PROCEDURE last\_ten\_req

IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Last 10:');

DECLARE

CURSOR Last\_Ten IS

select \* from(select req.Request\_Num,req.Date\_of\_Req ,req.Number\_of\_Days,b.Brand,

t.Car\_Type,m.Car\_Model,color.Color,c.Kilometers,c.Price\_Per\_Day,cli.Client\_Name,cit.City

from request req

join cars c

on req.Car\_Num = c.Car\_Num

join Car\_Model m

on c.Model\_Num = m.Model\_Num

join Car\_Brand b

on m.Car\_Brand\_Num = b.Car\_Brand\_Num

join Colors color

on c.Color\_Num = color.Color\_Num

join Car\_Types t

on c.Type\_Num = t.Type\_Num

join Clients cli

on req.Client\_Num = cli.Client\_Num

join Cities cit

on cli.City\_Num = cit.City\_Num

order by Date\_of\_Req desc)

where rownum <=10;

BEGIN

FOR Request in Last\_Ten

LOOP

DBMS\_OUTPUT.PUT\_LINE(Request.Request\_Num || ' ' || Request.Date\_of\_Req || ' ' || Request.Number\_of\_Days

|| ' ' || Request.Brand || ' ' || Request.Car\_Type || ' ' || Request.Car\_Model || ' ' ||Request.Color

|| ' ' || Request.Kilometers || ' ' || Request.Price\_per\_Day || ' ' || Request.Client\_Name || ' ' ||

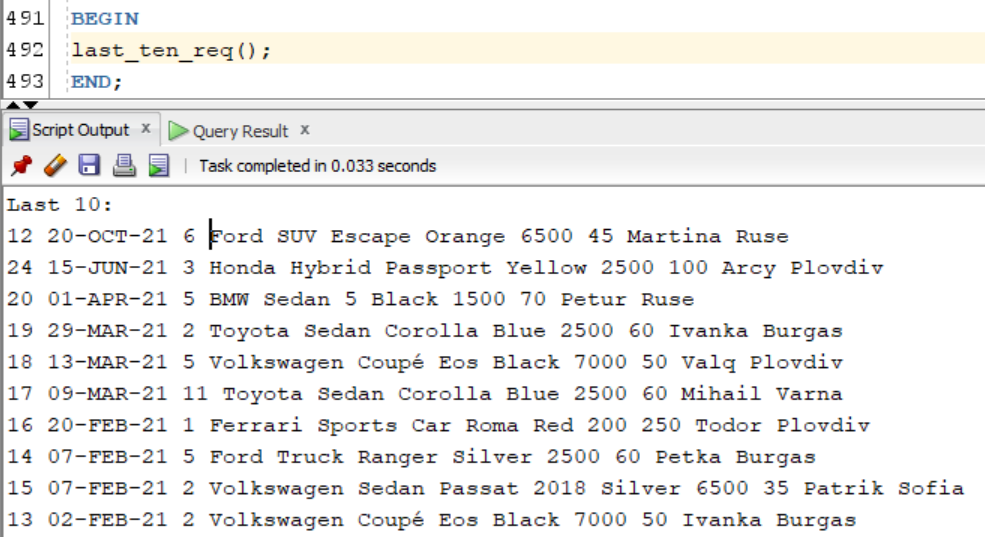
Request.City);

END LOOP;

END;

END;

**Излиза**



**cars\_from\_client**

create or replace PROCEDURE cars\_from\_client(Client\_Name varchar2)

IS

Client\_Name\_new varchar2(20);

BEGIN

Client\_Name\_new :='%'||Client\_Name||'%';

DECLARE

CURSOR Car\_Cli IS

select req.Date\_of\_Req, req.Number\_of\_Days,cli.Client\_Name,cit.City,b.Brand,

t.Car\_Type,m.Car\_Model,color.Color,c.Price\_per\_Day,c.Kilometers,c.Created\_Year

from Request req

join Clients cli

on req.Client\_Num = cli.Client\_Num

join Cities cit

on cli.City\_Num = cit.City\_Num

join cars c

on req.Car\_Num = c.Car\_Num

join Car\_Model m

on c.Model\_Num = m.Model\_Num

join Car\_Brand b

on m.Car\_Brand\_Num = b.Car\_Brand\_Num

join Colors color

on c.Color\_Num = color.Color\_Num

join Car\_Types t

on c.Type\_Num = t.Type\_Num

where cli.Client\_Name like Client\_Name\_new

order by Date\_of\_Req;

BEGIN

FOR Cars in Car\_Cli

LOOP

DBMS\_OUTPUT.PUT\_LINE(Cars.Date\_of\_Req || ' ' || Cars.Number\_of\_Days || ' ' ||Cars.Client\_Name || ' ' ||Cars.City || ' ' ||

Cars.Brand || ' ' || Cars.Car\_Type || ' ' || Cars.Car\_Model || ' ' || Cars.Color || ' ' || Cars.Price\_per\_Day || ' ' ||

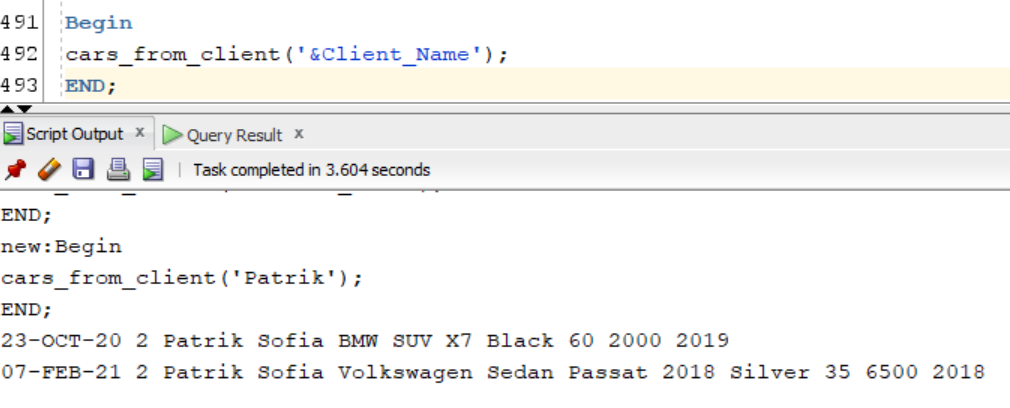
Cars.Kilometers || ' ' || Cars.Created\_Year);

END LOOP;

END;

END;

**При въвеждане на Patrik**



**cars\_between\_date**

create or replace PROCEDURE cars\_between\_date(Str\_Date Request.Date\_of\_Req%Type,End\_Date Request.Date\_of\_Req%Type)

IS

BEGIN

DECLARE

CURSOR Car\_Date IS

select req.Date\_of\_Req,req.Number\_of\_Days,cli.Client\_Name,cit.City,b.Brand,

t.Car\_Type,m.Car\_Model,color.Color,c.Kilometers,c.Price\_Per\_Day,e.Employee\_Name,p.Position\_Name

from request req

join Clients cli

on req.Client\_Num = cli.Client\_Num

join Cities cit

on cli.City\_Num = cit.City\_Num

join cars c

on req.Car\_Num = c.Car\_Num

join Car\_Model m

on c.Model\_Num = m.Model\_Num

join Car\_Brand b

on m.Car\_Brand\_Num = b.Car\_Brand\_Num

join Colors color

on c.Color\_Num = color.Color\_Num

join Car\_Types t

on c.Type\_Num = t.Type\_Num

join Employee e

on req.Employee\_Num = e.Employee\_Num

join Positions p

on e.Position\_Num = p.Position\_Num

where req.Date\_of\_Req >= Str\_Date AND req.Date\_of\_Req <= End\_Date

order by cli.Client\_Num;

BEGIN

FOR Cars in Car\_Date

LOOP

DBMS\_OUTPUT.PUT\_LINE(Cars.Date\_of\_Req || ' ' || Cars.Number\_of\_Days || ' ' ||Cars.Client\_Name || ' ' ||

Cars.City || ' ' ||Cars.Brand || ' ' || Cars.Car\_Type || ' ' || Cars.Car\_Model || ' ' || Cars.Color

|| ' ' || Cars.Kilometers || ' ' ||Cars.Price\_per\_Day || ' ' || Cars.Employee\_Name || ' ' ||

Cars.Position\_Name);

END LOOP;

END;

END;

**При въведени дати ('5-NOV-20','4-MAR-21')**

