

**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 храниться 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 вернуть 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`.

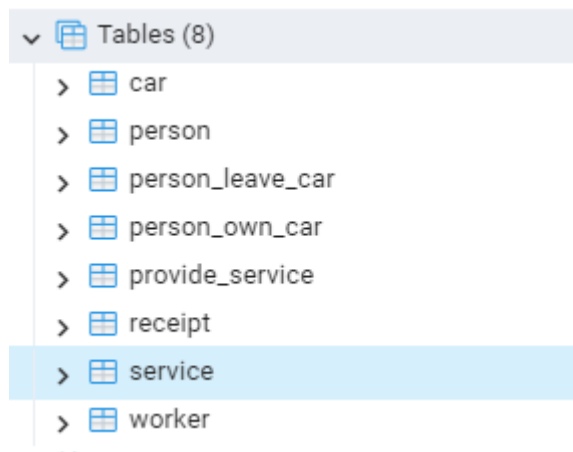


Рисунок 1 – созданные таблицы

```
CREATE TABLE car
(
    id serial primary key,
    brand varchar (30),
    color varchar (30),
    release_year int,
    plate varchar (10)
);

CREATE TABLE person
(
    id serial primary key,
    passport varchar (20),
    name varchar (60),
    birth_date date,
    registration varchar(200)
);

CREATE TABLE worker
(
    id serial primary key,
    name varchar (60),
    speciality varchar (60),
    category int,
    experience_years int
);
```

Рисунок 2 – пример создания таблиц

```
insert into car(brand, color, release_year, plate)
values
('Porsche', 'Black', 2018, 'C439PK'),
('Ford', 'Gray', 2003, 'K126KM'),
('Audi', 'Red', 2015, 'A321AH'),
('Renault', 'Yellow', 2022, 'T777YT'),
('Mercedes', 'Silver', 2010, 'O456OP'),
('Ford', 'Blue', 2019, 'Y890KO'),
('Renault', 'White', 2007, 'E222EB'),
('Toyota', 'Green', 2021, 'A555AT'),
('Nissan', 'Orange', 2017, 'K999KH'),
('Porsche', 'Purple', 2013, 'P777PK'),
('Ford', 'Silver', 2016, 'C777AC'),
('Toyota', 'Blue', 2014, 'H123HP'),
('Audi', 'Gray', 2008, 'E456AB'),
('Mercedes', 'Red', 2012, 'O999TM');

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),
('Jonah 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, Iver, 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', 'Gerald Rivian', '1985-12-15', 'Russia, Moscow, Rivia street 1'),
('4123456789', 'Yennifer 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');
```

Рисунок 3 – пример заполнения таблиц

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

```
1 select p.name, p.registration from person as p
2 join person_own_car as poc on p.id = poc.id_person
3 join car as c on poc.id_car = c.id
4 where c.plate = 'P777РЖ'
```

	name character varying (60)	registration character varying (80)
1	Geralt Rivian	USA, New York, Rivia street 1

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

```
1 select c.brand, c.release_year from car as c
2 join person_own_car as poc on c.id = poc.id_car
3 join person as p on poc.id_person = p.id
4 where p.id = 5
```

	brand character varying (30)	release_year integer
1	Mercedes	2010

Рисунок 5 - марка и год выпуска автомобиля данного владельца

```
1 select s.malfunction_type from service as s
2 join receipt as r on s.receipt_id = r.id
3 join person_leave_car as plc on r.id = plc.id_receipt
4 join person as p on plc.id_person = p.id
5 join person_own_car as poc on p.id = poc.id_person
6 where poc.id_person = 7
```

	malfunction_type character varying (120)
1	Mechanical
2	Electrical

Рисунок 6 – типы устраненных неисправностей в автомобиле данного клиента

```

1 select w.name, s.duration_hours from worker as w
2 join provide_service as ps on ps.id_worker = w.id
3 join service as s on s.id = ps.id_service
4 join receipt as r on r.id = s.receipt_id
5 join person_leave_car as plc on plc.id_receipt = r.id
6 where s.malfunction_type = 'Mechanical' and plc.id_person = 2

```

	name character varying (60)	duration_hours integer
1	Jorah Mormont	2

Рисунок 7 – имя работника устранившего данную неисправность в автомобиле данного клиента и время устранения

```

1 select p.name from person as p
2 join person_leave_car as plc on plc.id_person = p.id
3 join receipt as r on r.id = plc.id_receipt
4 join service as s on s.receipt_id = r.id
5 where s.malfunction_type = 'Mechanical'

```

	name character varying (60)
1	Jaime Lannister
2	Olenna Tyrell
3	Addison Tollett
4	Eddard Stark
5	Ciri Cintra
6	Harry Potter
7	Oberyn Martell
8	Olenna Tyrell
9	Eddard Stark
10	Yennefer Vengerberg
11	Ciri Cintra

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

## 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, 'C439PK'),
('Ford', 'Gray', 2003, 'K126KM'),
('Audi', 'Red', 2015, 'A321AH'),
('Renault', 'Yellow', 2022, 'T777YT'),
('Mercedes', 'Silver', 2010, 'O456OP'),
('Ford', 'Blue', 2019, 'Y890KO'),
('Renault', 'White', 2007, 'E222EB'),
('Toyota', 'Green', 2021, 'A555AT'),
('Nissan', 'Orange', 2017, 'K999KH'),
('Porsche', 'Purple', 2013, 'P777PK'),
('Ford', 'Silver', 2016, 'C777AC'),
('Toyota', 'Blue', 2014, 'H123HP'),
('Audi', 'Gray', 2008, 'E456AB'),
('Mercedes', 'Red', 2012, 'O999TM');

```

```

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 = 'P777PЖ';
```

```
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'
```