**Описание предметной области**

В некоторой библиотеке хранятся книги. Каждая книга имеет название, относится к одному жанру, имеет одного автора, издана в определенном издательстве. В библиотеке может быть несколько экземпляров одной книги.

Пользователями библиотеки являются читатели, библиотекари и администраторы, о которых хранится информация о фамилии, имени, отчестве, адресе проживания и номере телефона. Читатель может взять несколько книг на чтение, посмотреть какие книги есть в библиотеке и посмотреть какие книги он брал. Библиотекарь может выдавать книги читателю, просматривать наличие книг в библиотеке, редактировать информацию о читателе, просматривать список читателей, просматривать список должников (читателей, которые не сдали книгу). Администратор может редактировать список книг в библиотеке, редактировать информацию о библиотекаре, редактировать информацию о читателях.

**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

**Концептуальная модель**

В предметной области выделены следующие информационные объекты, которые необходимо хранить в базе данных: КНИГА, ЖАНР, ИЗДАТЕЛЬСТВО, АВТОР, ВЫДАЧА КНИГ, ПОСТАВКА, ПОЛЬЗОВАТЕЛЬ.

Каждый из выделенных информационных объектов имеет следующие атрибуты:

КНИГА – название книги, год издания, id автора, id жанра, цена книги, количество экземпляров;

ЖАНР – название жанра;

ИЗДАТЕЛЬСТВО – название издательства, город;

АВТОР – фамилия, имя, отчество автора;

ВЫДАЧА КНИГ – id читателя, id библиотекаря, дата выдачи книги, наименование книги;

ПОСТАВКА – id издательства, id администратора, дата поставки;

ПОЛЬЗОВАТЕЛЬ – имя, фамилия, отчество, адрес проживания, номер телефона.

Пользователь может иметь следующие роли: читатель, библиотекарь, администратор.

Таблица 1 – Типы и права пользователей

|  |  |
| --- | --- |
| **Пользователь** | **Задачи** |
| Библиотекарь | * Просматривать каталог книг * Редактировать и просматривать информацию о читателях * Редактировать и просматривать информацию о выдаче книг * Просмотр книг взятых, конкретным читателем * Просмотр читателей, у которых подошел срок возврата книг в библиотеку |
| Читатель | * Просматривать каталог книг * Просматривать книги, которые он взял из библиотеки |
| Администратор | * Редактировать и просматривать каталог книг * Редактировать и просматривать информацию о читателях * Редактировать и просматривать список издательств * Просмотр книг взятых, конкретным читателем * Просмотр читателей, у которых подошел срок возврата книг в библиотеку * Просмотр книг, выпущенных конкретным изданием |

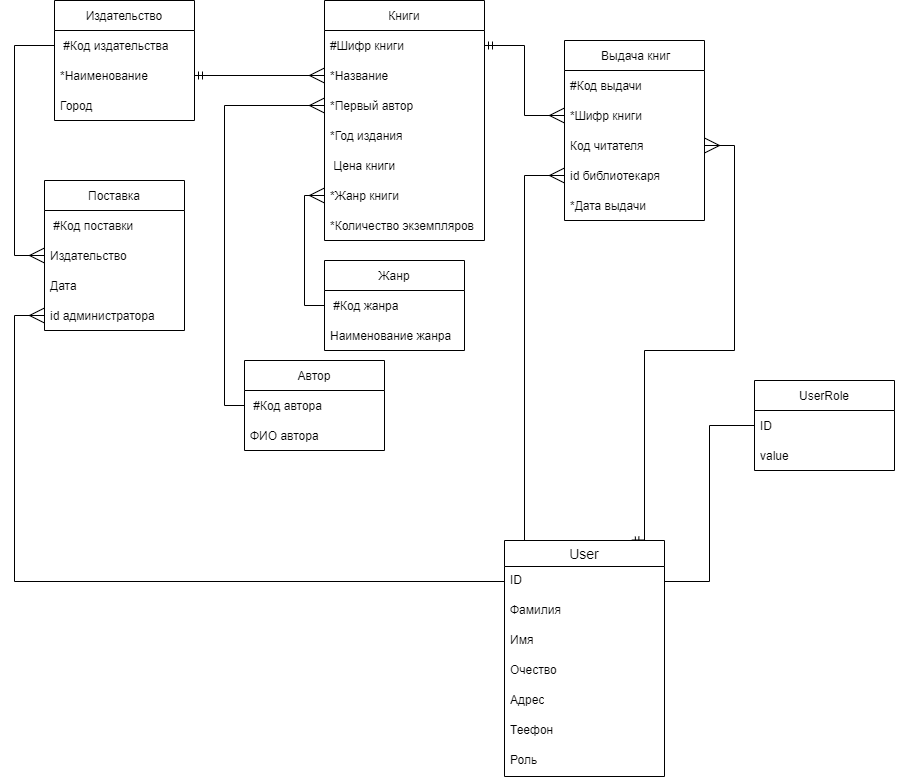
**Логическая модель**

Рисунок 1 - Логическая схема данных "Библиотека"

**Физическая модель**

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.

*# Создаем таблицу Users*cursor.execute('''  
CREATE TABLE IF NOT EXISTS Users (  
id INTEGER PRIMARY KEY,  
lastname TEXT NOT NULL,  
firstname TEXT NOT NULL,  
fathername TEXT,  
Date\_of\_Birth DATA NOT NULL,  
adress TEXT NOT NULL,  
phone\_number INTEGER NOT NULL,  
rool INTEGER NOT NULL  
)  
''')  
  
*# Создаем таблицу User\_role*cursor.execute('''  
CREATE TABLE IF NOT EXISTS User\_role (  
id INTEGER PRIMARY KEY,  
value TEXT NOT NULL  
)  
''')  
  
*# Создаем таблицу Автор*cursor.execute('''  
CREATE TABLE IF NOT EXISTS Author (  
id INTEGER PRIMARY KEY,  
fullname TEXT NOT NULL  
)  
''')  
  
*# Создаем таблицу Жанр*cursor.execute('''  
CREATE TABLE IF NOT EXISTS Genre (  
id INTEGER PRIMARY KEY,  
name\_genre TEXT NOT NULL  
)  
''')  
  
*# Создаем таблицу Издательство*cursor.execute('''  
CREATE TABLE IF NOT EXISTS Publishing\_house (  
id INTEGER PRIMARY KEY,  
name\_Publishing\_house TEXT NOT NULL,  
city TEXT NOT NULL  
)  
''')  
  
*# Создаем таблицу Поставка*cursor.execute('''  
CREATE TABLE IF NOT EXISTS Delivery (  
id INTEGER PRIMARY KEY,  
id\_admin INTEGER NOT NULL,  
id\_Publishing\_house INTEGER NOT NULL,  
data DATA NOT NULL  
)  
''')

**Приложение А**

*# Заполняем таблицу User\_role*user\_role\_data = [(1, 'администратор'),  
 (2, 'библиотекарь'),  
 (3, 'читатель')]  
*#cursor.executemany('''INSERT INTO User\_role (id, value) VALUES (?, ?);''', user\_role\_data)*print(cursor.execute("SELECT \* FROM User\_role").fetchall())  
  
*# Заполняем таблицу Users*user\_data = [(1, 'Козырина', 'Ольга', 'Константиновна', '1994-11-01', 'город Челябинск', 89000809006, 1),  
 (2, 'Морева', 'Светлана', 'Андреевна', '1994-07-19', 'город Челябинск', 89990006688, 1),  
 (3, 'Черепанова', 'Ирина', 'Александровна', '2001-05-25', 'город Калининград', 89045557788, 2),  
 (4, 'Евсеенко', 'Наталья', 'Васильевна', '1975-08-12', 'город Екатеринбург', 89065053396, 2),  
 (5, 'Дектеренко', 'Никита', 'Александрович', '1980-01-24', 'город Екатеринбург', 89961023231, 2),  
 (6, 'Лучевников', 'Егор', 'Борисович', '2010-09-27', 'город Владивосток', 84561234758, 3),  
 (7, 'Загриценко', 'Тимофей', 'Сергеевич', '2009-09-27', 'город Новосибирск', 89552365544, 3),  
 (8, 'Бодаева', 'Вероника', 'Евгеневна', '2008-09-27', 'город Хабаровск', 89995558877, 3),  
 (9, 'Аушкина', 'Елена', 'Олеговна', '2012-06-06', 'город Красноярск', 89223215458, 3),  
 (10, 'Семенова', 'Татьяна', 'Сергеевга', '2007-05-10', 'город Артем', 89428552525, 3)]  
*#cursor.executemany('''INSERT INTO Users (id, lastname, firstname, fathername, Date\_of\_Birth, adress, phone\_number,  
#rool) VALUES (?, ?, ?, ?, ?, ?, ?, ?);''', user\_data)*print(cursor.execute("SELECT \* FROM Users").fetchall())  
  
*# Заполняем таблицу Жанр*genre\_data = [(1, 'детектив'),  
 (2, 'боевик'),  
 (3, 'детские'),  
 (4, 'история'),  
 (5, 'классика'),  
 (6, 'психология'),  
 (7, 'поэзия'),  
 (8, 'фантастика'),  
 (9, 'приключения'),  
 (10, 'ужасы'),  
 (11, 'фэнтези'),  
 (12, 'юмор')]  
*#cursor.executemany('''INSERT INTO Genre (id, name\_genre) VALUES (?, ?);''', genre\_data)*print(cursor.execute("SELECT \* FROM Genre").fetchall())  
  
*# Заполняем таблицу Автор*author\_data = [(1, 'Эрих Мария Ремарк'),  
 (2, 'Николай Гоголь'),  
 (3, 'Лев Толстой'),  
 (4, 'Нил Гейман'),  
 (5, 'Джейн Остин'),  
 (6, 'Дженифер Арментроут'),  
 (7, 'Сара Маас'),  
 (8, 'Агата Кристи'),  
 (9, 'Эдуард Успенский'),  
 (10, 'Корней Иванович Чуковский'),  
 (11, 'Стефани Майер'),  
 (12, 'Джоан Роулинг')]  
*#cursor.executemany('''INSERT INTO Author (id, fullname) VALUES (?, ?);''', author\_data)*print(cursor.execute("SELECT \* FROM Author").fetchall())  
  
*# Заполняем таблицу Издательство*publishing\_house\_data = [(1, 'АСТ', 'Москва'),  
 (2, 'Эксмо', 'Санкт-Петербург'),  
 (3, 'Росмен', 'Екатеринбург'),  
 (4, 'Азбука', 'Новосибирск'),  
 (5, 'Магистраль', 'Москва'),  
 (6, 'Freedom', 'Москва'),  
 (7, 'Clever', 'Екатеринбург')]  
*#cursor.executemany('''INSERT INTO Publishing\_house (id, name\_Publishing\_house, city) VALUES (?, ?, ?);''', publishing\_house\_data)*print(cursor.execute("SELECT \* FROM Publishing\_house").fetchall())  
  
*# Заполняем таблицу Поставка*delivery\_data = [(1, 1, 1, '2015-10-22'),  
 (2, 2, 2, '2017-12-12'),  
 (3, 1, 4, '2018-02-01'),  
 (4, 2, 5, '2019-06-06'),  
 (5, 1, 2, '2022-07-07'),  
 (6, 2, 2, '2023-11-01'),  
 (7, 1, 1, '2021-12-16')]  
*#cursor.executemany('''INSERT INTO Delivery (id, id\_admin, id\_Publishing\_house, data) VALUES (?, ?, ?, ?);''', delivery\_data)*print(cursor.execute("SELECT \* FROM Delivery").fetchall())  
  
*# Заполняем таблицу Книга*book\_data = [(1, 'Убийство в Восточном Экспрессе', 1, 8, 1, '2015-01-01', 500, 10),  
 (2, 'Оникс', 8, 6, 1, '2016-01-01', 400, 15),  
 (3, 'Благие знамения', 8, 4, 2, '2017-05-05', 800, 20),  
 (4, 'Отражение', 8, 6, 2, '2019-08-08', 700, 25),  
 (5, 'Гарри Поттер и Филосовский камень', 11, 12, 3, '2010-11-01', 850, 30),  
 (6, 'Стеклянный трон', 11, 7, 4, '2020-01-01', 350, 35),  
 (7, 'Кролевство крыльев и руин', 11, 7, 4, '2015-01-06', 745, 40),  
 (8, 'Из Крови и пепела', 11, 6, 2, '2017-01-25', 300, 15),  
 (9, 'Три товарища', 5, 1, 5, '2016-05-05', 900, 10),  
 (10, 'Гордость и предубеждение', 5, 5, 5, '2015-05-01', 800, 5),  
 (11, 'На западном фронте без перемен', 5, 1, 5, '2007-07-06', 200, 20),  
 (12, 'Гарри Поттер и Тайная комната', 11, 12, 3, '2007-03-03', 500, 10),  
 (13, 'Триумфальная арка', 5, 1, 5, '2005-03-06', 450, 10),  
 (14, 'Мертвые души', 5, 2, 5, '1990-08-02', 100, 20),  
 (15, 'Анна Каренина', 5, 3, 2, '1990-07-05', 200, 15),  
 (16, 'Смерть на Ниле', 1, 8, 7, '2015-07-07', 300, 25),  
 (17, 'Чебурашка', 3, 9, 7, '2012-02-02', 500, 30),  
 (18, 'Ай Болит', 3, 10, 7, '2010-02-02', 450, 10),  
 (19, 'Сумерки', 11, 11, 6, '2010-04-03', 300, 5),  
 (20, 'Сумерки.Новолуние', 11, 11, 6, '2018-06-06', 600, 12)]  
*#cursor.executemany('''INSERT INTO Book (id, name, id\_genre, id\_author, id\_Publishing\_house, Date, Price, number\_of\_copies)  
 #VALUES (?, ?, ?, ?, ?, ?, ?, ?);''', book\_data)*print(cursor.execute("SELECT \* FROM Book").fetchall())  
  
*# Заполняем таблицу Выдача книг*book\_issuance\_data = [(1, 6, 1, 1, '2023-10-22'),  
 (2, 7, 2, 3, '2023-12-12'),  
 (3, 6, 3, 20, '2023-02-01'),  
 (4, 8, 1, 5, '2023-06-06'),  
 (5, 9, 2, 7, '2023-07-07'),  
 (6, 9, 3, 15, '2023-11-01'),  
 (7, 10, 1, 17, '2023-12-16'),  
 (8, 9, 2, 19, '2023-07-07'),  
 (9, 7, 3, 9, '2023-11-01'),  
 (10, 10, 1, 9, '2023-12-16')]  
*#cursor.executemany('''INSERT INTO bool\_issuance (id, id\_reader, id\_bibl, id\_book, Date) VALUES (?, ?, ?, ?, ?);''', book\_issuance\_data)*print(cursor.execute("SELECT \* FROM bool\_issuance").fetchall())