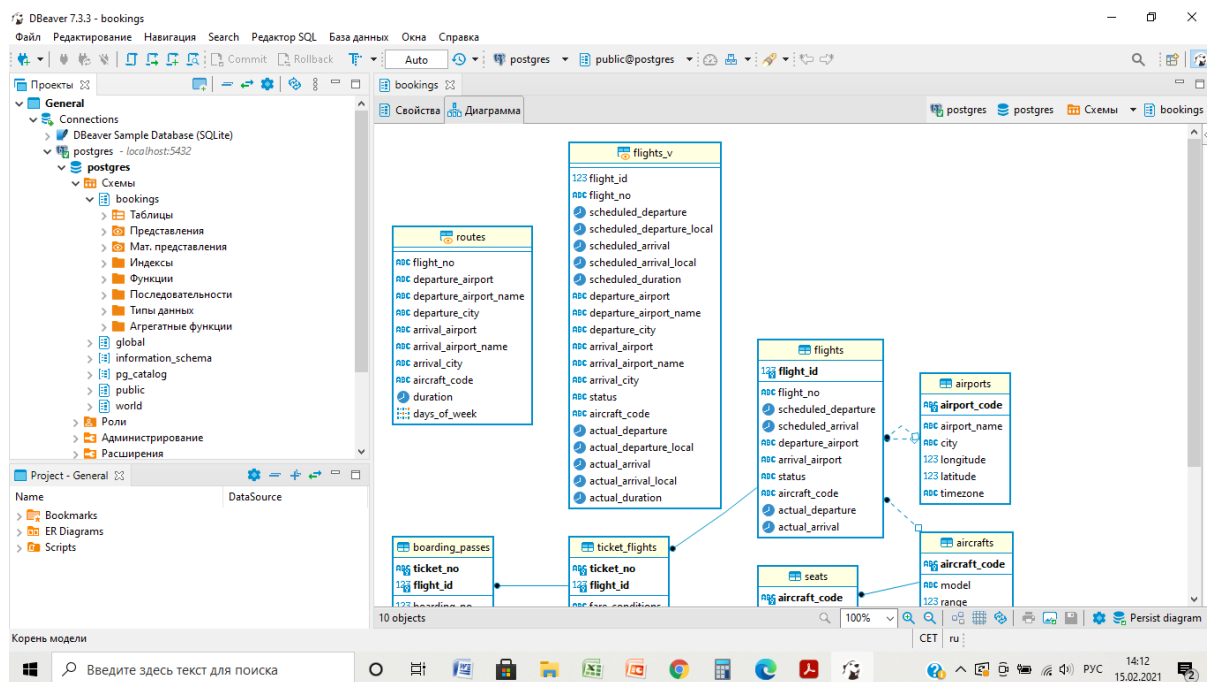


Проектная работа по модулю “SQL и получение данных”

Итоговая работа Сулеймановой Наили. Февраль 2021.

Приложение №1

1. В работе использовался **локальный** тип подключения, база была восстановлена из *.backur файла.
2. Скриншот ER-диаграммы из DBeaver`а.



3. Краткое описание БД - из каких таблиц и представлений состоит.

Имя отношения	Тип	Описание
aircrafts	таблица	Самолеты
airports	таблица	Аэропорты
boarding_passes	таблица	Посадочные талоны
bookings	таблица	Бронирования
flights	таблица	Рейсы
seats	таблица	Места
ticket_flights	таблица	Перелеты
tickets	таблица	Билеты
flights_v	представление	Рейсы с доп.информацией
routes	материальное представление	Маршруты

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

- ❑ **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). 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)
- ❑ 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.

- ❑ 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_date).
 - ❑ 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)

- ❑ **Представление "bookings.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).
- ❑ **Материализованное представление bookings.routes** - 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.

Бизнес задачи, которые можно решить, используя БД:

1. Проанализировав даты бронирования и продажи билетов, можно разработать алгоритм ценообразования (чем ближе к вылету, тем дороже билет, и чем меньше продано билетов ближе к вылету, тем меньше удорожание).
 2. Анализ времени вылета план/факт - для выявления причин задержек рейсов и их устранения.
 3. Анализ продажи билетов в зависимости от заполняемости самолета и класса обслуживания - принимать решения какой тип самолета более рентабелен для конкретного рейса/маршрута, увеличить/уменьшить кол-во кресел комфорт и бизнес класса.
 4. Анализ загруженности маршрутов - для проведения маркетинговых активностей, скидки и т.д.
 5. Проанализировать бронирования на предмет "пересадок" - для возможного открытия новых маршрутов (напр, прямые перелеты между периферийными городами, вместо пересадки в Мск).
5. Список SQL запросов из приложения №2 с описанием логики их выполнения - **приложены в SQL файле.**

