**Министерство транспорта Российской Федерации**

**Федеральное государственное автономное образовательное  
учреждение высшего образования  
«РУТ (МИИТ»**

**Институт транспортной техники и систем управления**

**Кафедра «Управление и защита информации»**

КУРСОВОЙ ПРОЕКТ

по дисциплине

**«Основы построения защищенных баз данных»**

**на тему  
«СОЗДАНИЕ РЕЛЯЦИОННОЙ БД “Т УРИСТИЧЕСКОЕ БЮРО”»**

Выполнил: ст. гр. ТКИ–511  
Тяжко Д.И.  
Проверила: Васильева М.А.

Москва 2021

Оглавление

[ЦЕЛЬ КУРСОВОГО ПРОЕКТА 5](#_Toc60177508)

[ЗАДАНИЕ НА КУРСОВОВОЙ ПРОЕКТ 5](#_Toc60177509)

[ПРОЕКТИРОВАНИЕ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ 5](#_Toc60177510)

[1 Инфологическое проектирование 5](#_Toc60177511)

[1.1 Анализ предметной области 5](#_Toc60177512)

[1.2 Анализ информационных задач и круга пользователей системы 8](#_Toc60177513)

[2 Логическое проектирование реляционной БД 9](#_Toc60177514)

[2.1 Составление реляционных отношений 9](#_Toc60177515)

[2.2 Нормализация полученных отношений 16](#_Toc60177516)

[3 Физическое проектирование БД 17](#_Toc60177517)

[3.1 Разработка скриптов на создание базы данных и таблиц 17](#_Toc60177518)

[3.2 Разработка скриптов на добавление данных в таблицы 21](#_Toc60177519)

[3.3 Разработка необходимых запросов, представлений (view), процедур, функций и триггеров 25](#_Toc60177520)

[Список литературы 28](#_Toc60177521)

# ЦЕЛЬ КУРСОВОГО ПРОЕКТА

Целью курсового проекта является изучение методов и закрепление знаний в проектировании реляционных баз данных (РБД) в системе управления базами данных (СУБД) Microsoft SQL Server.

# ЗАДАНИЕ НА КУРСОВОВОЙ ПРОЕКТ

В данном курсовом проекте ставится задача разработать РБД в СУБД Microsoft SQL Server по заданной теме «Туристическое бюро». Проектирование РБД проводится с помощью метода «Сущность-связь». Проверка построенной модели РБД осуществляется с помощью метода нормализации отношений.

Описание предметной области. БД создается для информационного обслуживания сотрудников фирмы. Фирма оказывает услуги в туристическом бизнесе. Клиентам предлагаются путевки в разные страны, города. Путевки отличаются содержанием программы (отдых, экскурсии, туризм и т.д.), имеется возможность выбора путевки по цене (в зависимости от места проживания, все включено и т.д.). для постоянных клиентов и детей до 12 лет имеются скидки.

Готовые запросы:

1. Показать все купленные программы путевок в данный город.
2. Выдавать список стран, популярных у определенного возраста клиентов.
3. Показать список заказанных путёвок с определённой программой.
4. Показать данные всех клиентов, попадающих под категорию "Дети младше 12-ти лет", а также страны и города, которые они посещали.
5. Показать все цены на путёвки (без скидок) в порядке убывания.

# ПРОЕКТИРОВАНИЕ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ

## Инфологическое проектирование

### Анализ предметной области

База данных создается для информационного обслуживания сотрудников фирмы. Фирма оказывает услуги в туристическом бизнесе. Клиентам предлагаются путевки в разные страны, города. Путевки отличаются содержанием программы (отдых, экскурсии, туризм и т.д.), имеется возможность выбора путевки по цене (в зависимости от места проживания, все включено и т.д.). для постоянных клиентов и детей до 12 лет имеются скидки.

Выделим базовые сущности этой предметной области:

* Путёвки (клиенты, страны, города, программы, дата начала, дата окончания, цена)

Выделим дополнительные сущности этой предметной области:

* Клиенты (скидка, фамилия, имя, отчество, возраст, паспорт, загранпаспорт, медицинское страхование).
* Страны (наименование, город).
* Города (наименование).
* Программы (наименование).
* Скидки (наименование).

ER-диаграмма БД, выполненная в среде визуального построения диаграмм (Micrososoft Visio 2013) приведена на Рисунке 1. На данной ER-диаграмме присутствуют связи один - ко - многим.

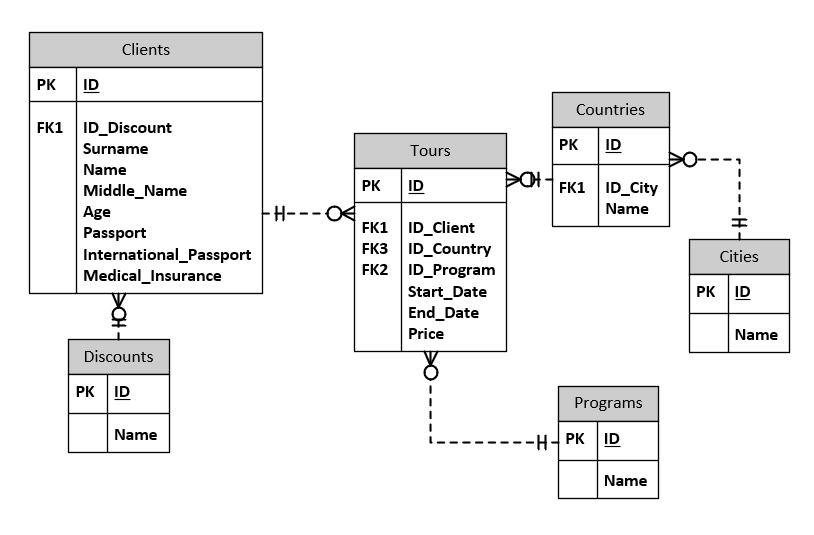


Рисунок 1 ‑ ER-диаграмма

В полученной ER-диаграмме присутствуют только связи один-ко-многим. Такие связи легко реализуются средствами СУБД.

### Анализ информационных задач и круга пользователей системы

Система создаётся для обслуживания следующих групп пользователей:

* администрация фирмы;
* заказчики;
* сотрудники фирмы.

Определим границы информационной поддержки пользователей:

**1) Функциональные возможности:**

* ведение БД (запись, чтение, модификация, удаление в архив);
* обеспечение логической непротиворечивости БД;
* обеспечение защиты данных от несанкционированного или случайного доступа (определение прав доступа);
* реализация наиболее часто встречающихся запросов в готовом виде.

**2) Готовые запросы:**

* Показать все купленные программы путевок в данный город.
* Выдавать список стран, популярных у определенного возраста клиентов.
* Показать список заказанных путёвок с определённой программой.
* Показать данные всех клиентов, попадающих под категорию "Дети младше 12-ти лет", а также страны и города, которые они посещали.
* Показать все цены на путёвки (без скидок) в порядке убывания.

## Логическое проектирование реляционной БД

### Составление реляционных отношений

Каждое реляционное отношение соответствует одной сущности (объекту ПрО), и в него вносятся все атрибуты сущности. Для каждого отношения необходимо определить первичный ключ и внешние ключи (если они есть). В том случае, если базовое отношение не имеет потенциальных ключей, вводится *суррогатный первичный ключ*, который не несёт смысловой нагрузки и служит только для идентификации записей.

1. Для сущности **Путевки** (**Tours**) в качестве ключевого атрибута введем суррогатный ключ ID.

**Таблица 1 – Схема отношения** **Путевки** (**Tours**)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID | Целое число | первичный ключ |
| ID Клиента | ID\_Client | Целое число | обязательное поле, внешний ключ, ссылается на Client(ID) |
| ID Страны | ID\_Country | Целое число | обязательное поле, внешний ключ, ссылается на Country(ID) |
| ID Города | ID\_City | Целое число | обязательное поле, внешний ключ, ссылается на City(ID) |
| ID Программы | ID\_Program | Целое число | обязательное поле, внешний ключ, ссылается на Program(ID) |
| Дата начала | [Start\_Date] | Дата | обязательное поле |
| Дата окончания | [End\_Date] | Дата | обязательное поле |
| Цена | Price | Денежный тип | обязательное поле |

1. Для сущности **Клиенты** (**Clients**) в качестве ключевого атрибута введем суррогатный ключ ID.

Таблица 2 – Схема отношения Клиенты (Clients)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID | Целое число | первичный ключ |
| ID Скидки | ID\_Discount | Целое число | обязательное поле, внешний ключ, ссылается на Discount(ID) |
| Фамилия | Surname | Строка(50) | обязательное поле |
| Имя | [Name] | Строка(50) | обязательное поле |
| Отчество | Middle\_Name | Строка(50) | обязательное поле |
| Возраст | Age | Целое число | обязательное поле |
| Паспортные данные | Passport | Строка(50) | обязательное, уникальное поле |
| Данные загранпаспорта | International\_Passport | Строка(50) | обязательное, уникальное поле |
| Медицинское страхование | Medical\_Insurance | Строка(50) | обязательное, уникальное поле |

1. Для сущности **Страны (Countries)** потенциальным ключом является атрибут «Название», т.к. его значение является уникальным. Но значениями этого атрибута будут, символьные строки. Введем для этой сущности суррогатный ключ ID.

Таблица 3 – Схема отношения Страны (Countries)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID | Целое число | первичный ключ |
| ID Города | ID\_City | Целое число | обязательное поле, внешний ключ, ссылается на Cities(ID) |
| Название | [Name] | Строка (50) | обязательное, уникальное поле |

1. Для сущности **Города** (**Cities**) потенциальным ключом является атрибут «Название», т.к. его значение является уникальным. Но значениями этого атрибута будут, символьные строки. Введем для этой сущности суррогатный ключ ID.

**Таблица 4 – Схема отношения** **Города** (**Cities**)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID | Целое число | первичный ключ |
| Название | [Name] | Строка (50) | обязательное, уникальное поле |

1. Для сущности **Программы** (**Programs**) потенциальным ключом является атрибут «Название», т.к. его значение является уникальным. Но значениями этого атрибута будут, символьные строки. Введем для этой сущности суррогатный ключ ID.

**Таблица 5 – Схема отношения** **Программы** (**Programs**)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID | Целое число | первичный ключ |
| Название | [Name] | Строка (50) | обязательное, уникальное поле |

1. Для сущности **Скидки** (**Discounts**) потенциальным ключом является атрибут «Название», т.к. его значение является уникальным. Но значениями этого атрибута будут, символьные строки. Введем для этой сущности суррогатный ключ ID.

**Таблица 6 – Схема отношения** **Скидки** (**Discounts**)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID | Целое число | первичный ключ |
| Название | [Name] | Строка (50) | обязательное, уникальное поле |

### Нормализация полученных отношений

**1НФ.** Для приведения таблиц к 1НФ требуется составить таблицы (один атрибут – один столбец) и разбить сложные атрибуты на простые, а многозначные атрибуты вынести в отдельные отношения.  
В наших таблицах все атрибуты простые.

**2НФ.** Все отношения находятся во 2НФ, так как для каждого из них введен уникальный ключ.

**3НФ**. Все неключевые атрибуты зависят только от первичного ключа и не зависят друг от друга. Поэтому спроектированная **база данных находится в третьей нормальной форме**.

## Физическое проектирование БД

### Разработка скриптов на создание базы данных и таблиц

БД «Туристическое бюро» проектируется в среде Microsoft SQL Server Management Studio.

*Создание базы данных «*Туристическое бюро*»:*

CREATE DATABASE TravelAgency;

*Создание таблицы* *Путевки:*

CREATE TABLE Tours

(

ID INTEGER PRIMARY KEY

, ID\_Client INTEGER NOT NULL FOREIGN KEY REFERENCES Clients(ID)

, ID\_Country INTEGER NOT NULL FOREIGN KEY REFERENCES Countries(ID)

, ID\_Program INTEGER NOT NULL FOREIGN KEY REFERENCES Programs(ID)

, [Start\_Date] NVARCHAR(50) NOT NULL

, [End\_Date] NVARCHAR(50) NOT NULL

, Price MONEY NOT NULL

);

*Создание таблицы Клиенты:*

CREATE TABLE Clients

(

ID INTEGER PRIMARY KEY

, ID\_Discount INTEGER NOT NULL FOREIGN KEY REFERENCES Discounts(ID)

, Surname NVARCHAR(50) NOT NULL

, [Name] NVARCHAR(50) NOT NULL

, Middle\_Name NVARCHAR(50) NOT NULL

, Age INTEGER NOT NULL

, Passport NVARCHAR(50) NOT NULL UNIQUE

, International\_Passport NVARCHAR(50) NOT NULL UNIQUE

, Medical\_Insurance NVARCHAR(50) NOT NULL UNIQUE

);

*Создание таблицы* *Страны:*

CREATE TABLE Countries

(

ID INTEGER PRIMARY KEY

, ID\_City INTEGER NOT NULL FOREIGN KEY REFERENCES Cities(ID)

, [Name] NVARCHAR(50) NOT NULL

);

*Создание таблицы Города:*

CREATE TABLE Cities

(

ID INTEGER PRIMARY KEY

, [Name] NVARCHAR(50) NOT NULL UNIQUE

);

*Создание таблицы* *Программы:*

CREATE TABLE Programs

(

ID INTEGER PRIMARY KEY

, [Name] NVARCHAR(50) NOT NULL UNIQUE

);

*Создание таблицы Скидки:*

CREATE TABLE Discounts

(

ID INTEGER PRIMARY KEY

, [Name] NVARCHAR(50) NOT NULL UNIQUE

);

Построим диаграмму спроектированной базы данных (Рисунок 3).

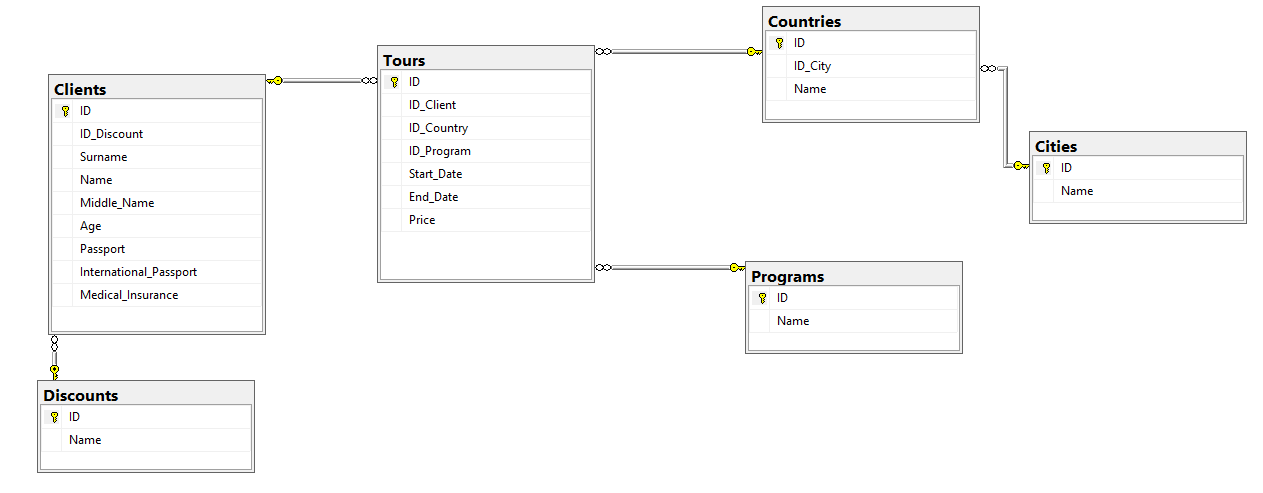


Рисунок 3 ‑ Диаграмма базы данных

### Разработка скриптов на добавление данных в таблицы

*Заполнение таблицы* *Путевки:*

INSERT INTO Tours

VALUES

(1, 1, 1, 4, '24.09.2021', '28.09.2021', 75000)

, (2, 2, 1, 3, '03.05.2021', '09.05.2021', 150000)

, (3, 3, 1, 3, '03.05.2021', '09.05.2021', 150000)

, (4, 4, 1, 3, '03.05.2021', '09.05.2021', 127500)

, (5, 5, 1, 5, '23.08.2021', '30.08.2021', 300000)

, (6, 6, 1, 5, '23.08.2021', '30.08.2021', 300000)

, (7, 7, 1, 6, '15.07.2021', '22.07.2021', 187500)

, (8, 8, 1, 6, '15.07.2021', '22.07.2021', 187500)

, (9, 9, 1, 6, '15.07.2021', '22.07.2021', 187500)

, (10, 10, 1, 6, '15.07.2021', '22.07.2021', 187500)

, (11, 11, 1, 6, '15.07.2021', '22.07.2021', 187500)

, (12, 12, 1, 6, '15.07.2021', '22.07.2021', 187500)

, (13, 13, 1, 6, '15.07.2021', '22.07.2021', 187500)

, (14, 14, 1, 6, '15.07.2021', '22.07.2021', 187500)

, (15, 15, 2, 2, '08.06.2021', '15.06.2021', 150000)

, (16, 16, 2, 2, '08.06.2021', '15.06.2021', 150000)

, (17, 17, 2, 2, '04.06.2021', '04.06.2021', 127500)

, (18, 18, 3, 4, '17.02.2021', '20.02.2021', 100000)

, (19, 19, 3, 4, '17.02.2021', '20.02.2021', 100000)

, (20, 20, 3, 5, '06.04.2021', '14.04.2021', 150000)

, (21, 21, 3, 5, '06.04.2021', '14.04.2021', 150000)

, (22, 22, 4, 2, '12.08.2021', '19.08.2021', 300000)

, (23, 23, 4, 2, '12.08.2021', '19.08.2021', 300000)

, (24, 24, 4, 2, '12.08.2021', '19.08.2021', 300000)

, (25, 25, 4, 2, '12.08.2021', '19.08.2021', 255000)

, (26, 26, 4, 2, '12.08.2021', '19.08.2021', 255000)

, (27, 27, 5, 3, '02.09.2021', '09.09.2021', 130000)

, (28, 28, 5, 3, '02.09.2021', '09.09.2021', 130000)

, (29, 29, 5, 3, '02.09.2021', '09.09.2021', 130000)

, (30, 30, 5, 3, '02.09.2021', '09.09.2021', 130000)

, (31, 31, 5, 3, '02.09.2021', '09.09.2021', 130000)

, (32, 32, 5, 3, '02.09.2021', '09.09.2021', 130000)

, (33, 33, 6, 4, '10.01.2021', '13.01.2021', 112500)

, (34, 34, 6, 6, '20.03.2021', '27.03.2021', 170000)

, (35, 35, 6, 6, '20.03.2021', '27.03.2021', 170000)

, (36, 36, 6, 6, '20.03.2021', '27.03.2021', 170000)

, (37, 37, 7, 5, '10.07.2021', '17.07.2021', 300000)

, (38, 38, 7, 5, '10.07.2021', '17.07.2021', 300000)

, (39, 39, 7, 3, '16.09.2021', '23.09.2021', 212500)

, (40, 40, 7, 3, '16.09.2021', '23.09.2021', 212500)

, (41, 41, 7, 3, '16.09.2021', '23.09.2021', 212500)

, (42, 42, 7, 3, '16.09.2021', '23.09.2021', 250000)

, (43, 43, 7, 3, '16.09.2021', '23.09.2021', 250000)

, (44, 44, 7, 1, '13.04.2021', '20.04.2021', 187500)

, (45, 45, 7, 1, '13.04.2021', '20.04.2021', 187500)

, (46, 46, 7, 1, '13.04.2021', '20.04.2021', 187500)

, (47, 47, 7, 5, '11.06.2021', '18.06.2021', 250000)

, (48, 48, 7, 5, '11.06.2021', '18.06.2021', 250000)

, (49, 49, 8, 2, '01.07.2021', '07.07.2021', 400000)

, (50, 50, 8, 2, '01.07.2021', '07.07.2021', 400000)

, (51, 51, 8, 2, '01.07.2021', '07.07.2021', 400000)

, (52, 52, 8, 2, '03.09.2021', '10.09.2021', 340000)

, (53, 53, 8, 2, '03.09.2021', '10.09.2021', 340000)

, (54, 54, 8, 2, '03.09.2021', '10.09.2021', 400000)

, (55, 55, 8, 2, '03.09.2021', '10.09.2021', 400000)

, (56, 56, 9, 4, '30.09.2021', '02.10.2021', 112500)

, (57, 57, 10, 6, '30.06.2021', '07.07.2021', 200000)

, (58, 58, 10, 6, '30.06.2021', '07.07.2021', 200000)

, (59, 59, 10, 6, '30.06.2021', '07.07.2021', 200000)

, (60, 60, 11, 3, '20.05.2021', '27.05.2021', 200000)

, (61, 61, 11, 3, '20.05.2021', '27.05.2021', 200000)

, (62, 62, 11, 3, '20.05.2021', '27.05.2021', 200000)

, (63, 63, 11, 3, '20.05.2021', '27.05.2021', 200000)

, (64, 64, 11, 3, '20.05.2021', '27.05.2021', 200000)

, (65, 65, 11, 4, '15.05.2021', '18.05.2021', 112500)

, (66, 66, 11, 4, '20.02.2021', '23.02.2021', 170000)

, (67, 67, 11, 4, '20.02.2021', '23.02.2021', 170000)

, (68, 68, 11, 4, '20.02.2021', '23.02.2021', 170000)

, (69, 69, 11, 5, '04.03.2021', '11.03.2021', 250000)

, (70, 70, 11, 5, '04.03.2021', '11.03.2021', 250000)

, (71, 71, 12, 1, '01.08.2021', '08.08.2021', 150000)

, (72, 72, 12, 1, '01.08.2021', '08.08.2021', 150000)

, (73, 73, 12, 1, '01.08.2021', '08.08.2021', 150000);

*Заполнение таблицы* *Клиенты:*

INSERT INTO Clients

VALUES

(1, 2, 'Игнатов', 'Александр', 'Андреевич', 30, '4718 190567', '43 4563452', '8708691914100201')

, (2, 1, 'Карасев', 'Платон', 'Михайлович', 38, '4763 757066', '48 15377174', '9799038796219976')

, (3, 1, 'Карасева', 'Ольга', 'Степановна', 37, '4371 261018', '47 17696471', '5579722933936865')

, (4, 3, 'Карасев', 'Николай', 'Платонович', 10, '456123', '43 46378010', '5564722533337772')

, (5, 1, 'Трофимов', 'Тимофей', 'Львович' , 25, '4860 983114', '42 59936706', '9575045601930470')

, (6, 1, 'Трофимова', 'Мария', 'Евгеньевна' , 22, '4980 122316', '46 72984406', '6163248351309449')

, (7, 2, 'Аксенова', 'Вера', 'Тимуровна', 18, '4143 559885', '42 78625109', '9032572868745776')

, (8, 2, 'Борисова ', 'Мадина', 'Никитична', 19, '4576 228300', '43 20854550', '2002662422601133')

, (9, 2, 'Иванов', 'Никита', 'Матвеевич', 18, '4688 136261', '45 82408143', '2535416946746409')

, (10, 2, 'Громов', 'Николай ', 'Никитич', 18, '4680 583834', '49 65349032', '7086939399130643')

, (11, 2, 'Шаповалова', 'Валерия ', 'Дмитриевна', 19, '4186 328003', '49 59725778', '2113590700086206')

, (12, 2, 'Казаков', 'Григорий', 'Даниилович', 19, '4124 600699', '47 58531156', '3105091269128024')

, (13, 2, 'Захаров', 'Даниил', 'Семёнович', 19, '4414 918252', '43 68452056', '8821412384975702')

, (14, 2, 'Абрамова', 'Анастасия', 'Константиновна', 18, '4629 677282', '47 65970411', '7496929220855237')

, (15, 1, 'Жуков', 'Степан', 'Павлович', 30, '4836 593326', '47 49398028', '1025697733275592')

, (16, 1, 'Жукова', 'Елена', 'Николаевна', 28, '4357 739019', '49 16616987', '1333439542911946')

, (17, 3, 'Жукова', 'Милана', 'Степановна', 3, '784300', '49 34321801', '9240907272323968')

, (18, 1, 'Демидов', 'Артём', 'Фёдорович', 40, '4328 369592', '45 37406395', '7933687144890428')

, (19, 1, 'Федоров', 'Даниэль', 'Михайлович', 33, '4363 196274', '45 86366820', '9856221170630308')

, (20, 2, 'Сидоров', 'Илья', 'Матвеевич', 21, '4927 695403', '42 71867330', '7567121947649867')

, (21, 2, 'Сидорова', 'Ирина', 'Олеговна', 19, '4686 936035', '44 50436870', '1151208975818008')

, (22, 1, 'Лобанова', 'Любовь', 'Максимовна', 52, '4917 395645', '44 34802932', '5345904560759664')

, (23, 1, 'Кудрявцева', 'Виктория', 'Олеговна', 30, '4329 545012', '40 49640233', '6355144254397601')

, (24, 1, 'Кудрявцев', 'Игорь', 'Викторович', 35, '4451 362490', '49 75123954', '8980638961307705')

, (25, 3, 'Кудрявцева', 'Екатерина', 'Игоревна', 9, '332865', '', '6534164766315371')

, (26, 3, 'Кудрявцев', 'Евгений', 'Игоревич', 1, '906412', '48 97935960', '8036839861888439')

, (27, 1, 'Пастухова', 'Ксения', 'Владиславовна', 65, '4956 111134', '40 46560053', '9091449188068510')

, (28, 1, 'Пастухов', 'Павел', 'Сергеевич', 68, '4066 801563', '47 60574591', '7985885459464044')

, (29, 1, 'Герасимов', 'Матвей', 'Алексеевич', 63, '4978 480133', '47 87325371', '4918445260263980')

, (30, 1, 'Уварова', 'Виктория', 'Павловна', 59, '4957 568155', '45 17288047', '8197987955994905')

, (31, 1, 'Воробьева', 'Арина', 'Степановна', 67, '4373 192254', '47 34697400', '8370045122224838')

, (32, 1, 'Богданов', 'Даниил', 'Артёмович', 67, '4117 482375', '40 45551309', '8114096621982754')

, (33, 2, 'Верещагина', 'Александра', 'Егоровна', 38, '4746 240782', '43 21188652', '8488277071621270')

, (34, 2, 'Абрамова', 'Мадина', 'Тимуровна', 21, '4953 705318', '47 99974224', '3126942232716828')

, (35, 2, 'Воронина', 'Мирослава', 'Дмитриевна', 21, '4186 341497', '46 33716633', '1809238934423774')

, (36, 2, 'Макарова', 'Валерия', 'Константиновна', 22, '4962 333736', '40 83918116', '1145262627396732')

, (37, 1, 'Молчанов', 'Степан', 'Павлович', 35, '4743 478895', '46 40659926', '9140659161377700')

, (38, 1, 'Молчанова', 'Виктория', 'Семёновна', 33, '4437 543067', '47 11470946', '1997989343944937')

, (39, 3, 'Никитин', 'Ярослав', 'Антонович', 8, '321688', '42 66929019', '7985320401843638')

, (40, 3, 'Никитин', 'Лев', 'Антонович', 2, '096533', '43 74597054', '3307820853311569')

, (41, 3, 'Никитина', 'Надежда', 'Антоновна', 2, '564312', '44 61428948', '6595146070700139')

, (42, 1, 'Никитин', 'Антон', 'Михайлович', 29, '4610 786996', '42 86557529', '3275560140144080')

, (43, 1, 'Никитина', 'Евгения', 'Викторовна', 30, '4736 312260', '42 48418171', '7005909831263126')

, (44, 2, 'Максимова', 'Екатерина', 'Арсентьевна', 28, '4365 208335', '45 38306760', '6898015023209155')

, (45, 2, 'Фетисова', 'Анна', 'Ивановна', 33, '4412 664861', '47 93387732', '7786992393899710')

, (46, 2, 'Юдин', 'Арсений', 'Эрикович', 30, '4741 144983', '49 99738875', '5822649900335819')

, (47, 1, 'Акимова', 'Алина', 'Егоровна', 25, '4319 781096', '43 19827922', '7376402189023793')

, (48, 1, 'Акимов', 'Сергей', 'Сергеевич', 32, '4791 285875', '41 40890296', '5806151963770390')

, (49, 1, 'Волкова', 'Полина', 'Демидовна', 23, '4783 311225', '42 60674527', '5225202586036176')

, (50, 1, 'Иванова', 'Вера', 'Львовна', 22, '4429 406216', '46 82556841', '5698955565225333')

, (51, 1, 'Соловьева', 'Анна', 'Петровна', 23, '4256 619068', '42 51781383', '3846177686471492')

, (52, 3, 'Тарасов', 'Демид', 'Львович', 10, '654398', '49 24247216', '7036457333713771')

, (53, 3, 'Тарасов', 'Захар', 'Львович', 11, '986533', '41 51871910', '3345701481681317')

, (54, 1, 'Тарасов', 'Лев', 'Демидович', 39, '4893 785605', '46 82435763', '2131440168246627')

, (55, 1, 'Тарасова', 'Надежда', 'Захаровна', 37, '4144 357814', '45 29355884', '6938553766813130')

, (56, 2, 'Савина', 'Мадина', 'Тимофеевна', 42, '4857 507443', '40 32268932', '7276762118097395')

, (57, 1, 'Евсеев', 'Денис', 'Кириллович', 23, '4775 312957', '47 60539178', '1553273796103894')

, (58, 1, 'Чумаков', 'Тимур', 'Викторович', 19, '4122 501297', '41 43413446', '7505893536377699')

, (59, 1, 'Ларионов', 'Денис', 'Михайлович', 22, '4720 277420', '42 96197500', '9489394223317504')

, (60, 1, 'Егоров', 'Михаил', 'Маркович', 40, '4682 847123', '40 18307420', '9905255832243712')

, (61, 1, 'Егорова', 'Валерия', 'Ярославовна', 38, '4229 158819', '44 60360131', '7497071449179203')

, (62, 1, 'Егорова', 'Оксана', 'Михайловна', 17, '4771 635486', '40 43376000', '6683231344446541')

, (63, 1, 'Егорова', 'Марина', 'Михайловна', 16, '4785 800533', '45 29698680', '6137412894982845')

, (64, 1, 'Егоров', 'Константин', 'Михайлович', 14, '4272 894455', '46 49666906', '5173650097101927')

, (65, 2, 'Козырев', 'Никита', 'Максимович', 34, '4879 855666', '45 31449447', '2320512363221496')

, (66, 1, 'Михайлов', 'Артём', 'Данилович', 24, '4146 803573', '45 47707565', '8421695432160051')

, (67, 1, 'Рыбакова', 'Василиса', 'Тимофеевна', 41, '4846 342475', '48 65629117', '6382026652339846')

, (68, 1, 'Гаврилов', 'Егор', 'Владимирович', 35, '4167 104878', '40 78434350', '8790360203944147')

, (69, 1, 'Сорокин', 'Иван', 'Даниилович', 35, '4873 556932', '42 36948931', '8421793999150396')

, (70, 1, 'Сорокина', 'Алёна', 'Андреевна', 25, '4093 969418', '41 72324330', '5495001975912601')

, (71, 1, 'Комаров', 'Илья', 'Даниилович', 25, '4097 944185', '41 78569562', '9388467140495778')

, (72, 1, 'Васильева', 'Дарья', 'Леонидовна', 25, '4384 253644', '41 76304600', '1603601944632828')

, (73, 1, 'Архипов', 'Михаил', 'Владимирович', 25, '4174 614827', '44 69563399', '9284387619700284');

*Заполнение таблицы* *Страны:*

INSERT INTO Countries

VALUES

(1, 1, 'США')

, (2, 2, 'США')

, (3, 3, 'США')

, (4, 4, 'Египет')

, (5, 5, 'Китай')

, (6, 6, 'Таиланд')

, (7, 7, 'Греция')

, (8, 8, 'Япония')

, (9, 9, 'Япония')

, (10, 10, 'Италия')

, (11, 11, 'Италия')

, (12, 12, 'ОАЭ')

, (13, 13, 'Южная Корея')

, (14, 14, 'Индия')

, (15, 15, 'Испания')

, (16, 16, 'Испания')

, (17, 17, 'Мексика');

*Заполнение таблицы* *Города:*

INSERT INTO Cities

VALUES

(1, 'Нью-Йорк')

, (2, 'Майами')

, (3, 'Лас-Вегас')

, (4, 'Каир')

, (5, 'Шанхай')

, (6, 'Пхукет')

, (7, 'Афины')

, (8, 'Осака')

, (9, 'Токио')

, (10, 'Венеция')

, (11, 'Рим')

, (12, 'Дубай')

, (13, 'Сеул')

, (14, 'Дели')

, (15, 'Мадрид')

, (16, 'Барселона')

, (17, 'Канкун');

*Заполнение таблицы* *Программы:*

INSERT INTO Programs

VALUES

(1, 'Спортивный тур')

, (2, 'Курортно-оздоровительный тур')

, (3, 'Экскурсионный тур')

, (4, 'Деловой тур')

, (5, 'Свадебный тур')

, (6, 'Тематический тур');

*Заполнение таблицы* *Скидки:*

INSERT INTO Discounts

VALUES

(1, 'Нет скидки')

, (2, 'Скидка постоянному клиенту 25%')

, (3, 'Скидка детям до 12 лет 15%');

### Разработка необходимых запросов, представлений (view), процедур, функций и триггеров

*Показать все купленные программы путевок в данный город.*

SELECT DISTINCT Cities.[Name] AS City\_Name

, Countries.[Name] AS Country\_Name

, Programs.[Name] AS [Program\_Name]

FROM Tours

INNER JOIN Countries

ON Tours.ID\_Country = Countries.ID

INNER JOIN Cities

ON Countries.ID\_City = Cities.ID

INNER JOIN Programs

ON Tours.ID\_Program = Programs.ID

WHERE Cities.[Name] LIKE '%рим%';

*Выдавать список стран, популярных у определенного возраста клиентов.*

SELECT DISTINCT Countries.[Name] AS Country\_Name

FROM Countries

INNER JOIN Tours

ON Tours.ID\_Country = Countries.ID

INNER JOIN Clients

ON Tours.ID\_Client = Clients.ID

WHERE Clients.Age BETWEEN 14 AND 25;

*Показать список заказанных путёвок с определённой программой.*

SELECT Tours.ID

, Countries.[Name] AS Country\_Name

, Cities.[Name] AS City\_Name

, Programs.[Name] AS [Program\_Name]

FROM Tours

INNER JOIN Countries

ON Tours.ID\_Country = Countries.ID

INNER JOIN Cities

ON Countries.ID\_City = Cities.ID

INNER JOIN Programs

ON Tours.ID\_Program = Programs.ID

WHERE Programs.[Name] LIKE '%деловой%';

*Показать данные всех клиентов, попадающих под категорию "Дети младше 12-ти лет", а также страны и города, которые они посещали.*

SELECT Clients.ID

, Clients.Surname

, Clients.[Name]

, Clients.Middle\_Name

, Clients.Age

, Clients.Passport

, Clients.International\_Passport

, Clients.Medical\_Insurance

, Countries.[Name] AS Country\_Name

, Cities.[Name] AS City\_Name

FROM Clients

INNER JOIN Tours

ON Tours.ID\_Client = Clients.ID

INNER JOIN Countries

ON Tours.ID\_Country = Countries.ID

INNER JOIN Cities

ON Countries.ID\_City = Cities.ID

WHERE Clients.Age < 12;

*Показать все цены на путёвки (без скидок) в порядке убывания.*

SELECT DISTINCT Countries.[Name] AS Country\_Name

, Cities.[Name] AS City\_Name

, Programs.[Name] AS [Program\_Name]

, Price

FROM Tours

INNER JOIN Countries

ON Tours.ID\_Country = Countries.ID

INNER JOIN Cities

ON Countries.ID\_City = Cities.ID

INNER JOIN Programs

ON Tours.ID\_Program = Programs.ID

INNER JOIN Clients

ON Tours.ID\_Client = Clients.ID

INNER JOIN Discounts

ON Clients.ID\_Discount = Discounts.ID

WHERE Discounts.[Name] LIKE '%нет%'

ORDER BY Price DESC;

# Список литературы

**Visual Paradigm** Entity Relation [Online] // Visual Paradigm Online. - @2020 by Visual Paradigm, 2020. - 2 2, 2020. - https://online.visual-paradigm.com/app/diagrams/#diagram:proj=0&type=ERDiagram.

ГОСТ 7.32-2017. Межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления [В Интернете] // Официальный сайт компании «КонсультантПлюс». - 24 Март 2020 г.. - http://www.consultant.ru/document/cons\_doc\_LAW\_292293/.