**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №2**

**по дисциплине «Базы данных»**

Тема: Реализация базы данных в СУБД PostgreSQL

|  |  |  |
| --- | --- | --- |
| Студент гр. 1303 |  | Попандопуло А.Г. |
| Преподаватель |  | Заславский М.М. |

Санкт-Петербург,

2023

## Цель работы.

## Практическое изучение работы СУБД на примере PostgreSQL, реализация базы данных по ранее заготовленным моделям, практика в написании запросов к БД.

## Задание.

Вариант 17

Пусть требуется создать программную систему, предназначенную для диспетчера станции техобслуживания. Такая система должна обеспечивать хранение сведений об услугах, оказываемых станцией и их стоимости, о клиентах станции, о работниках станции и об автомобилях, которые они ремонтируют в текущий момент. Клиент станции – это человек, который хотя бы раз воспользовался услугами станции. О клиенте должны хранится следующие сведения: паспортные данные, включая фамилию, имя, отчество, дату рождения, прописку, а также даты обращения на станцию техобслуживания с указанием автомобилей, которые он сдавал в ремонт. Клиент сдает в ремонт необязательно автомобиль, владельцем которого он является. Сведения об автомобилях включают в себя марку автомобиля, его цвет, год выпуска, номер государственной регистрации, перечень неисправностей и данные о владельце. Сведения о работнике – это его фамилия, имя, отчество, специальность, разряд, стаж работы. Диспетчер заносит в БД сведения об автомобиле и о клиенте, если клиент обращается на станцию впервые. После этого диспетчер определяет рабочих, которые будут устранять имеющиеся в автомобиле неисправности. Оставляя автомобиль на станции техобслуживания, клиент получает расписку, в которой указано, когда автомобиль был поставлен на ремонт, какие он имеет неисправности, когда станция обязуется возвратить отремонтированный автомобиль. После возвращения автомобиля клиенту данные о произведенном ремонте помещаются в архив, клиент получает счет, в котором содержится перечень устраненных неисправностей с указанием времени работы, стоимости работы и стоимости запчастей. Возможно увольнение и прием на работу работников станции, изменение сведений о клиенте (клиент может поменять паспорт, права, адрес, телефон), номера государственной регистрации и цвета автомобиля. Диспетчеру могут потребоваться следующие сведения:

* фамилия, имя, отчество и адрес владельца автомобиля с данным номером государственной регистрации?
* Марка и год выпуска автомобиля данного владельца?
* Перечень устраненных неисправностей в автомобиле данного владельца?
* фамилия, имя, отчество работника станции, устранявшего данную неисправность в автомобиле данного клиента, и время ее устранения?
* фамилия, имя, отчество клиентов, сдавших в ремонт автомобили с указанным типом неисправности?

## Выполнение работы.

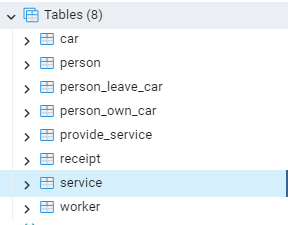
Была создана база данных service\_station с таблицами *car, person, person\_leave\_car, person\_own\_car, provide\_service, receipt, service, worker*, соответствующими разработанной ранее модели, написаны запросы на создание таблиц и вставку строк, далее приведены примеры для таблиц *car, person, worker.*

Рисунок – созданные таблицы

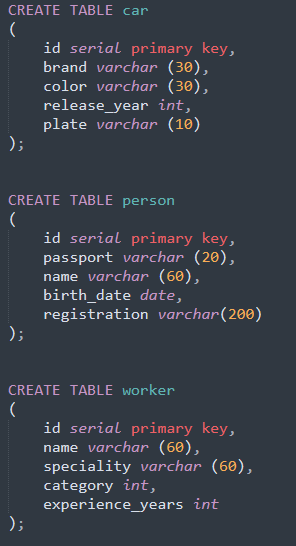
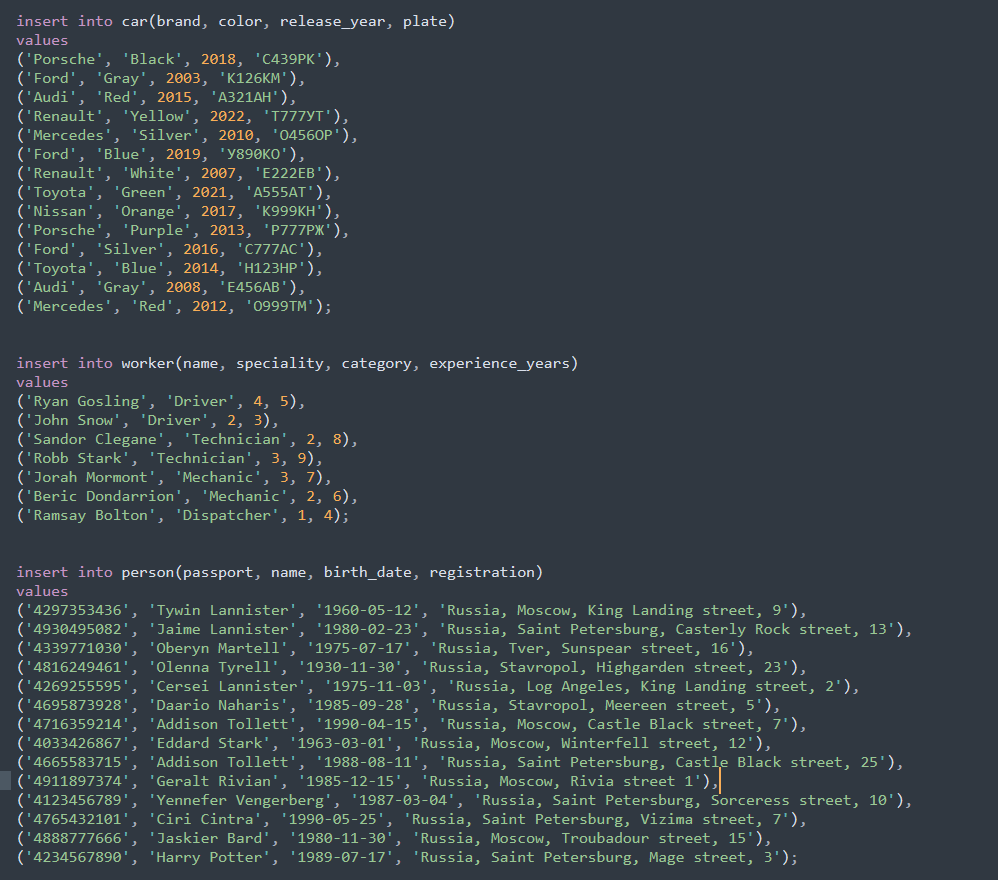


Рисунок 3 – пример заполнения таблиц

Рисунок 2 – пример создания таблиц

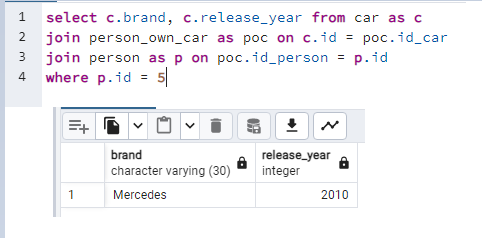
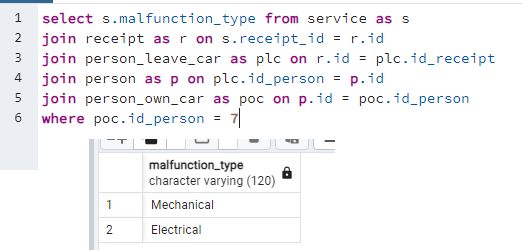
Приведем вывод указанных в условии данных, которые могут потребоваться диспетчеру станции технического обсуживания:

Рисунок 6 – типы устраненных неисправностей в автомобиле данного клиента

Рисунок 5 - марка и год выпуска автомобиля данного владельца

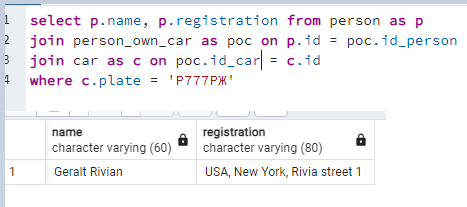


Рисунок 4 – имя и адрес владельца автомобиля с данным номером гос. регистрации

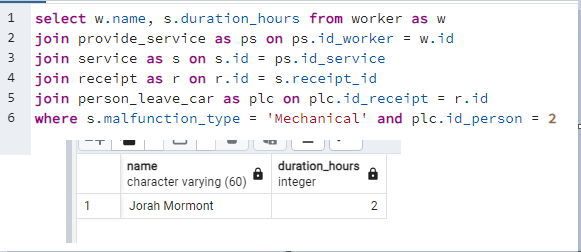
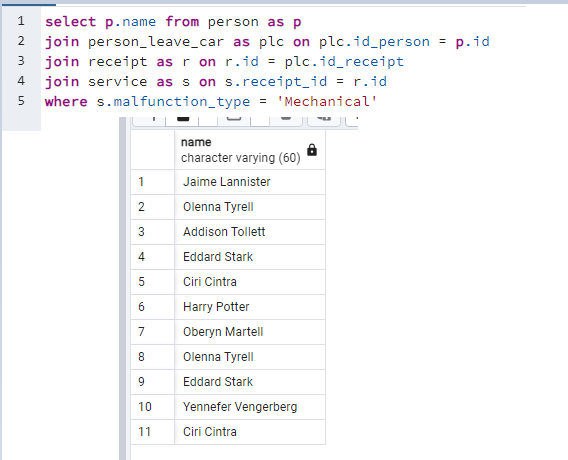


Рисунок 8 – имена клиентов, сдавших в ремонт автомобили с указанным типом неисправности

Рисунок 7 – имя работника устранявшего данную несправность в автомобиле данного клиента и время устранения

## Вывод.

## В ходе выполнения лабораторной работы была успешно реализована база данных, соответствующая разработанной ранее модели, в процессе был получен опыт в написании различных запросов для БД.

**Приложение А**

**Ссылки**

Pull Request: https://github.com/moevm/sql-2023-1303/pull/32

DB Fiddle: https://www.db-fiddle.com/f/5YBgM5bGUgZBLhNYgYWyhK/3

**Приложение Б**

**Исходный код**

Файл service\_station.sql:

CREATE TABLE car

(

id serial primary key,

brand varchar (30) not null,

color varchar (30) not null,

release\_year int not null,

plate varchar (10) not null

);

CREATE TABLE person

(

id serial primary key,

passport varchar (20) not null unique,

name varchar (60) not null,

birth\_date date not null,

registration varchar(200) not null

);

CREATE TABLE worker

(

id serial primary key,

name varchar (60) not null,

speciality varchar (60) not null,

category int not null,

experience\_years int not null

);

CREATE TABLE receipt

(

id serial primary key,

start\_work\_date date not null,

return\_date date not null,

request\_date date not null,

work\_duration\_hours int,

price\_work decimal(10,2),

price\_autoparts decimal(10,2)

);

CREATE TABLE service

(

id serial primary key,

name varchar(100) not null,

receipt\_id int not null,

malfunction\_type varchar (120) not null,

price decimal(10, 2) not null,

price\_autoparts decimal(10, 2) not null,

duration\_hours int not null,

foreign key (receipt\_id) references receipt(id) on delete cascade

);

CREATE TABLE provide\_service

(

id\_car int,

id\_worker int,

id\_service int,

primary key (id\_car, id\_worker, id\_service),

foreign key (id\_car) references car(id) on delete cascade,

foreign key (id\_worker) references worker(id) on delete cascade,

foreign key (id\_service) references service(id) on delete cascade

);

CREATE TABLE person\_leave\_car

(

id\_car int,

id\_person int,

id\_receipt int,

primary key (id\_car, id\_person, id\_receipt),

foreign key (id\_car) references car(id) on delete cascade,

foreign key (id\_person) references person(id) on delete cascade,

foreign key (id\_receipt) references receipt(id) on delete cascade

);

CREATE TABLE person\_own\_car

(

id\_person int,

id\_car int,

primary key (id\_person, id\_car),

foreign key (id\_car) references car(id) on delete cascade,

foreign key (id\_person) references person(id) on delete cascade

);

insert into car(brand, color, release\_year, plate)

values

('Porsche', 'Black', 2018, 'С439РК'),

('Ford', 'Gray', 2003, 'К126КМ'),

('Audi', 'Red', 2015, 'А321АН'),

('Renault', 'Yellow', 2022, 'Т777УТ'),

('Mercedes', 'Silver', 2010, 'О456ОР'),

('Ford', 'Blue', 2019, 'У890КО'),

('Renault', 'White', 2007, 'Е222ЕВ'),

('Toyota', 'Green', 2021, 'А555АТ'),

('Nissan', 'Orange', 2017, 'К999КН'),

('Porsche', 'Purple', 2013, 'Р777РЖ'),

('Ford', 'Silver', 2016, 'С777АС'),

('Toyota', 'Blue', 2014, 'Н123НР'),

('Audi', 'Gray', 2008, 'Е456АВ'),

('Mercedes', 'Red', 2012, 'О999ТМ');

insert into worker(name, speciality, category, experience\_years)

values

('Ryan Gosling', 'Driver', 4, 5),

('John Snow', 'Driver', 2, 3),

('Sandor Clegane', 'Technician', 2, 8),

('Robb Stark', 'Technician', 3, 9),

('Jorah Mormont', 'Mechanic', 3, 7),

('Beric Dondarrion', 'Mechanic', 2, 6),

('Ramsay Bolton', 'Dispatcher', 1, 4);

insert into person(passport, name, birth\_date, registration)

values

('4297353436', 'Tywin Lannister', '1960-05-12', 'Russia, Moscow, King Landing street, 9'),

('4930495082', 'Jaime Lannister', '1980-02-23', 'Russia, Saint Petersburg, Casterly Rock street, 13'),

('4339771030', 'Oberyn Martell', '1975-07-17', 'Russia, Tver, Sunspear street, 16'),

('4816249461', 'Olenna Tyrell', '1930-11-30', 'Russia, Stavropol, Highgarden street, 23'),

('4269255595', 'Cersei Lannister', '1975-11-03', 'Russia, Log Angeles, King Landing street, 2'),

('4695873928', 'Daario Naharis', '1985-09-28', 'Russia, Stavropol, Meereen street, 5'),

('4716359214', 'Addison Tollett', '1990-04-15', 'Russia, Moscow, Castle Black street, 7'),

('4033426867', 'Eddard Stark', '1963-03-01', 'Russia, Moscow, Winterfell street, 12'),

('4665583715', 'Addison Tollett', '1988-08-11', 'Russia, Saint Petersburg, Castle Black street, 25'),

('4911897374', 'Geralt Rivian', '1985-12-15', 'Russia, Moscow, Rivia street 1'),

('4123456789', 'Yennefer Vengerberg', '1987-03-04', 'Russia, Saint Petersburg, Sorceress street, 10'),

('4765432101', 'Ciri Cintra', '1990-05-25', 'Russia, Saint Petersburg, Vizima street, 7'),

('4888777666', 'Jaskier Bard', '1980-11-30', 'Russia, Moscow, Troubadour street, 15'),

('4234567890', 'Harry Potter', '1989-07-17', 'Russia, Saint Petersburg, Mage street, 3');

insert into receipt(start\_work\_date, return\_date, request\_date)

values

('2023-01-15', '2023-01-18', '2023-01-10'),

('2023-02-10', '2023-02-12', '2023-02-08'),

('2023-03-05', '2023-03-08', '2023-03-02'),

('2023-04-20', '2023-04-23', '2023-04-18'),

('2023-05-12', '2023-05-15', '2023-05-08'),

('2023-06-25', '2023-06-28', '2023-06-20'),

('2023-07-17', '2023-07-20', '2023-07-15'),

('2023-08-05', '2023-08-08', '2023-08-02'),

('2023-07-10', '2023-07-13', '2023-07-05'),

('2023-06-15', '2023-06-18', '2023-06-10'),

('2023-05-20', '2023-05-23', '2023-05-15'),

('2023-04-25', '2023-04-28', '2023-04-20'),

('2023-03-30', '2023-04-02', '2023-03-25'),

('2023-02-12', '2023-02-15', '2023-02-08');

insert into service(id, name, receipt\_id, malfunction\_type, price, price\_autoparts, duration\_hours)

values

(2, 'Brake Repair', 2, 'Mechanical', 129.99, 60.00, 2),

(4, 'Engine Tune-up', 4, 'Mechanical', 89.99, 40.00, 2),

(7, 'Transmission Flush', 7, 'Mechanical', 149.99, 75.00, 3),

(8, 'Brake Repair', 8, 'Mechanical', 129.99, 60.00, 2),

(12, 'Brake Repair', 12, 'Mechanical', 129.99, 60.00, 2),

(14, 'Engine Tune-up', 14, 'Mechanical', 89.99, 40.00, 2),

(17, 'Brake Repair', 3, 'Mechanical', 129.99, 60.00, 2),

(18, 'Transmission Flush', 4, 'Mechanical', 149.99, 75.00, 3),

(22, 'Engine Tune-up', 8, 'Mechanical', 89.99, 40.00, 2),

(25, 'Brake Repair', 11, 'Mechanical', 129.99, 60.00, 2),

(26, 'Transmission Flush', 12, 'Mechanical', 149.99, 75.00, 3),

(1, 'Oil Change', 1, 'Maintenance', 49.99, 25.00, 1),

(3, 'Tire Rotation', 3, 'Maintenance', 19.99, 10.00, 1),

(11, 'Oil Change', 11, 'Maintenance', 49.99, 25.00, 1),

(13, 'Tire Rotation', 13, 'Maintenance', 19.99, 10.00, 1),

(15, 'Oil Change', 1, 'Maintenance', 49.99, 25.00, 1),

(19, 'Oil Change', 5, 'Maintenance', 49.99, 25.00, 1),

(20, 'Tire Rotation', 6, 'Maintenance', 19.99, 10.00, 1),

(28, 'Oil Change', 14, 'Maintenance', 49.99, 25.00, 1),

(16, 'Wheel Alignment', 2, 'Suspension', 79.99, 35.00, 2),

(5, 'Wheel Alignment', 5, 'Suspension', 79.99, 35.00, 2),

(10, 'Wheel Alignment', 10, 'Suspension', 79.99, 35.00, 2),

(23, 'Wheel Alignment', 9, 'Suspension', 79.99, 35.00, 2),

(9, 'AC Service', 9, 'Electrical', 59.99, 30.00, 2),

(21, 'AC Service', 7, 'Electrical', 59.99, 30.00, 2),

(6, 'Car delivery', 6, '-', 49.99, 25.00, 1),

(24, 'Car delivery', 10, '-', 49.99, 25.00, 1),

(27, 'Car delivery', 13, '-', 19.99, 10.00, 1);

insert into person\_leave\_car(id\_car, id\_person, id\_receipt)

values

(1,1,1),

(2,2,2),

(3,3,3),

(4, 4, 4),

(5, 5, 5),

(6, 6, 6),

(7, 7, 7),

(8, 8, 8),

(9, 9, 9),

(10, 10, 10),

(11, 11, 11),

(12, 12, 12),

(13, 13, 13),

(14, 14, 14);

insert into person\_own\_car

values

(1, 1),

(2, 2),

(3, 3),

(4, 4),

(5, 5),

(6, 6),

(7, 7),

(8, 8),

(9, 9),

(10, 10),

(11, 11),

(12, 12),

(13, 13),

(14, 14);

insert into provide\_service(id\_car, id\_worker, id\_service)

values

(1,5,2),

(1,3,1),

(2,6,4),

(2,4,3),

(3,5,7),

(3,3,11),

(4,6,8),

(4,4,13),

(5,5,12),

(5,3,15),

(6,6,14),

(6,4,19),

(7,5,17),

(7,3,20),

(8,6,18),

(8,4,28),

(9,5,22),

(9,1,24),

(10,6,25),

(10,4,5),

(11,5,26),

(11,3,10),

(12,3,23),

(12,4,9),

(13,4,21),

(13,1,6),

(14,3,16),

(14,2,27);

UPDATE receipt

SET

work\_duration\_hours = (select sum(service.duration\_hours) from service where service.receipt\_id = receipt.id)

where receipt.id in (SELECT DISTINCT receipt\_id FROM service);

UPDATE receipt

SET

price\_work = (select sum(service.price) from service where service.receipt\_id = receipt.id)

where receipt.id in (SELECT DISTINCT receipt\_id FROM service);

UPDATE receipt

SET

price\_autoparts = (select sum(service.price\_autoparts) from service where service.receipt\_id = receipt.id)

where receipt.id in (SELECT DISTINCT receipt\_id FROM service);

select p.name, p.registration from person as p

join person\_own\_car as poc on p.id = poc.id\_person

join car as c on poc.id\_car = c.id

where c.plate = 'Р777РЖ';

select c.brand, c.release\_year from car as c

join person\_own\_car as poc on c.id = poc.id\_car

join person as p on poc.id\_person = p.id

where p.id = 5;

select s.malfunction\_type from service as s

join receipt as r on s.receipt\_id = r.id

join person\_leave\_car as plc on r.id = plc.id\_receipt

where plc.id\_person = 8;

select w.name, s.duration\_hours from worker as w

join provide\_service as ps on ps.id\_worker = w.id

join service as s on s.id = ps.id\_service

join receipt as r on r.id = s.receipt\_id

join person\_leave\_car as plc on plc.id\_receipt = r.id

where s.malfunction\_type = 'Mechanical' and plc.id\_person = 2;

select p.name from person as p

join person\_leave\_car as plc on plc.id\_person = p.id

join receipt as r on r.id = plc.id\_receipt

join service as s on s.receipt\_id = r.id

where s.malfunction\_type = 'Mechanical'