

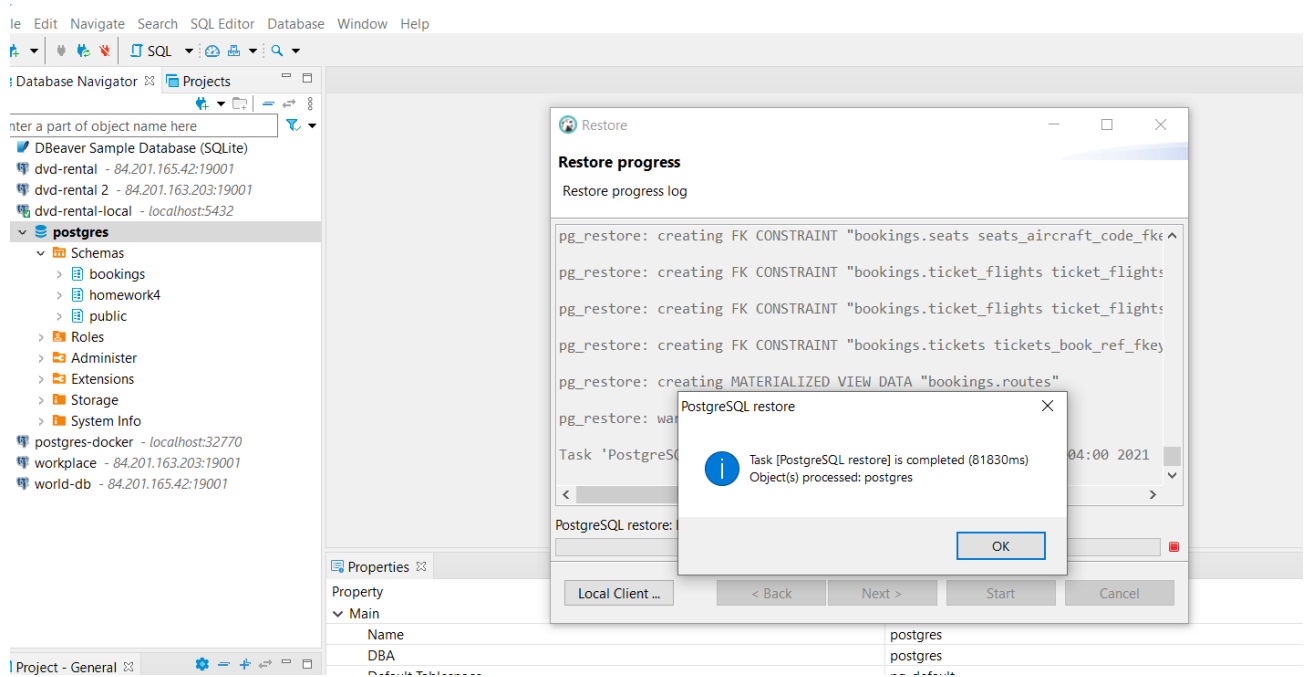
Нетология-групп

Итоговая работа по модулю “SQL и получение данных”

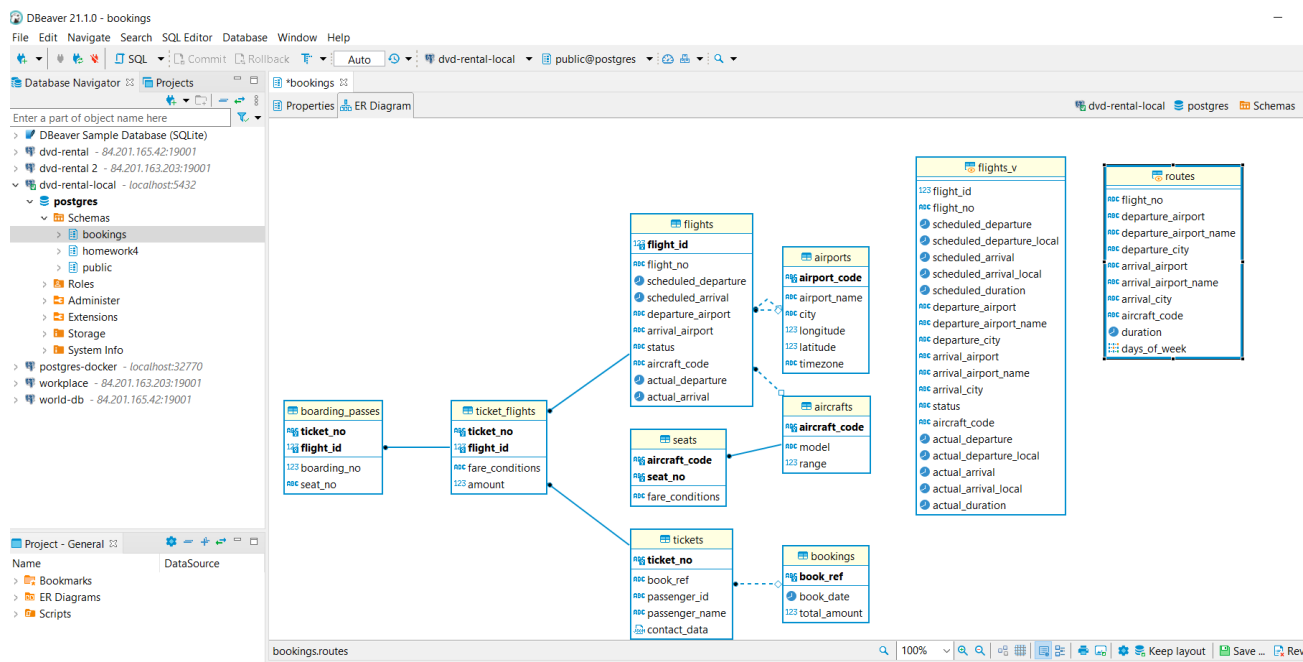
Иванов Александр
июнь, 2021

Задание можно найти по [ссылке](#)

1. В работе использовался локальный тип подключения.



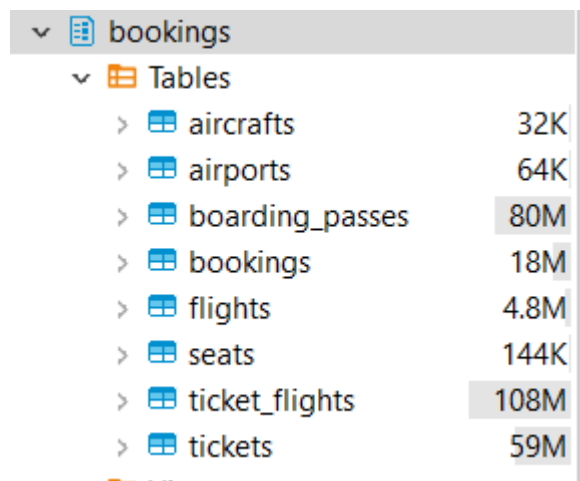
2. Скриншот ER-диаграммы из DBeaver`а согласно подключения.



3. Краткое описание БД (таблицы и представления).

Таблицы:

1. Aircrafts – код воздушного судна (ВС), модель ВС, максимальная дальность полёта (км)
2. Airports – код аэропорта, название аэропорта, город, координаты (долгота/широта), временная зона аэропорта
3. boarding_passes – номер билета, id рейса, номер посадочного, номер места
4. bookings – номер бронирования, дата бронирования, полная сумма бронирования
5. flights – id рейса, номер рейса, время вылета и прилета по расписанию, аэропорты отправления и прибытия, статус рейса, код ВС, фактическое время вылета и прилета
6. seats – код ВС, номер места, класс обслуживания
7. ticket_flights – номер билета, id рейса, класс обслуживания, стоимость перелета
8. tickets – номер билета, номер бронирования, id пассажира, ФИ пассажира, контактные данные пассажира

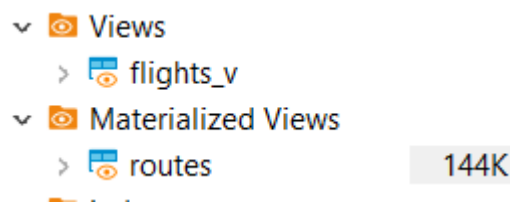


The screenshot shows a database management interface. At the top, the 'bookings' database is selected. Below it, a list of tables is displayed with their respective sizes. The tables are: aircrafts (32K), airports (64K), boarding_passes (80M), bookings (18M), flights (4.8M), seats (144K), ticket_flights (108M), and tickets (59M). The 'boarding_passes' and 'ticket_flights' tables are highlighted in grey.

Table	Size
aircrafts	32K
airports	64K
boarding_passes	80M
bookings	18M
flights	4.8M
seats	144K
ticket_flights	108M
tickets	59M

Представления

1. flights_v - идентификатор рейса, номер рейса, время вылета по расписанию + местное, время прилета по расписанию + местное, планируемая продолжительность полета, код аэропорта отправления, название аэропорта отправления, город отправления, код аэропорта прибытия, название аэропорта прибытия, город прибытия, статус рейса, код самолета, фактическое время вылета + местное, фактическое время прилета + местное, фактическая продолжительность полета
2. routes – материализованное – номер рейса, код аэропорта отправления, название аэропорта отправления, город отправления, код аэропорта прибытия, название аэропорта прибытия, город прибытия, код самолёта, продолжительность полета, дни недели, когда выполняется рейс



4. Развернутый анализ БД - описание таблиц, логики, связей и бизнес-области:

Таблица 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)*
- Внешние ключи:

TABLE "flights" FOREIGN KEY (aircraft_code) REFERENCES aircrafts(aircraft_code)

TABLE "seats" FOREIGN KEY (aircraft_code) REFERENCES aircrafts(aircraft_code) ON DELETE CASCADE

Таблица Airports:

- Аэропорт идентифицируется трехбуквенным кодом (*airport_code*) и имеет имя (*airport_name*).

Название города (*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)*

- Внешние ключи:

TABLE "flights" FOREIGN KEY (arrival_airport) REFERENCES airports(airport_code)

TABLE "flights" FOREIGN KEY (departure_airport) REFERENCES airports(airport_code)

Таблица 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:

- 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)
- Внешние ключи: TABLE "tickets" FOREIGN KEY (book_ref) REFERENCES bookings(book_ref)

Таблица flights:

- Основной 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 два аэропорта - аэропорт 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.
- 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)
- Внешние ключи: TABLE "ticket_flights" FOREIGN KEY (flight_id) REFERENCES flights(flight_id)

Таблица 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 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)
- Внешние ключи: TABLE "boarding_passes" FOREIGN KEY (ticket_no, flight_id) REFERENCES ticket_flights(ticket_no, flight_id)

Таблица 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).
- Индексы: PRIMARY KEY, btree (ticket_no)
- Ограничения внешнего ключа: FOREIGN KEY (book_ref) REFERENCES bookings(book_ref)
- Внешние ключи: TABLE "ticket_flights" FOREIGN KEY (ticket_no) REFERENCES tickets(ticket_no)

Бизнес-задачи, которые можно решить, используя БД:

1. Расчёт эффективности использования авиатранспорта исходя из дальности расстояния, средней загрузки и количества мест.
2. Увеличение либо сокращение полётов по определённым направлениям в зависимости от загруженности полёта.
3. Расчёт экономической целесообразности полёта по различным направлениям, в зависимости от выручки.
4. Увеличение эффективности работы аэропортов в плане анализа задержек рейсов.
5. Стимулирование пассажиров к повторным перелётам в виде бонусной системы на основе контактных данных пассажиров.