Министерство науки и высшего образования Российской Федерации  
Национальный исследовательский университет ИТМО  
Факультет программной инженерии и компьютерной техники

**Лабораторная работа №5**

Выполнил: **Дърлянов Христо Христов**

Вариант: 12256

Группа: **P3119**

Преподаватель:

г. Санкт-Петербург  
2024

*1. Задание:*

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса Movie, описание которого приведено ниже.

**Разработанная программа должна удовлетворять следующим требованиям:**

* Класс, коллекцией экземпляров которого управляет программа, должен реализовывать сортировку по умолчанию.
* Все требования к полям класса (указанные в виде комментариев) должны быть выполнены.
* Для хранения необходимо использовать коллекцию типа java.util.ArrayDequeue
* При запуске приложения коллекция должна автоматически заполняться значениями из файла.
* Имя файла должно передаваться программе с помощью: **аргумент командной строки**.
* Данные должны храниться в файле в формате xml
* Чтение данных из файла необходимо реализовать с помощью класса java.io.BufferedInputStream
* Запись данных в файл необходимо реализовать с помощью класса java.io.PrintWriter
* Все классы в программе должны быть задокументированы в формате javadoc.
* Программа должна корректно работать с неправильными данными (ошибки пользовательского ввода, отсутсвие прав доступа к файлу и т.п.).

**В интерактивном режиме программа должна поддерживать выполнение следующих команд:**

* help : вывести справку по доступным командам
* info : вывести в стандартный поток вывода информацию о коллекции (тип, дата инициализации, количество элементов и т.д.)
* show : вывести в стандартный поток вывода все элементы коллекции в строковом представлении
* add {element} : добавить новый элемент в коллекцию
* update id {element} : обновить значение элемента коллекции, id которого равен заданному
* remove\_by\_id id : удалить элемент из коллекции по его id
* clear : очистить коллекцию
* save : сохранить коллекцию в файл
* execute\_script file\_name : считать и исполнить скрипт из указанного файла. В скрипте содержатся команды в таком же виде, в котором их вводит пользователь в интерактивном режиме.
* exit : завершить программу (без сохранения в файл)
* remove\_head : вывести первый элемент коллекции и удалить его
* add\_if\_min {element} : добавить новый элемент в коллекцию, если его значение меньше, чем у наименьшего элемента этой коллекции
* remove\_lower {element} : удалить из коллекции все элементы, меньшие, чем заданный
* count\_by\_genre genre : вывести количество элементов, значение поля genre которых равно заданному
* count\_less\_than\_oscars\_count oscarsCount : вывести количество элементов, значение поля oscarsCount которых меньше заданного
* filter\_contains\_name name : вывести элементы, значение поля name которых содержит заданную подстроку

**Формат ввода команд:**

* Все аргументы команды, являющиеся стандартными типами данных (примитивные типы, классы-оболочки, String, классы для хранения дат), должны вводиться в той же строке, что и имя команды.
* Все составные типы данных (объекты классов, хранящиеся в коллекции) должны вводиться по одному полю в строку.
* При вводе составных типов данных пользователю должно показываться приглашение к вводу, содержащее имя поля (например, "Введите дату рождения:")
* Если поле является enum'ом, то вводится имя одной из его констант (при этом список констант должен быть предварительно выведен).
* При некорректном пользовательском вводе (введена строка, не являющаяся именем константы в enum'е; введена строка вместо числа; введённое число не входит в указанные границы и т.п.) должно быть показано сообщение об ошибке и предложено повторить ввод поля.
* Для ввода значений null использовать пустую строку.
* Поля с комментарием "Значение этого поля должно генерироваться автоматически" не должны вводиться пользователем вручную при добавлении.

**Описание хранимых в коллекции классов:**

public class Movie {

private Long id; //Поле не может быть null, Значение поля должно быть больше 0, Значение этого поля должно быть уникальным, Значение этого поля должно генерироваться автоматически

private String name; //Поле не может быть null, Строка не может быть пустой

private Coordinates coordinates; //Поле не может быть null

private java.time.LocalDateTime creationDate; //Поле не может быть null, Значение этого поля должно генерироваться автоматически

private long oscarsCount; //Значение поля должно быть больше 0

private Integer goldenPalmCount; //Значение поля должно быть больше 0, Поле может быть null

private int totalBoxOffice; //Значение поля должно быть больше 0

private MovieGenre genre; //Поле может быть null

private Person screenwriter;

}

public class Coordinates {

private Integer x; //Значение поля должно быть больше -500, Поле не может быть null

private Float y; //Поле не может быть null

}

public class Person {

private String name; //Поле не может быть null, Строка не может быть пустой

private java.time.ZonedDateTime birthday; //Поле не может быть null

private double weight; //Значение поля должно быть больше 0

private String passportID; //Длина строки должна быть не меньше 4, Значение этого поля должно быть уникальным, Поле может быть null

private Location location; //Поле может быть null

}

public class Location {

private Float x; //Поле не может быть null

private float y;

private float z;

}

public enum MovieGenre {

WESTERN,

DRAMA,

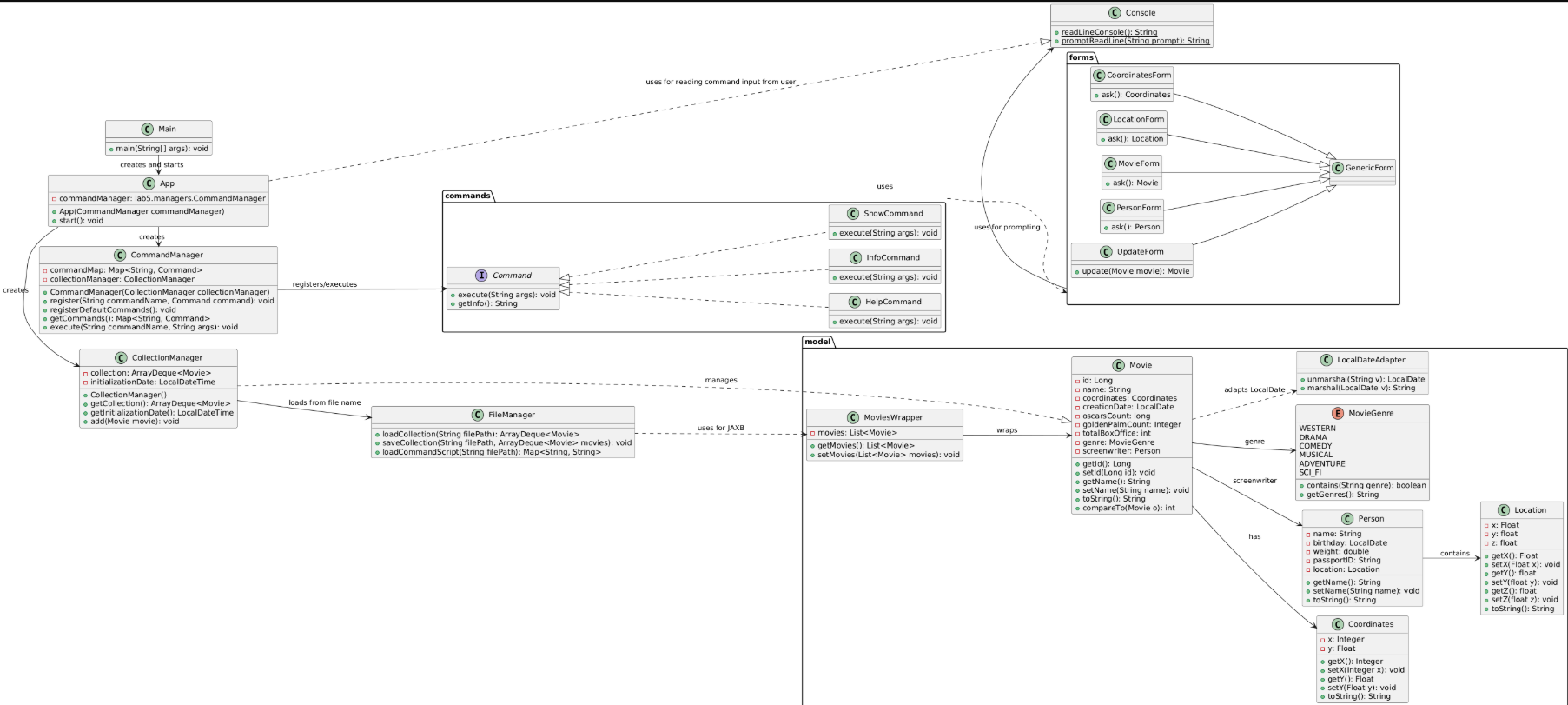
COMEDY,

MUSICAL,

ADVENTURE;

}

*2. Диаграма:*



*3. Исходный код программы*

<https://github.com/hristo-darlyanov/lab5>

*4. Bывод*

Научился работать с входными и выходными потоками, коллекциями, универсальными классами и подстановочными знаками. Узнал об автоматическом документировании с помощью javadoc.