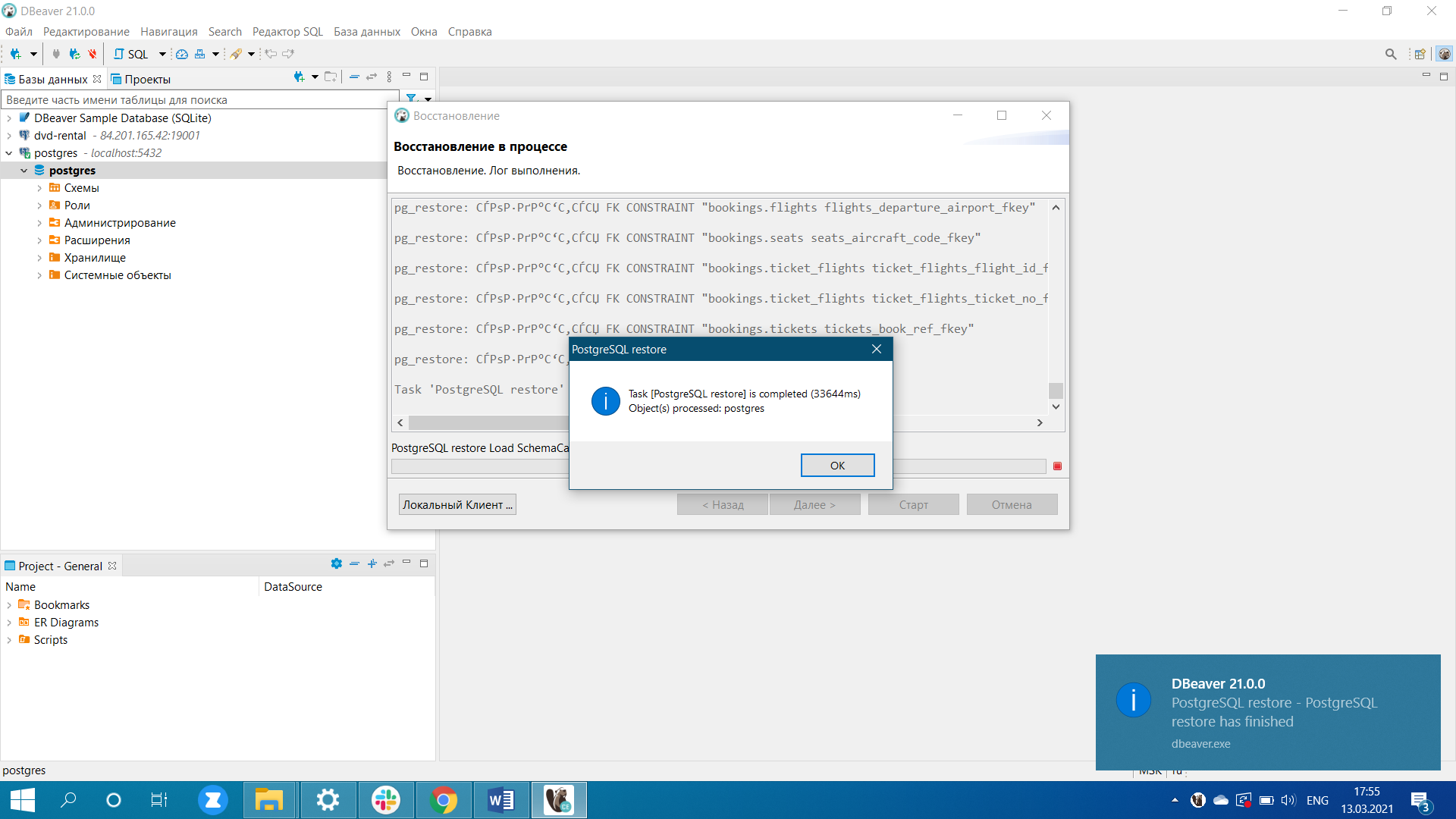
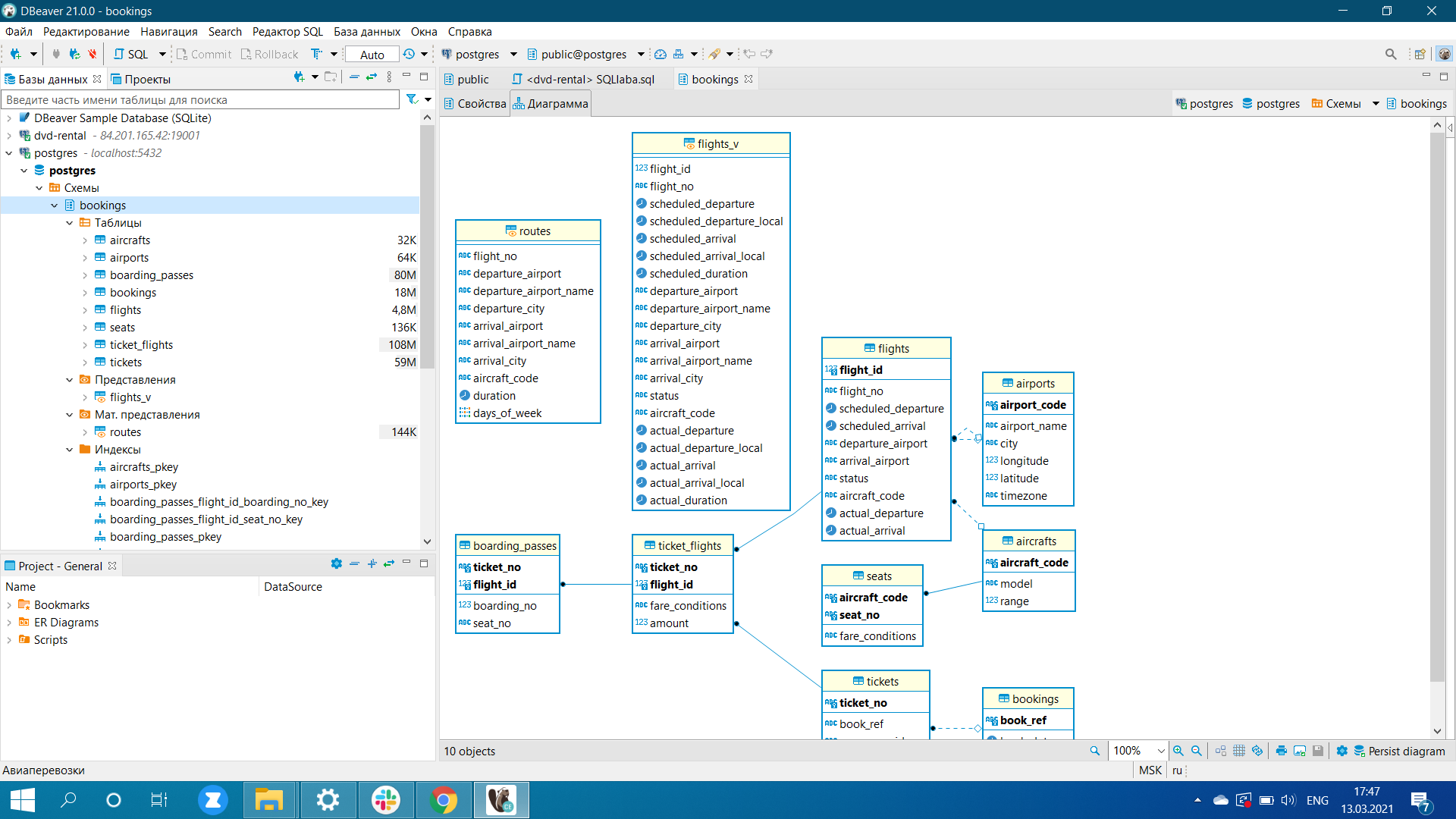
1. В работе использовался локальный тип подключения путем восстановления из backup файла



1. Скриншот ER диаграммы



3. Краткое описание БД

База данных состоит из 8 таблиц и 2 представлений. Таблицы: bookings, tickets, aircrafts, seats, airports, flights, ticket\_flights, boarding\_passes. Представления: flights\_v (view), routes (materialized view)

Таблицы:

Aircrafts – код воздушного судна (ВС), модель ВС, максимальная дальность полёта (км).

Airports – код аэропорта, название аэропорта, город, координаты (долгота/широта), временная зона аэропорта.

Boarding\_passes – номер билета, id рейса, номер посадочного, номер места

Bookings – номер бронирования, дата бронирования, полная сумма бронирования

Flights – id рейса, номер рейса, время вылета и прилета по расписанию, аэропорты отправления и прибытия, статус рейса, код ВС, фактическое время вылета и прилета

Seats – код ВС, номер места, класс обслуживания

Ticket\_flights – номер билета, id рейса, класс обслуживания, стоимость перелета

Tickets – номер билета, номер бронирования, id пассажира, ФИ пассажира, контактные данные пассажира b.

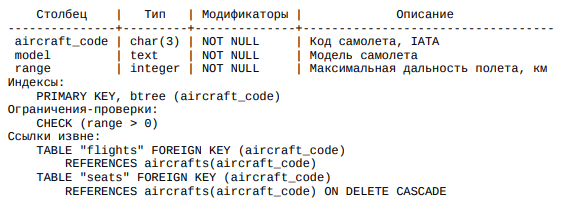
Представления:

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.

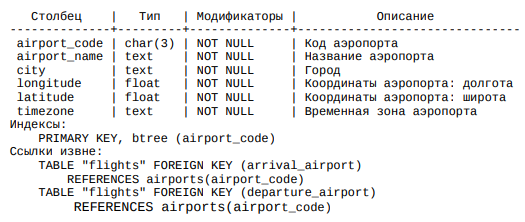
Routes – материализованное – номер рейса, код аэропорта отправления, название аэропорта отправления, город отправления, код аэропорта прибытия, название аэропорта прибытия, город прибытия, код самолёта, продолжительность полета, дни недели, когда выполняется рейс

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

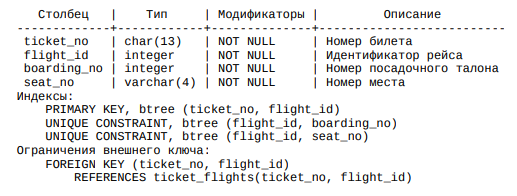
Aircrafts: Каждая модель воздушного судна идентифицируется своим трехзначным кодом (aircraft\_code). Указывается также название модели (model) и максимальная дальность полета в километрах (range). Индексы: PRIMARY KEY, btree (aircraft\_code). Ограничения-проверки: CHECK (range > 0). Ссылки извне: TABLE "flights" FOREIGN KEY (aircraft\_code) REFERENCES aircrafts(aircraft\_code) TABLE "seats" FOREIGN KEY (aircraft\_code) REFERENCES aircrafts(aircraft\_code) ON DELETE CASCADE



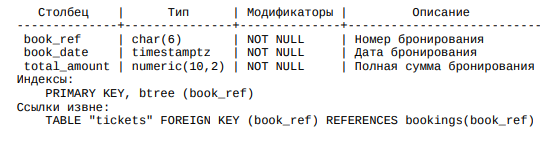
Airports: Аэропорт идентифицируется трехбуквенным кодом (airport\_code) и имеет свое имя (airport\_name). Название города (city) указывается и может служить для того, чтобы определить аэропорты одного города. Также указывается широта (longitude), долгота (latitude) и часовой пояс (timezone). Индексы: 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)



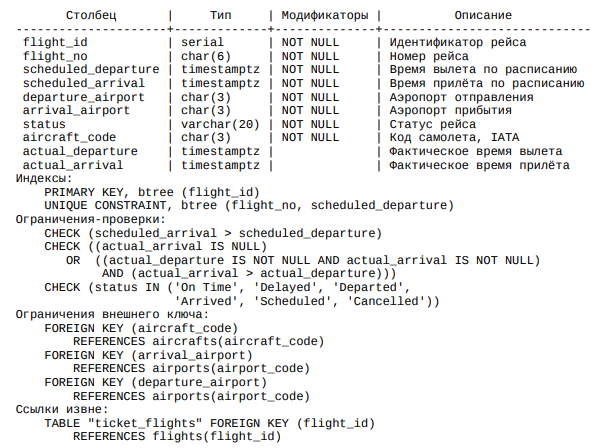
Boarding\_passes: При регистрации на рейс, которая возможна за сутки до плановой даты отправления, пассажиру выдается посадочный талон. Он идентифицируется также, как и перелет — номером билета и номером рейса. Посадочным талонам присваиваются последовательные номера (boarding\_no) в порядке регистрации 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)



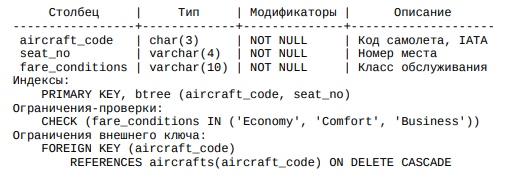
Bookings: Пассажир заранее (book\_date, максимум за месяц до рейса) бронирует билет себе и, возможно, нескольким другим пассажирам. Бронирование идентифицируется номером (book\_ref, шестизначная комбинация букв и цифр). Поле total\_amount хранит общую стоимость включенных в бронирование перелетов всех пассажиров. Индексы: PRIMARY KEY, btree (book\_ref) Ссылки извне: TABLE "tickets" FOREIGN KEY (book\_ref) REFERENCES bookings(book\_ref)



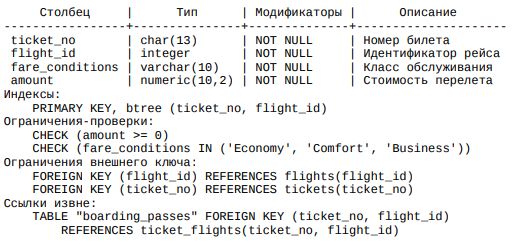
Flights: Естественный ключ таблицы рейсов состоит из двух полей — номера рейса (flight\_no) и даты отправления (scheduled\_departure). Чтобы сделать внешние ключи на эту таблицу компактнее, в качестве первичного используется суррогатный ключ (flight\_id). Рейс всегда соединяет две точки — аэропорты вылета (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)



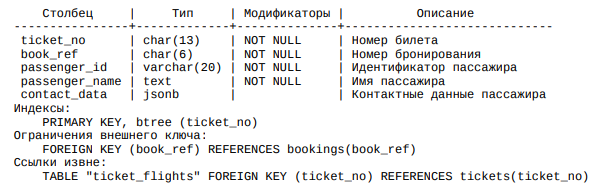
Seats: Места определяют схему салона каждой модели. Каждое место определяется своим номером (seat\_no) и имеет закрепленный за ним класс обслуживания (fare\_conditions) — Economy, Comfort или Business. Индексы: PRIMARY KEY, btree (aircraft\_code, seat\_no) Ограничения-проверки: CHECK (fare\_conditions IN ('Economy', 'Comfort', 'Business')) Ограничения внешнего ключа: FOREIGN KEY (aircraft\_code) REFERENCES aircrafts(aircraft\_code) ON DELETE CASCADE



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)



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)



Бизнес – задачи, которые можно решить, используя БД:

- Посчитать количество людей, которые летят до определенного города с пересадками с целью запустить отдельный прямой рейс, если количество таких пассажиров в среднем заполнит целый самолет и подобный рейс будет экономически выгодным

- Посчитать соотношение бронированных билетов к посадочным талонам, чтобы выявить средний процент пассажиров, не явившихся на рейс (Продавать больше билетов на рейс, чтобы полностью заполнить самолет даже с учетом неявившихся пассажиров. Сейчас такая практика часто встречается в авиакомпаниях)

- Посчитать соотношение броней бизнес класса и эконом класса, чтобы регулировать спрос-предложение и ценообразование в данной области (Допустим, что места эконом-класса в самолете заполнены почти на 100%, что говорит о том, что есть спрос на данный рейс, в то время как места бизнес-класса пустеют. Это означает, что необходимо провести анализ затрат на обслуживание бизнес-класса и провести изменение ценовой политики на данную категорию перевозок)

- Посчитать соотношение пассажиров рейса в целом к вместительности самолета с целью возможности замены самолета на меньший (в случае если самолет не заполняется хотя бы на 75%) или больший (в случае если заполняемость самолета стремится к 100%, что говорит о том, что на данный рейс имеется высокий спрос) (Замена на меньший с целью экономии средств на топливо и обслуживание самолета. Замена на больший с целью удовлетворить спрос на данное направление и, соответственно, увеличить прибыль)

- Провести тестирование. Будет ли увеличенная продолжительность рейса сказываться на спросе на билеты (Если нет, то можно увеличивать продолжительность полета, что будет выгодно, так как будет тратиться меньше топлива самолета)

- Посмотреть влияет ли время вылета и прилета на продажи билетов (запустить рейс в дневное время и посмотреть заполняемость самолета, затем запустить в ночное. Результаты сравнить и выбрать наиболее экономически выгодный)

- Посмотреть влияет ли день недели на продажи билетов (Подбирать самолеты побольше в дни ажиотажа и поменьше в дни послабления спроса)

- Сравнение статистики по общему количеству / сумме продаж / рейсов в месяц / год чтобы подстраиваться под спрос (Например, организовать большее количество рейсов в летнее время, так как многие улетают за границу в отпуск. На основании статистики прошлых периодов выявить необходимое количество рейсов для максимального извлечения прибыли)

- Вывести соотношение общих стоимостей брони рейса на 1 км пути, чтобы вычислить наиболее коммерчески выгодные рейсы / направления

- Выяснить, влияет ли модель самолета на продажи (Например, на рейсе одного направления, при одинаковых условиях, присутствуют 2 перевозчика. Если у второго перевозчика самолет лучше, то попробовать заменить самолет на тот, где будут лучшие условия по сравнению с другим перевозчиком. Если ситуация не меняется, то обратиться к другим показателям (Ценовая политика, маркетинг, время и даты и т. п.))

5. Список SQL запросов с описанием логики их выполнения: Комментарии с описанием логики даны в самих решениях, прилагаемых к данному отчёту.