

Проект

Разработка приложения для предметной области «Библиотека»

1. Описание предметной области

В некоторой библиотеке хранятся книги. Каждая книга имеет название, относится к одному жанру, имеет одного или нескольких авторов, издана в определенном издательстве. В библиотеке может быть несколько экземпляров одной книги. Каждый экземпляр имеет уникальный шифр.

Пользователями библиотеки являются читатели, о которых в карточках хранится информация о фамилии, адресе и телефоне. Каждый читатель может взять несколько экземпляров книг, которые имеются в наличии. При этом в личной карточке читателя сохраняются сведения о взятой книге и дате ее получения. При возврате книги в карточке читателя указывается дата возврата, и данный экземпляр книги становится доступным для других пользователей.

2. 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*

2.1. 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 «Библиотека», были выделены следующие информационные объекты, которые необходимо хранить в базе данных: КНИГА, ЖАНР, ИЗДАТЕЛЬСТВО, АВТОР, ЭКЗЕМПЛЯР, ЧИТАТЕЛЬ, УЛИЦА.

Каждый из выделенных информационных объектов имеет следующие атрибуты:

ЖАНР – название жанра;

ИЗДАТЕЛЬСТВО – название издательства;

АВТОР – фамилия, имя, отчество автора;

ЧИТАТЕЛЬ – фамилия, имя, отчество, телефон, адрес;

КНИГА – название книги, год издания;

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

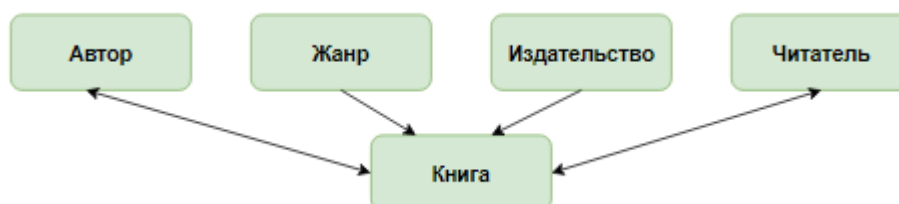
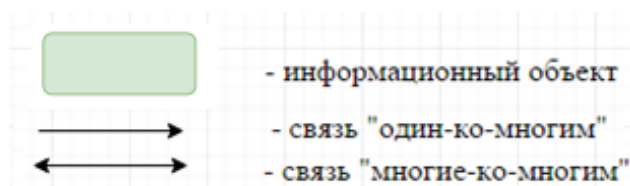


Рис. 1. Концептуальная схема базы данных «Библиотека»

Для этой диаграммы приняты следующие обозначения:



2.2. 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 «Библиотека».

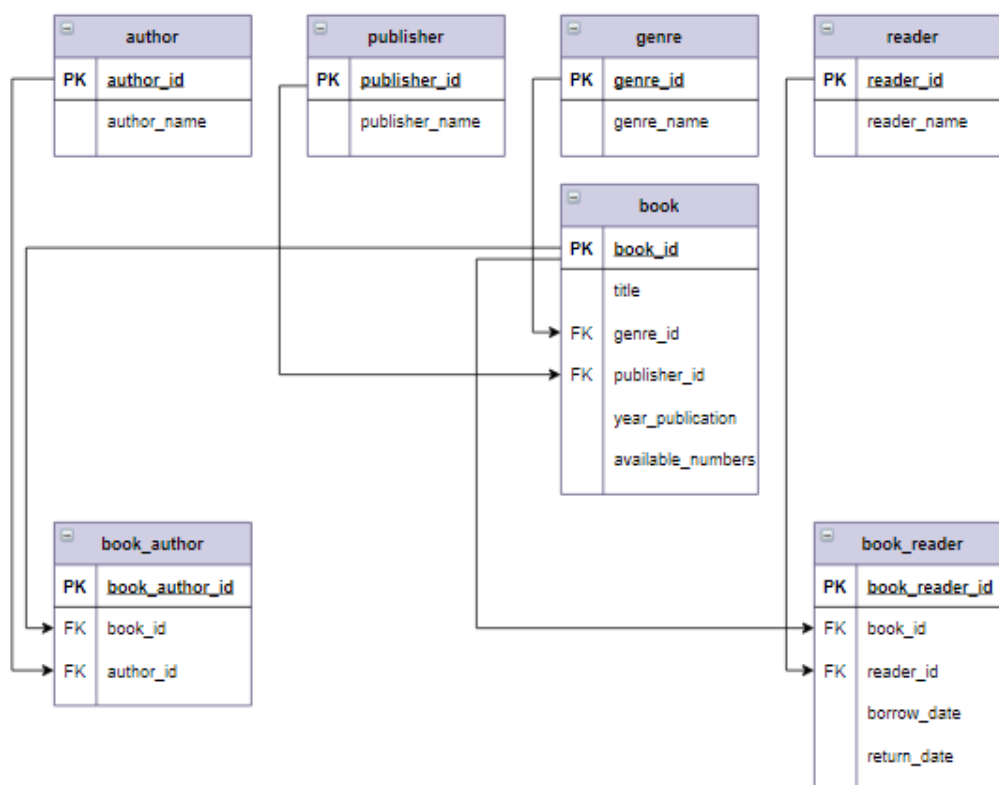


Рис. 2. Логическая схема базы данных «Библиотека»

2.3. 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 author (
    author_id INTEGER PRIMARY KEY AUTOINCREMENT,
    author_name VARCHAR(30)
);

CREATE TABLE IF NOT EXISTS genre (
    genre_id INTEGER PRIMARY KEY AUTOINCREMENT,
    genre_name VARCHAR(30)
);

CREATE TABLE IF NOT EXISTS publisher (
    publisher_id INTEGER PRIMARY KEY AUTOINCREMENT,
    publisher_name VARCHAR(40)
);

CREATE TABLE IF NOT EXISTS reader (
    reader_id INTEGER PRIMARY KEY AUTOINCREMENT,
    reader_name VARCHAR(30)
);

CREATE TABLE IF NOT EXISTS book (
    book_id INTEGER PRIMARY KEY AUTOINCREMENT,
    title VARCHAR(80),
    genre_id int,
    publisher_id INT,
    year_publication INT,
    available_numbers INT,
    FOREIGN KEY (genre_id) REFERENCES genre (genre_id) ON DELETE CASCADE,
    FOREIGN KEY (publisher_id) REFERENCES publisher (publisher_id) ON
DELETE CASCADE
);

CREATE TABLE IF NOT EXISTS book_author (
    book_author_id INTEGER PRIMARY KEY AUTOINCREMENT,
    book_id INT,
    author_id INT,
    FOREIGN KEY (book_id) REFERENCES book (book_id) ON DELETE CASCADE,
    FOREIGN KEY (author_id) REFERENCES author (author_id) ON DELETE
CASCADE
);

CREATE TABLE IF NOT EXISTS book_reader (
    book_reader_id INTEGER PRIMARY KEY AUTOINCREMENT,
    book_id int,
    reader_id INT,
    borrow_date DATE,
    return_date DATE,
    FOREIGN KEY (book_id) REFERENCES book (book_id) ON DELETE CASCADE,
    FOREIGN KEY (reader_id) REFERENCES reader (reader_id) ON DELETE
CASCADE
);

```

ПРИЛОЖЕНИЕ А. Наполнение таблиц базы данных

```
INSERT INTO author (author_name) VALUES
```

```
('Пушкин А.С. '),  
( 'Агата Кристи'),  
( 'Жюль Верн'),  
( 'Ильф И.А. '),  
( 'Петров Е.П. '),  
( 'Булгаков М.А. '),  
( 'Лермонтов М.Ю. '),  
( 'Стругацкий А.Н. '),  
( 'Стругацкий В.Н. '),  
( 'Дойл Артур Конан'),  
( 'Достоевский Ф.М. ');
```

```
INSERT INTO genre (genre_name) VALUES
```

```
('Роман'),  
( 'Приключения'),  
( 'Детектив'),  
( 'Лирика'),  
( 'Фантастика'),  
( 'Фэнтези');
```

```
INSERT INTO publisher (publisher_name) VALUES
```

```
('ЭКСМО'),  
( 'ПИТЕР'),  
( 'РОСМЭН'),  
( 'АЛЬФА-КНИГА'),  
( 'ДРОФА'),  
( 'АСТ'),  
( 'НАУКА');
```

```
INSERT INTO reader (reader_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 book(title, genre_id, publisher_id, year_publication, available_numbers) VALUES
```

```
('Двенадцать стульев', 1, 3, 2018, 1),  
( 'Золотой теленок', 1, 1, 2014, 3),  
( 'Мастер и Маргарита', 1, 5, 2014, 5),  
( 'Таинственный остров', 2, 4, 2015, 0),  
( 'Пуаро ведет следствие', 3, 5, 2008, 2),  
( 'Евгений Онегин', 1, 4, 2011, 8),  
( 'Бородино', 4, 6, 2015, 0),  
( 'Трудно быть богом', 5, 5, 2018, 2),  
( 'Пикник на обочине', 5, 3, 2018, 9),  
( 'Дубровский', 1, 5, 2020, 7),  
( 'Собачье сердце', 1, 5, 2015, 6),  
( 'Понедельник начинается в субботу', 5, 3, 2012, 2),  
( 'Вокруг света за 80 дней', 2, 5, 2019, 5),  
( 'Смерть на Ниле', 3, 1, 2017, 8),  
( 'Убийства по алфавиту', 3, 4, 2017, 9),  
( 'Загадочное происшествие', 3, 5, 2018, 5),  
( 'Капитанская дочка', 1, 4, 2016, 5),  
( 'Этюд в багровых тонах', 3, 4, 2020, 0),  
( 'Приключения Шерлока Холмса', 3, 3, 2013, 1),
```

```

('Записки о Шерлоке Холмсе', 3, 5, 2015, 1),
('Затерянный мир', 2, 2, 2020, 0),
('Стихи', 4, 1, 2019, 6),
('Поэмы', 4, 6, 2020, 8),
('Герой нашего времени', 1, 6, 2017, 2),
('Стихи', 4, 3, 2017, 5),
('Одноэтажная Америка', 1, 2, 2015, 3),
('Смерть поэта', 4, 3, 2020, 2),
('Поэмы', 4, 4, 2019, 5),
('Скрюченный домишко', 3, 5, 2018, 2);

```

```

INSERT INTO book_author(book_id, author_id) VALUES

```

```

(1, 4),
(1, 5),
(2, 4),
(2, 5),
(3, 6),
(4, 3),
(5, 2),
(6, 1),
(7, 7),
(8, 8),
(8, 9),
(9, 8),
(9, 9),
(10, 1),
(11, 6),
(12, 8),
(12, 9),
(13, 3),
(14, 2),
(15, 2),
(16, 2),
(17, 1),
(18, 10),
(19, 10),
(20, 10),
(21, 10),
(22, 1),
(23, 1),
(24, 7),
(25, 7),
(26, 4),
(26, 5),
(27, 7),
(28, 7),
(29, 2);

```

```

INSERT INTO book_reader(book_id, reader_id, borrow_date, return_date) VALUES

```

```

( 4, 4, '2020-09-11', '2020-09-24'),
( 12, 6, '2020-09-11', Null),
( 29, 5, '2020-09-17', '2020-10-10'),
( 27, 6, '2020-09-18', '2020-10-14'),
( 15, 4, '2020-09-18', '2020-10-04'),
( 18, 1, '2020-09-21', '2020-10-09'),
( 22, 4, '2020-09-25', '2020-10-10'),
( 1, 3, '2020-09-28', '2020-10-07'),
( 26, 2, '2020-09-30', '2020-10-08'),
( 27, 1, '2020-10-05', '2020-10-13'),
( 18, 2, '2020-10-06', Null),
( 26, 2, '2020-10-06', '2020-10-26'),
( 11, 1, '2020-10-06', '2020-10-13'),
( 3, 5, '2020-10-08', '2020-10-27'),
( 20, 6, '2020-10-08', '2020-10-23'),

```

```

( 20, 3, '2020-10-09', '2020-10-29'),
( 2, 2, '2020-10-09', '2020-10-12'),
( 27, 6, '2020-10-13', '2020-10-17'),
( 28, 4, '2020-10-15', '2020-11-04'),
( 5, 1, '2020-10-15', '2020-10-18'),
( 29, 1, '2020-10-15', '2020-10-29'),
( 3, 6, '2020-10-15', Null),
( 9, 6, '2020-10-16', '2020-10-19'),
( 15, 2, '2020-10-17', '2020-11-08'),
( 8, 6, '2020-10-19', Null),
( 18, 1, '2020-10-20', NULL),
( 23, 5, '2020-10-21', Null),
( 4, 5, '2020-10-22', '2020-10-28'),
( 26, 3, '2020-10-23', '2020-10-28'),
( 5, 6, '2020-10-24', '2020-11-01'),
( 8, 2, '2020-10-28', '2020-11-18'),
( 21, 5, '2020-10-30', Null),
( 20, 1, '2020-10-30', '2020-11-26'),
( 14, 4, '2020-10-30', '2020-11-10'),
( 8, 6, '2020-11-01', Null),
( 11, 4, '2020-11-06', '2020-11-26'),
( 28, 6, '2020-11-09', '2020-11-28'),
( 14, 3, '2020-11-14', Null),
( 24, 2, '2020-11-15', Null),
( 5, 6, '2020-11-17', Null),
( 11, 3, '2020-11-21', '2020-12-12'),
( 14, 1, '2020-11-23', Null),
( 4, 5, '2020-11-23', Null),
( 23, 5, '2020-11-28', Null),
( 21, 3, '2020-11-29', '2020-12-21');

```