



ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – ВАРНА

---

Факултет по изчислителна техника и автоматизация  
Катедра „СИТ“

# СЕМЕСТРИАЛНА ДОМАШНА РАБОТА

по дисциплината „Обектно-ориентирано програмиране проект  
1-част”

на тема: „Библиотека”

Вариант Проект 10

Изготвил: Диян Димитров

Проверил: гл.ас.Д.Стоянова

Специалност: СИТ

Група: 4а

Факултетен номер: 22621686

2024

## Съдържание

I. Увод .....	3
а) Описание и идея на проекта .....	3
б) Цел и задачи на разработката .....	3
с) Структура на документацията .....	3
II. Преглед на предметната дейност .....	5
I. Основни дефиниции и концепции .....	5
II. Дефиниране на проблеми и сложност на поставената задача .....	5
III. Проектиране.....	8
IV. Реализация, тестване.....	19
V. Реализация на условие D.....	<b>Error! Bookmark not defined.</b>
VI. Реализация на условие E .....	<b>Error! Bookmark not defined.</b>
VII. Реализация на условие F .....	<b>Error! Bookmark not defined.</b>
VIII. Реализация на условие ... - допълнение първо .....	<b>Error! Bookmark not defined.</b>
IX. Реализация на условие ... - допълнение второ .....	<b>Error! Bookmark not defined.</b>
X. Реализация на допълнение трето .....	<b>Error! Bookmark not defined.</b>
III. Упътване за употреба .....	<b>Error! Bookmark not defined.</b>
I. Впишете съответната част от проекта .....	<b>Error! Bookmark not defined.</b>
II. Впишете съответната част от проекта .....	<b>Error! Bookmark not defined.</b>
IV. Примерно действие на програмата.....	<b>Error! Bookmark not defined.</b>
I. Условие A .....	<b>Error! Bookmark not defined.</b>
II. Условие B .....	<b>Error! Bookmark not defined.</b>
III. Условие C .....	<b>Error! Bookmark not defined.</b>
IV. Условие D .....	<b>Error! Bookmark not defined.</b>
V. Условие E .....	<b>Error! Bookmark not defined.</b>
VI. Условие F.....	<b>Error! Bookmark not defined.</b>
VII. Допълнение първо .....	<b>Error! Bookmark not defined.</b>
VIII. Допълнение второ.....	<b>Error! Bookmark not defined.</b>
IX. Допълнение трето.....	<b>Error! Bookmark not defined.</b>

## I. Увод

### а) Описание и идея на проекта

Да се напише програма, реализираща информационна система, която поддържа библиотека. Програмата съхранява и обработва данни за наличните в момента книги във файл. Потребителят може да извършва операции върху отворените файлове, като добавя, редактира, също и да ги изтрива, както и да ги записва обратно в същия файл или в друг файл, който избере потребителя. При отваряне на файла се записват данните от файла, но ако няма такъв файл, той просто се създава и се затваря(по условие). Потребителя може да избере дали иска да запише данните в същия файл или от друг нов файл, като му подаде пътя. Всички промени, направени от потребителя, се пазят в паметта, но не се записват обратно, освен ако потребителят явно поиска това.

### б) Цел и задачи на разработката

Целта на проекта "Библиотека" е да предостави функционалност за отваряне, редактиране и запис на данни относно книги в една система. Освен основните операции като отваряне, затваряне, запис и помощ, приложението поддържа допълнителни операции. Това са командите, които проекта трябва да реализира: login – влизане на потребител ;  
logout - потребителят напуска системата (програмата продължава да работи);  
books all - извежда последователно за всяка книга следната информация:-заглавие, автор, жанр, персонален номер  
books info <isbn\_value>извежда на екрана подробна информация за книга с персонален номер равен на <isbn\_value>;  
books find <option> <option\_string> - <option>е едно от title, author, tag<option\_string>е стойността на критерия за търсене, може да съдържа интервали;  
books sort <option> [asc | desc] <option>е едно от title, author, year, rating asc означава възходящо сортиране (по подразбиране), а des означава низходящо сортиране; users add <user> <password> - добавя нов потребител с потребителско име <user>и парола <password>. Потребителят и неговата парола се записват във файл.;  
users remove<user> - изтрива потребителя с потребителско име <user>от файла.

### с) Структура на документацията

Документацията за проекта е структурирана в няколко глави, които представят последователно различни аспекти на разработката. "Увод", включва описание на проекта, неговата идея и целите на разработката. "Преглед на предметната област", представя основни дефиниции, концепции и алгоритми, които ще бъдат използвани.. "Проектиране", се разглежда общата структура на проекта и се представят диаграми и блок схеми, които

илюстрират структурата на проекта. "Реализация, тестване", се фокусира върху реализацията на класовете, включително важни моменти и кодови фрагменти, както и планирането и създаването на тестови сценарии. "Заклучение", представя обобщение на постигнатите цели и предлага насоки за бъдещо развитие и усъвършенстване на проекта

## II. Преглед на предметната дейност

### I. Основни дефиниции и концепции

#### ***TextFileReader(четец на текстови файлове)***

Посредством него данните се вмъкват в системата, като му се подава пътя на файла, от който искаме да се прочете. Той запълва класа Библиотека с данните от файла.

#### ***TextFileWriter(записвач на данни във файлове)***

От класа библиотека се вземат файловете и се подават на този записвач, той ги обработва и ги записва един по един в структуриран вид.

#### ***Command***

Посредством поведенческия шаблон Команда съм направил менюто, като за него също съм използвал Controller, който има задачата да свързва данните, подадени от командата и главните класове, които го обработват.

#### ***Library***

Класът, съдържащ колекцията от книги и също класът, където се развива бизнес логиката, свързана с книгите.

#### ***User***

В този клас се съдържат данните на потребителя.

### II. Дефиниране на проблеми и сложност на поставената задача

#### II.1. Реализиране на връзките между обектите

Разделил съм командите на подгрупи и спрямо тези групи успях да изградя системата на класовете. Едната подгрупа е тази, която отговаря за функционалността на библиотеката и извършва всички нейни дейности. Имам една група, която е свързана с командите, касаещи потребителите. Последната група засяга записването и четенето на данните във файл, като тук може да се спомене, че тази група се свързва с другите две и извършва своята дейност.

#### II.2. Четене и записване във файл

Четенто и записването е направено по начин, по който да отговаря на принципите на конкретния обект, който записва, тъй като изискват различен начин за записване на данните. Записването, което касае библиотеката може да се записва при желание на потребителя, като и има два варианта за записване – в същия файл или в нов, зададен от потребителя, докато данните за потребителите се записват в отделен файл, за да може при влизане на потребител да се проверява дали съществува и съответно при регистрация.

#### II.3. Подходи, методи за решаване на поставените проблеми

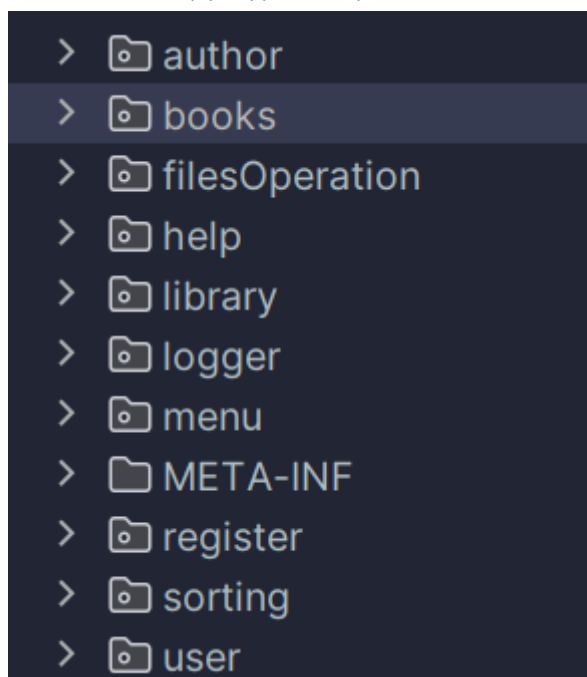
Подходите и начините на разрешаване на проблемите съм приложил в java doc





### III. Проектиране.

#### III.1. Обща структура на проекта пакети, които ще се реализират



#### III.2. Структура на класовете във всеки отделен пакет



> books



> author

## © Author

Ⓣ 🔒 *firstName* String

Ⓣ 🔒 *lastName* String

Ⓜ 🔒 *hashCode()* int

Ⓜ 🔒 *setFirstName(String)* void

Ⓜ 🔒 *toString()* String

Ⓜ 🔒 *getLastName()* String

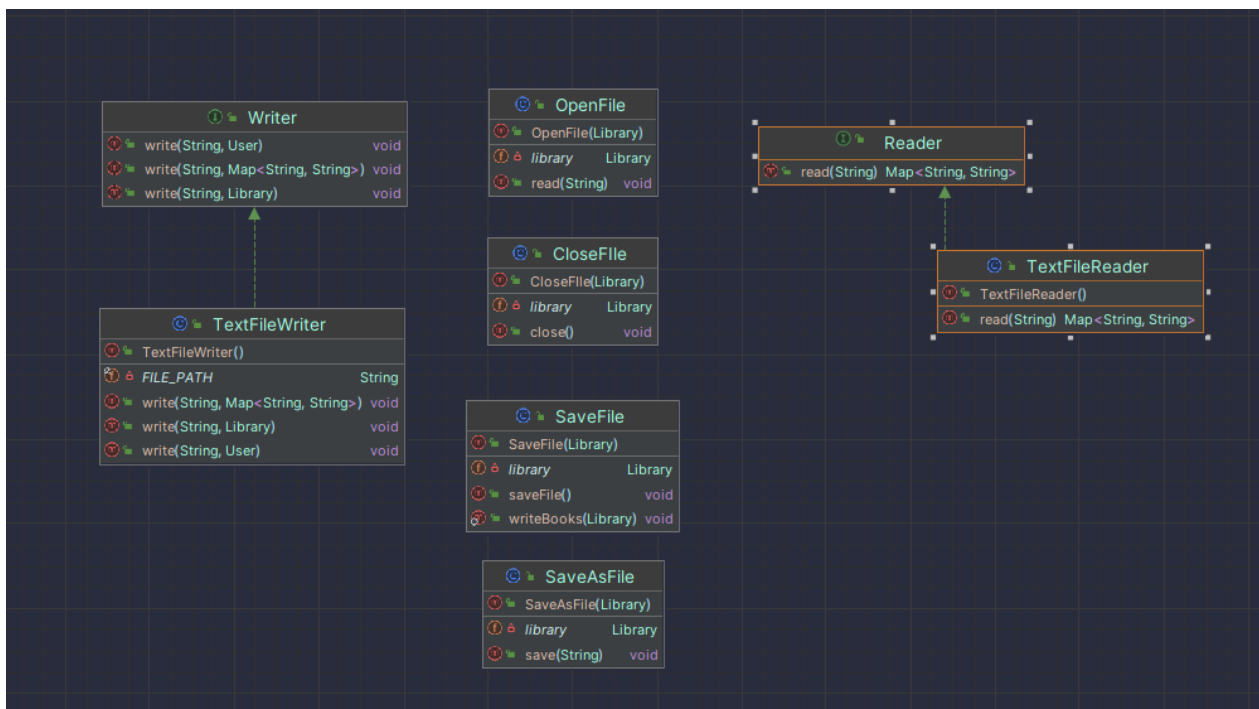
Ⓜ 🔒 *equals(Object)* boolean

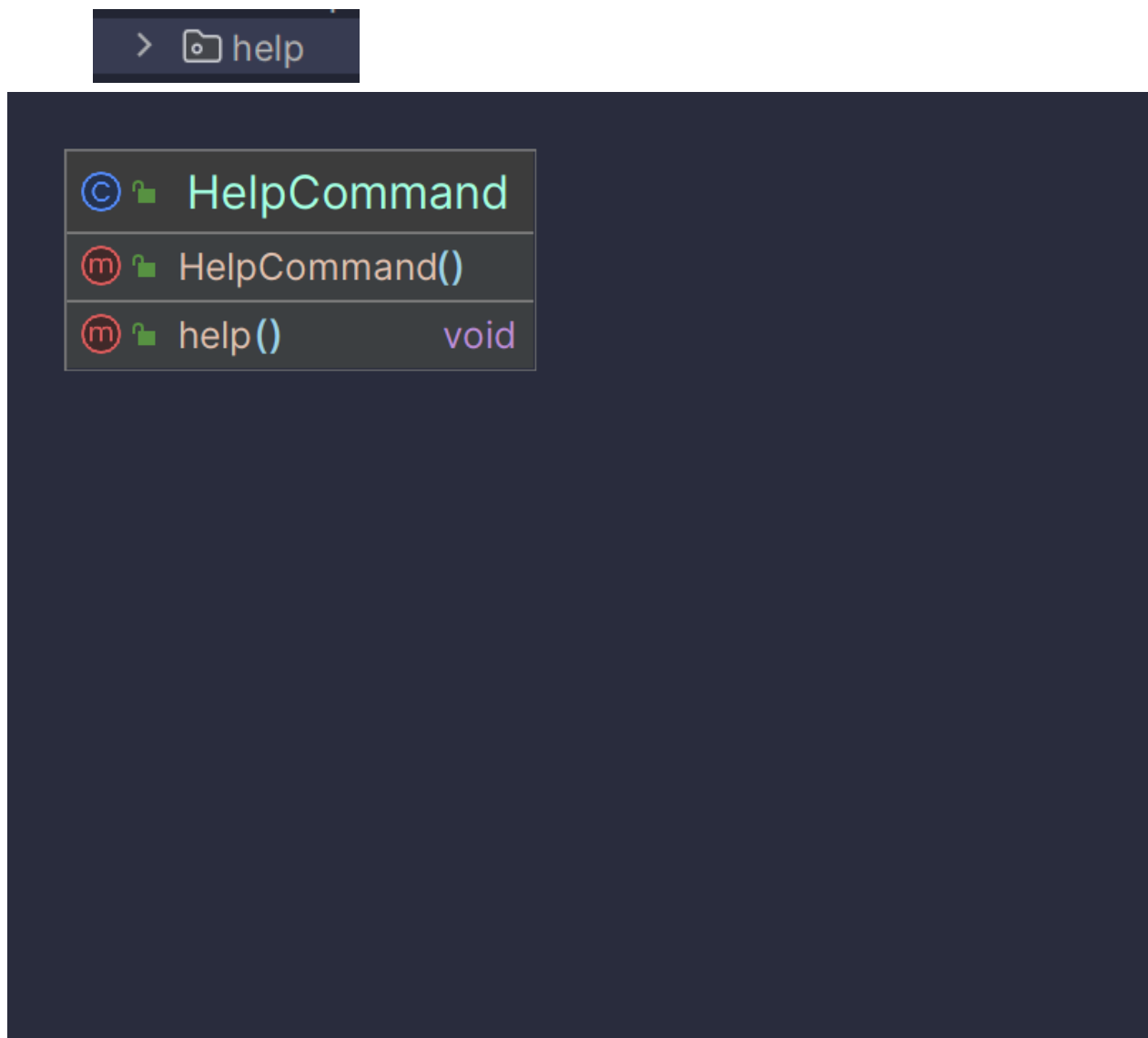
Ⓜ 🔒 *setLastName(String)* void

Ⓜ 🔒 *getFirstName()* String

Ⓜ 🔒 *compareTo(Author)* int

## > filesOperation






>  library

## © Library

f	⚡	<i>user</i>	User
f	⦿	<i>books</i>	List<Books>
f	⚡	<i>sorter</i>	Sorter
f	⚡	<i>path</i>	String
m	⚡	removeBook(String)	void
m	⚡	getSorter()	Sorter
m	⚡	booksFindAuthor(Author)	void
m	⚡	booksSort(String)	void
m	⚡	setBooks(List<Books>)	void
m	⚡	setUser(User)	void
m	⚡	setPath(String)	void
m	⚡	getPath()	String
m	⚡	booksFindTag(String)	void
m	⚡	booksSort(String, String)	void
m	⚡	booksAll()	void
m	⚡	addBook(Books)	void
m	⚡	setSorter(Sorter)	void
m	⚡	getUser()	User
m	⚡	booksInfo(String)	void
m	⚡	getBooks()	List<Books>
m	⚡	booksFindTitle(String)	void

> logger


## LogOperation


 login(User, String, String) boolean

 logout(User) void




## Logger

 Logger(Reader)

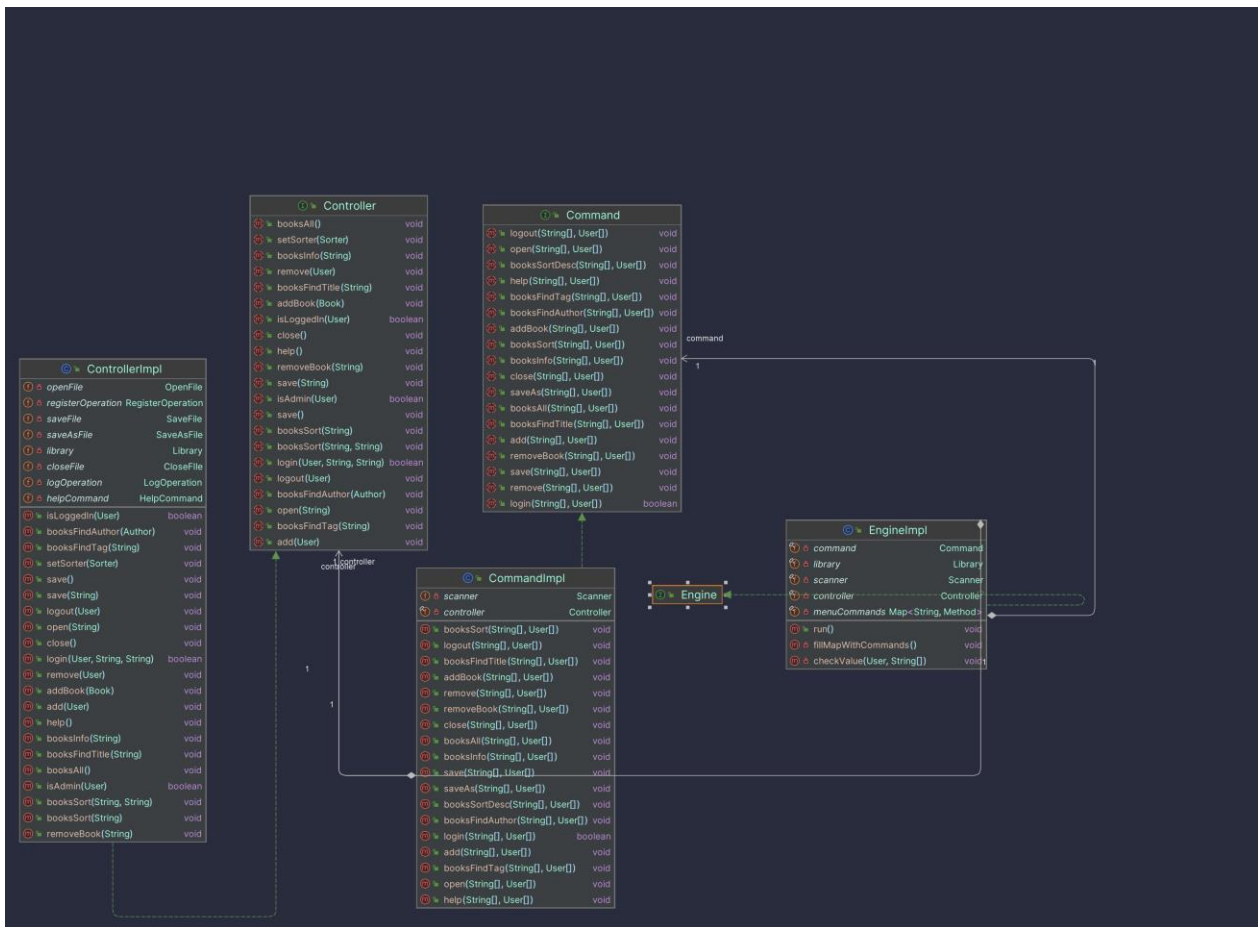
 FILE\_PATH String

 reader Reader

 login(User, String, String) boolean

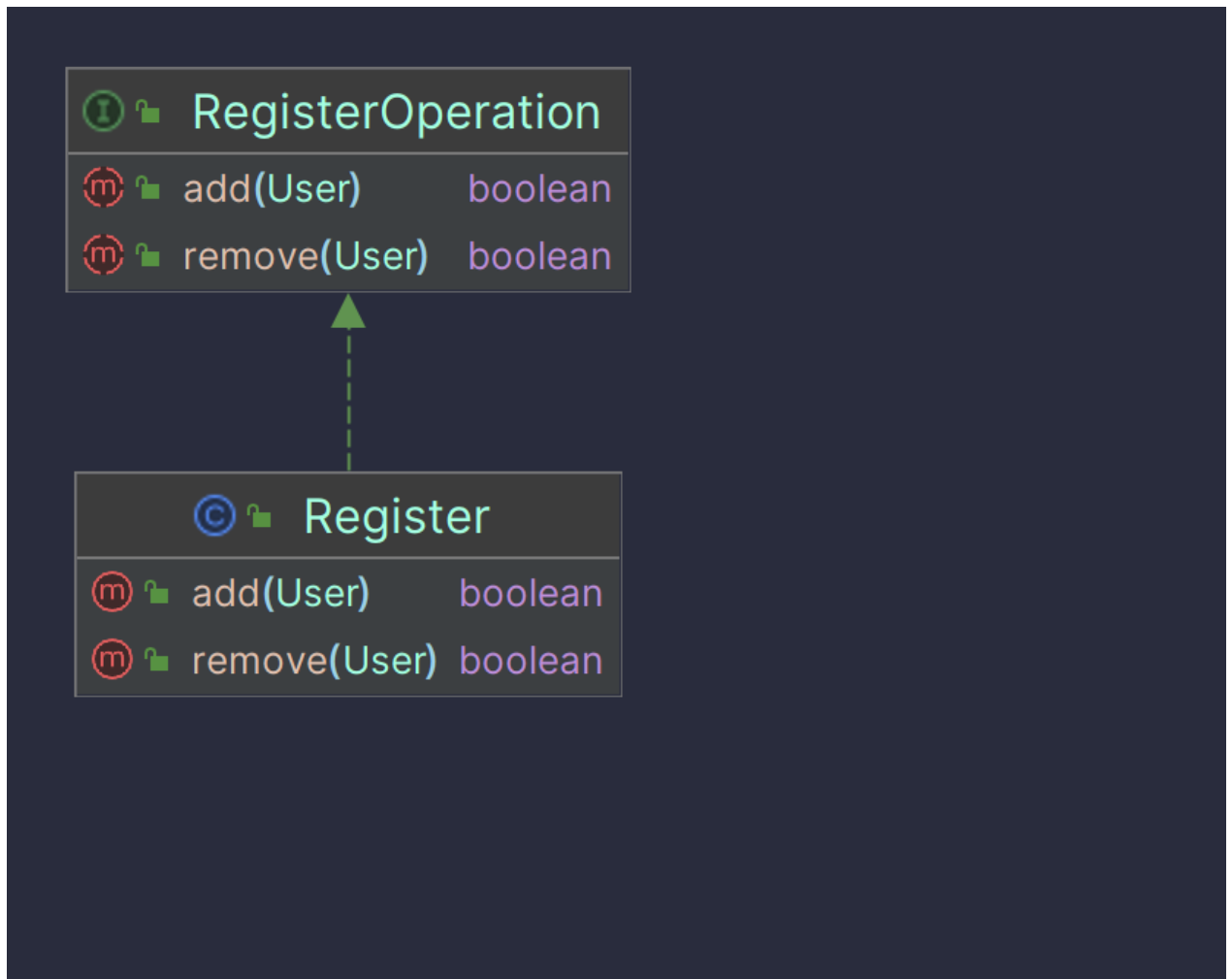
 logout(User) void

> menu

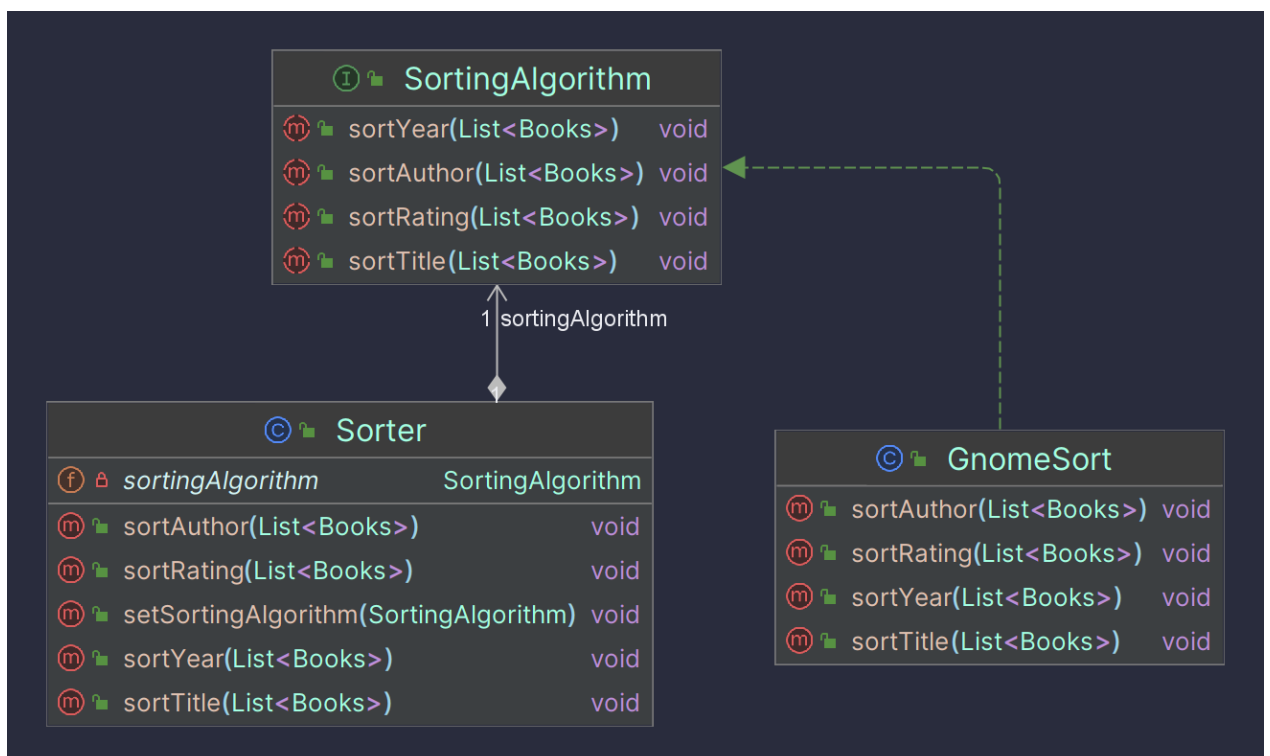


> register





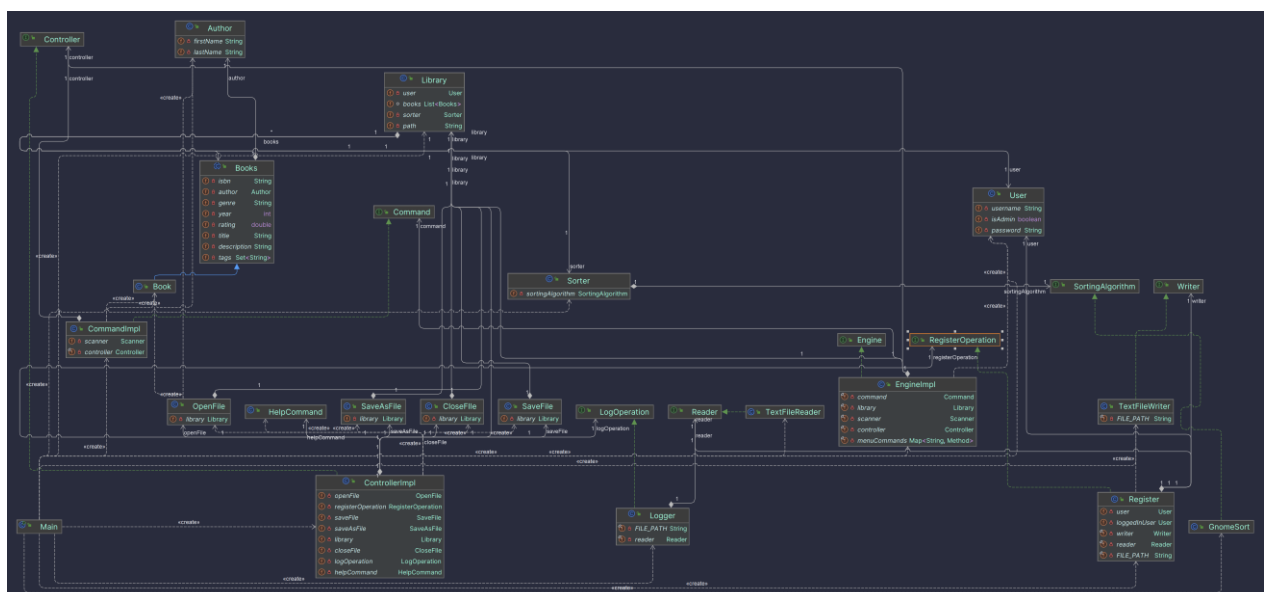
> sorting



> user

## © User

f	username	String
f	isAdmin	boolean
f	password	String
m	getPassword()	String
m	setPassword(String)	void
m	isAdmin()	boolean
m	setUsername(String)	void
m	setAdmin(boolean)	void
m	getUsername()	String



## IV. Реализация, тестване

### IV.1. Реализация на класове

Структурата на целия проект не е прекалено дълга и съответно не изпитах никакво затруднение там, но като най-интересното от целия проект мога да отбележа менюто. За него съм използвал шаблон Команд и рефлексия, реализираща се с класа Метод. Всички команди, които са заложили в проекта се реализират чрез помощен интерфейс и негова имплементация, наречен Controller. Там има поле от всички обекти, които съдържат дадена команда и контролера играе връзката между тях и Command. В имплементацията на Command има поле от Controller и така съм направил взаимодействието между Command и другите обекти.

### IV.2. Алгоритми и Оптимизации.

Единственият по-сложен алгоритъм в проекта е алгоритъм за сортиране, наречен Гном сорт. Интересно ми беше да се запозная как работи, тъй като до сега не бях чувал за него и понеже по условие имах да реализирам алгоритъм за сортиране различен от метода на мехурчето, направих проучване за няколко алгоритъма и се спрях конкретно на този.

### IV.3. Планиране, описание и създаване на тестови сценарии

Тестовите сценарии, разбира се, зависят от данните затова първо бях планирал да въведа примерни данни и да видя дали всяка една функционалност работи. По-долу ще представя снимка на тестовите варианти. Така изглежда менюто, когато се зареди първоначално.

```
The following commands are supported:
open <file>           opens <file>
close                 closes currently opened file
save                  saves the currently open file
saveas <file>         saves the currently open file in <file>
help                  prints this information
exit                  exists the program
login                 sign in the user
logout                sign out the user
books all              show the information about the books
books info <isbn_value> search a book by isbn
books find <option> <option_string> option is one of the following
                                     {title,author,tag}
                                     <option_string> is the criteria
books sort <option> [desc]
                                     <option> ={title,author,year,rating} by default asc
users add <user> <password>
                                     add a new user
users remove <user_name>
                                     remove a user by given user_name
books add              add a new book
books remove           remove a book
Next command:
```

Пример за принтиране на всички книги

```
books all
Title: Don Quixote |Author: firstName= Miguel lastName= Cervantes |Genre: Fiction |ISBN: 978-0060934347 |Year: 1605| Rating: 4,10
Title: Pride and Prejudice |Author: firstName= Jane lastName= Austen |Genre: Romance |ISBN: 978-1503290563 |Year: 1813| Rating: 4,80
Title: Frankenstein |Author: firstName= Mary lastName= Shelley |Genre: Horror |ISBN: 978-0486282114 |Year: 1818| Rating: 4,00
Title: Wuthering Heights |Author: firstName= Emily lastName= Bronte |Genre: Gothic |ISBN: 978-0141439556 |Year: 1847| Rating: 4,50
Title: Jane Eyre |Author: firstName= Charlotte lastName= Bronte |Genre: Romance |ISBN: 978-0141441146 |Year: 1847| Rating: 4,70
Title: The Fall of the House of Usher |Author: firstName= Edgar lastName= Allan Poe |Genre: Gothic |ISBN: 978-0486282145 |Year: 1849|
Rating: 4,60
Title: A Tale of Two Cities |Author: firstName= Charles lastName= Dickens |Genre: Historical |ISBN: 978-1503219700 |Year: 1859| Rating:
4,40
Title: Les Misérables |Author: firstName= Victor lastName= Hugo |Genre: Historical |ISBN: 978-0451419439 |Year: 1862| Rating: 4,80
Title: Crime and Punishment |Author: firstName= Fyodor lastName= Dostoevsky |Genre: Psychological |ISBN: 978-0486454115 |Year: 1866|
Rating: 4,50
Next command:
```

Пример за търсене на книга по ISBN номер

```
books info 978-0060934347
Title: Don Quixote |Author: firstName= Miguel lastName= Cervantes |Genre: Fiction |ISBN: 978-0060934347 |Year: 1605| Rating: 4,10
Next command:
```

Пример за търсене на книга по зададен таг

```
books find tag tragedy
Title: Frankenstein |Author: firstName= Mary lastName= Shelley |Genre: Horror |ISBN: 978-0486282114 |Year: 1818| Rating: 4,00
Title: Wuthering Heights |Author: firstName= Emily lastName= Bronte |Genre: Gothic |ISBN: 978-0141439556 |Year: 1847| Rating: 4,50
Next command:
```

Пример за сортиране по някаква опция

```
books sort
Choose option(author,title,year,rating):
year
Next command:
books all
Title: Don Quixote |Author: firstName= Miguel lastName= Cervantes |Genre: Fiction |ISBN: 978-0060934347 |Year: 1605| Rating: 4,10
Title: Pride and Prejudice |Author: firstName= Jane lastName= Austen |Genre: Romance |ISBN: 978-1503290563 |Year: 1813| Rating: 4,80
Title: Frankenstein |Author: firstName= Mary lastName= Shelley |Genre: Horror |ISBN: 978-0486282114 |Year: 1818| Rating: 4,00
Title: Wuthering Heights |Author: firstName= Emily lastName= Bronte |Genre: Gothic |ISBN: 978-0141439556 |Year: 1847| Rating: 4,50
Title: Jane Eyre |Author: firstName= Charlotte lastName= Bronte |Genre: Romance |ISBN: 978-0141441146 |Year: 1847| Rating: 4,70
Title: The Fall of the House of Usher |Author: firstName= Edgar lastName= Allan Poe |Genre: Gothic |ISBN: 978-0486282145 |Year: 1849|
Rating: 4,60
Title: A Tale of Two Cities |Author: firstName= Charles lastName= Dickens |Genre: Historical |ISBN: 978-1503219700 |Year: 1859| Rating:
4,40
Title: Les Misérables |Author: firstName= Victor lastName= Hugo |Genre: Historical |ISBN: 978-0451419439 |Year: 1862| Rating: 4,80
Title: Crime and Punishment |Author: firstName= Fyodor lastName= Dostoevsky |Genre: Psychological |ISBN: 978-0486454115 |Year: 1866|
Rating: 4,50
Next command:
```

Пример за това как трябва да си влязъл в системата, за да имаш достъп до някои функции.

```
logout
Next command:
books all
Need to log in first
Next command:
```

## V. Заключение

### V.1. Обобщение на изпълнението на началните цели

В обобщение мога да кажа, че са изпълнени всички команди, които бяха по заданието и са направени допълнителните, които са като бонус. Интересно в случая мога да добавя, че единия бонус, а именно да се замества символа, когато си пишеш паролата е невъзможен да се случи в конзолата. Направих проучване спрямо това и стигнах до компромисен вариант, а именно да се чете паролата, без да се изписва символа, но не го замества веднага, а след като се прочете паролата. Всички други цели са изпълнени изцяло и мога да кажа, че единствената трудност беше заместването на символите от паролата със звездичка.

## V.2. Насоки за бъдещо развитие и усъвършенстване

Проектът би могъл да се удължи още много, тъй като за момента е на много базово ниво. Предполагам в бъдеще бих искал да свържа база данни с проекта и да го усложня малко повече от към функционалност, понеже за момента разполага само с базови функции. Би било интересно всяка книга да може да се заема и да се следят количество от тази книга и дали може даден потребител да я заеме или не. Би било интересно също да се добавят раздели за оценки на книги от потребители и да имат възможност да оставят отзив. Мисля, че тези неща биха направили проекта малко по-интересен за осъществяване.