Задание по коллекциям Java

Цель задания: Изучить и применить на практике различные коллекции в Java, создавая собственное приложение, которое будет использовать эти коллекции для решения реальной задачи.

Задание:

1. Тема: Библиотека

2. Определение структуры данных:

Определите, какие данные вам нужно хранить. Для создания библиотеки, вам могут понадобиться классы Book, Author, и Library.

Коллекции использовать для хранения данных. Например:

•

Для хранения списка книг используйте ArrayListBook>.

Для хранения уникальных авторов используйте HashSet Author>.

Для хранения книг по жанрам используйте HashMap String, List<Book>>, где ключ жанр, а значение - список книг этого жанра.

3. Программа должна выводить и добавлять список книг, авторов и жанры

Код программы:

import java.util.\*;

class Book {

private String title;

private String genre;

private Author author;

public Book(String title, String genre, Author author) {

this.title = title;

this.genre = genre;

this.author = author;

}

public String getTitle() {

return title;

}

public String getGenre() {

return genre;

}

public Author getAuthor() {

return author;

}

}

class Author {

private String name;

public Author(String name) {

this.name = name;

}

public String getName() {

return name;

}

}

class Library {

private List<Book> books = new ArrayList<>();

private Set<Author> authors = new HashSet<>();

private Map<String, List<Book>> genres = new HashMap<>();

public void addBook(Book book) {

books.add(book);

authors.add(book.getAuthor());

genres.computeIfAbsent(book.getGenre(), g -> new ArrayList<>()).add(book);

}

public void showBooks() {

for (Book book : books) {

System.out.println("Book: " + book.getTitle() + ", Genre: " + book.getGenre() + ", Author: " + book.getAuthor().getName());

}

}

public void showAuthors() {

for (Author author : authors) {

System.out.println("Author: " + author.getName());

}

}

public void showBooksByGenre() {

for (String genre : genres.keySet()) {

System.out.println("Genre: " + genre);

for (Book book : genres.get(genre)) {

System.out.println(" Book: " + book.getTitle() + ", Author: " + book.getAuthor().getName());

}

}

}

}

public class Main {

public static void main(String[] args) {

Library library = new Library();

Author author1 = new Author("George Orwell");

Author author2 = new Author("Harper Lee");

library.addBook(new Book("Brave New World", "Dystopian", author1));

library.addBook(new Book("The Great Gatsby", "Fiction", author2));

library.addBook(new Book("Fahrenheit 451", "Dystopian", author1));

library.addBook(new Book("Pride and Prejudice", "Fiction", author2));

library.showBooks();

library.showAuthors();

library.showBooksByGenre();

}

}

Результаты:

