

Задание. На основе прикладной области «Автостоянка» выполнить следующие задачи:

- создать БД;
- создать таблицы БД, определить поля таблиц и тип данных полей;
- определить связи между таблицами и ограничения целостности, которые характерны для предметной области БД;
- создать схему БД;

Решение.

База данных была создана с использованием кода представленного ниже.

```
create database parking
with
owner = postgres
encoding = 'UTF8'
lc_collate = 'ru_RU.UTF-8'
lc_ctype = 'ru_RU.UTF-8'
tablespace = pg_default
connection limit = -1
is_template = false
template = template0;
```

Были созданы следующие таблицы, в пояснение к каждой таблице приведён код на языке *SQL*:

1) *Contracts*.

```
create table contracts (
contract_id smallint not null,
client_id smallint not null,
tariff_id smallint not null,
start_date date not null,
end_date date not null,
parking_space_id smallint not null,
entry_permit boolean not null,
constraint contracts_pk primary key (contract_id)
);
```

2) *Tariffs*.

```
create table tariffs (
tariff_id smallint not null,
tariff_price numeric(8, 2) not null,
constraint tariffs_pk primary key (tariff_id)
```

);

3) *Clients.*

```
create table clients (  
  client_id smallint not null,  
  client_name varchar(60) not null,  
  constraint clients_pk primary key (client_id)  
);
```

4) *Cars.*

```
create table cars (  
  car_id varchar(10) not null,  
  car_name varchar(30) not null,  
  constraint cars_pk primary key (car_id)  
);
```

5) *Payments.*

```
create table payments (  
  contract_id smallint not null,  
  payment_id integer not null,  
  payment_amount numeric(8, 2) not null,  
  payment_date date not null,  
  constraint payments_pk primary key (contract_id)  
);
```

6) *Parking_spaces.*

```
create table parking_spaces (  
  parking_space_id smallint not null,  
  parking_space_area smallint not null,  
  constraint parking_spaces_pk primary key (parking_space_id)  
);
```

7) *Passages.*

```
create table passages (  
  contract_id smallint not null,  
  passage_id integer not null,  
  passage_date date not null,  
  passage_type varchar(5) not null,  
  car_id varchar(10) not null,  
  constraint passages_pk primary key (contract_id, passage_id)  
);
```

Для поддержания целостности данных в таблицы были добавлены ограничения на внешние ключи. Каждое ограничение задаёт каскадное удаление и изменение данных между таблицами.

Введённые ограничения для таблиц и поясняющий код *SQL* для каждого ограничения:

1) Для таблицы *contracts* обозначены три внешних ключа: *client_id*, *tariff_id* и *parking_space_id*.

```
alter table contracts
add constraint contracts_fk1 foreign key (client_id)
references clients (client_id)
on delete cascade
on update cascade;
```

```
alter table contracts
add constraint contracts_fk2 foreign key (tariff_id)
references tariffs (tariff_id)
on delete cascade
on update cascade;
```

```
alter table contracts
add constraint contracts_fk3 foreign key (parking_space_id)
references parking_spaces (parking_space_id)
on delete cascade
on update cascade;
```

2) Для таблицы *payments* обозначен единственный внешний ключ *contract_id*.

```
alter table payments
add constraint payments_fk foreign key (contract_id)
references contracts (contract_id)
on delete cascade
on update cascade;
```

3) Для таблицы *passages* были обозначены два внешних ключа *contract_id* и *cars*.

```
alter table passages
add constraint passages_fk1 foreign key (contract_id)
references contracts (contract_id)
on delete cascade
on update cascade;
```

```
alter table passages
add constraint passages_fk2 foreign key (car_id)
references cars (car_id)
on delete cascade
```

on update cascade;

ERD, сгенерированная программой *pgAdmin4* показана на Рисунке 1.

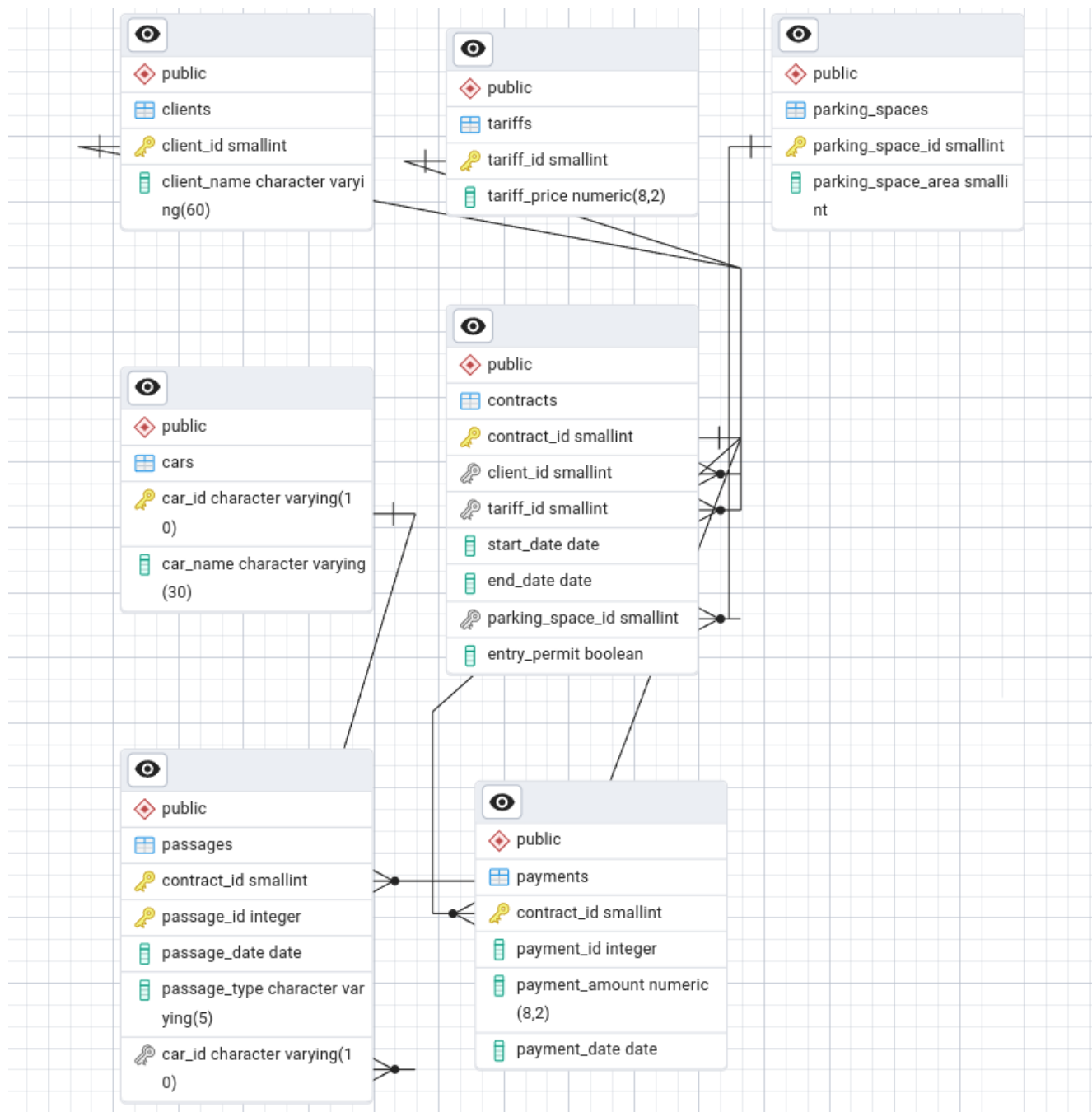


Рисунок 1 - ERD базы данных «Автостоянка»