

## **РАЗРАБОТКА ПРОГРАММЫ ДЛЯ УЧЕТА КНИГ В БИБЛИОТЕКЕ**

**Гатина Алина Алмазовна**

магистрант, Набережночелнинский институт (филиал) ФГАОУ ВО К(П)ФУ, РФ, г. Набережные Челны

**Ибрагимова Инзиля Фиркатовна**

студент, Набережночелнинский институт (филиал) ФГАОУ ВО К(П)ФУ, РФ, г. Набережные Челны

**Закиев Ильнар Азгамович**

студент, Набережночелнинский институт (филиал) ФГАОУ ВО К(П)ФУ, РФ, г. Набережные Челны

Объектно-ориентированное программирование – это новый подход к созданию программ. По мере развития вычислительной техники возникали разные методики программирования. На каждом этапе создавался новый подход, который помогал программистам справляться с растущим усложнением программ.

Были разработаны принципы объектно-ориентированного программирования. Объектно-ориентированное программирование позволяет разложить проблему на составные части. Каждая составляющая становится самостоятельным объектом, содержащим свои собственные коды и данные, которые относятся к этому объекту. Все языки ООП основаны на трёх основополагающих концепциях, называемых инкапсуляцией, полиморфизмом и наследованием.

В данной работе на основе принципов ООП разработано приложение для учета книг в библиотеке.

Приложение состоит из следующих основных классов:

- Book;
- Library;
- Utils.

### **Класс Book**

Класс Book предназначен для хранения и обработки информации о книгах. Данный класс содержит следующие поля:

- SName - фамилия и инициалы автора книги,
- Name- название книги,
- Year- год издания книги,
- Kol - количество экземпляров данной книги, хранящихся в библиотеке.

Класс Book, как и любой класс в программировании, имеет собственный конструктор, который предназначен для инициализации полей класса.

## **Класс Library**

Следующим классом является класс библиотеки – Library. Этот класс предназначен для управления записями книг. Класс содержит статическое поле af, который определяет порядковый номер записи в таблице. Для управления записями учета книг были реализованы следующие методы:

- ShowBook();
- CheckBook();
- AddBook();
- Delete Book().

Метод ShowBook() предназначен для вывода всей имеющейся информации о записях в графическое окно программы.

Метод CheckBook(), который в качестве аргумента которого передается объект класса Book, выполняет проверку на наличие указанного объекта среди существующих записей.

Метод AddBook() служит для добавления новой записи – книги в таблицу путем создания нового объекта класса Book.

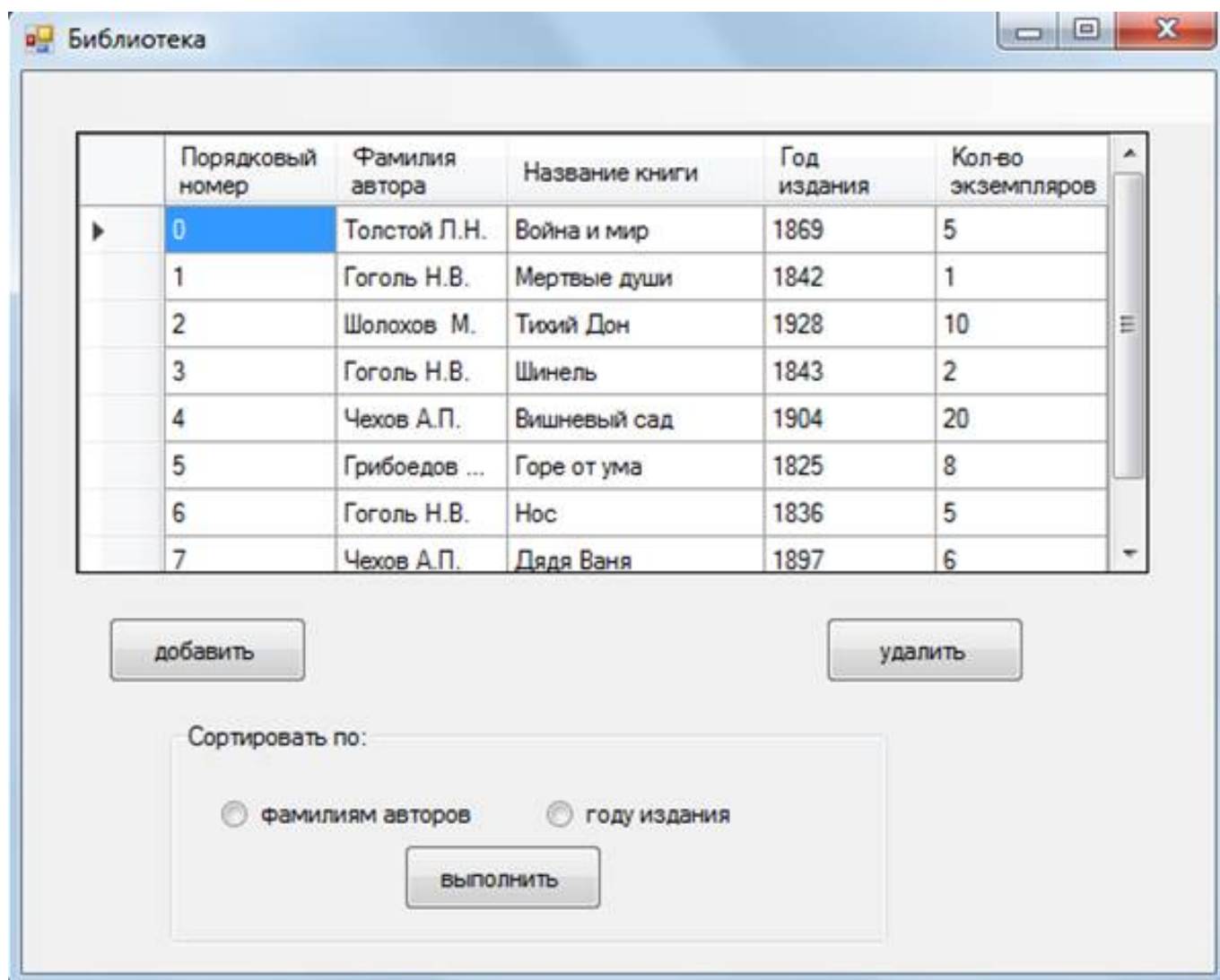
Метод Delete Book() является деструктором, который предназначен для удаления записи о книге по его индексу (в данном случае это порядковый номер записи).

## **Класс Utils**

Класс Utils является вспомогательным классом и содержит следующие поля:

- render\_book\_list() – метод, предназначенный для считывания необходимых данных из текстового файла;
- addition() – метод, при вызове которого создается объект класса Book и добавляется в таблицу с помощью метода Add Book();
- delete() – метод для удаления выделенной строки (деструктор);
- sort() – метод, который реализует сортировку таблицы записей книг методом прямого выбора. Данная сортировка выполняется по столбцу «фамилия автора», либо по столбцу «год издания».

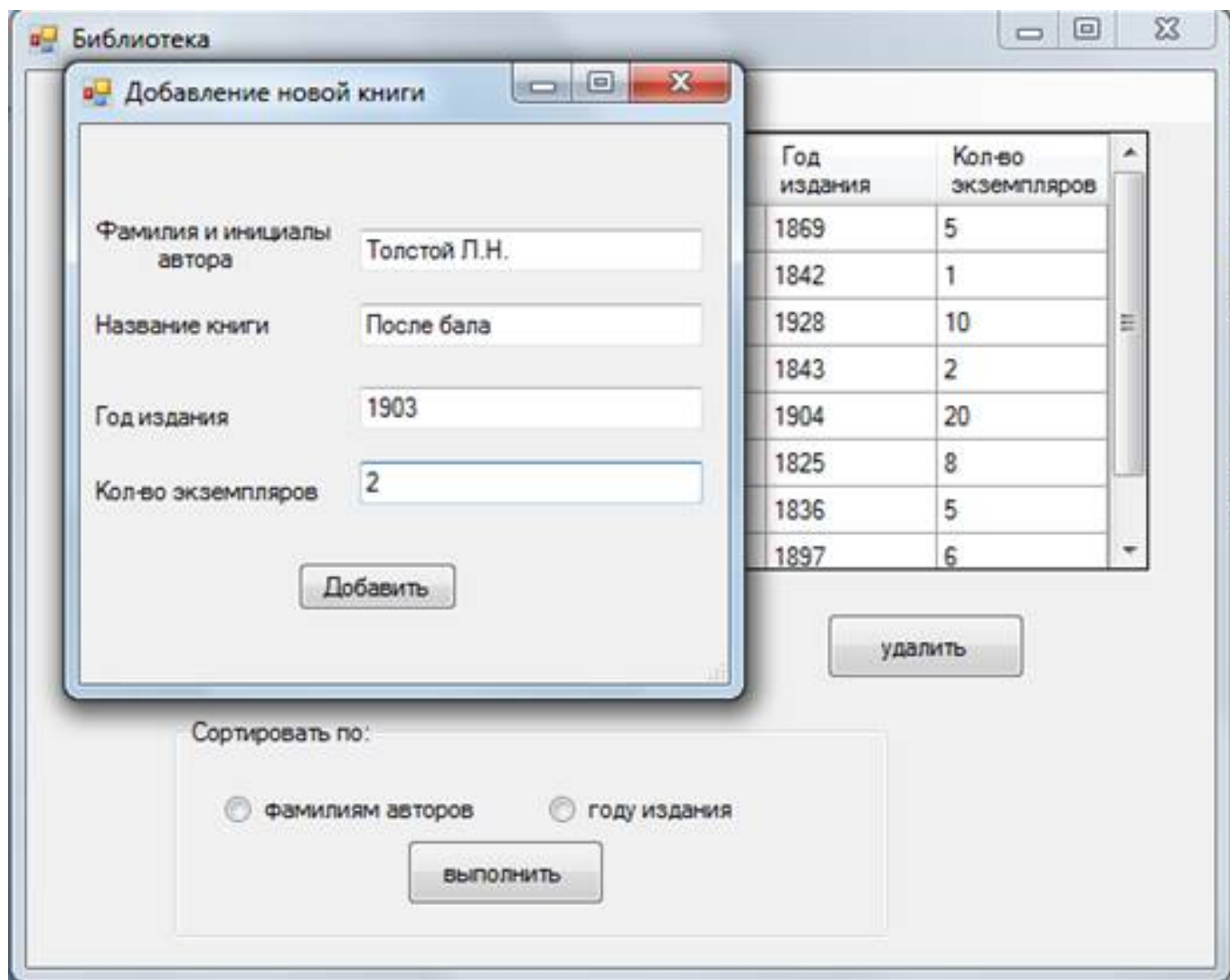
Данное приложение было реализовано на языке программирования C# с помощью среды Visual Studio. Основное графическое окно приложения для учета книг в библиотеке представлено на рисунке 1.



**Рисунок 1. Главное окно приложения**

Пользователю представлена область, в которой он видит информацию о существующих книгах в библиотеке, если они имеются. Информация о записях книг сгруппирована в небольшую таблицу, что делает ее восприятие очень удобным. Так же пользователь графического приложения может добавить новую запись либо удалить, выбрав одну из существующих. Для этого предназначены кнопки «добавить» и «удалить» соответственно.

Окно добавления записи о книге представлено на рисунке 2.



**Рисунок 2. Окно добавления записи**

В процессе добавления новой записи реализована проверка на корректность введенных данных, и в случае отрицательного результата проверки, приложение выдаст пользователю сообщение о некорректности введенных им данных.

При необходимости можно выполнить сортировку существующих записей по фамилии авторов либо по году издания книги, и провести необходимый пользователю анализ получившегося результата.

Результатом выполнения данной работы является приложение с графическим пользовательским интерфейсом, предназначенное для ведения учёта книг в библиотеке. Сведения о книгах содержат фамилию и инициалы автора, название книги, год издания и количество экземпляров данной книги в библиотеке.

Разработанная программа обеспечивает выполнение следующих функций:

1. добавление данных о книгах, вновь поступающих в библиотеку;
2. удаление данных о списываемых книгах;
3. выдача сведений обо всех книгах, упорядоченных по фамилиям авторов;
4. выдача сведений обо всех книгах, упорядоченных по годам издания.