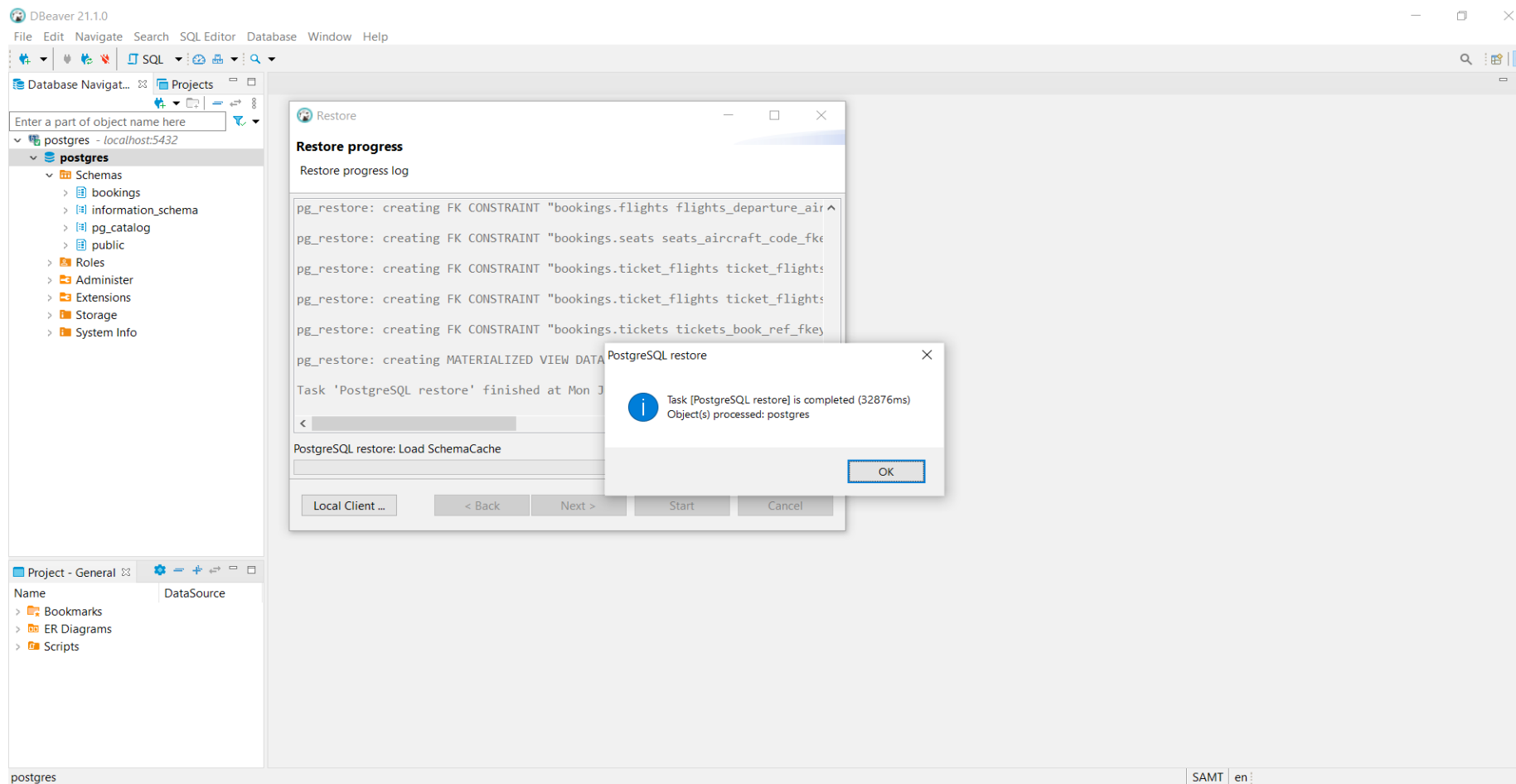


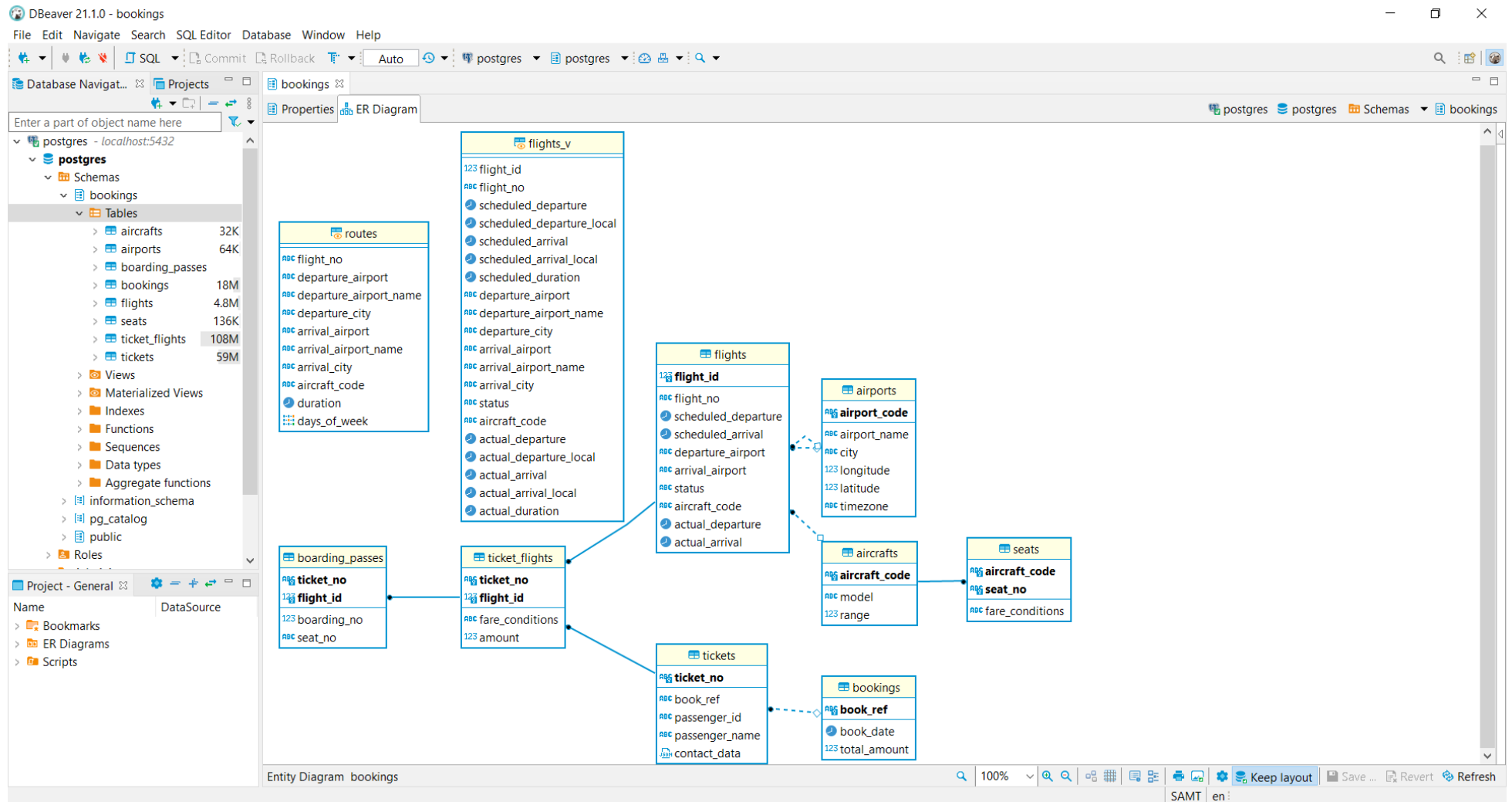
**Итоговая работа по блоку
«SQL и получение данных»**

Афремова Ильи

1 В работе использовался локальный тип подключения



2 Скриншот ER-диаграммы



3 Краткое описание БД

База данных (БД) «Авиаперевозки» содержит сведения об авиаперевозках по России за месяц. БД состоит из 8 таблиц и 2-х представлений.

| Название | Тип | Содержит сведения |
|------------------------|---------------------------------|-------------------|
| aircrafts | таблица | Самолеты |
| airports | таблица | Аэропорты |
| boarding_passes | таблица | Посадочные талоны |
| bookings | таблица | Бронирования |
| flights | таблица | Рейсы |
| flights_v | представление | Рейсы |
| routes | материализованное представление | Маршруты |
| seats | таблица | Места |
| ticket_flights | таблица | Перелеты |
| tickets | таблица | Билеты |

4 Развернутый анализ БД

4.1 Логика схемы

Основной сущностью является бронирование (bookings). В одно бронирование можно включить несколько пассажиров, каждому из которых выписывается отдельный билет (tickets). Билет имеет уникальный номер и содержит информацию о пассажире.

Как таковой пассажир не является отдельной сущностью. Как имя, так и номер документа пассажира могут меняться с течением времени, так что невозможно однозначно найти все билеты одного человека; для простоты можно считать, что все пассажиры уникальны.

Билет включает один или несколько перелетов (`ticket_flights`). Несколько перелетов могут включаться в билет в случаях, когда нет прямого рейса, соединяющего пункты отправления и назначения (полет с пересадками), либо когда билет взят «туда и обратно».

В схеме данных нет жесткого ограничения, но предполагается, что все билеты в одном бронировании имеют одинаковый набор перелетов.

Каждый рейс (`flights`) следует из одного аэропорта (`airports`) в другой. Рейсы с одним номером имеют одинаковые пункты вылета и назначения, но будут отличаться датой отправления.

При регистрации на рейс пассажиру выдается посадочный талон (`boarding_passes`), в котором указано место в самолете. Пассажир может зарегистрироваться только на тот рейс, который есть у него в билете. Комбинация рейса и места в самолете должна быть уникальной, чтобы не допустить выдачу двух посадочных талонов на одно место.

Количество мест (`seats`) в самолете и их распределение по классам обслуживания зависит от модели самолета (`aircrafts`), выполняющего рейс. Предполагается, что каждая модель самолета имеет только одну компоновку салона. Схема данных не контролирует, что места в посадочных талонах соответствуют имеющимся в самолете (такая проверка может быть сделана с использованием табличных триггеров или в приложении).

4.2 Особенности данных

Схема имеет множество ограничений и связей между таблицами. Избыточность данных минимизирована и для оптимизации работы создано отдельное материализованное представление (`routes`).

Данные таблиц не являются полными. Вероятно, из-за того, что в представленной БД содержится лишь временной срез и, например, значения атрибутов `actual_departure` и `actual_arriving` в ряде кортежей не указаны. Также, нет сведений по ряду рейсов в таблице `boarding_passes`.

Кроме того, база содержит ошибки. В частности, имеется рейс, помеченный как отмененный (Cancelled), однако, при этом у этого рейса имеются не-NULL значение атрибутов actual_departure и actual_arriving.

4.3 Применимость в бизнесе

Представленную БД можно использовать в бизнесе для:

- оптимизации расходов за счет определения наиболее и наименее загруженных маршрутов и оптимизации заполняемости самолетов (например, выбор моделей с большим или меньшим кол-вом мест);
- повышения выручки за счет предоставления более гибких цен на основе анализа востребованности того или иного класса обслуживания;
- улучшения логистики на основе анализа какой или какие наборы рейсов в билете встречаются чаще (открыть новые прямые сообщения);
- своевременного обращения внимания на проблемы в аэропортах на основе анализа задержек вылетов;
- отслеживания изменения пассажиропотока в сравнении с другими аналогичными этому временными срезами, с целью выявления факторов сезонности и пр.

4.4 Подробная структура

4.4.1 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)

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

4.4.2 Таблица *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)

4.4.3 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)

4.4.4 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)

Ссылки извне:

TABLE "tickets" FOREIGN KEY (book_ref) REFERENCES bookings(book_ref)

4.4.5 Таблица *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:

| Статус | Значение |
|------------------|---|
| 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)

4.4.6 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

4.4.7 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)

Ссылки извне:

TABLE "boarding_passes" FOREIGN KEY (ticket_no, flight_id) REFERENCES ticket_flights(ticket_no, flight_id)

4.4.8 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).

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)

4.4.9 Представление 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).

4.4.10 Материализованное представление 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.

5 Список SQL-запросов (см. приложение: файл tasks.sql)