**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра МО ЭВМ**

**ОТЧЕТ**

**по лабораторной работе №1**

**по дисциплине «Базы данных»**

**Тема: Проектирование ER модели и структуры БД по текстовому**

**описанию предметной области.**

| Студент гр. 2300 |  | Бакланов О.К. |
| --- | --- | --- |
| Преподаватель |  | Заславский М.М. |

Санкт-Петербург

2024

## Цель работы.

Проектирование *ER* модели и структуры БД по текстовому описанию предметной области.

## Задание.

Вариант 2

Пусть требуется создать программную систему, предназначенную для работников библиотеки. Такая система должна обеспечивать хранение сведений об имеющихся в библиотеке книгах, о читателях библиотеки и читальных залах. Для каждой книги в БД должны храниться следующие сведения: название книги, автор (ы), издательство, год издания, число экземпляров этой книги в каждом зале библиотеки, а также шифр книги и дата закрепления книги за читателем. Сведения о читателях библиотеки должны включать номер читательского билета, фамилию читателя, номер паспорта, дату рождения, адрес, номер телефон, образование, наличие ученой степени. Читатели закрепляются за определенным залом и могут записываться и выписываться из библиотеки. Библиотека имеет несколько читальных залов, которые характеризуются номером, названием и вместимостью, то есть количеством людей, которые могут одновременно работать в зале. . Библиотека может получать новые книги и списывать старые. Шифр книги может измениться в результате переклассификации, а номер читательского билета в результате перерегистрации. Библиотекарю могут потребоваться следующие сведения о текущем состоянии библиотеки:

* Какие книги закреплены за определенным читателем?
* Как называется книга с заданным шифром?
* Какой шифр у книги с заданным названием?
* Когда книга была закреплена за читателем?
* Кто из читателей взял книгу более месяца тому назад?
* За кем из читателей закреплены книги, количество экземпляров которых в библиотеке не превышает 2?
* Какое число читателей пользуется библиотекой?
* Сколько в библиотеке читателей младше 20 лет?

## Выполнение работы.

В ходе выполнения работы была спроектирована *ER* модель, представленная на рисунке 1.

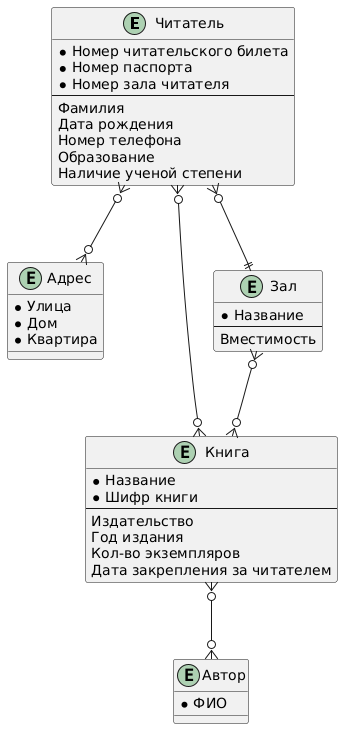


Рисунок 1 – *ER* модель.

Для характеризации сущности *Читатель* используются атрибуты: *номер читательского билета, номер паспорта, номер зала читателя, фамилия, дата рождения, номер телефона, образование, наличие ученой степени*.

Для характеризации сущностей *Зал* используются атрибуты: *название, вместимость.*

Для характеризации сущности *Адрес* используются атрибуты: *улица*, *дом* и *квартира*.

Для характеризации сущности *Книга* используются атрибуты: *название, шифр книги, издательство, год издания, кол-во экземпляров, дата закрепления за читателем.*

Для характеризации сущностей *Автор* используются атрибут: *ФИО.*

Пояснение по составлению ER модели:

* Читатель может взять одну или много книг.
* Книга может принадлежать нескольким читателям в разное время.
* Читатель может быть записан только в один читальный зал, а в читальном зале очевидно может находиться множество читателей.
* В зале находится множество книг и несколько экземпляров одной книги может быть в нескольких залах.
* У книги есть автор или несколько авторов, а автор может написать несколько книг.
* У читателя может быть несколько адресов, а по одному адресу могут жить несколько читателей.

Далее *ER* модель была преобразована в реляционную модель, представленную на рисунке 2.

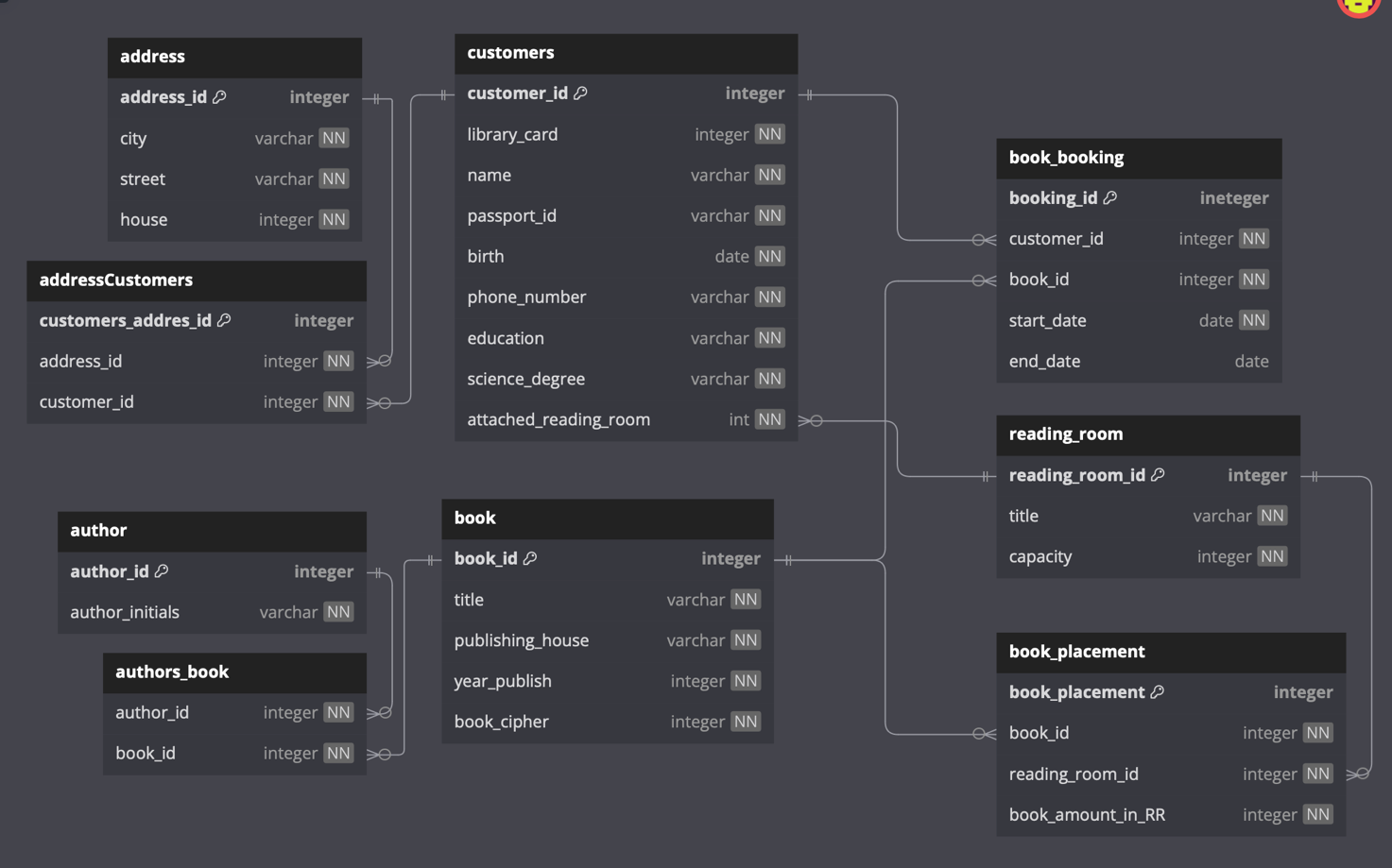


Рисунок 2 – Реляционная модель.

Во всех сущностях первичными ключами были сделаны идентификаторы. Для реализации ER модели, а точнее many-to-many отношений были добавлены новые таблицы: book\_placement, book\_booking, authors\_book, addressCustomers.

Реляционная модель имеет следующие связи:

Ref: book\_booking.customer\_id > customers.customer\_id

Ref: book\_booking.book\_id > book.book\_id

Ref: addressCustomers.address\_id > address.address\_id

Ref: addressCustomers.customer\_id > customers.customer\_id

Ref: book\_placement.book\_id > book.book\_id

Ref: book\_placement.reading\_room\_id > reading\_room.reading\_room\_id

Ref: customers.attached\_reading\_room > reading\_room.reading\_room\_id

Ref: authors\_book.author\_id > author.author\_id

Ref: authors\_book.book\_id > book.book\_id

Проверим находятся ли текущие отношения в НФБК:

1. Отношение customers имеет следующие функциональные зависимости:

(customer\_id) → (library\_card, name, passport\_id, birth, phone\_number, education, science\_degree, attached\_reading\_room),

(library\_card) → (customer\_id, name, passport\_id, birth, phone\_number, education, science\_degree, attached\_reading\_room),

(passport\_id) → (customer\_id, library\_card, name, birth, phone\_number, education, science\_degree, attached\_reading\_room). Чтобы идентифицировать читателя (помимо customer\_id) необходим его паспорт или номер читательского билета (при условии, что паспорт и номер уникальны). ФИО, дата рождения, номер телефона, образование, привязанная к читателю комната по понятным причинам не могут быть потенциальными ключами. (cinema\_id), (library\_card), (passport\_id) – потенциальные ключи. Потенциальными ключами являются атрибуты (cinema\_id), (library\_card), и (passport\_id), поскольку все остальные атрибуты зависят от этих ключей, и отсутствуют транзитивные зависимости между не ключевыми атрибутами. Отношение соответствует НФБК.

1. Отношение book имеет следующие функциональные зависимости:

Функциональная зависимость: (book\_id) → (title, publishing\_house, year\_publish, book\_cipher). Обратная зависимость неверна, поскольку названия, издательства и годы издания могут совпадать. Книжный шифр также может изменяться в результате переклассификации. Все атрибуты зависят от book\_id, и между не ключевыми атрибутами отсутствуют транзитивные зависимости. Отношение соответствует НФБК.

1. Отношение book\_booking имеет следующие функциональные зависимости:

Функциональная зависимости: (booking\_id) → (customer\_id, book\_id, start\_date, end\_date), (customer\_id, book\_id) → (booking\_id, start\_date, end\_date). Обратная зависимость неверна, так как несколько людей могут забирать и возвращать книги в одинаковые даты. Потенциальными ключами являются booking\_id и комбинация (customer\_id, book\_id). Все атрибуты зависят от этих ключей, и отсутствуют транзитивные зависимости между не ключевыми атрибутами. Отношение соответствует НФБК. Отношение соответствует НФБК.

1. Отношение reading\_room имеет следующие функциональные зависимости:

Функциональная зависимость: (reading\_room\_id) → (title, capacity). Обратная зависимость неверна, так как названия залов могут повторяться. Все атрибуты зависят от reading\_room\_id, и отсутствуют транзитивные зависимости между не ключевыми атрибутами. Отношение соответствует НФБК.

1. Отношение book\_placement имеет следующие функциональные зависимости:

Функциональная зависимость: (book\_placement) → (reading\_room\_id, book\_id, book\_amount\_in\_RR), (reading\_room\_id, book\_id) → (book\_placement, book\_amount\_in\_RR). Зная читальный зал и шифр книги, можно определить book\_placement и количество книг в зале. Потенциальными ключами являются book\_placement и комбинация (reading\_room\_id, book\_id). Все остальные атрибуты зависят от этих ключей, и отсутствуют транзитивные зависимости между не ключевыми атрибутами. Отношение соответствует НФБК.

1. Отношения author имеет следующие функциональные зависимости:

Функциональная зависимость: (author\_id) → (author\_initials). Обратная зависимость неверна, так как в теории разные авторы могут иметь одинаковые инициалы. Все атрибуты зависят от author\_id, и отсутствуют транзитивные зависимости между не ключевыми атрибутами. Отношение соответствует НФБК.

1. Отношения authors\_book имеет следующие функциональные зависимости:

Это отношение состоит из двух внешних ключей (FK). Атрибуты не зависят друг от друга и образуют составной ключ. Отношение соответствует НФБК

1. Отношение addressCustomers имеет следующие функциональные зависимости:

Функциональная зависимость: (customers\_address\_id) → (address\_id, customers\_id). Все атрибуты зависят от customers\_address\_id, и отсутствуют транзитивные зависимости между не ключевыми атрибутами. Отношение соответствует НФБК.

1. Отношение address имеет следующие функциональные зависимости:

Функциональная зависимость: (address\_id) → (city, street, house). Обратная зависимость неверна, так как в одном городе может быть много адресов, на одной улице могут находиться дома в разных городах, а один дом может быть расположен на разных улицах в разных городах. Все атрибуты зависят от address\_id, и транзитивных зависимостей нет. Отношение соответствует НФБК.

Доказано, что отношения находятся в НФБК.

## Вывод.

Приобретен навык построения схемы базы данных на основе построения ER модели, а также изучения связи между таблицами. Изучены основы НФ и НФБК в частности. Полученные знания применены на практике.

# Приложение А Ссылка на PR

Ссылка на PR: