

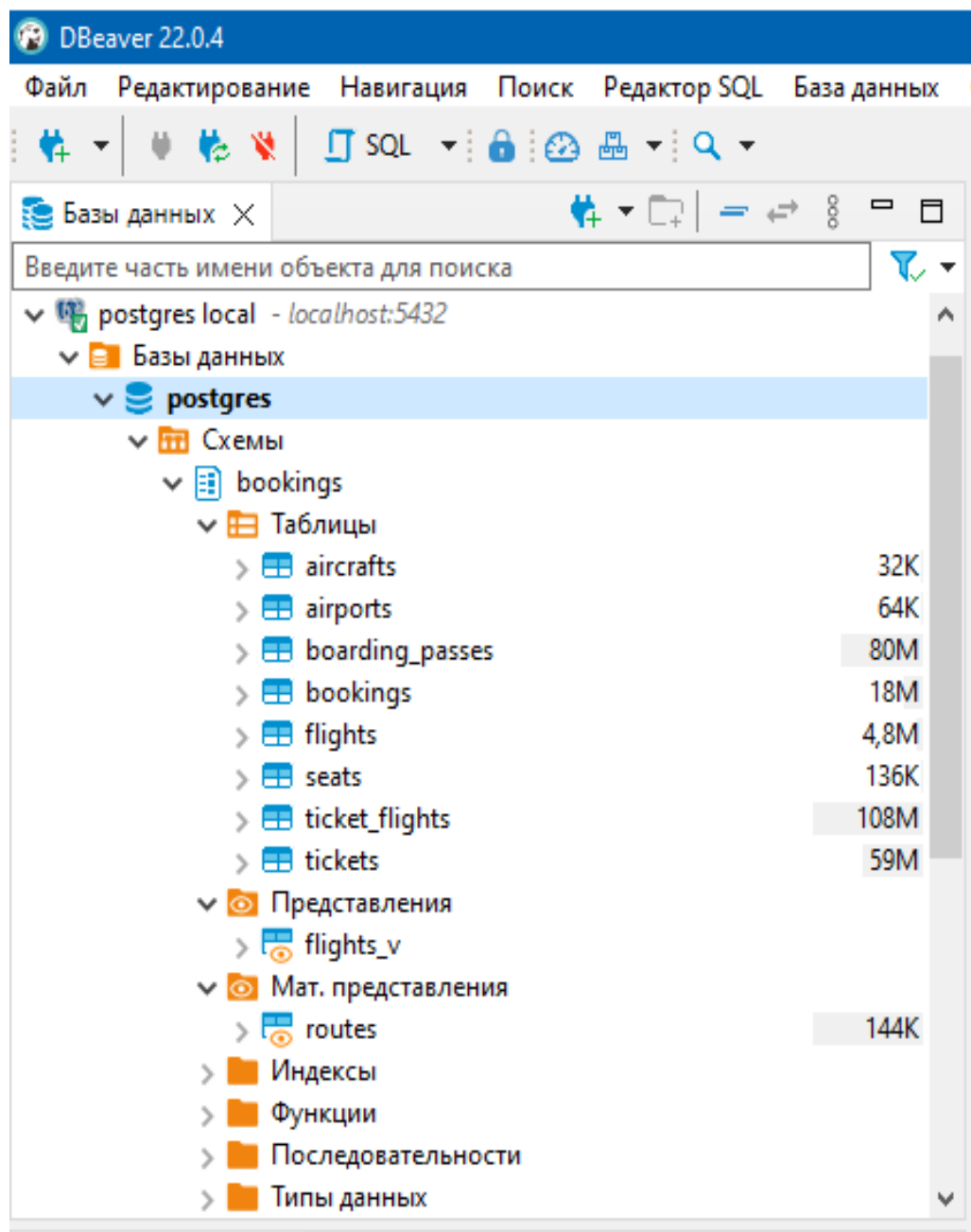
Итоговая работа

по модулю

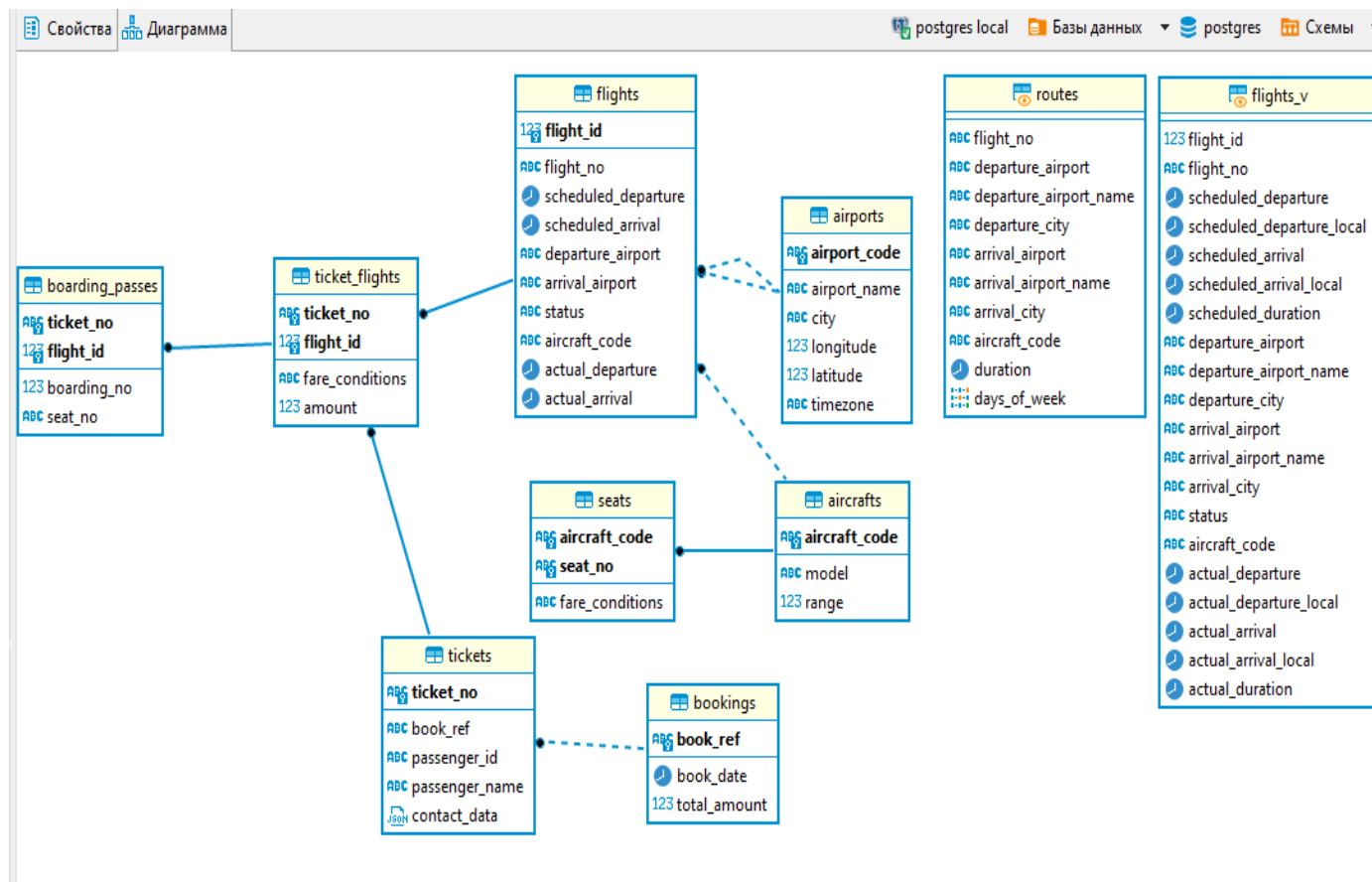
“SQL и получение данных”

В работе использовался локальный тип подключения:

- скриншот подключения:



- Скриншот ER-диаграммы:



КРАТКОЕ ОПИСАНИЕ БАЗЫ ДАННЫХ

База данных “Авиаперевозки” содержит данные по полетам по России

Список объектов базы данных:

- **Таблицы:**
 - Bookings
 - Tickets
 - Ticket_flights
 - Boarding_passes
 - Flights
 - Airports
 - Aircrafts
 - Seats
- **Представление:**
 - Flights_v
- **Материализованное представление:**
 - Routes

АНАЛИЗ БАЗЫ ДАННЫХ

I. Описание объектов

Таблица Bookings (Бронирования)

Основная сущность.

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 Tickets (Билеты)

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)

Таблица Tcket_flights (Перелеты)

Отношение билетов к перелетам.

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 Boarding_passes (Посадочные талоны)

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 Flights (Рейсы)

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.

Рейсы с одним номером имеют одинаковые пункты вылета и назначения, но будут отличаться датой отправления.

Один рейс обслуживается одним типом самолета.

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):

Scheduled - доступен для бронирования. Это происходит за месяц до плановой даты вылета; до этого запись о рейсе не существует в базе данных.

On Time - доступен для регистрации (за сутки до плановой даты вылета) и не задержан.

Delayed - 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 - отменен.

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).

- 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 Airports (Аэропорты)

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 Aircrafts (Самолеты)

Каждая модель воздушного судна идентифицируется своим трехзначным кодом (aircraft_code). Указывается также название модели (model) и максимальная дальность полета в километрах (range). Каждая модель самолета имеет только одну компоновку салона.

- 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 Seats (Места)

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

Представление Bookings.flights_v

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).

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. Именно такая информация и составляет материализованное представление routes.

II. Бизнес задачи

Информацию, которая содержится в базе данных “Авиаперевозки”, можно использовать для решения ряда бизнес задач. Комплексный анализ поможет выявить проблемы и найти пути их решения. Например:

- Анализ заполняемости рейсов ставит вопрос об эффективном использовании авиапарка, оптимальности затрат и прибыльности рейсов с низкой заполняемостью. В результате этого можно рассмотреть вопрос об отправке самолетов меньшей вместимостью на определенные направления, снизить частоту рейсов когда это невыгодно.
- Расчет расстояния между аэропортами позволит использовать маршрут с минимальным расстоянием для экономии топлива.
- Анализ количества перевезенных пассажиров позволит изучить спрос по направлениям, принять решение по сокращению или расширению количества прямых рейсов, оптимизировать рейсы с пересадками, увеличить частоту рейсов на популярных маршрутах.
- Анализ потока пассажиров по каждому аэропорту перераспределить рейсы между аэропортами тем самым сняв нагрузку с переполненных.
- Изучение пассажирских предпочтений и покупательской способности поможет правильно спланировать таргетированную рекламу. Привлечь новых клиентов и удержать старых тем самым увеличив прибыль.
- Анализ сроков задержки рейсов и изучение их закономерностей, могут привести к выявлению системных проблем в компании.