

# Проектная работа по модулю “SQL и получение данных”



**Выполнил:**  
Хильчук Максим Игоревич  
**Группа:**  
SQL-38

февраль – март

2022

## 1. Тип подключения.

В работе использовался локальный тип подключения (рисунок 1):

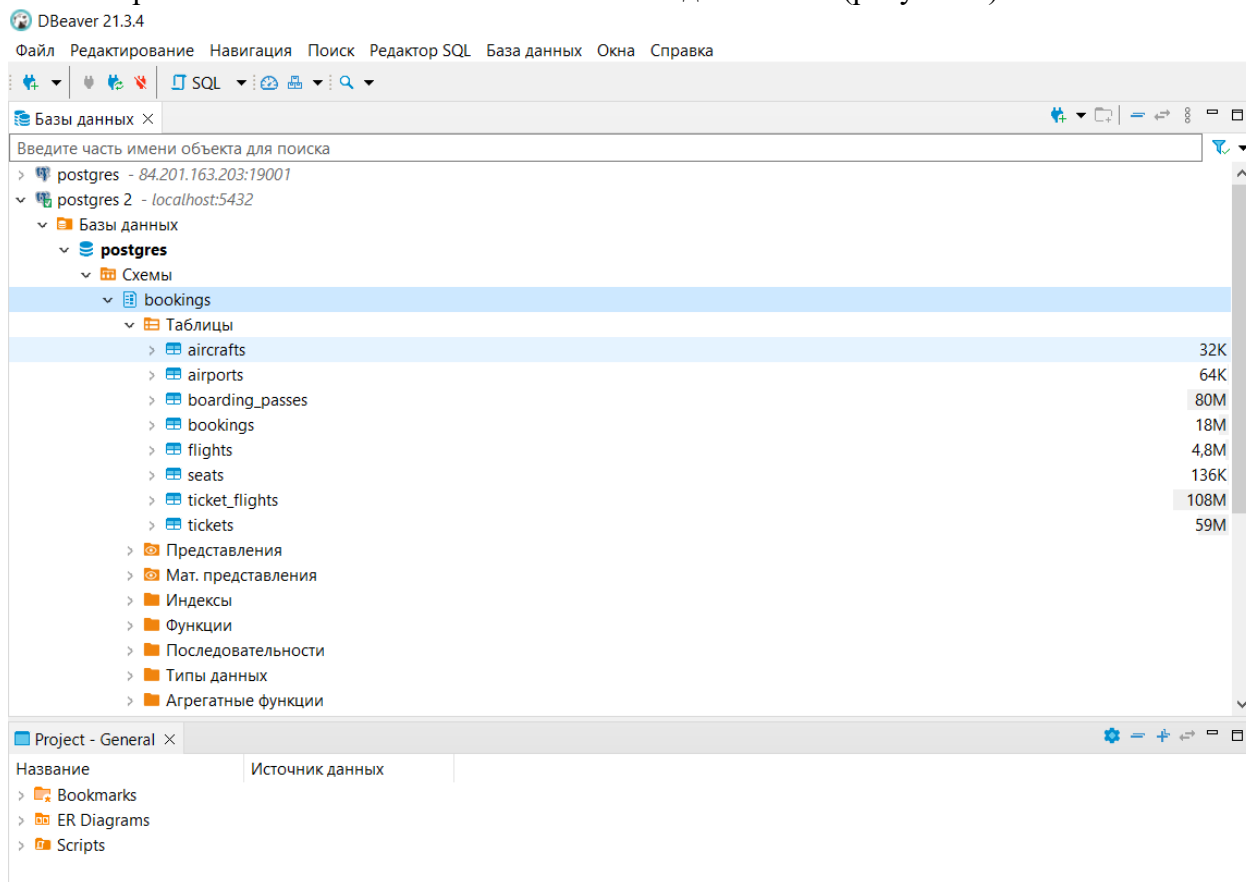


Рисунок 1. Скриншот древа базы данных bookings с локальным типом подключения.

## 2. ER-диаграмма

Скриншот ER-диаграммы из DBeaver'a согласно типу подключения представлен на рисунке 2:

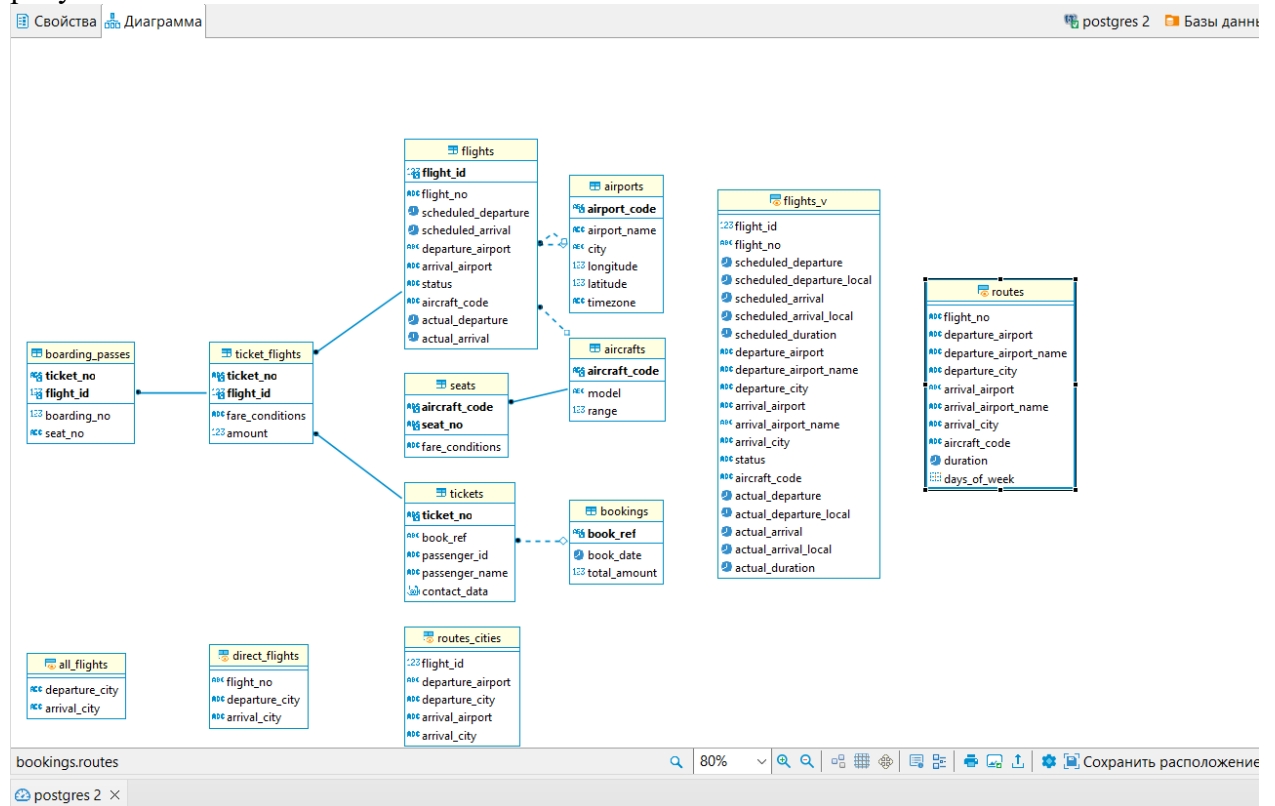


Рисунок 2. Скриншот ER-диаграммы схемы "bookings"

### 3. Краткое описание базы данных.

#### *Таблицы:*

- **aircrafts:** в таблице имеется информация о уникальном коде воздушного судна (aircraft\_code), названии модели (model) и максимальной дальности полета (range);
- **airports:** в таблице имеется информация о уникальном коде аэропорта (airport\_code), его наименовании (airport\_name), названии города аэропорта (city), координатах аэропорта (широта - longitude, долгота - latitude) и часовом поясе аэропорта (timezone);
- **boarding\_passes:** в таблице содержится информация о номере посадочного талона (boarding\_no), номере места пассажира (seat\_no), номере билета (ticket\_no) и уникальном идентификаторе полета (flight\_id);
- **bookings:** в таблице содержится информация о дате бронирования (book\_date), номере бронирования (book\_ref), а также полной стоимости бронирования (total\_amount);
- **flights:** в таблице содержится информация о уникальном идентификаторе полета (flight\_id), номере рейса (flight\_no), времени вылета и прилета по расписанию (scheduled\_departure, scheduled\_arrival) и фактические (actual\_departure, actual\_arrival), аэропорте вылета и прилета (departure\_airport, arrival\_airport), статусе рейса (status), а также коде самолета (aircraft\_code);
- **seats:** в таблице имеется информация о коде самолета (aircraft\_code), номере места (seat\_no) и классе обслуживания (fare\_conditions);
- **ticket\_flights:** в таблице имеется информация о номере билета (ticket\_no), уникальном идентификаторе рейса (flight\_id), стоимости перелета (amount) и классе обслуживания (fare\_conditions);
- **tickets:** в таблице содержится информация о номере билета (ticket\_no), номере бронирования (book\_ref), идентификаторе пассажира (passenger\_id), имени пассажира (passenger\_name), а также его контактных данных (contact\_data).

#### *Представления:*

- **flights\_v:** содержит обширный набор данных, дополняющих информацию таблицы flights. Имеется информация о идентификаторе рейса (flight\_id), номере рейса (flight\_no), времени вылета и прилета по расписанию (scheduled\_departure, scheduled\_arrival) и фактические (actual\_departure, actual\_arrival), а также времени вылета и прилета по расписанию (scheduled\_departure\_local, scheduled\_arrival\_local) и фактические (actual\_departure\_local, actual\_arrival\_local) по местному времени в пункте отправления, планируемой и фактической продолжительности полета (scheduled\_duration, actual\_duration), коде и названии аэропорта отправления (departure\_airport, departure\_airport\_name) и аэропорте прибытия (arrival\_airport,

arrival\_airport\_name), городе отправления и прибытия (departure\_city, arrival\_city), коде самолета (aircraft\_code) и статусе рейса (status);

- **материализованное представление routes:** содержит информацию о номере рейса (flight\_no), коде и названии аэропорта отправления (departure\_airport, departure\_airport\_name) и аэропорте прибытия (arrival\_airport, arrival\_airport\_name), городе отправления и прибытия (departure\_city, arrival\_city), коде самолета (aircraft\_code), продолжительности полета (duration) и днях недели выполнения рейсов (days\_of\_week).

## 4. Развернутый анализ базы данных. Бизнес задачи.

### 4.1. Aircrafts.

Каждая модель воздушного судна идентифицируется своим трехзначным кодом (aircraft\_code). Указывается также название модели (model) и максимальная дальность полета в километрах (range).

#### Индексы:

- PRIMARY KEY, btree (aircraft\_code).

#### Ограничения-проверки:

- CHECK (range > 0).

#### Ссылки извне:

- TABLE "flights"
  - FOREIGN KEY (aircraft\_code) REFERENCES aircrafts(aircraft\_code)
- TABLE "seats"
  - FOREIGN KEY (aircraft\_code) REFERENCES aircrafts(aircraft\_code) ON DELETE CASCADE

### 4.2. Airports.

Аэропорт идентифицируется трехбуквенным кодом (airport\_code) и имеет свое имя (airport\_name). Для города не предусмотрено отдельной сущности, но название (city) указывается и может служить для того, чтобы определить аэропорты одного города. Также указывается широта (longitude), долгота (latitude) и часовой пояс (timezone).

#### Индексы:

- PRIMARY KEY, btree (airport\_code).

#### Ссылки извне:

- TABLE "flights"
  - FOREIGN KEY (arrival\_airport) REFERENCES airports(airport\_code)
- TABLE "flights"
  - FOREIGN KEY (departure\_airport) REFERENCES airports(airport\_code).

### 4.3. Boarding\_passes.

При регистрации на рейс, которая возможна за сутки до плановой даты отправления, пассажиру выдается посадочный талон. Он идентифицируется также, как и перелет - номером билета и номером рейса. Посадочным талонам присваиваются последовательные номера (boarding\_no) в порядке регистрации пассажиров на рейс (этот номер будет

уникальным только в пределах данного рейса). В посадочном талоне указывается номер места (seat\_no).

Индексы:

- PRIMARY KEY, btree (ticket\_no, flight\_id)
  - UNIQUE CONSTRAINT, btree (flight\_id, boarding\_no);
  - UNIQUE CONSTRAINT, btree (flight\_id, seat\_no).

Ограничения внешнего ключа:

- FOREIGN KEY (ticket\_no, flight\_id) REFERENCES ticket\_flights(ticket\_no, flight\_id).

#### 4.4. Bookings.

Пассажир заранее (book\_date, максимум за месяц до рейса) бронирует билет себе и, возможно, нескольким другим пассажирам. Бронирование идентифицируется номером (book\_ref, шестизначная комбинация букв и цифр). Поле total\_amount хранит общую стоимость включенных в бронирование перелетов всех пассажиров.

Индексы:

- PRIMARY KEY, btree (book\_ref).

Ссылки извне:

- TABLE "tickets"
  - FOREIGN KEY (book\_ref) REFERENCES bookings(book\_ref).

#### 4.5. Flights.

Естественный ключ таблицы рейсов состоит из двух полей — номера рейса (flight\_no) и даты отправления (scheduled\_departure). Чтобы сделать внешние ключи на эту таблицу компактнее, в качестве первичного используется суппругатный ключ (flight\_id). Рейс всегда соединяет две точки — аэропорты вылета (departure\_airport) и прибытия (arrival\_airport). Такое понятие, как «рейс с пересадками» отсутствует: если из одного аэропорта до другого нет прямого рейса, в билет просто включаются несколько необходимых рейсов. У каждого рейса есть запланированные дата и время вылета (scheduled\_departure) и прибытия (scheduled\_arrival). Реальные время вылета (actual\_departure) и прибытия (actual\_arrival) могут отличаться: обычно не сильно, но иногда и на несколько часов, если рейс задержан. Статус рейса (status) может принимать одно из следующих значений: Scheduled, On Time, Delayed, Departed, Arrived, Cancelled.

Индексы:

- PRIMARY KEY, btree (flight\_id)
  - UNIQUE CONSTRAINT, btree (flight\_no, scheduled\_departure).

Ограничения-проверки:

- CHECK (scheduled\_arrival > scheduled\_departure);
- CHECK ((actual\_arrival IS NULL) OR ((actual\_departure IS NOT NULL AND actual\_arrival IS NOT NULL) AND (actual\_arrival > actual\_departure)));
- CHECK (status IN ('On Time', 'Delayed', 'Departed', 'Arrived', 'Scheduled', 'Cancelled')).

Ограничения внешнего ключа:

- FOREIGN KEY (aircraft\_code) REFERENCES aircrafts(aircraft\_code);
- FOREIGN KEY (arrival\_airport) REFERENCES airports(airport\_code);
- FOREIGN KEY (departure\_airport) REFERENCES airports(airport\_code)

Ссылки извне:

- TABLE "ticket\_flights"
  - FOREIGN KEY (flight\_id) REFERENCES flights(flight\_id).

#### 4.6. Seats.

Места определяют схему салона каждой модели. Каждое место определяется своим номером (seat\_no) и имеет закрепленный за ним класс обслуживания (fare\_conditions) — Economy, Comfort или Business.

Индексы:

- PRIMARY KEY, btree (aircraft\_code, seat\_no).

Ограничения-проверки:

- CHECK (fare\_conditions IN ('Economy', 'Comfort', 'Business')).

Ограничения внешнего ключа:

- FOREIGN KEY (aircraft\_code) REFERENCES aircrafts(aircraft\_code) ON DELETE CASCADE.

#### 4.7. Ticket\_flights.

Перелет соединяет билет с рейсом и идентифицируется их номерами. Для каждого перелета указываются его стоимость (amount) и класс обслуживания (fare\_conditions).

Индексы:

- PRIMARY KEY, btree (ticket\_no, flight\_id).

Ограничения-проверки:

- CHECK (amount >= 0);
- CHECK (fare\_conditions IN ('Economy', 'Comfort', 'Business')).

Ограничения внешнего ключа:



- FOREIGN KEY (flight\_id) REFERENCES flights(flight\_id);
- FOREIGN KEY (ticket\_no) REFERENCES tickets(ticket\_no).

Ссылки извне:

- TABLE "boarding\_passes"
  - FOREIGN KEY (ticket\_no, flight\_id) REFERENCES ticket\_flights(ticket\_no, flight\_id).

#### 4.8. Tickets.

Билет имеет уникальный номер (ticket\_no), состоящий из 13 цифр. Билет содержит идентификатор пассажира (passenger\_id) — номер документа, удостоверяющего личность, — его фамилию и имя (passenger\_name) и контактную информацию (contact\_date). Ни идентификатор пассажира, ни имя не являются постоянными (можно поменять паспорт, можно сменить фамилию), поэтому однозначно найти все билеты одного и того же пассажира невозможно.

Индексы:

- PRIMARY KEY, btree (ticket\_no).

Ограничения внешнего ключа:

- FOREIGN KEY (book\_ref) REFERENCES bookings(book\_ref).

Ссылки извне:

- TABLE "ticket\_flights"
  - FOREIGN KEY (ticket\_no) REFERENCES tickets(ticket\_no).

#### 4.9. Представления и функции.

Над таблицей flights создано представление **flights\_v**, содержащее дополнительную информацию: расшифровку данных об аэропорте вылета (departure\_airport, departure\_airport\_name, departure\_city), расшифровку данных об аэропорте прибытия (arrival\_airport, arrival\_airport\_name, arrival\_city), местное время вылета (scheduled\_departure\_local, actual\_departure\_local), местное время прибытия (scheduled\_arrival\_local, actual\_arrival\_local), продолжительность полета (scheduled\_duration, actual\_duration). Столбцы представления flights\_v с типами данных и их кратким описанием приведены на рисунке 3:

Столбец	Тип	Описание
flight_id	integer	Идентификатор рейса
flight_no	char(6)	Номер рейса
scheduled_departure	timestampz	Время вылета по расписанию
scheduled_departure_local	timestamp	Время вылета по расписанию, местное время в пункте отправления
scheduled_arrival	timestampz	Время прилёта по расписанию
scheduled_arrival_local	timestamp	Время прилёта по расписанию, местное время в пункте прибытия
scheduled_duration	interval	Планируемая продолжительность полета
departure_airport	char(3)	Код аэропорта отправления
departure_airport_name	text	Название аэропорта отправления
departure_city	text	Город отправления
arrival_airport	char(3)	Код аэропорта прибытия
arrival_airport_name	text	Название аэропорта прибытия
arrival_city	text	Город прибытия
status	varchar(20)	Статус рейса
aircraft_code	char(3)	Код самолета, IATA
actual_departure	timestampz	Фактическое время вылета
actual_departure_local	timestamp	Фактическое время вылета, местное время в пункте отправления
actual_arrival	timestampz	Фактическое время прилёта
actual_arrival_local	timestamp	Фактическое время прилёта, местное время в пункте прибытия
actual_duration	interval	Фактическая продолжительность полета

Рисунок 3. Содержимое представления flights\_v

Таблица рейсов содержит избыточность: из нее можно было бы выделить информацию о маршруте (номер рейса, аэропорты отправления и назначения), которая не зависит от конкретных дат рейсов. Именно такая информация и составляет материализованное представление **routes**. Столбцы материализованного представления **routes** с типами данных и их кратким описанием приведены на рисунке 4:

Столбец	Тип	Описание
flight_no	char(6)	Номер рейса
departure_airport	char(3)	Код аэропорта отправления
departure_airport_name	text	Название аэропорта отправления
departure_city	text	Город отправления
arrival_airport	char(3)	Код аэропорта прибытия
arrival_airport_name	text	Название аэропорта прибытия
arrival_city	text	Город прибытия
aircraft_code	char(3)	Код самолета, IATA
duration	interval	Продолжительность полета
days_of_week	integer[]	Дни недели, когда выполняются рейсы

Рисунок 4. Содержимое материализованного представления routes

Демонстрационная база содержит временной «срез» данных — так, как будто в некоторый момент была сделана резервная копия реальной системы. Например, если некоторый рейс имеет статус Departed, это означает, что в момент резервного копирования самолет вылетел и находился в воздухе. Позиция «среза» сохранена в функции **bookings.now()**. Ей можно пользоваться в запросах там, где в обычной жизни использовалась бы функция **now()**.

#### 4.10. **Бизнес задачи, которые можно решить, используя базу данных.**

Благодаря использованию настоящей базы данных, могут быть решены следующие бизнес задачи:

- Анализ данных о задержках и переносах рейсов, выполнении рейсов согласно расписанию, составление прогнозов надежности;
- Оценка рентабельности выполнения рейсов по определенным маршрутам, корректировка их количества в зависимости от заполняемости (популярности);
- Оценка перспективных направлений перелетов при отсутствии прямых рейсов между городами;
- Составление рейтинга пассажиров по количеству бронирований для составления индивидуальных спецпредложений и формирования привилегий для клиентов, а также рейтинга надежности каждого клиента по информации о неиспользованных бронированиях мест (в случае, если пассажир не получает посадочный талон на рейс);
- Составление рейтинга самолетов по количеству перелетов, суммарной прибыли за рейс для авиакомпаний, формирующих собственный авиапарк;
- Оценка загруженности аэропорта по количеству рейсов и пассажиров в течение определенного срока для корректировки расписания или необходимости проведения реорганизации аэропорта;
- Оценка благосостояния клиентов по данным о бронировании билетов комфорт- и бизнес-класса.

## **5. Список SQL-запросов.**

SQL-запросы приведены в файле “Khilchuk\_final\_requests.sql” (Приложение 2). Описание логики выполнения приведено внутри файла в текстовых комментариях перед каждым запросом или напротив строк запроса.