**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

**«Информационное обеспечение систем управления»**

на тему **«РБД «Туристическое бюро»**

Выполнил: ст. гр. ТУУ-411  
Гусева О.О.  
Проверил: доцент, к.т.н. Васильева М.А.

Москва 2023

Оглавление

[ЦЕЛЬ КУРСОВОГО ПРОЕКТА\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 3](#_Toc130762116)

[ЗАДАНИЕ НА КУРСОВОВОЙ ПРОЕКТ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 3](#_Toc130762117)

[ПРОЕКТИРОВАНИЕ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ\_\_\_\_ \_\_\_\_\_4](#_Toc130762118)

[1 Инфологическое проектирование\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_4](#_Toc130762119)

[1.1 Анализ предметной области\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_4](#_Toc130762120)

[1.2 Анализ информационных задач и круга пользователей системы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_4](#_Toc130762121)

[2 Логическое проектирование реляционной БД\_\_\_\_\_\_\_\_\_\_\_\_\_\_5](#_Toc130762122)

[2.1 Составление реляционных отношений\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_5](#_Toc130762123)

[2.2 Нормализация полученных отношений\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_8](#_Toc130762124)

[3 Физическое проектирование БД\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_8](#_Toc130762125)

[3.1 Разработка скриптов на создание базы данных и таблиц\_\_\_\_8](#_Toc130762126)

[3.2 Разработка скриптов на добавление данных в таблицы\_\_\_\_\_10](#_Toc130762127)

[3.3 Разработка необходимых запросов, представлений (view), процедур, функций и триггеров\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_11](#_Toc130762128)

[Список литературы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 14](#_Toc130762129)

# 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 КУРСОВОЙ 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.

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.).

Готовые запросы:

1. Выдавать список стран.
2. Выдавать список городов.
3. Показывать весь ассортимент путевок в данное место.
4. Показывать список самых популярных путевок по содержанию.

# ПРОЕКТИРОВАНИЕ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ

## Инфологическое проектирование

### Анализ предметной области

База данных создаётся для информационного обслуживания сотрудников фирмы, оказывающей услуги в туристическом бизнесе.

Выделим базовые сущности этой предметной области:

* Страны
* Города
* Путёвки
* Типы отдыха
* Клиенты

ER-диаграмма БД приведена на рисунке 1.

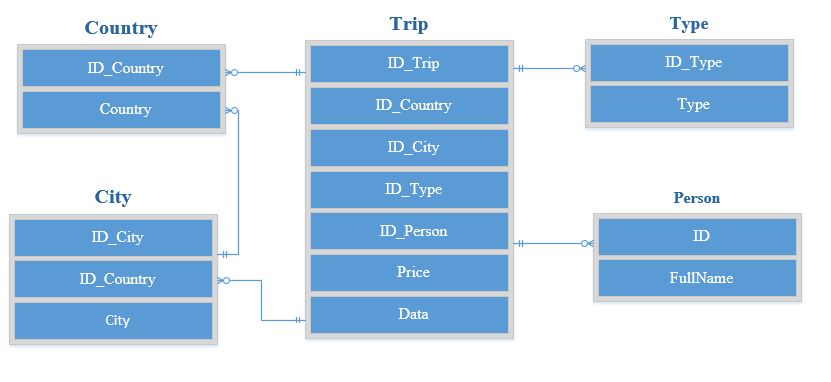


Рисунок 1 – ER-диаграмма

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

### Анализ информационных задач и круга пользователей системы

Определим границы функциональной возможности пользователей:

* ведение БД (запись, чтение, модификация, удаление в архив);
* обеспечение логической непротиворечивости БД;
* реализация наиболее часто встречающихся запросов в готовом виде.

## 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 **Города** (**City**) совокупность атрибутов ID\_Country и City является уникальной и является потенциальным ключом. Для единообразия именования для этой сущности в качестве ключевого атрибута введем суррогатный ключ ID\_City, который будет являться ключом для атрибута «ID\_City» сущности **Путёвки** (**Trip**).
2. Для сущности **Страны** (**Country**) потенциальным ключом является атрибут «Country», т.к. его значение является уникальным. Но значениями этого атрибута будут символьные строки, поэтому введем для этой сущности суррогатный ключ ID\_Country, который будет являться ключом для атрибута «ID\_Country» сущности **Путёвки** (**Trip**) и ключом для атрибута «ID\_Country» сущности **Города** (**City**).
3. Для сущности **Типы отдыха** (**Type**) потенциальным ключом является атрибут «Type», т.к. его значение является уникальным. Но значениями этого атрибута будут символьные строки, поэтому введем для этой сущности суррогатный ключ ID\_Type, который будет являться ключом для атрибута «ID\_Type» сущности **Путёвки** (**Trip**).
4. Для сущности **Клиенты** (**Person**) потенциальным ключом является атрибут «FullName», т.к. его значение является уникальным. Но значениями этого атрибута будут символьные строки, поэтому введем для этой сущности суррогатный ключ ID, который будет являться ключом для атрибута «ID\_Person» сущности **Путёвки** (**Trip**).
5. Для сущности **Путёвки** (**Trip**) совокупность атрибутов является уникальной и является потенциальным ключом. Для единообразия именования для этой сущности в качестве ключевого атрибута введем суррогатный ключ ID\_Trip.

**Таблица 1 – Схема отношения Города** (**City**)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID\_City | Целое число | первичный ключ |
| Страна | ID\_Country | Целое число | обязательное поле |
| Город | City | Строка (30) | уникальное, обязательное поле |

**Таблица 2 – Схема отношения Страны** (**Country**)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID\_Country | Целое число | первичный ключ |
| Страна | Country | Строка (30) | уникальное, обязательное поле |

**Таблица 3 – Схема отношения Типы отдыха** (**Type**)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID\_Type | Целое число | первичный ключ |
| Тип отдыха | Type | Строка (30) | уникальное, обязательное поле |

**Таблица 4 – Схема отношения Клиенты** (**Person**)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID | Целое число | первичный ключ |
| Полное имя | FullName | Строка (100) | обязательное поле |

Таблица 5 – Схема отношения Путёвки (Trip)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | ID | Целое число | первичный ключ |
| Страна | ID\_Country | Целое число |  |
| Город | ID\_City | Целое число |  |
| Тип отдыха | ID\_Type | Целое число |  |
| Клиент | ID\_Person | Целое число |  |
| Цена | Price | Десятичное число |  |
| Дата | Data | Дата |  |

### 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 «TravelAgency»:*

CREATE DATABASE TravelAgency;

*--Создание таблицы Страны:*

CREATE TABLE Country  
(  
 ID\_Country INT PRIMARY KEY  
 , Country NVARCHAR(30) NOT NULL UNIQUE  
);

*--Создание таблицы Города:*

CREATE TABLE City  
(  
 ID\_City INT PRIMARY KEY

, ID\_Country INT FOREIGN KEY REFERENCES Country(ID\_Country)  
 , City NVARCHAR(30) NOT NULL UNIQUE  
);

*--Создание таблицы Типы отдыха:*

CREATE TABLE [Type]  
(  
 ID\_Type INT PRIMARY KEY  
 , Type NVARCHAR(30) NOT NULL UNIQUE  
);

*--Создание таблицы Клиенты:*

CREATE TABLE Person  
(  
 ID INT PRIMARY KEY  
 , FullName NVARCHAR(100) NOT NULL  
);

*--Создание таблицы Путёвки:*

CREATE TABLE Trip  
(  
 ID\_Trip INT PRIMARY KEY  
 , ID\_Country INT FOREIGN KEY REFERENCES Country(ID\_Country)  
 , ID\_City INT FOREIGN KEY REFERENCES City (ID\_City)  
 , ID\_Type INT FOREIGN KEY REFERENCES Type (ID\_Type)  
 , ID\_Person INT FOREIGN KEY REFERENCES Person (ID)  
 , Price MONEY  
 , [Data] DATE  
);

Построим диаграмму спроектированной базы данных (рисунок 2).

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 2 - Диаграмма базы данных

### Разработка скриптов на добавление данных в таблицы

*--Заполнение таблицы Страны:*

INSERT INTO Country VALUES  
(1, N'Россия'), (2, N'Беларусь'), (3, N'Казахстан');

*--Заполнение таблицы Города:*

INSERT INTO City VALUES

(1, 1, N'Москва'), (2, 1, N'Екатеринбруг')

, (3, 1, N'Мурманск'), (4, 1, N'Воронеж')

, (5, 2, N'Минск'), (6, 2, N'Гомель')

, (7, 2, N'Брест'), (8, 3, N'Астана')

, (9, 3, N'Казалинск'), (10, 3, N'Сергеевка');

*--Заполнение таблицы Типы отдыха:*

INSERT INTO [Type] VALUES

(1, N'Экскурсия'), (2, N'Туризм')

, (3, N'Спортивный'), (4, N'Лечебный'), (5, N'Пляжный');

*--Заполнение таблицы Клиенты:*

INSERT INTO Person VALUES

(1, N'Александр Сергеевич Пушкин')

, (2, N'Фёдор Михайлович Достоевский')

, (3, N'Николай Васильевич Гоголь')

, (4, N'Лев Николаевич Толстой')

, (5, N'Иван Сергеевич Тургенев');

*--Заполнение таблицы Путёвки:*

INSERT INTO Trip VALUES

(1, 2, 7, 2, 5, 24500.50, '2023-07-26')

, (2, 3, 10,1, 1, 18400, '2023-02-14')

, (3, 1, 4, 1, 2, 9670.20, '2023-12-02')

, (4, 3, 8, 5, 3, 30800, '2024-10-11')

, (5, 2, 5, 2, 1, 5700.60, '2024-05-07')

, (6, 1, 3, 3, 4, 12100.10, '2013-09-19')

, (7, 1, 1, 1, 5, 7770.40, '2024-11-22')

, (8, 2, 7,2, 2, 32000, '2023-06-09')

, (9, 1, 3, 4, 1, 24900, '2023-03-17');

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

*Представление (Countries) Страны (просмотр всех стран, в которые есть путёвки):*

CREATE VIEW Countries AS

SELECT Country FROM Country;

Результат выполнения следующего скрипта представлен на рисунке 3.

SELECT \* FROM Countries;

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 3 – Список стран

*Представление (Cities) Города (просмотр всех городов, в которые есть путёвки):*

CREATE VIEW Cities AS

SELECT City FROM City;

Результат выполнения следующего скрипта представлен на рисунке 4.

SELECT \* FROM Cities;

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 4 – Список городов

*Функция (FindTrip) Путёвки (просмотр путёвок в определённом городе):*

CREATE FUNCTION FindTrip ( @City VARCHAR(30))

RETURNS TABLE AS RETURN

SELECT Country, City, Type, Price, Data

FROM Trip, Country, City, Type

WHERE Trip.ID\_City=City.ID\_City

AND Trip.ID\_Country=Country.ID\_Country

AND Trip.ID\_Type=Type.ID\_Type

AND City = @City;

Результат выполнения следующего скрипта представлен на рисунке 5.

SELECT \* FROM FindTrip(N'Мурманск');

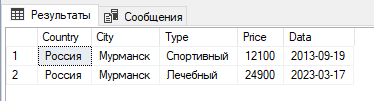


Рисунок 5 – Список путёвок

*Процедура (TopTypes) Топ видов (просмотр самых популярных путёвок по виду отдыха):*

CREATE PROCEDURE TopTypes AS

BEGIN

SELECT \* FROM

(

SELECT Type, COUNT(\*)AS TripsCount

FROM Trip, Type WHERE Trip.ID\_Type=Type.ID\_Type

GROUP BY Trip.ID\_Type, Type

)AS counts

WHERE TripsCount=

(

SELECT TOP 1 COUNT(\*)

FROM Trip

GROUP BY ID\_Type

ORDER BY COUNT(\*) DESC

)

END;

Результат выполнения следующего скрипта представлен на рисунке 6.

EXEC TopTypes;

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 6 – Список популярных типов отдыха

*Триггер (PreventDropTable) предотвращает удаление таблиц из БД:*

CREATE TRIGGER PreventDropTable

ON DATABASE FOR DROP\_TABLE

AS

RAISERROR (N'Попытка удаления таблицы', 10, 1);

ROLLBACK;

Результат выполнения следующего скрипта представлен на рисунке 7.

DROP TABLE Trip;

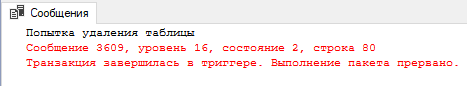


Рисунок 7 – Попытка удаления таблицы

# 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/.