

Проектная работа
по дисциплине «Хранение и обработка данных»
«Тема 15 "Расписание электричек»>

Выполнила:
Бунковская Анна
ИСУ: 465304

1 Описание предметной области

Система предназначена для хранения и обработки расписания электричек вокзалов Санкт-Петербурга.

1.1 Хранимая информация

В системе хранится следующая информация:

- **Вокзалы** - основные транспортные узлы
- **Станции** - железнодорожные станции города и области
- **Маршруты** - маршруты следования электричек
- **Привязка маршрутов к станциям** - информация о том, какие маршруты проходят через какие станции
- **Время прибытия/отправления** - точное время прибытия и отправления электричек
- **Дни действия маршрутов** - расписание для будних дней и выходных

Пользователь системы может:

- Получать расписание на конкретный день
- Просматривать расписание для конкретного маршрута
- Ознакамливаться с расписанием для конкретного вокзала
- Фильтровать расписание по дням недели (будни/выходные)

2 Структура базы данных

2.1 Таблицы базы данных

- **vokzals** — таблица вокзалов
- **stations** — таблица станций (с привязкой к вокзалам)
- **routes** — таблица маршрутов
- **route_stations** — расписание маршрутов (остановки, порядок, время)
- **trains** — таблица поездов (опционально)
- **days** — таблица дней действия маршрутов (если реализовано отдельно)

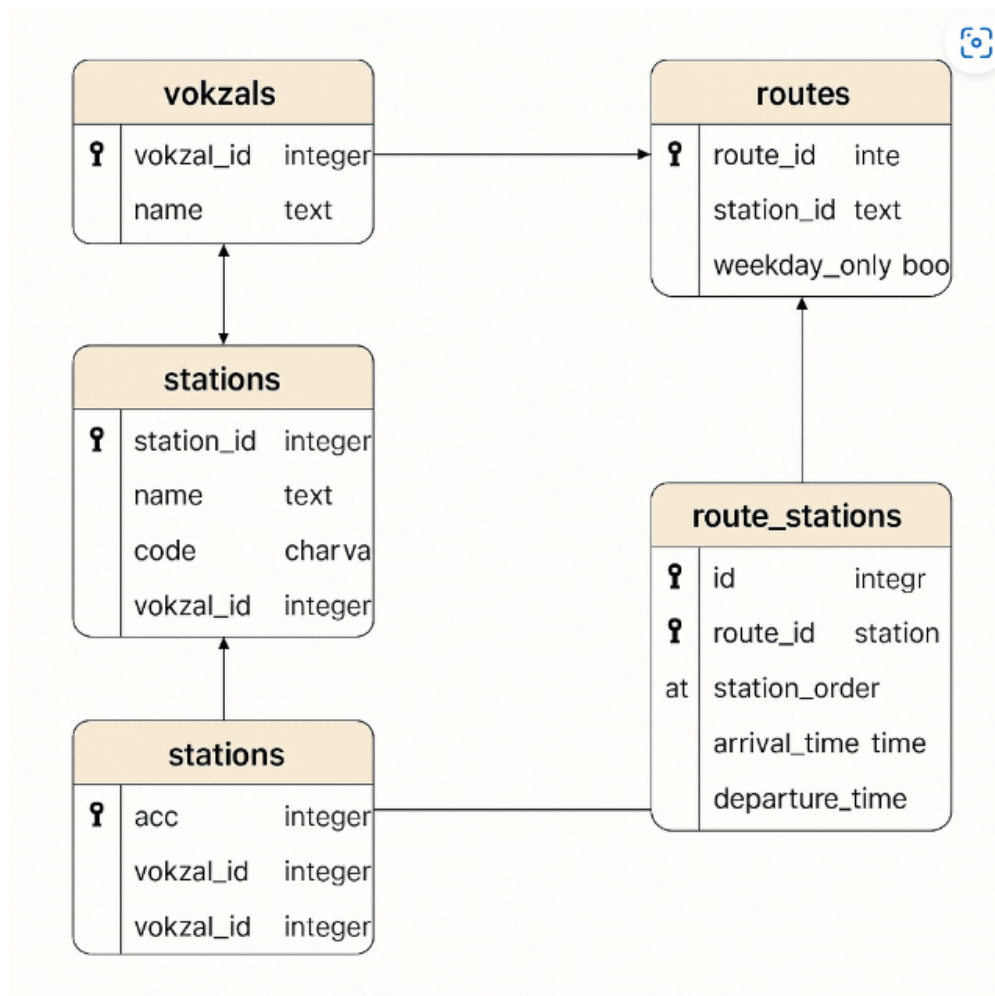


Рис. 1: ER-диаграмма базы данных расписания электричек

2.2 ER-диаграмма

3 Заполнение данными

3.1 Генерация тестовых данных

Для заполнения базы данных использовались следующие файлы:

- `add_stations.py` — Python-скрипт для заполнения `stations` в БД
- `add_routes.py` — Python-скрипт для заполнения `routes` в БД
- `add_route_stations.py` — Python-скрипт для заполнения `route.stations` в БД

3.2 Скрипт создание БД в pgAdmin

```

1 -- Создание таблицы вокзалов
2 CREATE TABLE vokzals (
3     vokzal_id SERIAL PRIMARY KEY,
4     name VARCHAR(100) NOT NULL,
5     city VARCHAR(50) NOT NULL
6 );
7
  
```

```

8  -- Создание таблицы типов недели
9  CREATE TABLE week_types (
10     week_type_id SERIAL PRIMARY KEY,
11     name VARCHAR(20) NOT NULL
12 );
13
14 -- Создание таблицы поездов
15 CREATE TABLE trains (
16     train_id SERIAL PRIMARY KEY,
17     model VARCHAR(50) NOT NULL,
18     capacity INTEGER NOT NULL
19 );
20
21 -- Создание таблицы станций
22 CREATE TABLE stations (
23     station_id SERIAL PRIMARY KEY,
24     name VARCHAR(100) NOT NULL,
25     code VARCHAR(10) NOT NULL,
26     vokzal_id INTEGER REFERENCES vokzals(vokzal_id)
27 );
28
29 -- Создание таблицы маршрутов с флагами дней
30 CREATE TABLE routes (
31     route_id SERIAL PRIMARY KEY,
32     name VARCHAR(100) NOT NULL,
33     train_id INTEGER REFERENCES trains(train_id),
34     weekday_only BOOLEAN DEFAULT FALSE,
35     weekend_only BOOLEAN DEFAULT FALSE
36 );
37
38 -- Создание таблицы остановок маршрута
39 CREATE TABLE route_stations (
40     route_station_id SERIAL PRIMARY KEY,
41     route_id INTEGER REFERENCES routes(route_id),
42     station_id INTEGER REFERENCES stations(station_id),
43     stop_order INTEGER NOT NULL,
44     arrival_time TIME,
45     departure_time TIME
46 );
47
48 -- Создание таблицы расписания
49 CREATE TABLE schedules (
50     schedule_id SERIAL PRIMARY KEY,
51     route_id INTEGER REFERENCES routes(route_id),
52     week_type_id INTEGER REFERENCES week_types(week_type_id)
53 );
54
55 -- Вставка вокзалов
56 INSERT INTO vokzals (name, city) VALUES
57 ('Финляндский вокзал', 'Санкт-Петербург'),
58 ('Московский вокзал', 'Санкт-Петербург'),
59 ('Витебский вокзал', 'Санкт-Петербург');
60

```

```

61 -- Вставка типов недель
62 INSERT INTO week_types (name) VALUES
63 ('Рабочие дни'), ('Выходные дни');
64
65 -- Вставка поездов
66 INSERT INTO trains (model, capacity) VALUES
67 ('ЭД4М', 800),
68 ('Ласточка', 350),
69 ('ЭР2Т', 700);
70
71 -- Вставка станций
72 INSERT INTO stations (name, code, vokzal_id) VALUES
73 ('Ладужская', 'LDZ', 1),
74 ('Девяткино', 'DVT', 1),
75 ('Токсово', 'TKS', 1),
76 ('Пискаревка', 'PSK', 1),
77 ('Пушкин', 'PSH', 2),
78 ('Павловск', 'PVL', 2),
79 ('Тосно', 'TSN', 2),
80 ('Гатчина', 'GCH', 3),
81 ('Дно', 'DNO', 3),
82 ('Луга', 'LGA', 3);
83
84 -- Вставка маршрутов с указанием дней
85 INSERT INTO routes (name, train_id, weekday_only, weekend_only)
86     VALUES
87     ('СПб Токсово', 1, FALSE, FALSE), -- Работает все дни
88     ('СПб Павловск', 2, TRUE, FALSE), -- Только рабочие дни
89     ('СПб Луга', 3, FALSE, TRUE); -- Только выходные
90
91 -- Привязка маршрутов к станциям
92 INSERT INTO route_stations (route_id, station_id, stop_order,
93     arrival_time, departure_time) VALUES
94     -- Маршрут СПб Токсово
95     (1, 1, 1, '06:00', '06:05'),
96     (1, 2, 2, '06:30', '06:32'),
97     (1, 3, 3, '07:00', NULL),
98
99     -- Маршрут СПб Павловск
100    (2, 4, 1, '08:00', '08:05'),
101    (2, 5, 2, '08:25', '08:30'),
102    (2, 6, 3, '09:00', NULL),
103
104    -- Маршрут СПб Луга
105    (3, 7, 1, '10:00', '10:05'),
106    (3, 8, 2, '11:00', '11:10'),
107    (3, 9, 3, '12:00', NULL);
108
109 -- Привязка расписания к неделе
110 INSERT INTO schedules (route_id, week_type_id) VALUES
111 (1, 1), (1, 2), -- Маршрут 1 работает все дни
112 (2, 1), -- Маршрут 2 только рабочие
113 (3, 2); -- Маршрут 3 только выходные

```

4 Типовые SQL-запросы

4.1 Примеры запросов к базе данных

```
1 SELECT rs.station_order, s.name, rs.arrival_time, rs.departure_time
2 FROM route_stations rs
3 JOIN stations s ON rs.station_id = s.station_id
4 WHERE rs.route_id = 5
5 ORDER BY rs.station_order;
```

Листинг 2: Расписание маршрута с остановками

```
1 SELECT * FROM stations WHERE вокзал_id = 1;
```

Листинг 3: Список станций для конкретного вокзала:

```
1 SELECT v.name, COUNT(DISTINCT r.route_id)
2 FROM вокзалы v
3 JOIN stations s ON v.вокзал_id = s.вокзал_id
4 JOIN route_stations rs ON s.station_id = rs.station_id
5 JOIN routes r ON rs.route_id = r.route_id
6 GROUP BY v.name;
```

Листинг 4: Все вокзалы с количеством маршрутов:

```
1 SELECT * FROM routes WHERE weekday_only = TRUE;
```

Листинг 5: Все маршруты с отправлением в рабочие дни:

```
1 SELECT * FROM вокзалы;
```

Листинг 6: Все вокзалы

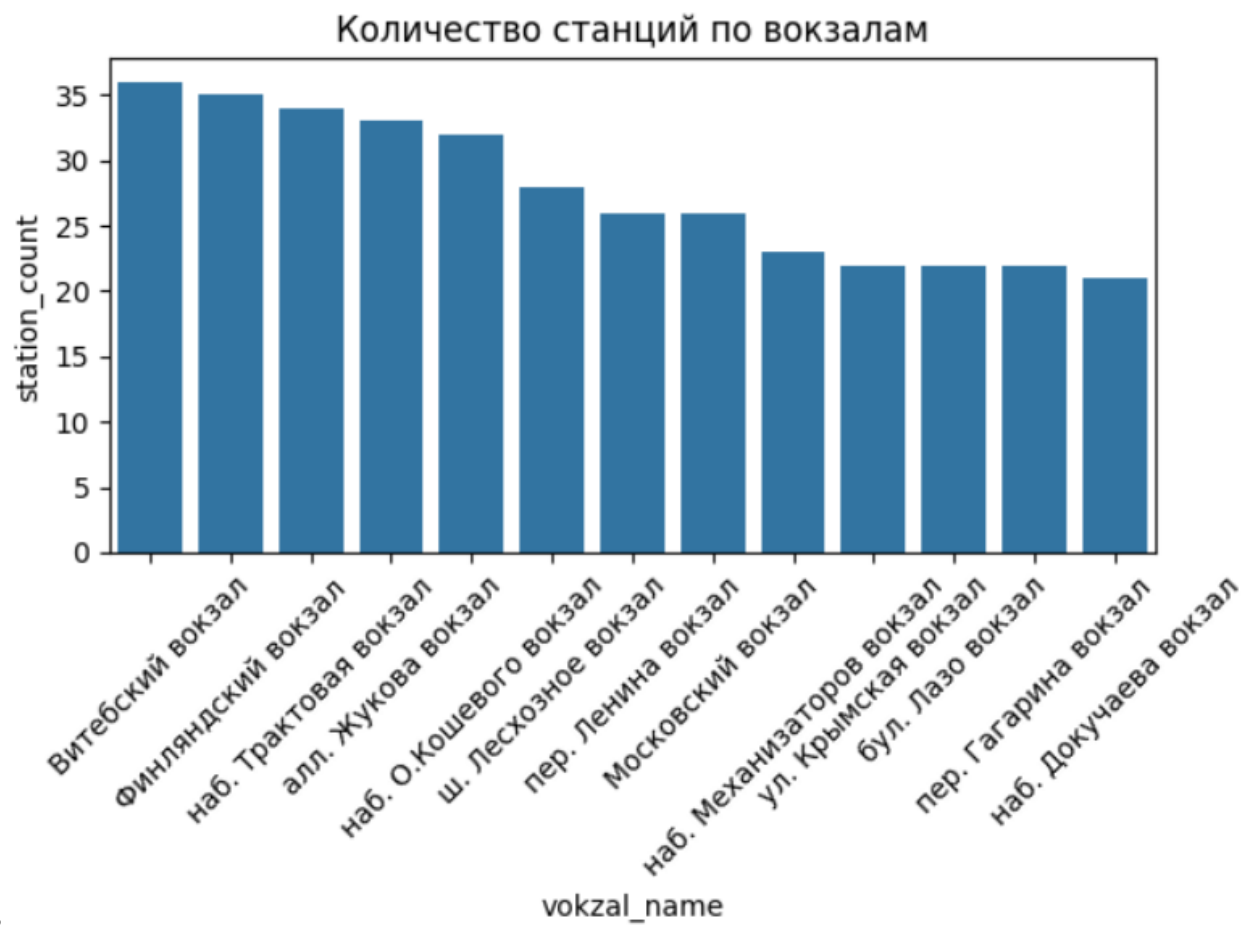
```
1 SELECT s.name
2 FROM route_stations rs
3 JOIN stations s ON rs.station_id = s.station_id
4 WHERE rs.route_id = 5
5 ORDER BY rs.station_order;
```

Листинг 7: Все станции маршрута №5 (с названиями)

```
1 SELECT route_id, COUNT(*) AS num_stops
2 FROM route_stations
3 GROUP BY route_id
4 HAVING COUNT(*) > 5;
```

Листинг 8: Маршруты с +5 остановками

4.2 Визуал на Питоне



0.3

Распределение маршрутов по дням недели

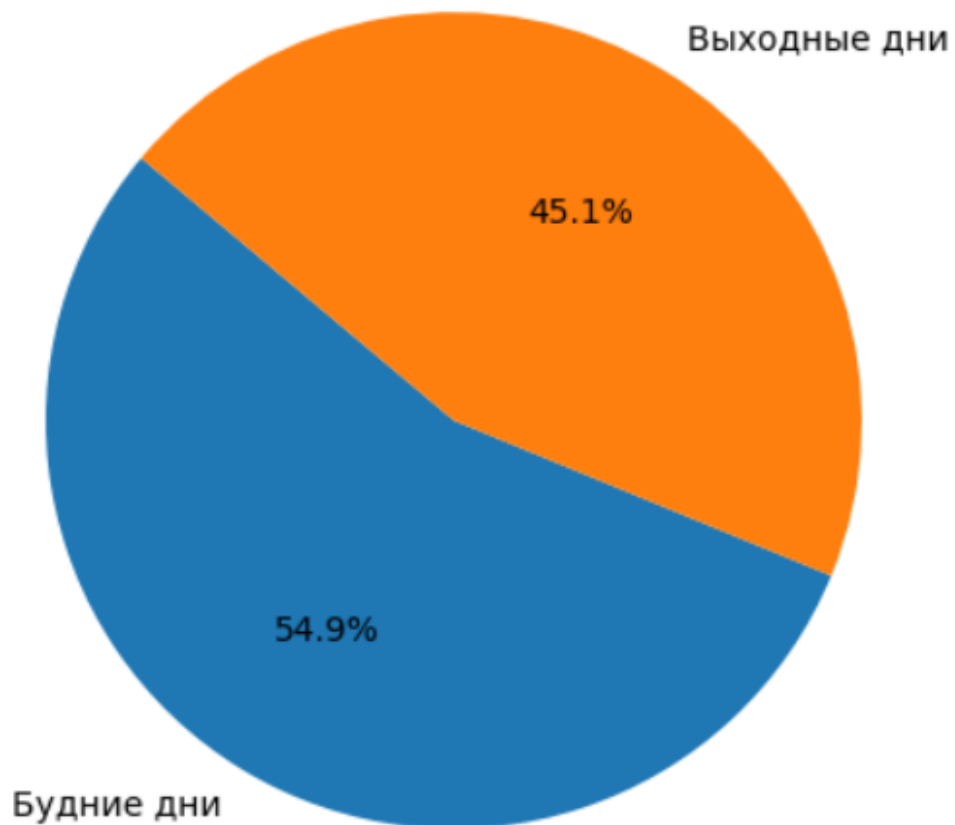


Рис. 2: Диаграмма 1

Количество маршрутов на вокзал

