Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра автоматизованих систем управління

Зображення, що містить текст

Автоматично згенерований опис

**Звiт**

до лабораторної роботи № 2

з дисципліни

# “ Прикладне програмування”

Виконав: студент групи ОІ-11СП

**Богдан Заяць**

Прийняв:

**МельникР.В.**

Львів – 2024

**Завдання:**

1. Створити проект, що складається з двох класів: основного (Main) та класу для представлення об’єкта відповідно специфікації, що наведена у варіанті індивідуального завдання. Кожний клас повинен бути розміщений у окремому пакеті. У створеному класі визначити приватні поля для зберігання указаних даних, конструктори для створення об’єктів та відкриті методи setValue(), getValue(), toString() для доступу до полів об’єкту.

2. В основному класі програми визначити методи, що створюють масив об'єктів. Задати критерії вибору даних та вивести ці дані на консоль ( використати метод toString() ). Для кожного критерію створити окремий метод.

3. Виконати програму, та пересвідчитись, що дані зберігаються та коректно виводяться на екран відповідно до вказаних критеріїв.

**Book**: id, Назва, Автор, Видавництво, Рік видання, Кількість сторінок, Ціна.

Скласти масив об'єктів. Вивести:

a) список книг заданого автора;

b) список книг, що видані заданим видавництвом;

c) список книг, що випущені після заданого року.

**Результати:**

Вивід масиву:

Book{bookId=8556210, author='Dan Brown', name='Angels and demons', publishingHouse='Morning', publicationYear=2014, pageCount=625}

Book{bookId=4852142, author='Dan Brown', name='Davinci Code', publishingHouse='CFL', publicationYear=2015, pageCount=645}

Book{bookId=4585545, author='Dan Brown', name='The Source', publishingHouse='CFL', publicationYear=2016, pageCount=855}

Book{bookId=4456210, author='Dan Brown', name='The lost symbol', publishingHouse='CFL', publicationYear=2016, pageCount=555}

Book{bookId=4455850, author='Dan Brown', name='Inferno', publishingHouse='CFL', publicationYear=2018, pageCount=750}

Book{bookId=7474545, author='Stephen King', name='Pet Cemetery', publishingHouse='CFL', publicationYear=2017, pageCount=750}

Selected

Book{bookId=4455850, author='Dan Brown', name='Inferno', publishingHouse='CFL', publicationYear=2018, pageCount=750}

a) список книг заданого автора;

I dont want to be buried in a

Book{bookId=7474545, author='Stephen King', name='Pet Cemetery', publishingHouse='CFL', publicationYear=2017, pageCount=750}

b) список книг, що видані заданим видавництвом і автором;

"Dan Brown's boo,selected from with the Morning publishing house"

Book{bookId=8556210, author='Dan Brown', name='Angels and demons', publishingHouse='Morning', publicationYear=2014, pageCount=625}

Books with 750 pages

Book{bookId=4455850, author='Dan Brown', name='Inferno', publishingHouse='CFL', publicationYear=2018, pageCount=750}

Book{bookId=7474545, author='Stephen King', name='Pet Cemetery', publishingHouse='CFL', publicationYear=2017, pageCount=750}

c) список книг, що випущені після заданого року.

Books newer than 2015

Book{bookId=4585545, author='Dan Brown', name='The Source', publishingHouse='CFL', publicationYear=2016, pageCount=855}

Book{bookId=4456210, author='Dan Brown', name='The lost symbol', publishingHouse='CFL', publicationYear=2016, pageCount=555}

Book{bookId=4455850, author='Dan Brown', name='Inferno', publishingHouse='CFL', publicationYear=2018, pageCount=750}

Book{bookId=7474545, author='Stephen King', name='Pet Cemetery', publishingHouse='CFL', publicationYear=2017, pageCount=750}

Програма:

Main.java

package edu.bogdan.zaiats.lab2.main;

import edu.bogdan.zaiats.lab2.book.Book;

import java.util.List;

class Main {

private static final List<Book> bookShelf = List.of(

new Book(8556210, "Dan Brown", "Angels and demons", "Morning", 2014, 625),

new Book(4852142, "Dan Brown", "Davinci Code", "CFL", 2015, 645),

new Book(4585545, "Dan Brown", "The Source", "CFL", 2016, 855),

new Book(4456210, "Dan Brown", "The lost symbol", "CFL", 2016, 555),

new Book(4455850, "Dan Brown", "Inferno", "CFL", 2018, 750),

new Book(7474545, "Stephen King", "Pet Cemetery", "CFL", 2017, 750)

);

public static void main(String[] args) {

for (var book : bookShelf){

System.out.println(book.toString());

}

var selected = new BookSelector(bookShelf).withPublicationYear(2018).getBooks();

System.out.println("Selected");

for (var book : selected) {

System.out.println(book.toString());

}

System.out.println("Dan Brown's boo,selected from with the Morning publishing house");

var withDanBrown = new BookSelector(bookShelf).withAuthor("Dan Brown").withPublishingHouse("Morning") .getBooks() ;

for (var book : withDanBrown) {

System.out.println(book.toString());

}

System.out.println("Books with 750 pages");

var selectedWithPages = new BookSelector(bookShelf).withPageCount(750).getBooks();

selectedWithPages.forEach(book -> System.out.println(book.toString()));

System.out.println("Books newer than 2015");

var newerBooks = new BookSelector(bookShelf).newerThanYear(2015).getBooks();

newerBooks.forEach(book -> System.out.println(book.toString()));

System.out.println("I dont want to be buried in a");

var fromStephenKing = new BookSelector(bookShelf).withAuthor("Stephen King").getBooks();

fromStephenKing.forEach(book -> System.out.println(book.toString()));

}

}

Book.java

package edu.bogdan.zaiats.lab2.book;

public class Book {

private int bookId;

private String author;

private String name;

private String publishingHouse;

private int publicationYear;

private int pageCount;

public Book(int bookId, String author, String name, String publishingHouse, int publicationYear, int pageCount) {

this.bookId = bookId;

this.author = author;

this.name = name;

this.publishingHouse = publishingHouse;

this.publicationYear = publicationYear;

this.pageCount = pageCount;

}

public int getBookId() {

return bookId;

}

public void setBookId(int bookId) {

this.bookId = bookId;

}

public String getAuthor() {

return author;

}

public void setAuthor(String author) {

this.author = author;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getPublishingHouse() {

return publishingHouse;

}

public void setPublishingHouse(String publishingHouse) {

this.publishingHouse = publishingHouse;

}

public int getPublicationYear() {

return publicationYear;

}

public void setPublicationYear(int publicationYear) {

this.publicationYear = publicationYear;

}

public int getPageCount() {

return pageCount;

}

public void setPageCount(int pageCount) {

this.pageCount = pageCount;

}

@Override

public String toString() {

return "Book{" +

"bookId=" + bookId +

", author='" + author + '\'' +

", name='" + name + '\'' +

", publishingHouse='" + publishingHouse + '\'' +

", publicationYear=" + publicationYear +

", pageCount=" + pageCount +

'}';

}

}

BookSelector.java

package edu.bogdan.zaiats.lab2.main;

import edu.bogdan.zaiats.lab2.book.Book;

import java.util.List;

public class BookSelector {

private List<Book> books;

public BookSelector (List<Book> books) {

this.books = books;

}

public BookSelector withAuthor(String author) {

this.books = this.books.stream().filter(book -> book.getAuthor().equals(author)).toList();

return this;

}

public BookSelector withPublicationYear(int year) {

this.books = this.books.stream().filter(book -> book.getPublicationYear() == year).toList();

return this;

}

public BookSelector newerThanYear(int year) {

this.books = this.books.stream().filter(book -> book.getPublicationYear() > year).toList();

return this;

}

public BookSelector withPublishingHouse(String publishingHouse) {

this.books = this.books.stream().filter(book -> book.getPublishingHouse().equals(publishingHouse)).toList();

return this;

}

public BookSelector withPageCount(int pageCount) {

this.books = this.books.stream().filter(book -> book.getPageCount() == pageCount).toList();

return this;

}

public List<Book> getBooks() {

return this.books;

}

}

**Репозиторій:**

<https://github.com/zayats1/Labs.git>

**Висновок:**

У цій роботі я навчився працювати з класами та пакетами.