Проект

Разработка приложения для предметной области «путешественник»

1. **Описание предметной области**

**В городах есть достопримечательности, и у каждого места есть название. Есть один или несколько путешественников. Человек в этом месте является путешественником, и в записи хранится такая информация, как фамилия, адрес и т. д. И путешественник был в нескольких местах, а информация о путешественнике и дата сохраняются в записи.**

1. **Проектирование базы данных**

**Процесс проектирования базы данных представляет собой последовательность переходов от словесного описания информационной структуры предметной области к**

**формализованному описанию объектов предметной области в терминах некоторой модели. В общем случае выделяют следующие этапы проектирования:**

**1. Анализ и описание предметной области информационной системы.**

**2. Концептуальное моделирование.**

**3. Построение логической модели**

**4. Построение физической модели**

* 1. **Концептуальная модель**

**В этом разделе необходимо дать частично формализованное описание объектов**

**предметной области в терминах некоторой семантической модели, например, в терминах ER-модели (модели «сущность-связь»).**

**Данная концептуальная модель обеспечивает интегральное представление о предметной области и имеет слабо формализованный характер, отображает информационные объекты, свойства и связи между ними.**

**Согласно ER-модели, информационный объект изображается на диаграмме в виде прямоугольника, содержащего имя объекта. Связь изображается линией, которая**

**связывает два информационных объекта, участвующих в отношении. Степень конца связи указывается графически, множественность связи изображается в виде «стрелки» на**

**конце связи. Таким образом, для связи «один-к-одному» используется линия без стрелки,**

**для связи «один-ко-многим» – линия со стрелкой на стороне второго объекта, для связи «многие-ко-многим» – линия со стрелками на сторонах обоих объектов. При необходимости Атрибуты информационного объекта записываются внутри прямоугольника, его**

**изображающего.**

**Для построения ER-диаграммы можно использовать online средство www.draw.io,**

**находящееся в открытом доступе.**

**На основе анализа предметной области «путешественник», были выделены**

**следующие информационные объекты, которые необходимо хранить в базе**

**данных: city, country, transport, traveler, place.**

**Каждый из выделенных информационных объектов имеет следующие**

**атрибуты:**

**city– название city;**

**country – название country;**

**transport – название transport;**

**traveler - фамилия, имя, отчество, адрес;**

**place – название place;**

**На рисунке 1 приведена концептуальная модель предметной области «путешественник» в виде ER-диаграммы.**

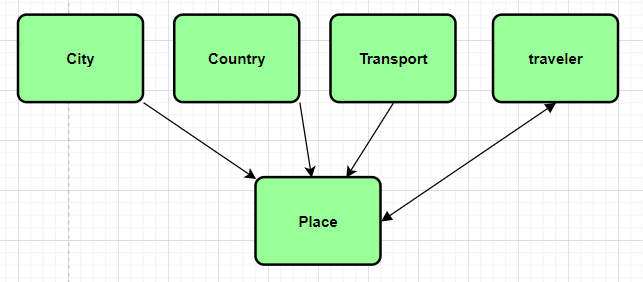
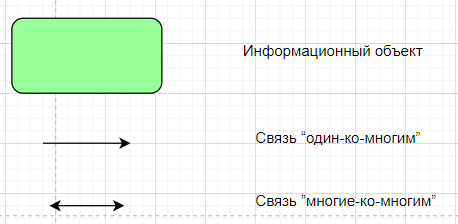


Рис. 1Концептуальная схема базы данных «путешественник»

Для этой диаграммы приняты следующие обозначения:



* 1. **Логическая модель**

Логическое проектирование базы данных – это описание базы данных в терминах принятой логической модели данных, в нашем случае – реляционной. Результат – схема реляционной базы данных, в которой информационные объекты и связи, полученные на этапе концептуального проектирования, представляются в виде, допустимом для реализации средствами реляционной модели. Полученные информационные объекты должны отвечать требованиям нормализации.

**На рисунке 2 приведена логическая модель базы данных «путешественник»**

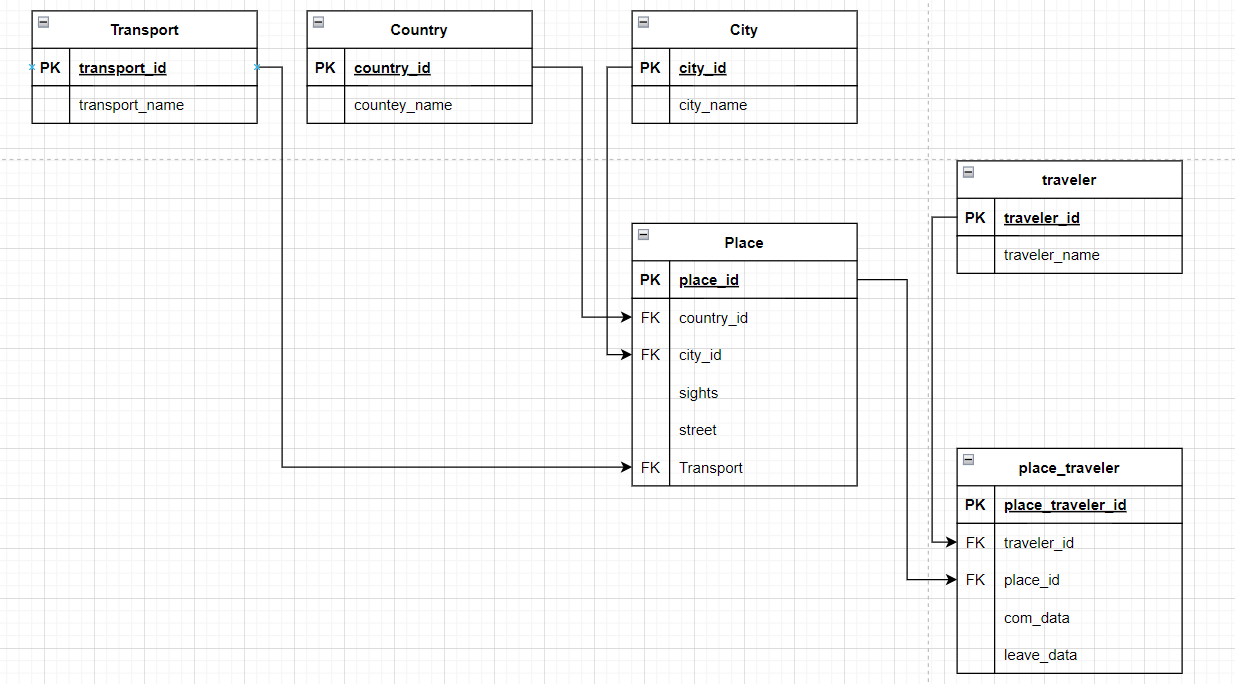


Рис. 2Логическая схема базы данных «путешественник»

* 1. **Физическая модель**

**В данном разделе приводится подробное описание каждого информационного объекта: атрибутов и их типов, а также указываются первичные и внешние ключи.**

**В результате проектирования базы данных были получены информационные объекты, реализованные в виде реляционных таблиц. Ниже приведены**

**SQL-запросы для их создания, в Приложении А приведено наполнение таблиц базы данных.**

CREATE TABLE if not exists city (

city\_id INTEGER PRIMARY KEY AUTOINCREMENT,

city\_name VARCHAR(30)

);

CREATE TABLE if not exists country (

country\_id INTEGER PRIMARY KEY AUTOINCREMENT,

country\_name VARCHAR(40)

);

CREATE TABLE if not exists transport (

transport\_id INTEGER PRIMARY KEY AUTOINCREMENT,

transport\_name VARCHAR(30)

);

CREATE TABLE if not exists traveler (

traveler\_id INTEGER PRIMARY KEY AUTOINCREMENT,

traveler\_name VARCHAR(30)

);

CREATE TABLE if not exists place (

place\_id INTEGER PRIMARY KEY AUTOINCREMENT,

country\_id INT,

city\_id int,

street VARCHAR(80),

sights VARCHAR(80),

transport\_id int,

FOREIGN KEY (country\_id) REFERENCES country (country\_id) ON DELETE CASCADE,

FOREIGN KEY (city\_id) REFERENCES city (city\_id) ON DELETE CASCADE

);

CREATE TABLE if not exists place\_traveler (

place\_traveler\_id INTEGER PRIMARY KEY AUTOINCREMENT,

place\_id int,

traveler\_id INT,

com\_date DATE,

leave\_date DATE,

FOREIGN KEY (place\_id) REFERENCES place (place\_id) ON DELETE CASCADE,

FOREIGN KEY (traveler\_id) REFERENCES traveler (traveler\_id) ON DELETE CASCADE

);

ПРИЛОЖЕНИЕ A. Наполнение таблиц базы данных

INSERT INTO city (city\_name) VALUES

('Shang Hai'),

('Ha Erbing'),

('Cheng Du'),

('Bei Jing'),

('Lan Zhou'),

('Moscow'),

('Vladivostok'),

('Saint Petersburg'),

('Hui Chun'),

('Hai Nan'),

('Chong Qing');

INSERT INTO country (country\_name) VALUES

('China'),

('Russia');

INSERT INTO transport (transport\_name) VALUES

('plain'),

('car'),

('train');

INSERT INTO traveler (traveler\_name) VALUES

('Иванов М.С.'),

('Петров Ф.С.'),

('Федоров П.Р.'),

('Абрамова А.А.'),

('Самарин С.С.'),

('Туполев И.Д.'),

('Баранов П.В.');

INSERT INTO place(country\_id, city\_id, street, sights, transport\_id) VALUES

(1, 1, 'The Bund', 60000, 1),

(1, 2, 'Central Avenue', 10000, 2),

(1, 3, 'Du jiang yan', 30000, 1),

(1, 4, 'Wangfujing Street', 50000, 1),

(1, 5, 'Baita Mountain', 40000, 1),

(2, 6, 'Red Square', 80000, 3),

(2, 7, 'Oceanarium', 40000, 1),

(2, 8, 'Winter Palace', 80000, 1),

(1, 9, 'Fangchuan Scenic Area', 30000, 3),

(1, 10, 'Nanshan Temple', 80000, 1),

(1, 11, 'Fengdu Ghost Town', 40000, 1);

INSERT INTO place\_traveler(place\_id, traveler\_id, com\_date, leave\_date) VALUES

( 2, 4, '2020-09-11', '2020-09-24'),

( 11, 6, '2020-09-11', '2020-09-18'),

( 3, 5, '2020-09-17', '2020-10-10'),

( 4, 6, '2020-09-20', '2020-10-14'),

( 5, 4, '2020-09-26', '2020-10-04'),

( 11, 1, '2020-09-21', '2020-10-09'),

( 10, 4, '2020-10-10', '2020-10-25'),

( 1, 3, '2020-09-28', '2020-10-07'),

( 2, 2, '2020-09-30', '2020-10-08'),

( 7, 1, '2020-10-15', '2020-10-25'),

( 9, 4, '2020-10-27', '2020-11-25'),

( 6, 7, '2020-10-06', '2020-10-26'),

( 11, 1, '2020-11-06', '2020-11-13'),

( 4, 3, '2020-10-18', '2020-10-29'),

( 7, 2, '2020-10-15', '2020-10-27'),

( 2, 7, '2020-10-29', '2020-11-12'),

( 3, 5, '2020-10-14', '2020-10-27');