ИТОГОВАЯ РАБОТА

По модулю «SQL и получение данных»

Выполнил студент:

Группа: SQL-41

Степанов Сергей Васильевич

Санкт-Петербург

2022 год

Оглавление

[Приложение №1 3](#_Toc107228341)

[1. Тип подключения. 3](#_Toc107228342)

[2. Скриншот ER-диаграммы. 4](#_Toc107228343)

[3. Краткое описание БД - из каких таблиц и представлений состоит. 4](#_Toc107228344)

[4.1 Развернутый анализ БД - описание таблиц, логики, связей и бизнес области (частично можно взять из описания базы данных, оформленной в виде анализа базы данных): 5](#_Toc107228345)

[4.2. Бизнес задачи, которые можно решить, используя БД: 8](#_Toc107228346)

[5. Список SQL запросов из приложения №2 с описанием логики их выполнения. 9](#_Toc107228347)

[Приложение №2 10](#_Toc107228348)

Приложение №1

1. Тип подключения.

В работе использовался локальный тип подключения:

Рисунок 1. «Лог процесса восстановления»

Изображение выглядит как текст

Автоматически созданное описание

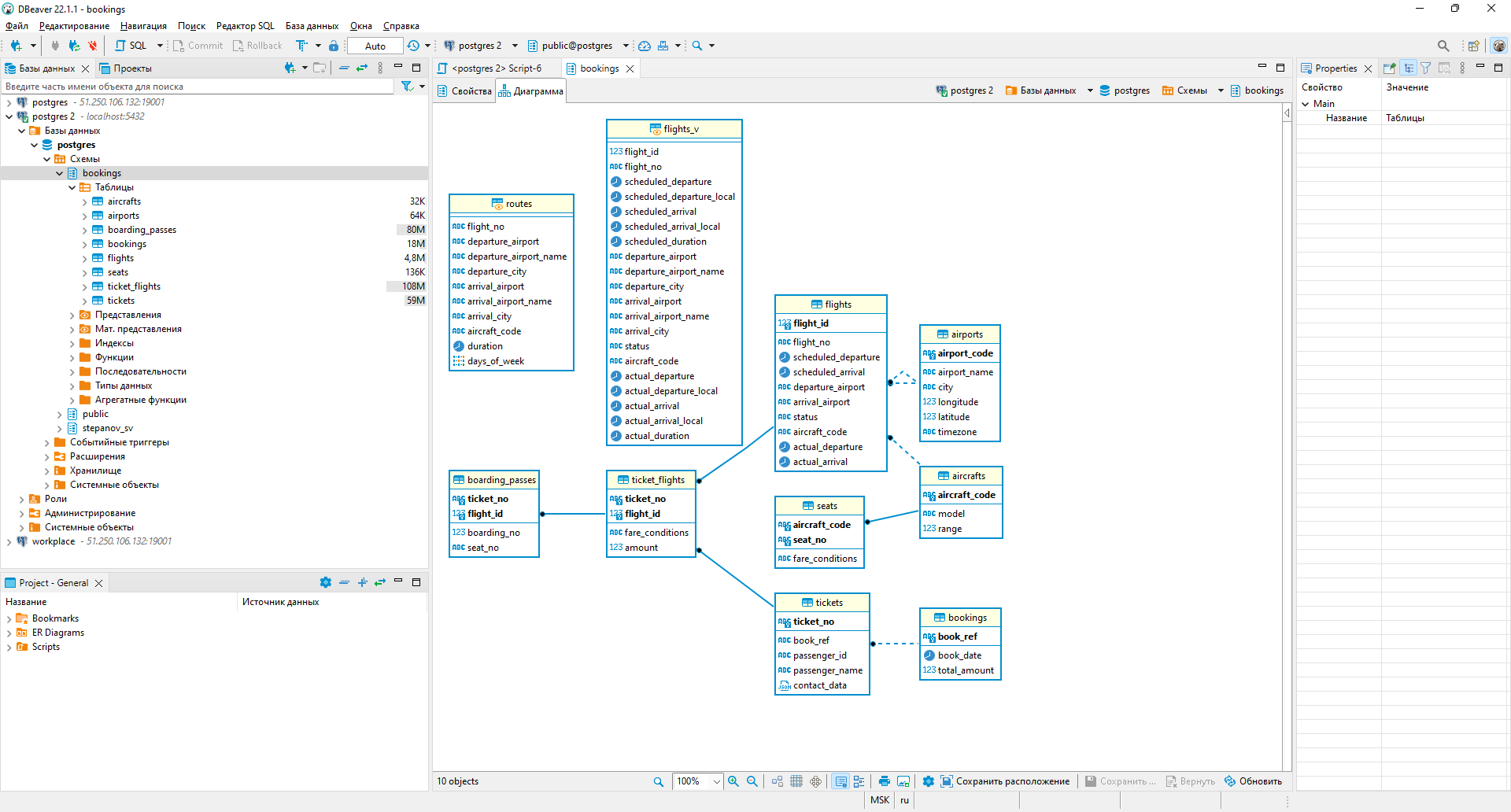
Рисунок 2. «Общий вид на рабочий стол DBeaver`a с восстановленной базой»

Изображение выглядит как текст

Автоматически созданное описание

2. Скриншот ER-диаграммы.

Рисунок 3. «Скриншот ER- диаграммы из DBeaver`a»



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

3.1 База включает 8 таблиц:

3.1.1 boarding\_passes – состоит из столбцов: номер билета, идентификатор рейса, номер посадочного талона, номер места;

3.1.2 ticket\_flights – состоит из 4 столбцов: номер билета, идентификатор рейса, класс обслуживания, стоимость перелета;

3.1.3 flights – состоит из 10 столбцов: идентификатор рейса, номер рейса, время вылета по расписанию, время прилета по расписанию, аэропорт отправления, аэропорт прибытия, статус рейса, код самолета, фактическое время вылета, фактическое время прилета;

3.1.4 seats – состоит из 3 столбцов: код самолета, номер места, класс обслуживания;

3.1.5 tickets – состоит из 5 столбцов: номер билета, номер бронирования, идентификатор пассажира, имя пассажира, контактные данные пассажира;

3.1.6 airports – состоит из столбцов: код самолета, модель самолета и максимальная дальность полета (км);

3.1.7 aircrafts – состоит из столбцов: код аэропорта, название аэропорта, город, координаты аэропорта (долгота), координаты аэропорта (широта), временная зона аэропорта;

3.1.8 bookings – состоит из столбцов: номер бронирования, дата бронирования, полная сумма бронирования.

3.2 База данных включает 2-а представления:

3.2.1 flights\_v – состоит из 20 столбцов: идентификатор рейса, номер рейса, время вылета по расписанию, время вылета по расписанию (местное время в пункте отправления), время прилёта по расписанию, время прилёта по расписанию (местное время в пункте прибытия), 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, фактическое время вылета (местное время в пункте отправления), фактическое время прилёта, фактическое время прилёта (местное время в пункте прибытия), фактическая продолжительность полета;

3.2.2 routes – состоит из 10 столбцов: номер рейса, код аэропорта отправления, название аэропорта отправления, город отправления, код аэропорта прибытия, название аэропорта прибытия, город прибытия, код самолета, продолжительность полета, дни недели, когда выполняются рейсы

4.1 Развернутый анализ БД - описание таблиц, логики, связей и бизнес области (частично можно взять из описания базы данных, оформленной в виде анализа базы данных):

4.1.1 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).

4.1.2 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).

4.1.3 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:

• 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).

4.1.4 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.

4.1.5 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).

– 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).

4.1.6 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).

4.1.7 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.

4.1.8 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)

4.1.9. 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).

4.1.10 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.

4.2. Бизнес задачи, которые можно решить, используя БД:

Исходя из представленных данных я могу предположить следующие бизнес-задачи, которые можно решить:

4.2.1 Определить загруженность самолетов пассажирами, что позволит оценить экономический эффект от перевозок;

4.2.2 Определить, по каким рейсам были задержки вылетов, их динамику и продолжительность. Это поможет выявить закономерности подобных задержек и минимизировать их в будущем, что также позволит получить дополнительный экономический эффект (сокращение простоев самолетов) и социальный (снизит недовольство пассажиров);

4.2.3 Определить направления, куда предпочитают летать бизнес-классом, для увеличения посадочных мест, что позволит получить дополнительный экономический эффект (увеличение стоимости перелета;

4.2.4 Определить направления, где перелет бизнес-классом стоит дешевле или наравне с эконом-классом. Позволит исправить цену и получить дополнительный экономический эффект.

P.S. Я уверен, что задач может решаться гораздо больше, описанных выше, т.к. я не затрагиваю техническую сторону вопроса (определение оптимальных транспортных маршрутов, использование разных типов самолетов и многое другое).

5. Список SQL запросов из приложения №2 с описанием логики их выполнения.

Список SQL запросов направлен отдельным файлом.

Приложение №2