|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ \_\_Лингвистика\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА Романо-германские языки\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К КУРСОВОМУ ПРОЕКТУ***

***НА ТЕМУ:***

***«База данных туристического агентства»***

Студент Л4-71 **\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_**  Е. В. Звягинцева**\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель курсового проекта **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_**А.Б. Домрачева\_\_\_\_\_

(Подпись, дата) (И.О.Фамилия)

Консультант **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О.Фамилия)

*2020*

**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технический университет имени Н.Э. Баумана**

**(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

УТВЕРЖДАЮ

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_

(Индекс)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(И.О.Фамилия)

« \_\_\_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_ 20 \_\_\_\_ г.

**ЗАДАНИЕ**

**на выполнение курсового проекта**

по дисциплине \_\_Базы данных\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Студент группы \_\_Л4-71\_\_\_\_\_\_\_\_

\_\_Звягинцева Евгения Владимировна\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, имя, отчество)

Тема курсового проекта \_\_База данных туристического агентства \_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Направленность КП (учебный, исследовательский, практический, производственный, др.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Источник тематики (кафедра, предприятие, НИР) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

График выполнения проекта: 25% к нед., 50% к нед., 75% к нед., 100% к нед.

***Задание***  Обзор литературы, описание предметной области. Разработка инфологической модели предметной области (ER-диаграмма). Разработка даталогической модели предметной области (реляционная модель). Реализация баз данных и организация запросов для получения нужных  
пользователю сведений из баз данных. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Оформление курсового проекта:***

Расчетно-пояснительная записка на \_36\_\_\_ листах формата А4.

Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата выдачи задания « \_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.

**Руководитель курсового проекта**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_А.Б. Домрачева\_\_\_\_\_

(Подпись, дата) (И.О.Фамилия)

**Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_**Е.В. Звягинцева**\_\_\_**

(Подпись, дата) (И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

**СОДЕРЖАНИЕ**

Введение ………………………………………………………………….…..5

Глава 1. Инфологическое проектирование……………………….....………7

1.1 Обзор литературы…………………………………………………………7

1.2 Анализ предметной области …………………………………….……….7  
1.3 Входные данные………………………………………………………….8  
Глава 2. Даталогическое проектирование……...……………….………….12

Глава 3. Физическое проектирование……………..……………………….16

Заключение …………………………………………………………….……25

Список литературы …………………………………………………..……..26

ПРИЛОЖЕНИЕ А…………………………………………………….……..27

**ВВЕДЕНИЕ**

В настоящее время жизнь человека зависит от различного рода информации, для управления которой требуются создания огромного количества баз и банков данных различного назначения. Последнее десятилетие связано с развитием информационных систем различного вида предназначений, например, транспортные системы, электронные дневники для школьников или студентов, различные виды регистратуры и т.д. Использование компьютерной техники и всевозможных устройств глубоко проникло в повседневную жизнь людей. В связи с этим, все большую актуальность и широкое распространение получают базы данных (БД) и системы управления базами данных (СУБД), которые используются для обработки разного рода информации.

Базы данных (БД)- это информационная модель, позволяющая хранить данные в структурированном виде. Цель базы данных — помочь людям и организациям вести учет определенных вещей. Под предметной областью принято понимать некоторую область человеческой деятельности или область реального мира, подлежащих изучению для организации управления и автоматизации, например, предприятие, вуз и.т.д.

Современные компании и организации вынуждены функционировать в условиях большого объема постоянно изменяющейся информации, которую необходимо оперативно анализировать. Как большие корпорации, так и маленькие предприятия зависят и от уровня развития IT-технологий, скорости и качества обработки информации, чтобы быть конкурентоспособными и увеличивать прибыль.

**Цель работы**: проектирование объектов базы данных средствами СУБД.

Для достижения поставленной цели были определены следующие **задачи** исследования:

1. Изучить научную литературу по заданной теме;
2. Проанализировать предметную область, определить цели и пользователей;
3. Осуществить реализацию спроектированных объектов базы данных средствами СУБД.

**Глава 1. Инфологическое проектирование**

Предполагается, что на этом этапе база данных и её предметная область описываются на наиболее высоком уровне абстракции.

Задачами этапа являются: определение название предметной области с указанием цели создания БД, указание пользователей БД, проектирование отчета, построение ER-диаграмм: сущности БД должны быть выбраны в соответствии с информацией, хранимой в отчете*.*

Описание каждой сущности производится при помощи данных, называемых атрибутами или свойствами сущности. Сущности имеют связи друг с другом при посредстве включенных в них атрибутов.

* 1. **Обзор литературы**

В рамках данного курсового проекта была использована литература, посвященная разработке и созданию баз данных.

Хорошая теоретическая база позволяет избежать ошибок в написании работы, а так же правильно организовать работу БД и хранения в ней данных. Проектирование баз данных требует четкого понимания различных моделей, на каких этапах и для чего применяется та или иная модель, какие бывают сущности и атрибуты.

В процессе написания работы были непосредственно использованы различные учебные и справочные материалы, а так же электронные ресурсы и руководства.

* 1. **Анализ предметной области**

В качестве предметной области данного курсового проекта рассматривается база данных «Туристического агентства», занимающегося продажей туров. Туризм представляет собой одну из самых крупных и динамичных отраслей экономики.

Данная таблица данных будет в основном использоваться сотрудниками фирмы, например, для финансовых отчетов, быстрого поиска или составления дальнейших планов по развитию новых направлений.

Сначала формируется база с турами, в которую входит:

1. Номер тура
2. Название
3. Цена
4. Дата
5. Тип
6. Клиент

Работа с клиентами компании организована следующим образом: у каждого клиента, собираются некоторые стандартные данные и присваивается тур:

1. Фамилия Имя Отчество
2. Паспорт
3. Номер телефона
4. E-mail

После этого сотрудники выясняют у клиента, куда он хотел бы поехать отдыхать, и составляют договор. Далее, если клиент приобретает тур, сотрудники заносят данные в графу "продажи", где указывается номер договора и дата продажи. У данного турагентства есть несколько отделений, работающие в разных городах.

**1.3 Входные данные**

Проектирование базы данных предусматривает входные данные, которые представлены в следующих таблицах:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tour\_id | Tour\_name | Client\_id | Price | Type\_of\_tour |
| 1 | Дублин | 64116690 | 400$ | языковые курсы |
| 2 | Мюнхен | 76159087 | 550$ | шоппинг |
| 3 | Рим | 54126865 | 320$ | байк тур |
| 4 | Копенгаген | 98156756 | 600$ | музеи |
| 5 | Прага | 13166546 | 290$ | эконом тур |
| 6 | Амстердам | 63164686 | 650$ | активный отдых |
| 7 | Будапешт | 54146577 | 500$ | музыкальный фестиваль |
| 8 | Лос- Анджелес | 78197827 | 2000$ | путешествие по штату |
| 9 | Иераклион | 23126782 | 510$ | экологический туризм |
| 10 | Барселона | 45109383 | 700$ | гастрономический |

Таблица №1 «Tour»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Client\_id | C\_name | C\_email | C\_phone | Tour\_id |
| 64116690 | Дарья Владимировна Лебедева | [daria@gmail.com](mailto:daria@gmail.com) | 89273345676 | 1 |
| 76159087 | Денис Владимирович Гонтарь | [denis@gmail.com](mailto:denis@gmail.com) | 84997856356 | 2 |
| 54126865 | Артем Романов Ненашев | [artem@gmail.com](mailto:artem@gmail.com) | 85674567433 | 3 |
| 98156756 | Анастасия Игоревна Миланова | [anastacia@gmail.com](mailto:anastacia@gmail.com) | 89276656743 | 4 |
| 13166546 | Полина Дмитриевна Шишкина | [polina@gmail.com](mailto:polina@gmail.com) | 89108769022 | 5 |
| 63164686 | Владимир Алесеевич Пушкан | [vladimir@gmail.com](mailto:vladimir@gmail.com) | 87651542371 | 6 |
| 54146577 | Ангелина Сергеевна Дантес | [angelina@gmail.com](mailto:angelina@gmail.com) | 89456783456 | 7 |
| 78197827 | Ярослав Олегович Морозов | [yaroslav@gmail.com](mailto:yaroslav@gmail.com) | 89233245643 | 8 |
| 23126782 | Елена Владимировна Волкова | [elena@gmail.com](mailto:elena@gmail.com) | 89031267832 | 9 |
| 45109383 | Александра Игоревна Гусева | [alex@gmail.com](mailto:alex@gmail.com) | 89026252826 | 10 |

Таблица №2 «Client»

|  |  |  |  |
| --- | --- | --- | --- |
| Emp\_id | E\_name | Salary | E\_email |
| 34678799 | Ирина Владимировна Мартышкина | 1500 | irinaangent@gmail.com |
| 40155342 | Ольга Дмитриевна Войкова | 1550 | olgaangent@gmail.com |
| 78993212 | Екатерина Сергеевна Самаркина | 1600 | ekaterinaangent@gmail.com |
| 64116999 | Анастасия Оскаровна Вулах | 2000 | anastasiiaangent@gmail.com |
| 77152087 | Елизавета Алексеевна Екатеринина | 2500 | elizavetaangent@gmail.com |
| 94126875 | Николай Борисов Никитин | 1500 | nikolayangent@gmail.com |
| 98346756 | Даниил Игоревич Бандарчук | 1600 | daniilangent@gmail.com |
| 13166648 | Константин Дмитриевич Волков | 2100 | konsatntinangent@gmail.com |
| 64164686 | Михаил Ильич Киров | 3000 | michailangent@gmail.com |
| 54147877 | Владимир Робертович Денисов | 1780 | vladimirangent@gmail.com |

Таблица №3 «Employee»

|  |  |
| --- | --- |
| branch\_id | location |
| 200 | Moscow |
| 201 | Saint Petersburg |
| 202 | Kazan |
| 203 | Samara |
| 204 | Ekaterenburg |
| 205 | Vladivostok |
| 206 | Sochi |
| 207 | Murmansk |
| 208 | Saratov |
| 209 | Ufa |

Таблица №4 «Branch»

Средством моделирования предметной области на этапе концептуального проектирования является модель "сущность–связь". Часто ее называют ER-моделью (Entity – сущность, Relation – связь).

Построим ER-диаграмму:

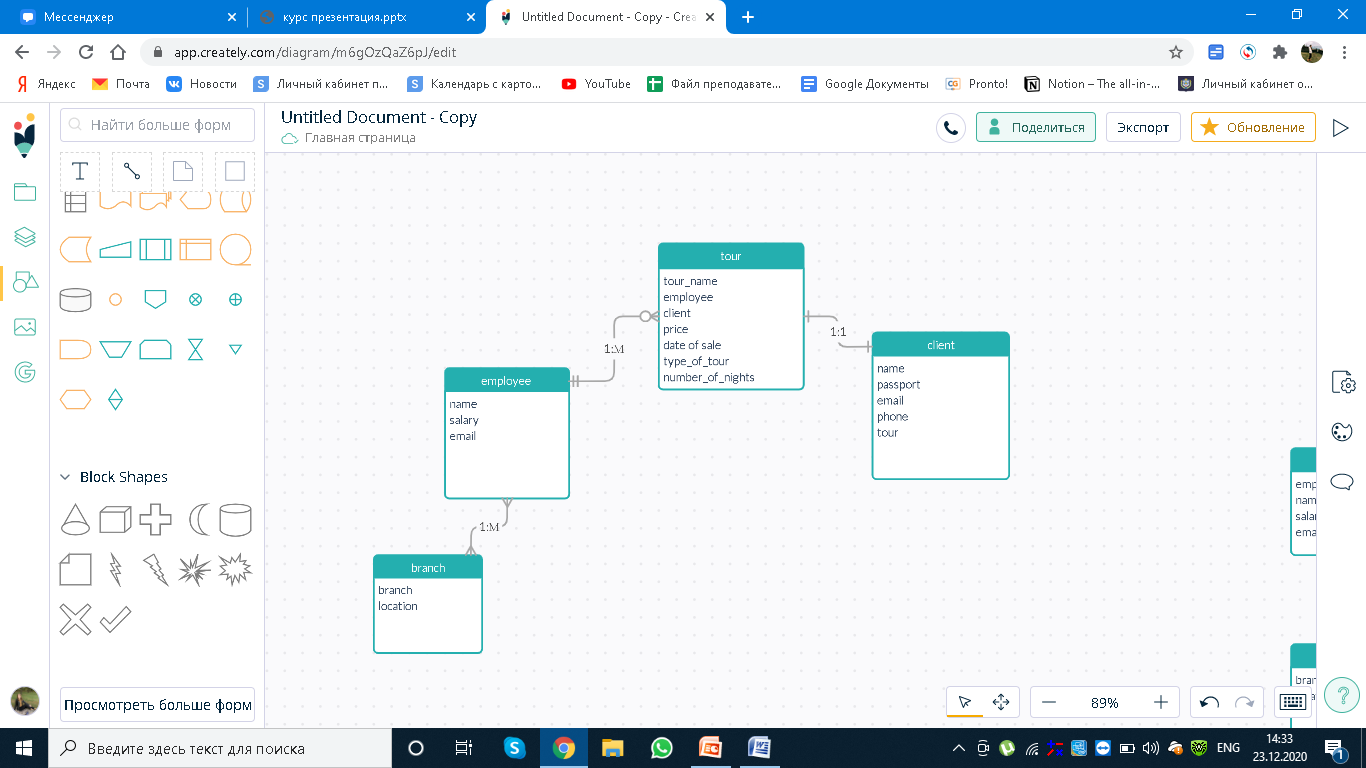


Рисунок №1. Результат инфологического проектирования. ER-диаграмма

**Глава 2 Даталогическое проектирование**

Даталогическое проектирование — создание схемы базы данных на основе конкретной модели данных, например, реляционной модели данных. Для реляционной модели данных даталогическая модель — набор схем отношений, обычно с указанием первичных ключей, а также «связей» между отношениями, представляющих собой внешние ключ. В результате получаем реляционную модель.

Сначала проведем трансформацию связей:

Tour и Client. Тип связи «один — к — одному» (1:1)

В данном случае используем первое правило. Принадлежность обязательна, отношение остается неизменным. Первичным ключом может быть ключ одной из сущностей.

Tour и Sales. Тип связи « много — к — одному» (М:1)

В данном случае может быть много туров, прибыль собирается в одну таблицу по каждому туру.

Принадлежность обязательна. При степени связи 1:М (М:1) между сущностями и обязательном классе принадлежности М-связной сущности достаточно сформировать 2 отношения (каждое отношение соответствует сущности). Первичными ключами таких отношений будут ключи этих сущностей. Кроме того, в отношение, которое соответствует М-связной сущности, нужно добавить в качестве атрибута (внешнего ключа) ключ односвязной сущности.

Employee и Tour . Тип связи « много — к — одному» (М:1)

Одним туром могут наниматься несколько сотрудников. Принадлежность необязательна, используем 5 правило. При степени связи 1:М (или М:1) и необязательном классе принадлежности М-связной сущности нужно сформировать 3 отношения. Два из них будут соответствовать обоим сущностям, ключи которых будут первичными для этих отношений. Третье отношение «Tour\_Employee» связывает первые два и имеет ключ, который объединяет ключевые атрибуты первых двух отношений.

Branch и Employee. Тип связи « много — к — одному» (М:1)

В одном офисе могут числиться несколько сотрудников. Принадлежность обязательна, используем 4 правило. При степени связи 1:М (М:1) между сущностями и обязательном классе принадлежности М-связной сущности достаточно сформировать 2 отношения (каждое отношение соответствует сущности). Первичными ключами таких отношений будут ключи этих сущностей.

Таблицы реляционной БД должны отвечать требованиям нормализации отношений. Под нормализацией понимается  процесс организации данных в базе данных. В нее входят: создание таблиц, установка отношений между таблицами в соответствии с правилами, предназначенными для защиты данных и обеспечения большей гибкости базы данных за счет исключения избыточности и несогласованности зависимости.

Отношения находятся в 1НФ тогда и только тогда, когда в любом допустимом отношении любой кортеж содержит только одно значение для каждого атрибута. Значения атрибутов атомарны. Исходя из этого правила, данная база данных находится в 1НФ.

Отношения находятся во второй нормальной форме тогда и только тогда, когда находятся в первой, каждый не ключевой элемент функционально полно зависит от его ключевого элемента.

Отношения находятся в третьей нормальной форме тогда и только тогда, когда находятся во второй, отсутствуют транзитивные нефункциональные зависимости ключевых элементов одной таблицы от не ключевых элементов другой таблицы.

В таблице «tour» дублируются не ключевые поля «date\_of\_sale». Таким образом ее можно разбить на две таблицы «tour» и «sales». Добавляем Внешний ключ «tour\_id» типа «один — ко— многим»

Итого получаем:

|  |  |  |
| --- | --- | --- |
| Sales\_id | Date\_of\_sale | tour\_id |
| 701 | 14/01/20 | 1 |
| 702 | 15/09/20 | 2 |
| 703 | 13/02/20 | 3 |
| 704 | 09/10/20 | 4 |
| 705 | 20/05/20 | 5 |
| 706 | 14/04/20 | 6 |
| 707 | 23/08/20 | 7 |
| 708 | 16/07/20 | 8 |
| 709 | 24/03/20 | 9 |
| 710 | 25/09/20 | 10 |

Таблица №5 «Sales»

Согласно правилам нормализации, выделенные и описанные атрибуты таблиц создаваемой базы данных находятся в третьей нормальной форме.

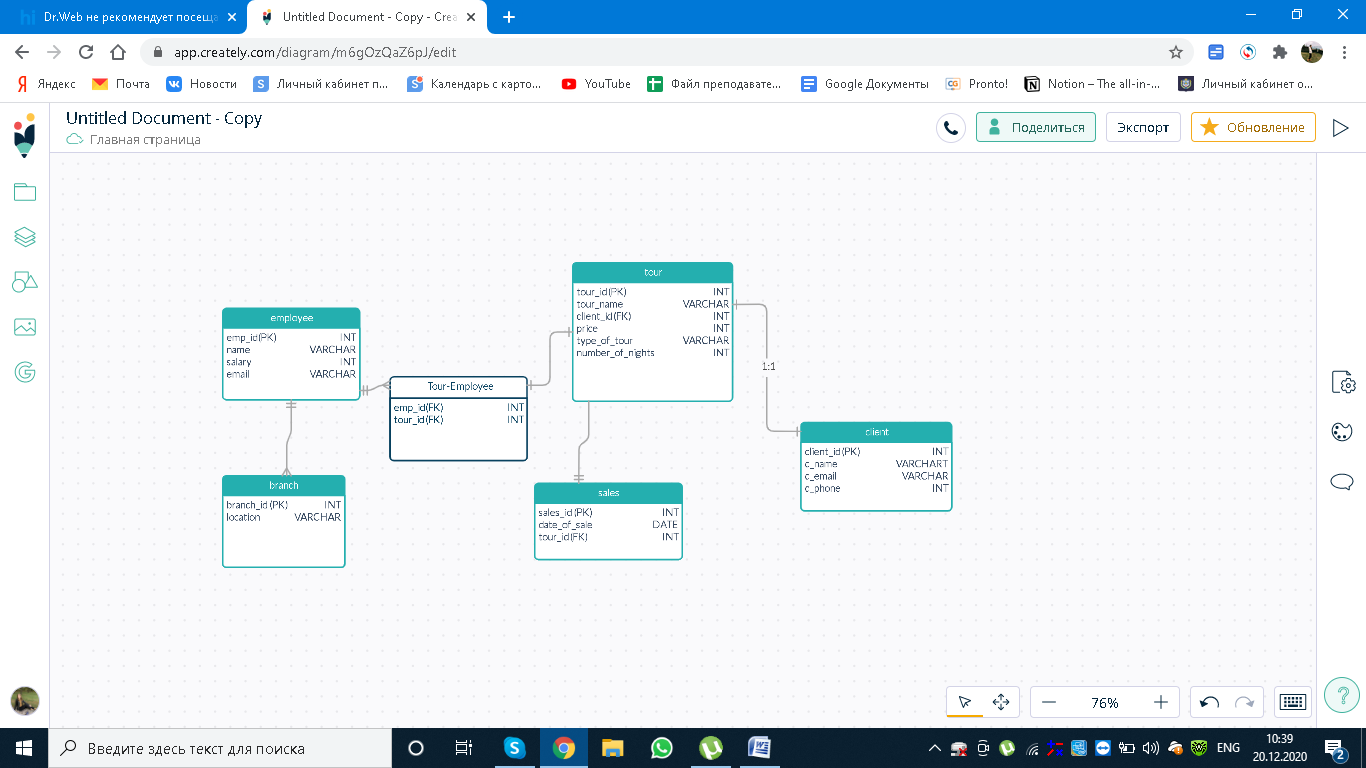


Рисунок №2 Результат даталогического проектирования, реляционная база данных

**Глава 3 Физическое проектирование**

При проектировании базы данных, был создан файл «tour.db» и сделаны запросы на языке SQLite.

Опишем таблицы проектируемой БД:

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип данных | Примечание |
| Код тура | Tour\_id | INT | Не нулевое поле  Первичный ключ |
| Название тура | Tour\_name | VARCHAR | Не нулевое поле |
| Паспорт клиента | Client\_id | INT | Не нулевое поле  Внешний ключ таблицы «Client» |
| Цена | Price | INT | Не нулевое поле |
| Тип | Type\_of\_tour | VARCHAR | Не нулевое поле |

Таблица №6 Схема сущность-связь таблицы «Tour»

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип данных | Примечание |
| Паспорт сотрудника | Emp\_id | INT | Не нулевое поле  Первичный ключ |
| Имя сотрудника | Name | VARCHAR | Не нулевое поле |
| Зарплата | Salary | INT | Не нулевое поле |
| Почта | Email | VARCHAR | Не нулевое поле |

Таблица №7 Схема сущность-связь таблицы «Employee»

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип данных | Примечание |
| Номер сети | Branch\_id | INT | Не нулевое поле  Первичный ключ |
| Локация | Location | VARCHAR | Не нулевое поле |

Таблица №7 Схема сущность-связь таблицы «Branch»

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип данных | Примечание |
| Номер договора | Sales\_id | INT | Не нулевое поле  Первичный ключ |
| Дата продажи тура | Date\_of\_sale | VARCHAR | Не нулевое поле |
| Код тура | Tour\_id | INT | Не нулевое поле  Внешний ключ таблицы «tour» |

Таблица №7 Схема сущность-связь таблицы «Sales»

|  |  |  |  |
| --- | --- | --- | --- |
| Содержание поля | Имя поля | Тип данных | Примечание |
| Паспорт клиента | Client\_id | INT | Не нулевое поле  Первичный ключ |
| Имя клиета | C\_name | VARCHAR | Не нулевое поле |
| Почта клиента | C\_email | VARCHAR | Не нулевое поле |
| Номер телефона клиента | C\_phone | INT | Не нулевое поле |

Таблица №7 Схема сущность-связь таблицы «Client»

Для создания таблиц используется следующий код запроса :

CREATE TABLE database\_name.table\_name(

column1 datatype PRIMARY KEY(one or more columns),

column2 datatype,

column3 datatype,

.....

columnN datatype

);

Для добавления данных используется следующий код запроса:

INSERT INTO table (column1,column2 ,..)

VALUES( value1, value2 ,...);

Получаем 5 таблиц:

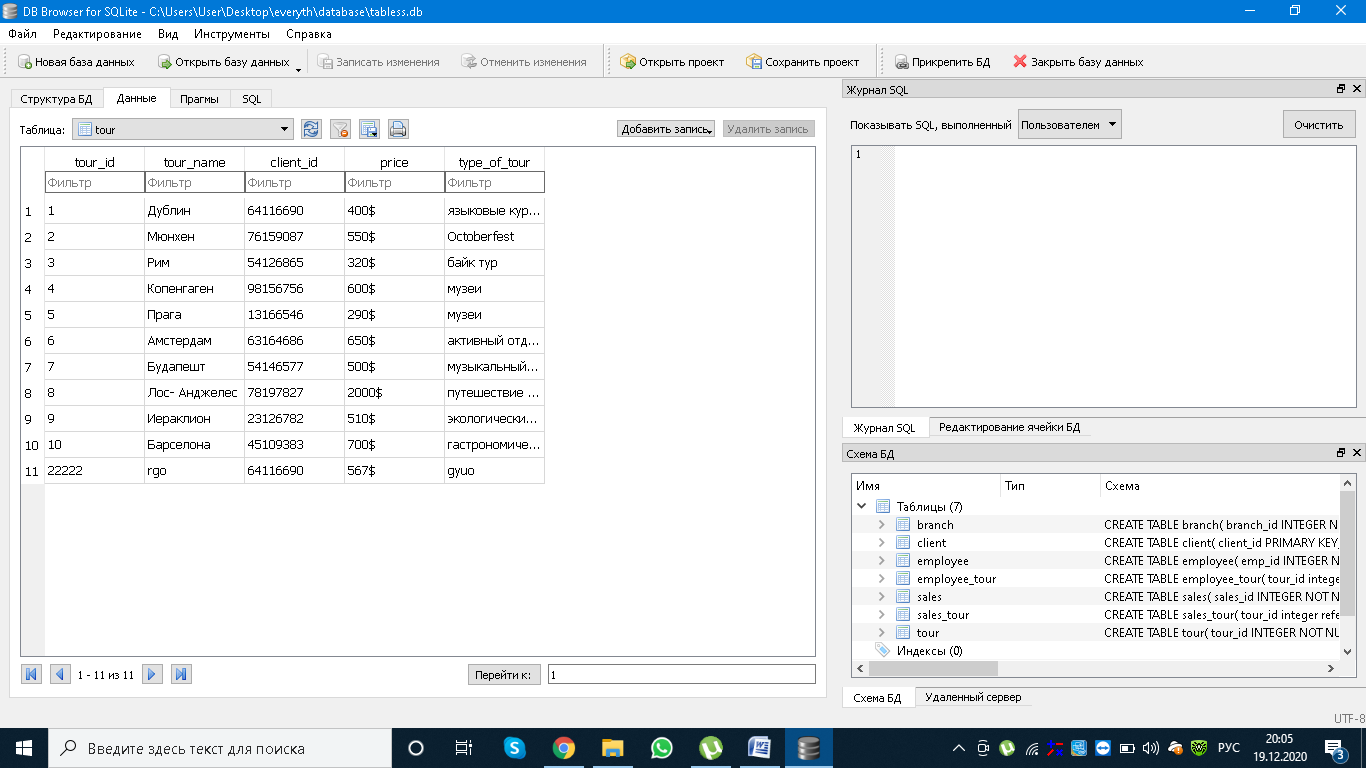


Рисунок №3 Таблица «Tour» в BD Browser

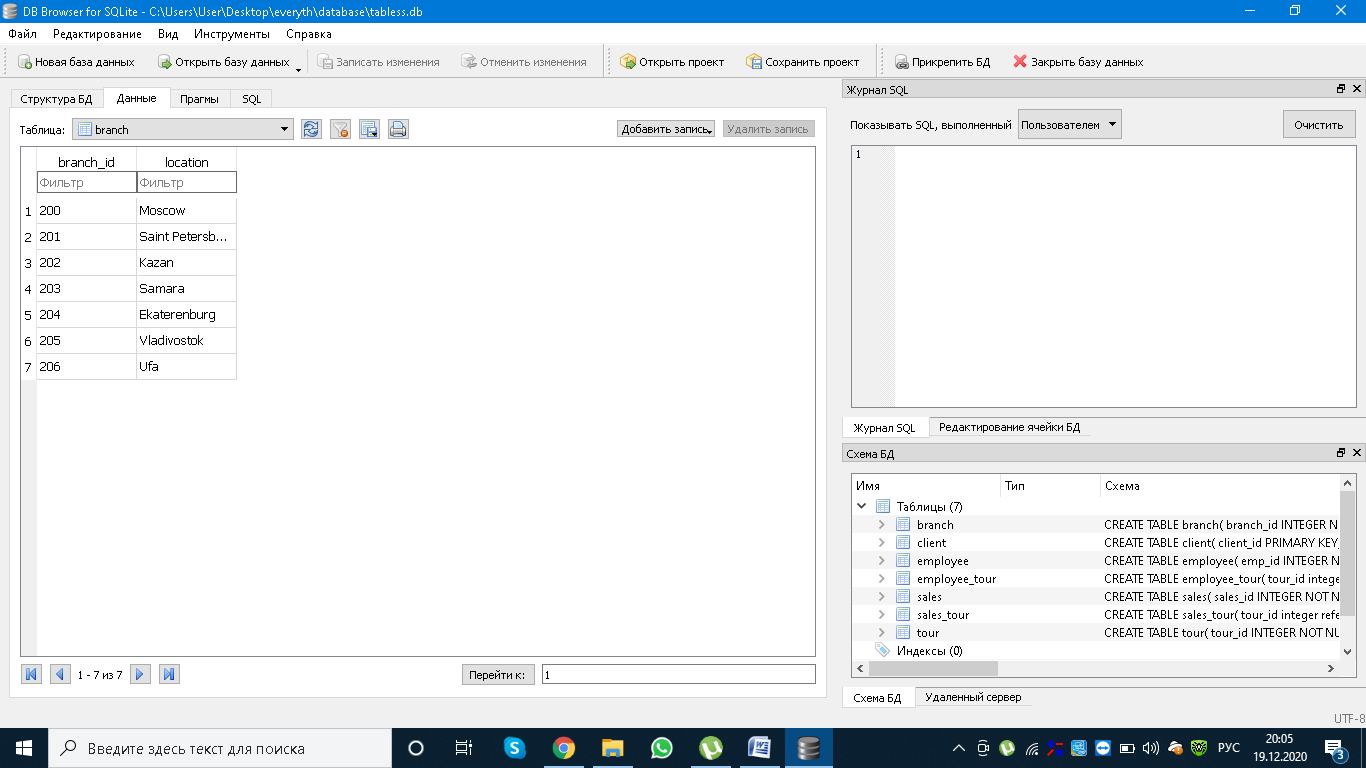


Рисунок №4 Таблица «Branch» в BD Browser

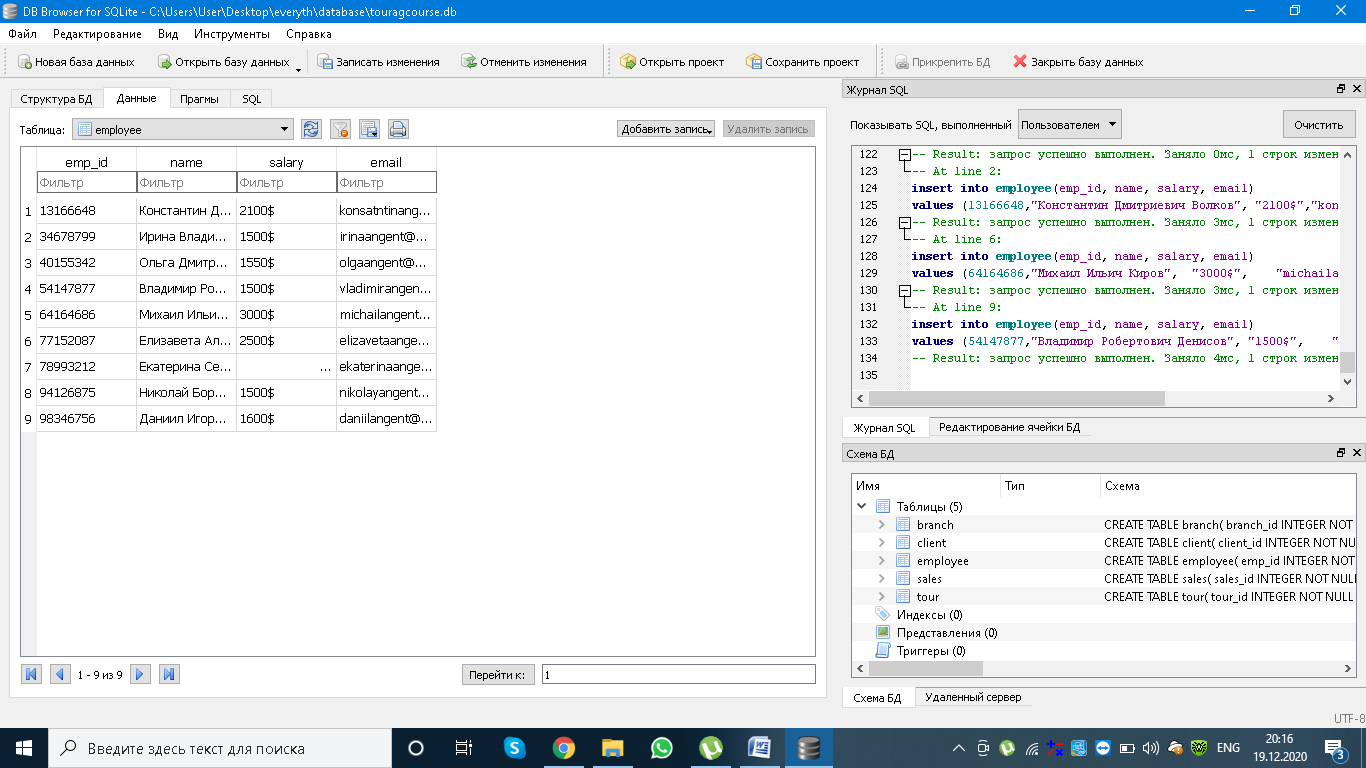


Рисунок №5 Таблица «Employee» в BD Browser

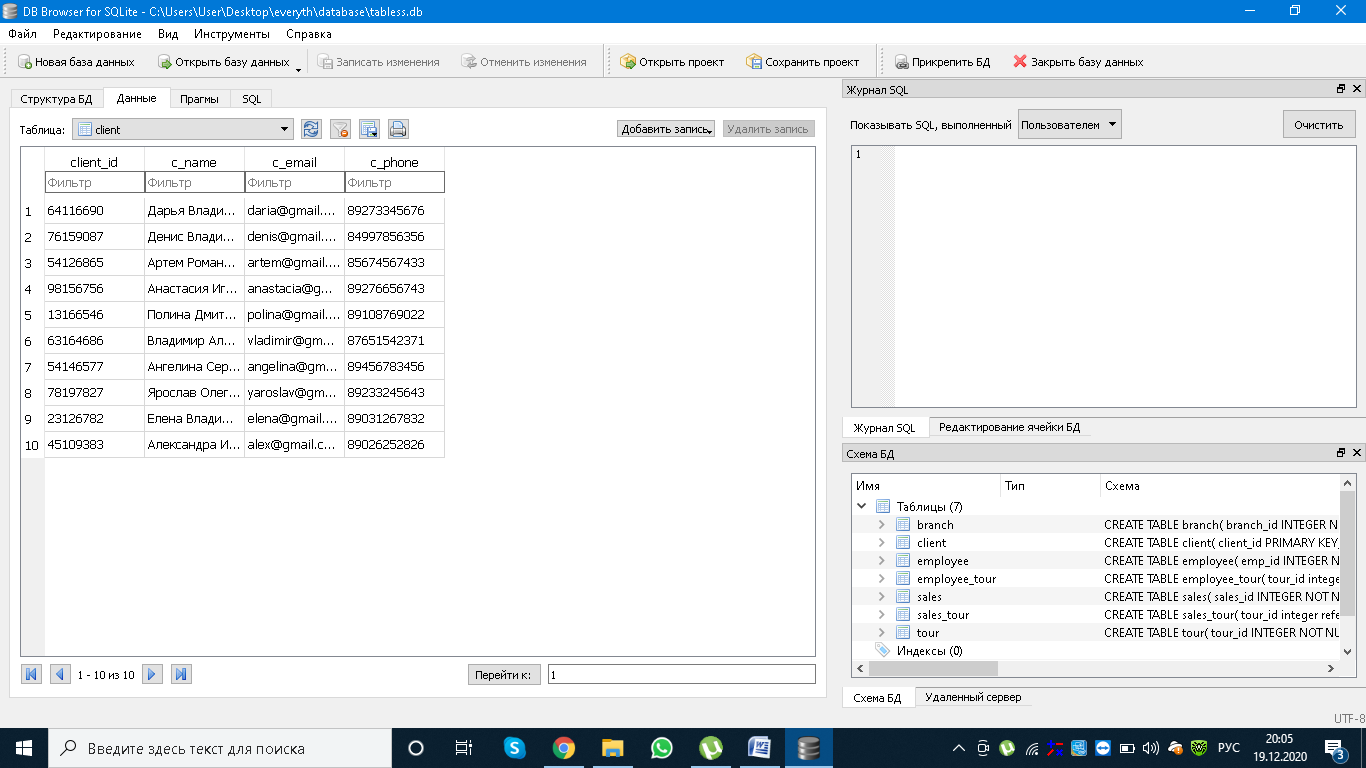


Рисунок №6 Таблица «Client» в BD Browser

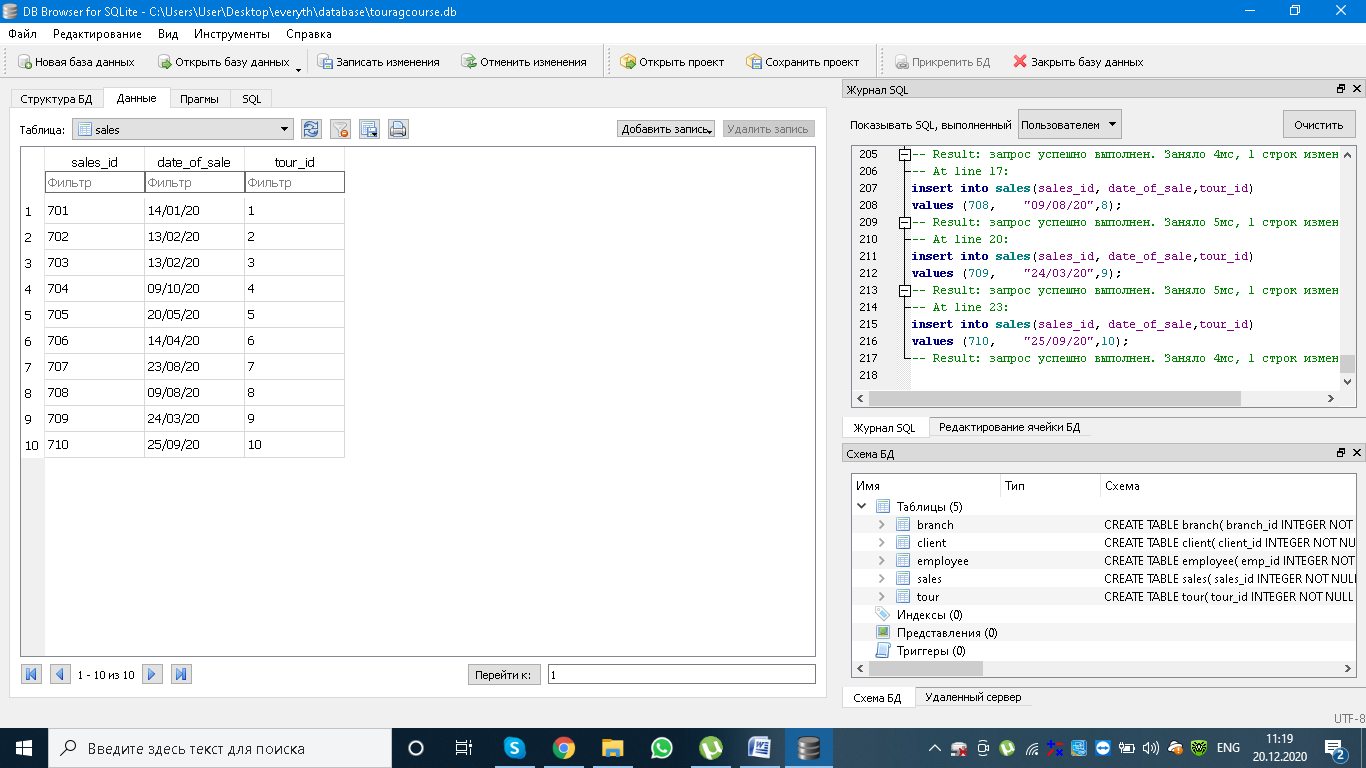


Рисунок №7 Таблица «Sales» в BD Browser

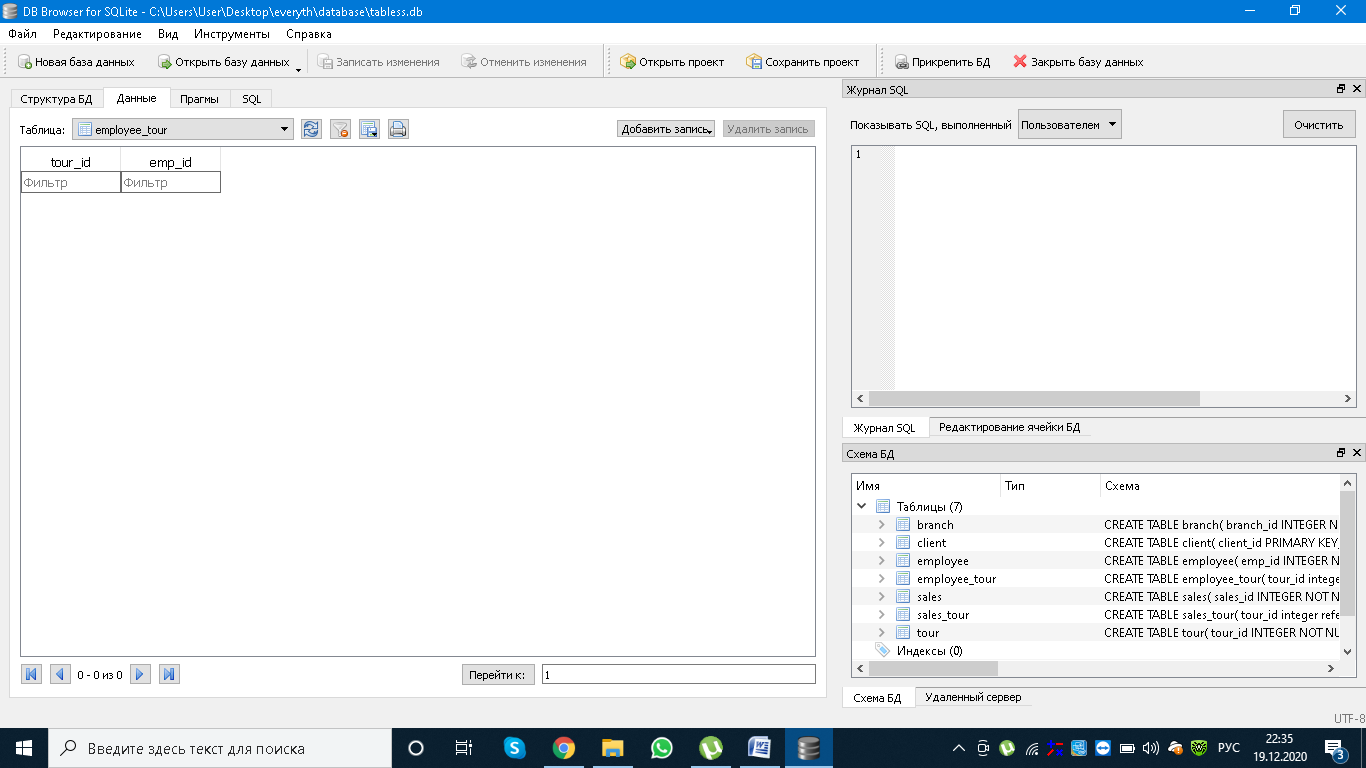


Рисунок №8 Таблица «Employee\_Tour» в BD Browser

После создания и внесения данных в таблицу, осуществим несколько запросов:

1. Выборка

Для выборки данных воспользуемся командой SELECT.

В результате из таблицы будут извлечены все строки.

Данная функция помогает быстро найти нужный тур и получить необходимую информацию.

В работе данной базой данных это важно для быстрого поиска информации, сортировки и корректировки.

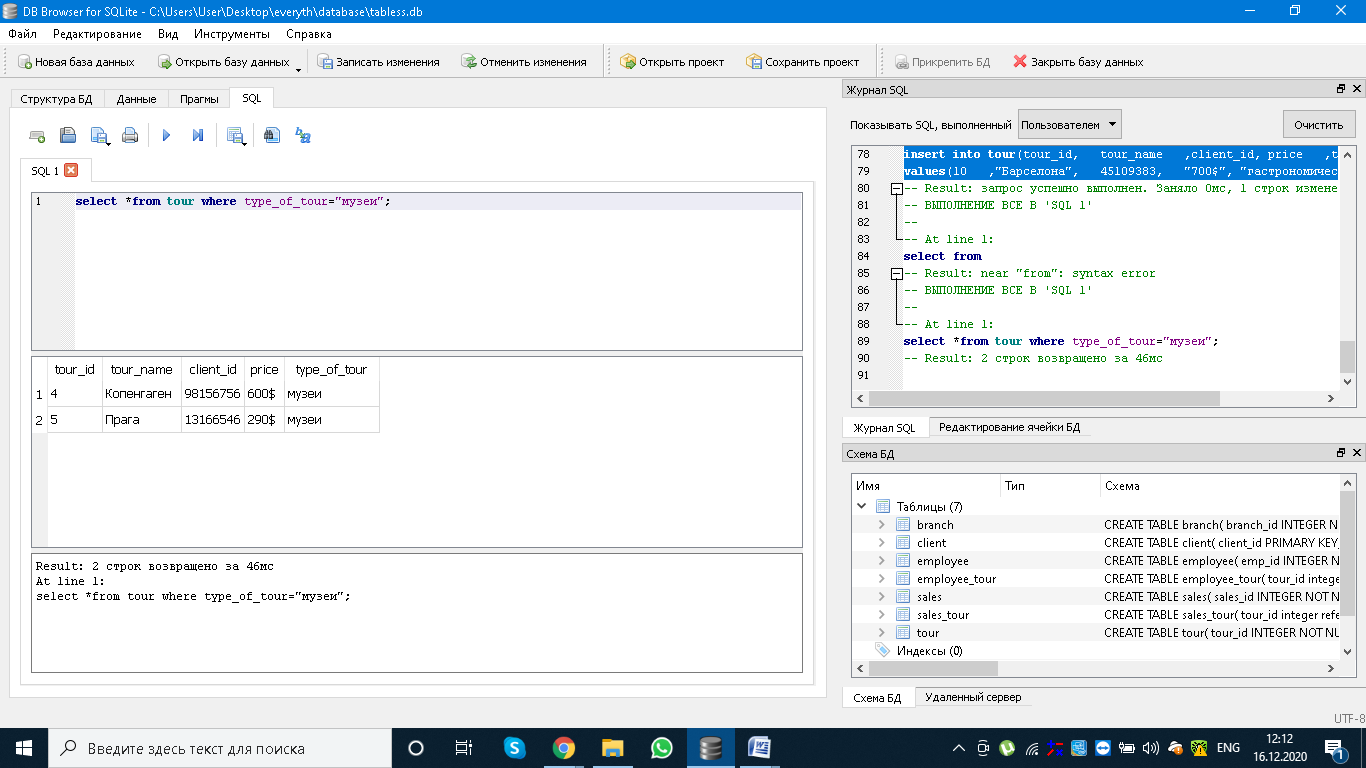


Рисунок №9 Результат команды SELECT

1. Сортировка

Команда **ORDER BY** позволяет сортировать записи по определенному полю при выборе из базы данных. Упорядочим таблицу «Client»так, чтобы ФИО клиентов располагались в алфавитном порядке. Это поможет осуществить более быстрый ручной поиск по таблице, нежели если бы сортировали по номеру паспорта.

Данная команда имеет следующий синтаксис:

SELECT \* FROM имя\_таблицы WHERE условие ORDER BY поле\_для\_сортировки

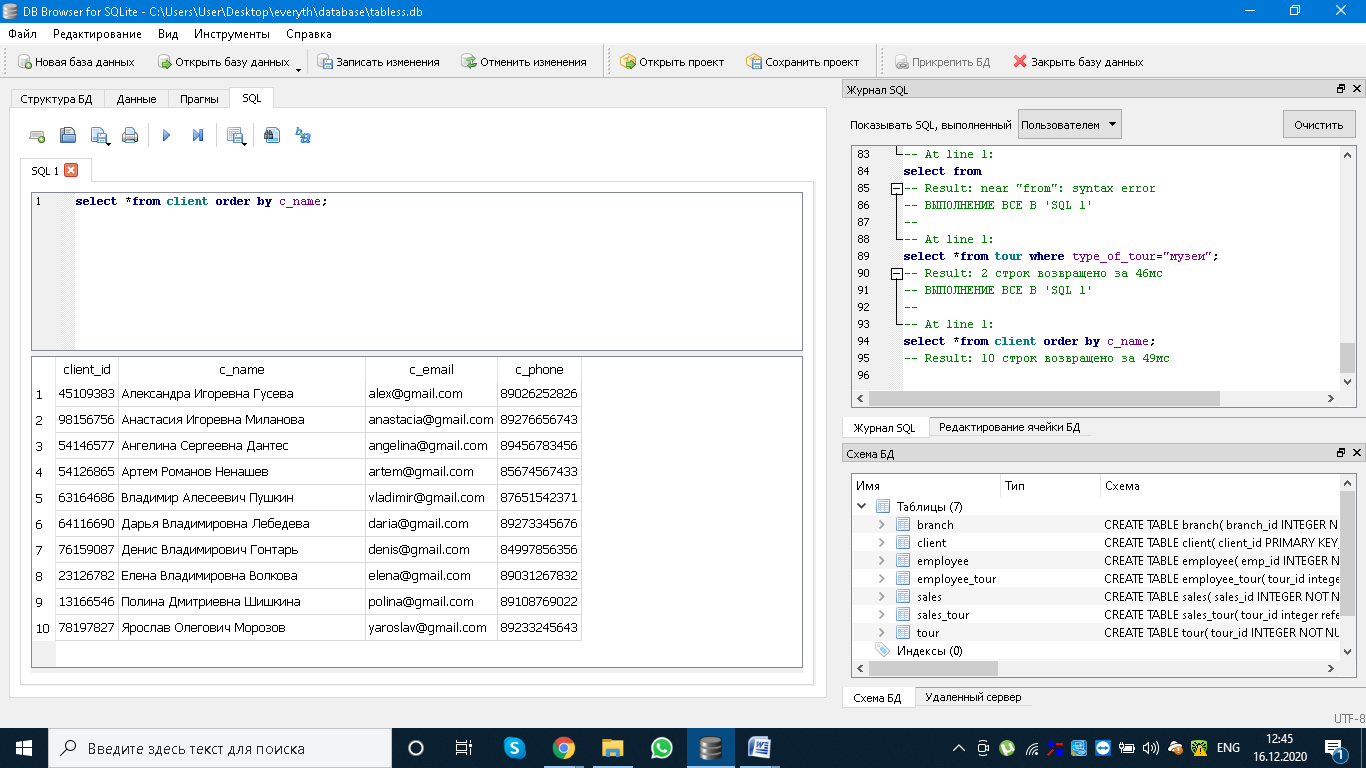


Рисунок №10 Результат команды ORDER BY

1. Удаление элемента

Оператор SQL DELETE предназначен для удаления данных из таблицы. Он имеет следующий синтаксис:

DELETE FROM ИМЯ\_ТАБЛИЦЫ

WHERE УСЛОВИЕ ;

Если не указывать условие, из таблицы будут удалены все строки.

Удалим ненужный элемент из таблицы «Tour»:

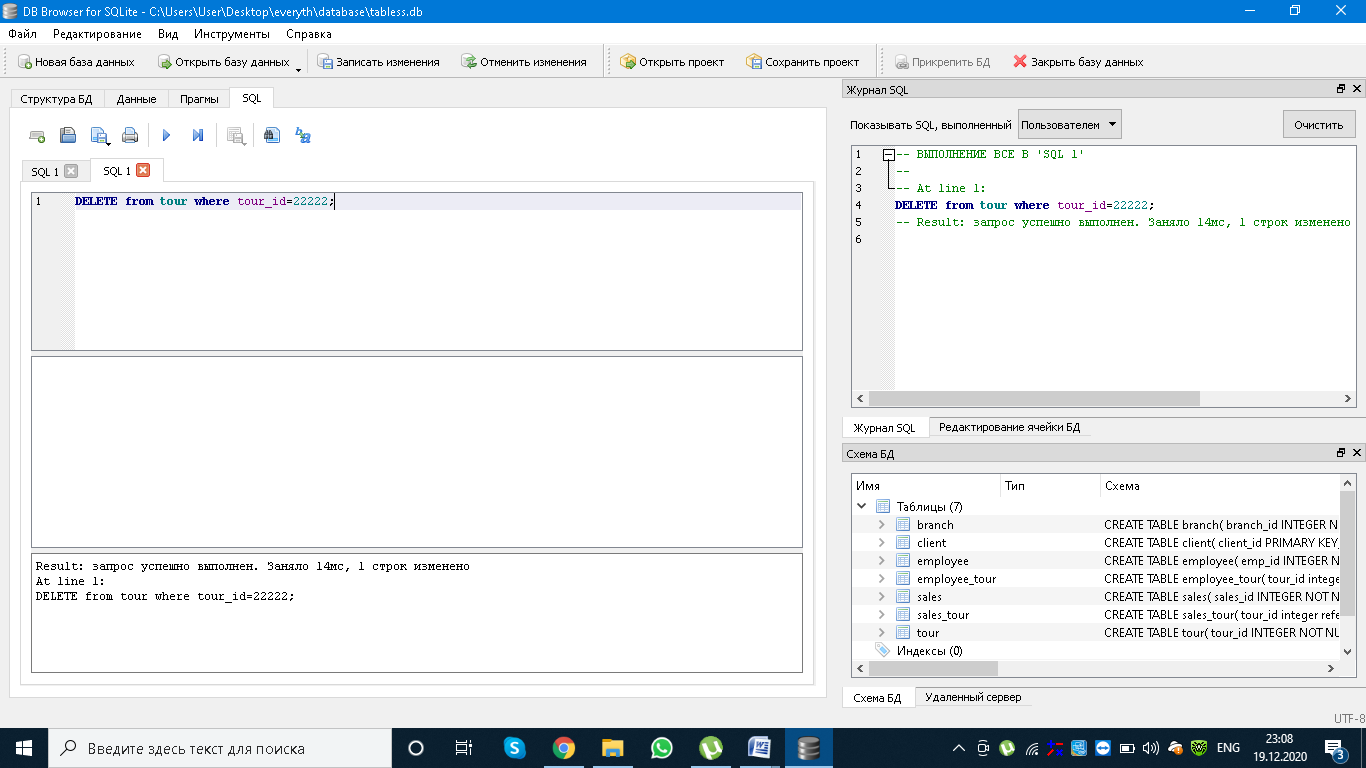


Рисунок №11 Результат оператора DELETE

1. Обновление

Оператор UPDATE используется для обновления данных в таблице. Он применяется, когда в той или иной строке таблицы уже записаны некоторые данные и нужно внести в них изменения. Оператор UPDATE имеет следующий синтаксис:

UPDATE ИМЯ\_ТАБЛИЦЫ SET

ИМЯ\_СТОЛБЦА\_1=ЗНАЧЕНИЕ, ...,

ИМЯ\_СТОЛБЦА\_N=ЗНАЧЕНИЕ

[WHERE УСЛОВИЕ]

В таблице «Employee» в столбце «salary» неправильно отображается зарплата одного из сотрудников.

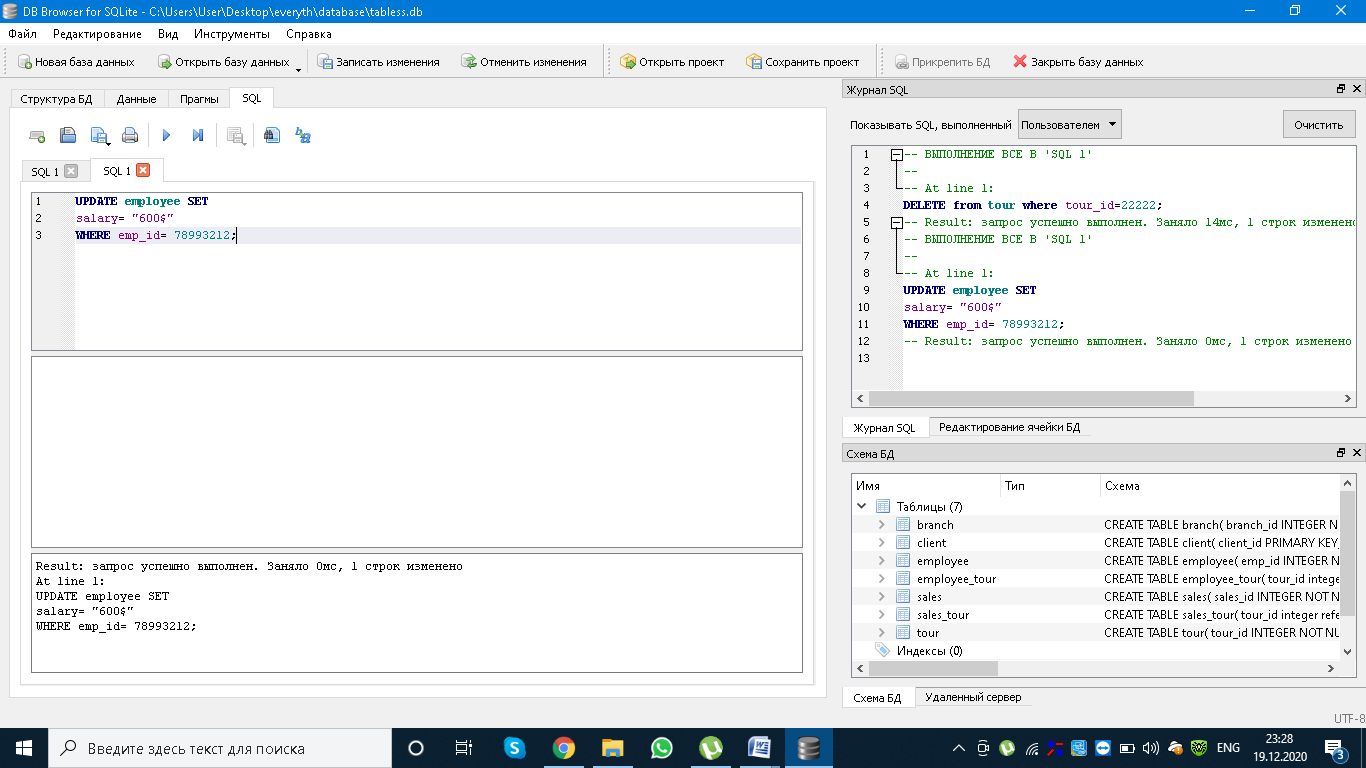


Рисунок №12 Результат оператора UPDATE

1. Изменение структуры

Команда ALTER TABLE используется для добавления, удаления или модификации колонки в уже существующей таблице.

Например, код добавления столбца:

ALTER TABLE table\_name ADD column\_name datatype;

Добавим в таблицу «Tour» раздел «number\_of\_nights»:

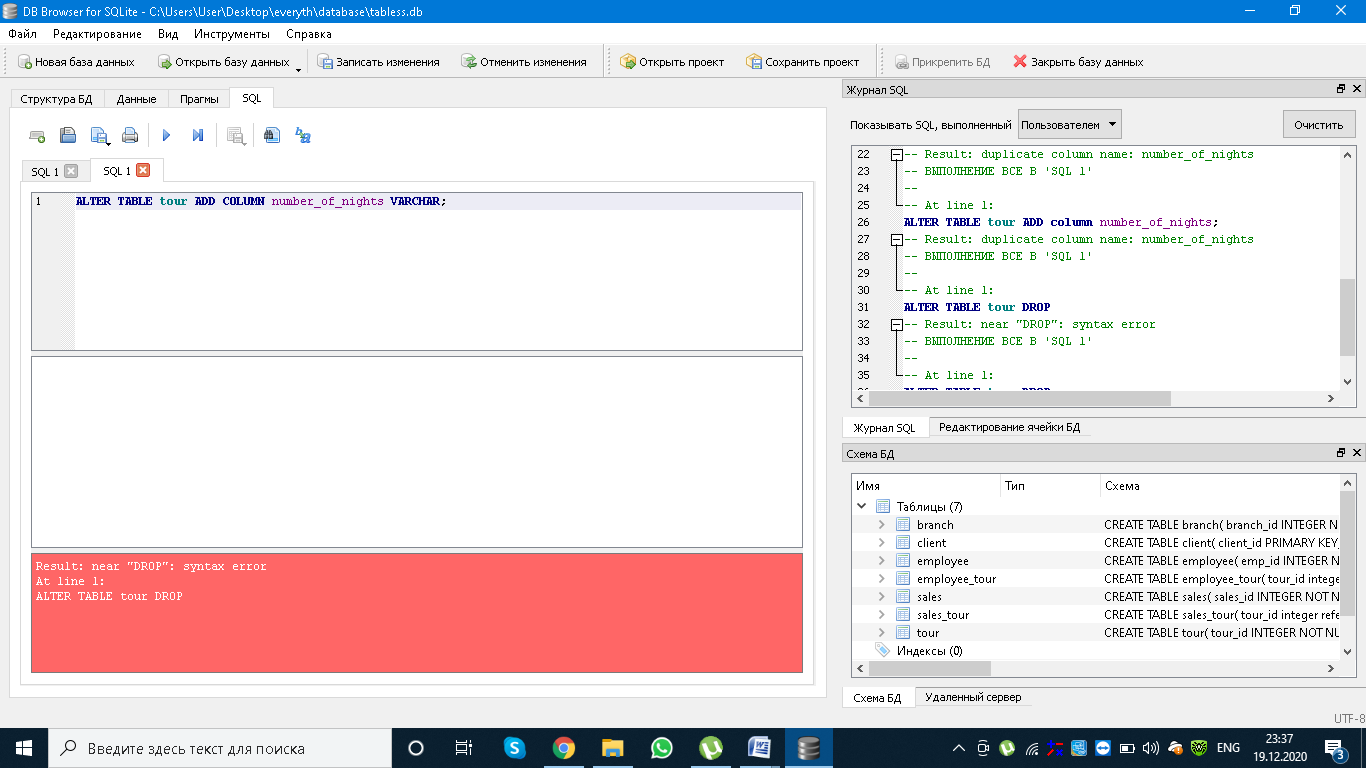


Рисунок №13 Результат оператора ALTER TABLE

**ЗАКЛЮЧЕНИЕ**

В результате проект был выполнен в соответствии с поставленными целью и задачами. Была спроектирована база данных средствами СУБД, проанализирована предметная область, определены цели и пользователи.

Данная база данных позволяет осуществлять эффективную работу предприятия. Повышена оперативность работы с клиентами, в перспективе работа с еще большим количеством. При работе значительно будет уменьшено количество времени и ошибок при работе, упрощение процесса добавления и удаления, поскольку БД структурированная.

В перспективе возможно расширение и добавление новых элементов, разработка веб-приложения.

# Список использованных источников

* + - 1. Бойко В.В., Савинков В.М. Проектирование баз данных информационных систем / Под ред. В.В. Бойко. - М.: Финансы и статистика, 2007. - 351 с.
      2. Связи между таблицами базы данных // habr.com URL: https://habr.com/ru/post/488054/ (дата обращения: 16.10.20).
      3. Правила формирования отношений // spravochnick.ru URL: https://spravochnick.ru/bazy\_dannyh/metod\_suschnost-svyaz\_osnovnye\_ponyatiya\_metoda/pravila\_formirovaniya\_otnosheniy/ . Взято со страницы: https://spravochnick.ru/bazy\_dannyh/metod\_suschnost-svyaz\_osnovnye\_ponyatiya\_metoda/pravila\_formirovaniya\_otnosheniy/ (дата обращения: 16.10.20).
      4. Оператор SQL DELETE для удаления данных из таблицы // function-x.ru URL: function-x.ru (дата обращения: 11.12.20).
      5. Язык запросов SQL // sql-language.ru URL: http://sql-language.ru/alter-table.html (дата обращения: 11.12.20).

6. Функции языка SQL. // it-black.ru URL: https://it-black.ru/funktsii-yazyka-sql/ (дата обращения: 15.12.20).

7. app.creately.com URL: https://app.creately.com/diagram/m6gOzQaZ6pJ/edit (дата обращения: 15.09.20).

8. Атре Ш. Структурный подход к организации баз данных / Под ред. Ш. Атре. - М.: Финансы и статистика, 2006. - 320 с.

**ПРИЛОЖЕНИЕ А**

create table tour(

tour\_id INTEGER NOT NULL Primary key,

tour\_name VARCHAR(15) NOT NULL,

client\_id INTEGER NOT NULL,

price VARCHAR(10) NOT NULL,

type\_of\_tour VARCHAR(25) NOT NULL,

FOREIGN KEY(client\_id) REFERENCES client(client\_id) ON DELETE SET NULL,

);

CREATE TABLE sales(

sales\_id INTEGER NOT NULL PRIMARY KEY,

date\_of\_sale VARCHAR(20) NOT NULL,

tour\_id INTEGER,

FOREIGN KEY (tour\_id) REFERENCES tour(tour\_id)

);

create table employee(

emp\_id INTEGER NOT NULL PRIMARY KEY,

name VARCHAR(255) NOT NULL,

salary VARCHAR(10) NOT NULL,

email VARCHAR(255) NOT NULL,

FOREIGN KEY (emp\_id) REFERENCES employee(emp\_id)

);

create table branch(

branch\_id INTEGER NOT NULL PRIMARY KEY,

location VARCHAR(30),

FOREIGN KEY (branch\_id) REFERENCES branch(branch\_id)

);

CREATE TABLE client(

client\_id INTEGER NOT NULL PRIMARY KEY,

c\_name VARCHAR (255) NOT NULL,

c\_email VARCHAR (25) NOT NULL,

c\_phone INTEGER NOT NULL

);

create table employee\_tour(

tour\_id integer references employee(emp\_id),

emp\_id integer references tour(tour\_id));

create table sales\_tour(

tour\_id integer references sales(sales\_id),

sales\_id integer references tour(tour\_id));

insert into employee(emp\_id, name, salary, email)

values (34678799, "Ирина Владимировна Мартышкина", "1500$", "irinaangent@gmail.com");

insert into employee(emp\_id, name, salary, email)

values (40155342, "Ольга Дмитриевна Войкова", "1550$","olgaangent@gmail.com");

insert into employee(emp\_id, name, salary, email)

values (78993212,"Екатерина Сергеевна Самаркина","1600$","ekaterinaangent@gmail.com");

insert into employee(emp\_id, name, salary, email)

values (64116999,"Анастасия Оскаровна Вулах", "2000$","anastasiiaangent@gmail.com");

insert into employee(emp\_id, name, salary, email)

values (77152087, "Елизавета Алексеевна Екатеринина", "2500$", "elizavetaangent@gmail.com");

insert into employee(emp\_id, name, salary, email)

values (94126875, "Николай Борисов Никитин", "1500$","nikolayangent@gmail.com");

insert into employee(emp\_id, name, salary, email)

values (98346756, "Даниил Игоревич Бандарчук", "1600$", "daniilangent@gmail.com");

insert into employee(emp\_id, name, salary, email)

values (13166648,"Константин Дмитриевич Волков", "2100$","konsatntinangent@gmail.com");

insert into employee(emp\_id, name, salary, email)

values (64164686,"Михаил Ильич Киров", "3000$", "michailangent@gmail.com");

insert into employee(emp\_id, name, salary, email)

values (54147877,"Владимир Робертович Денисов", "1500$", "vladimirangent@gmail.com");

insert into branch(branch\_id, location)

values(200, "Moscow");

insert into branch(branch\_id, location)

values(202, "Kazan");

- At line 1:

insert into branch(branch\_id, location)

values(202, "Kazan");

- At line 1:

insert into branch(branch\_id, location)

values(203, "Samara");

- At line 1:

insert into branch(branch\_id, location)

values(204, "Ekaterenburg");

- At line 1:

insert into branch(branch\_id, location)

values(205, "Vladivostok");

insert into sales(sales\_id, date\_of\_sale,tour\_id)

values (702, "13/02/20",2);

insert into sales(sales\_id, date\_of\_sale,tour\_id)

values (703, "13/02/20",3);

insert into sales(sales\_id, date\_of\_sale,tour\_id)

values (704, "09/10/20",4);

insert into sales(sales\_id, date\_of\_sale,tour\_id)

values (705, "20/05/20",5);

insert into sales(sales\_id, date\_of\_sale,tour\_id)

values (706, "14/04/20",6);

insert into sales(sales\_id, date\_of\_sale,tour\_id)

values (707, "23/08/20",7);

insert into sales(sales\_id, date\_of\_sale,tour\_id)

values (708, "09/08/20",8);

insert into sales(sales\_id, date\_of\_sale,tour\_id)

values (709, "24/03/20",9);

insert into sales(sales\_id, date\_of\_sale,tour\_id)

values (710, "25/09/20",10);

insert into client(client\_id, c\_name,c\_email, c\_phone)

values (64116690, "Дарья Владимировна Лебедева ","daria@gmail.com" ,89273345676);

insert into client(client\_id, c\_name,c\_email, c\_phone)

values(54126865, "Артем Романов Ненашев", "artem@gmail.com", 85674567433);

-- Result: запрос успешно выполнен. Заняло 137мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into client(client\_id, c\_name,c\_email, c\_phone)

values(98156756, "Анастасия Игоревна Миланова", "anastacia@gmail.com", 89276656743);

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into client(client\_id, c\_name,c\_email, c\_phone)

values(13166546, "Полина Дмитриевна Шишкина", "polina@gmail.com", 89108769022);

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into client(client\_id, c\_name,c\_email, c\_phone)

values(63164686,"Владимир Алесеевич Пушкин", "vladimir@gmail.com", 87651542371);

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into client(client\_id, c\_name,c\_email, c\_phone)

values(54146577, "Ангелина Сергеевна Дантес", "angelina@gmail.com", 89456783456);

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into client(client\_id, c\_name,c\_email, c\_phone)

values(78197827, "Ярослав Олегович Морозов", "yaroslav@gmail.com", 89233245643);

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into client(client\_id, c\_name,c\_email, c\_phone)

values(23126782, "Елена Владимировна Волкова", "elena@gmail.com", 89031267832);

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into client(client\_id, c\_name,c\_email, c\_phone)

values(45109383, "Александра Игоревна Гусева","alex@gmail.com", 89026252826);

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(1 ,"Дублин", 64116690 ,"400$", "языковые курсы");

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(2, "Мюнхен", 76159087, "550$", "Octoberfest");

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(3, "Рим", 54126865 ,"320$", "байк тур");

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(4 ,"Копенгаген", 98156756 ,"600$" ,"музеи");

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(5 ,"Прага", 13166546 ,"290$","музеи");

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(6, "Амстердам", 63164686, "650$", "активный отдых");

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(7, "Будапешт", 54146577 ,"500$", "музыкальный фестиваль");

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(8, "Лос- Анджелес", 78197827 ,"2000$", "путешествие по штату");

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(9, "Иераклион", 23126782, "510$", "экологический туризм");

-- Result: запрос успешно выполнен. Заняло 0мс, 1 строк изменено

-- ВЫПОЛНЕНИЕ ВСЕ В 'SQL 1'

--

-- At line 1:

insert into tour(tour\_id, tour\_name ,client\_id, price ,type\_of\_tour)

values(10 ,"Барселона", 45109383, "700$", "гастрономический");

DELETE FROM employee WHERE salary= 54;

DELETE from tour where tour\_id=22222;