

**Проектная работа по модулю**

**“SQL и получение данных”**

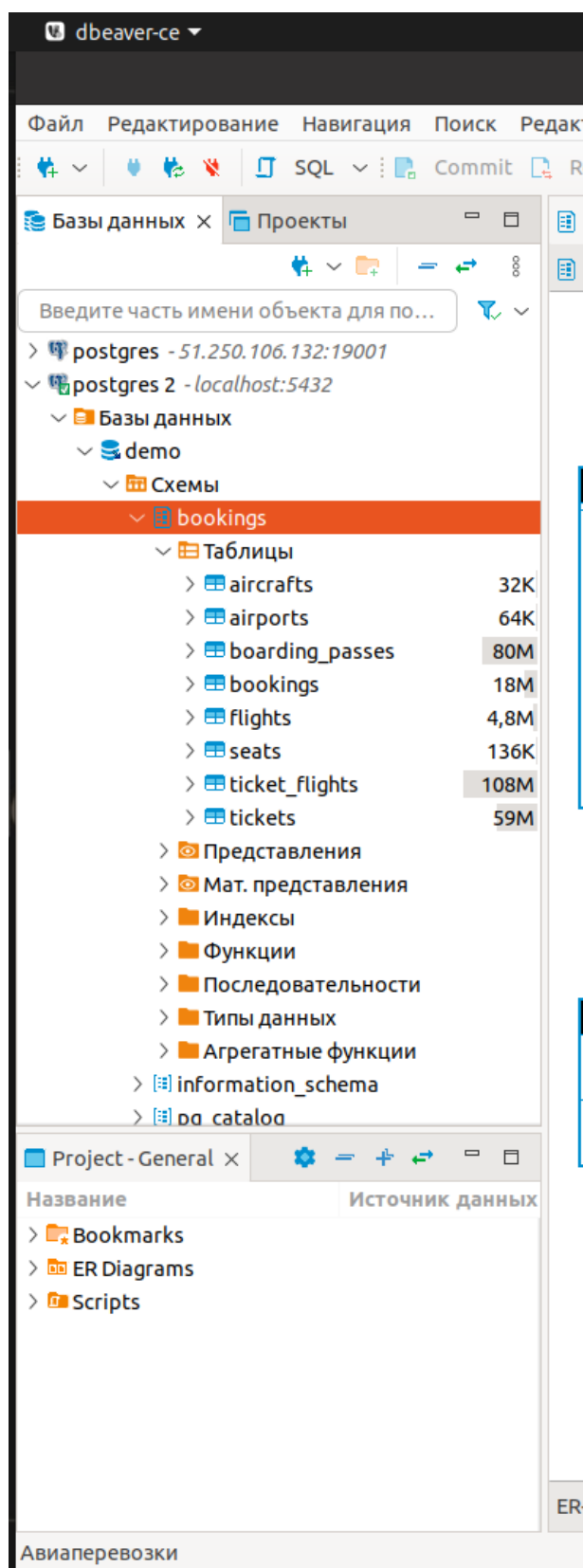


**Вертелецкая Александра**

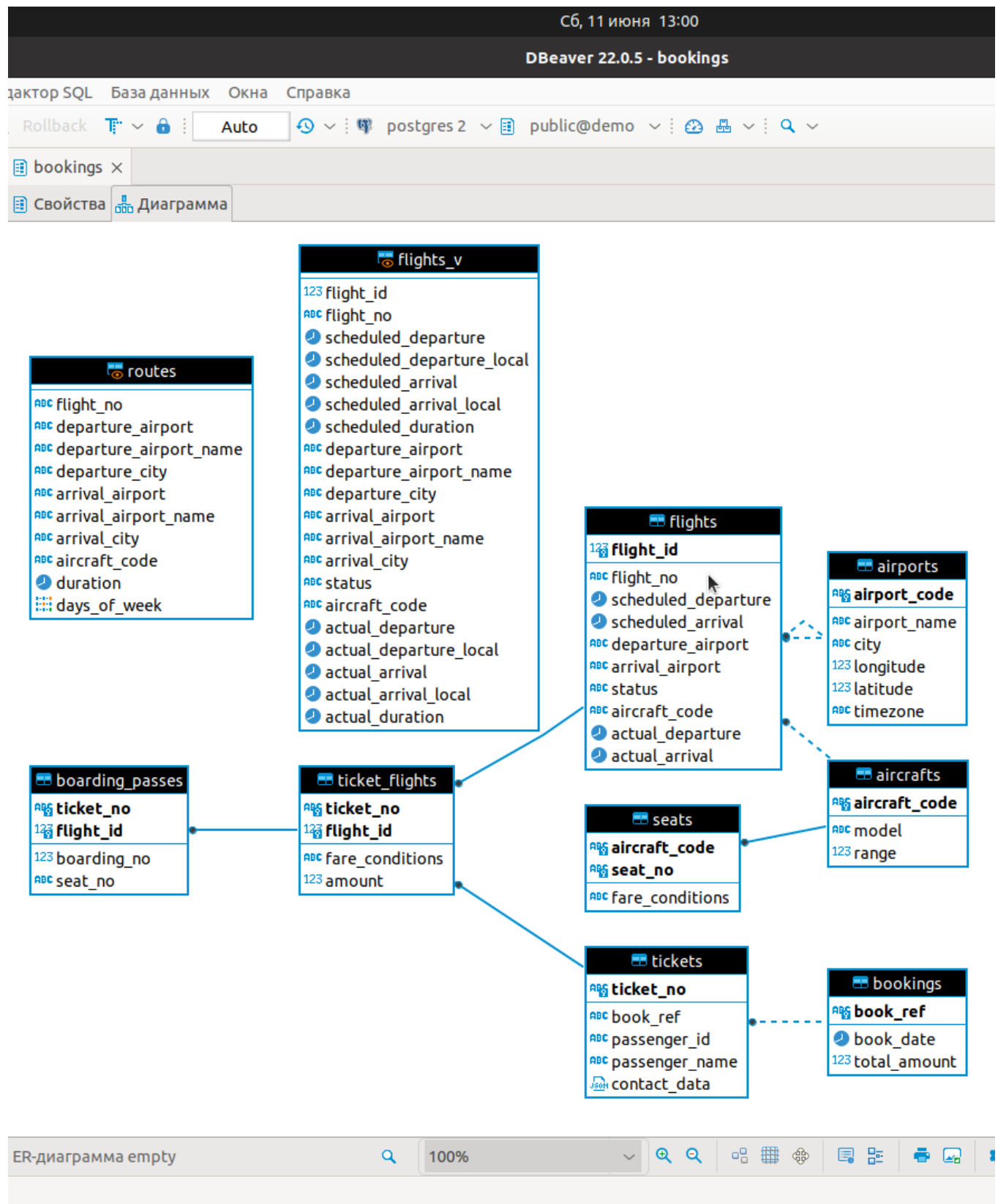
Группы SQL-41, DAU-33

Июнь, 2022

1. В работе использовался локальный тип подключения.



## 2. Скриншот ER-диаграммы из DBeaver`а согласно моего подключения.



### 3. Краткое описание БД

Таблицы:

- aircrafts (Самолёты)
- airports (Аэропорты)
- boarding\_passes (Посадочные талоны)
- bookings (Бронирования)
- flights (Рейсы)
- seats (Места)
- ticket\_flights (Перелёты)
- tickets (Билеты)

Представления:

- flights\_v (Рейсы)

Материализованные представления:

- routes (Маршруты)

#### 4. 1 Развернутый анализ БД - описание таблиц, логики, связей и бизнес области (частично можно взять из описания базы данных, оформленной в виде анализа базы данных).

##### Таблица bookings.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

##### Таблица bookings.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)

##### Таблица bookings.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)

## Таблица bookings.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)

## Таблица bookings.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 суррогатный 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
  - Рейс доступен для бронирования. Это происходит за месяц до плановой даты вылета; до этого запись о рейсе не существует в базе данных.
- On Time
  - Рейс доступен для регистрации (за сутки до плановой даты вылета) и не задержан.
- Delayed
  - Рейс доступен для регистрации (за сутки до плановой даты вылета), но задержан.
- Departed
  - Самолет уже вылетел и находится в воздухе.

- Arrived  
Самолет прибыл в пункт назначения.
- Cancelled  
Рейс отменен.
- 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)

## Таблица bookings.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

## Таблица bookings.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)

## Таблица bookings.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\_data). 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)



## 4.2 Бизнес задачи, которые можно решить, используя БД

Анализируя данные по бронированиям и перелётам из БД bookings, можно решить следующие задачи:

- Выявлять наиболее популярные маршруты
- Выявлять какие модели самолётов используются чаще остальных
- Подбирать наиболее подходящую модель самолёта для построения нового рейса, с учётом длительности перелёта и предполагаемой заполненности пассажирами
- Отслеживать бронирования, заполненность самолётов, рейсы
- Получать информацию о пассажирах по конкретным номерам билетов, бронированиям
- Выявлять какие рейсы чаще остальных отменяются
- Выявлять разницу между фактическим и плановым временем вылета/прилёта, для их дальнейшей оптимизации
- Оперативно получать информацию о существующих рейсах

5. Список SQL запросов из приложения №2 с описанием логики их выполнения.

Запросы с комментариями выполнены в отдельном sql-файле.