Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерных технологий

Лабораторная работа №5

Вариант 4361

Выполнил:

Кузнецов Андрей Романович

Группа Р3108

Проверил:

Миху Вадим Дмитриевич

Санкт-Петербург 2025

## Содержание

[Содержание 2](#_Toc198049922)

[Задание 3](#_Toc198049923)

[Исходный код программы 7](#_Toc198049924)

[UML-диаграмма классов 7](#_Toc198049925)

[Вывод 7](#_Toc198049926)

## 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.HashSet
* 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 json
* Чтение данных из файла необходимо реализовать с помощью класса java.io.BufferedInputStream
* Запись данных в файл необходимо реализовать с помощью класса java.io.BufferedWriter
* 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)
* add\_if\_max {element} : добавить новый элемент в коллекцию, если его значение превышает значение наибольшего элемента этой коллекции
* 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
* min\_by\_usa\_box\_office : вывести любой объект из коллекции, значение поля usaBoxOffice которого является минимальным
* count\_less\_than\_length length : вывести количество элементов, значение поля length которых меньше заданного
* filter\_starts\_with\_name name : вывести элементы, значение поля name которых начинается с заданной подстроки

**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 int oscarsCount; //Значение поля должно быть больше 0

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

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

private MovieGenre genre; //Поле может быть null

private Person director; //Поле может быть null

}

public class Coordinates {

private Float x; //Значение поля должно быть больше -959, Поле не может быть null

private int y; //Значение поля должно быть больше -822

}

public class Person {

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

private java.time.LocalDate birthday; //Поле может быть null

private Double height; //Поле не может быть null, Значение поля должно быть больше 0

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

}

public enum MovieGenre {

ACTION,

THRILLER,

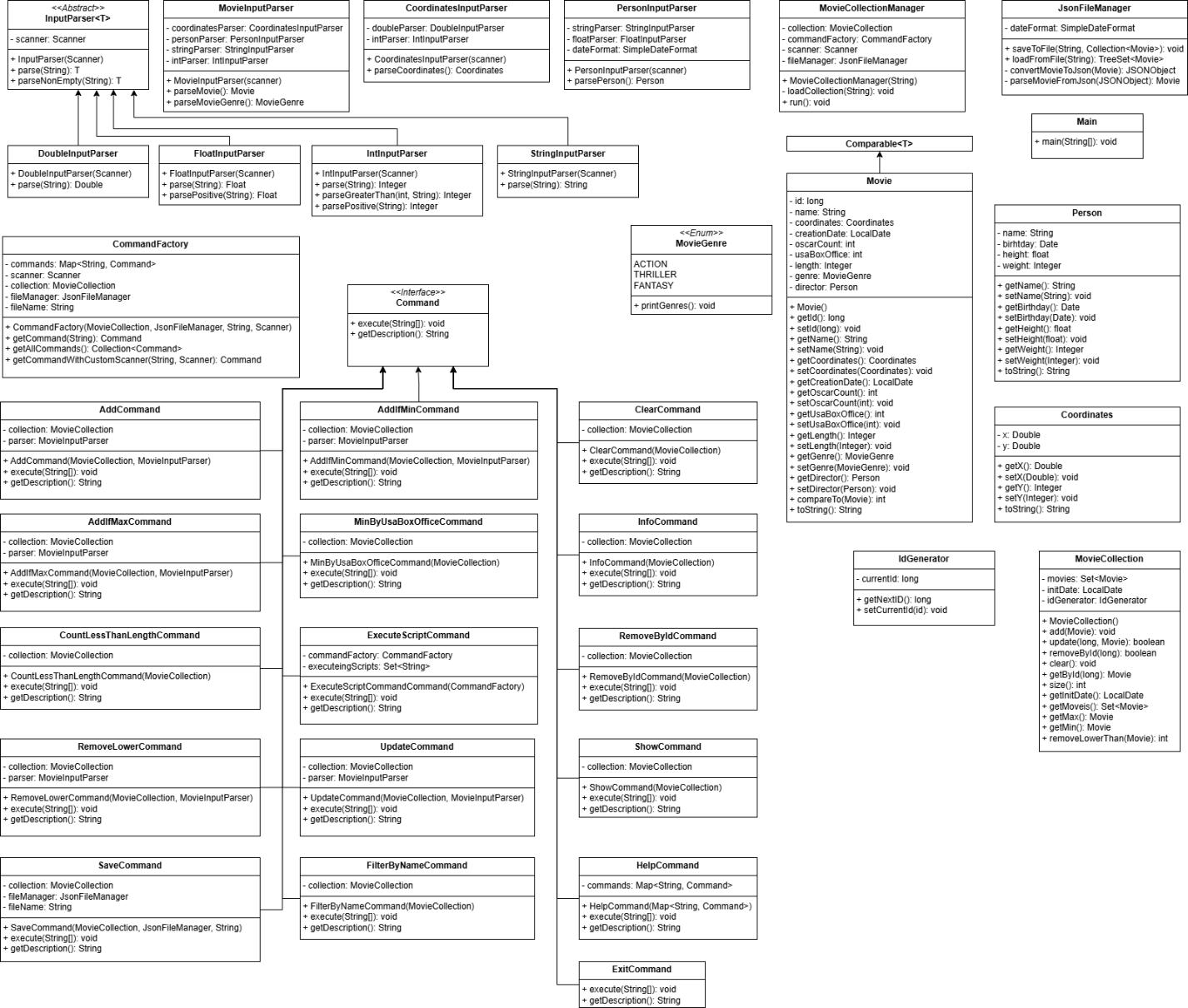
FANTASY;

}

## Исходный код программы

Ссылка на исходный код программы: <https://disk.yandex.ru/d/RgCAx6EgCi_cdg>

## UML-диаграмма классов



## Вывод

В ходе выполнения лабораторной работы я научился работать с коллекциями, подробнее узнал о работе с файлами и вводом/выводом. Изучил Javadoc и его особенности.