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

Факультет Программной Инженерии и Компьютерной Техники

**Лабораторная работа по программированию №5**

Вариант: 3117500

Преподаватель: Письмак Алексей Евгеньевич

Выполнил: Барабанщиков Андрей Дмитриевич

Группа: Р3117

Санкт-Петербург, 2022г

# 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.PriorityQueue
* При запуске приложения коллекция должна автоматически заполняться значениями из файла.
* Имя файла должно передаваться программе с помощью: **переменная окружения**.
* Данные должны храниться в файле в формате json
* Чтение данных из файла необходимо реализовать с помощью класса java.io.InputStreamReader
* 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)
* add\_if\_min {element} : добавить новый элемент в коллекцию, если его значение меньше, чем у наименьшего элемента этой коллекции
* remove\_greater {element} : удалить из коллекции все элементы, превышающие заданный
* remove\_lower {element} : удалить из коллекции все элементы, меньшие, чем заданный
* filter\_less\_than\_oscars\_count oscarsCount : вывести элементы, значение поля oscarsCount которых меньше заданного
* filter\_greater\_than\_director director : вывести элементы, значение поля director которых больше заданного
* print\_unique\_oscars\_count : вывести уникальные значения поля oscarsCount всех элементов в коллекции

**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 int id; //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.LocalDate creationDate; //Поле не может быть null, Значение этого поля должно генерироваться автоматически

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

private MovieGenre genre; //Поле не может быть null

private MpaaRating mpaaRating; //Поле может быть null

private Person director; //Поле может быть null

}

public class Coordinates {

private Float x; //Поле не может быть null

private long y; //Значение поля должно быть больше -863

}

public class Person {

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

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

private Color hairColor; //Поле не может быть null

private Location location; //Поле не может быть null

}

public class Location {

private Double x; //Поле не может быть null

private double y;

private String name; //Поле может быть null

}

public enum MovieGenre {

ADVENTURE,

THRILLER,

HORROR;

}

public enum MpaaRating {

G,

PG,

PG\_13,

NC\_17;

}

public enum Color {

RED,

BLUE,

YELLOW,

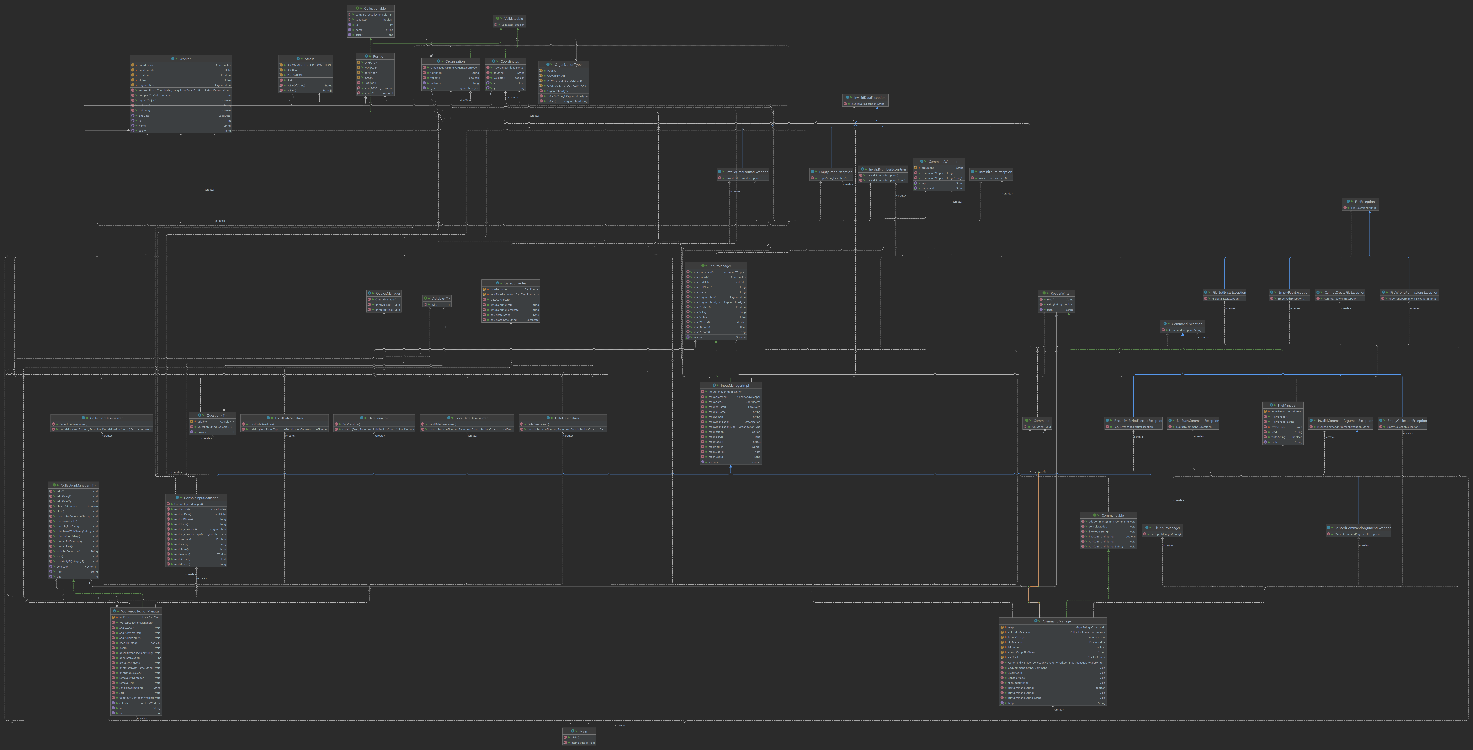
ORANGE,

WHITE;

}

# Ход работы

*Диаграмма классов программы:*



Документация Javadoс:

<https://dimaledentsov.github.io/lab5_javadoc.github.io/>

Исходный код:

<https://github.com/DimaLedentsov/ITMO/tree/main/sem_2/lab5>

# Вывод

Во время выполнения лабораторной работы я написал типо полноценное консольное приложение. Научился работать с библиотеками ввода/вывода, библиотекой GSON для парсинга json.