

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

Факультет программной инженерии и компьютерной техники

Лабораторная работа №5  
по дисциплине «Программирование»  
Вариант 5186

Выполнил  
Разгоняев Максим Витальевич  
Группа Р3131  
Проверил  
Обляшевский Севастьян Александрович

Санкт-Петербург 2025

## Оглавление

Задание.....	2
Диаграмма классов разработанной программы.....	6
Исходный код программы .....	6
Выводы по работе.....	6

### 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.LinkedList`
- ☐ 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.BufferedReader`
- ☐ Запись данных в файл необходимо реализовать с помощью класса `java.io.BufferedOutputStream`
- ☐ Все классы в программе должны быть задокументированы в формате `javadoc`.
- ☐ Программа должна корректно работать с неправильными данными (ошибки пользовательского ввода, отсутствие 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_at index` : удалить элемент, находящийся в заданной позиции коллекции (`index`)
- ☐ `remove_first` : удалить первый элемент из коллекции
- ☐ `reorder` : отсортировать коллекцию в порядке, обратном нынешнему
- ☐ `sum_of_length` : вывести сумму значений поля `length` для всех элементов коллекции
- ☐ `count_by_operator operator` : вывести количество элементов, значение поля `operator` которых равно заданному
- ☐ `print_field_descending_oscars_count` : вывести значения поля `oscarsCount` всех элементов в порядке убывания

#### **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'e; 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.LocalDate creationDate; //Поле не может быть null, Значение этого
поля должно генерироваться автоматически
    private int oscarsCount; //Значение поля должно быть больше 0
    private Integer length; //Поле не может быть null, Значение поля должно быть
больше 0
    private MovieGenre genre; //Поле может быть null
    private Mpaarating mpaarating; //Поле может быть null
    private Person operator; //Поле может быть null
}
public class Coordinates {
    private double x; //Значение поля должно быть больше -817
    private Long y; //Поле не может быть null
}
public class Person {
    private String name; //Поле не может быть null, Строка не может быть пустой
    private Long height; //Поле не может быть null, Значение поля должно быть больше
0
    private float weight; //Значение поля должно быть больше 0
```

```
private Color eyeColor; //Поле может быть null
private Country nationality; //Поле может быть null
}

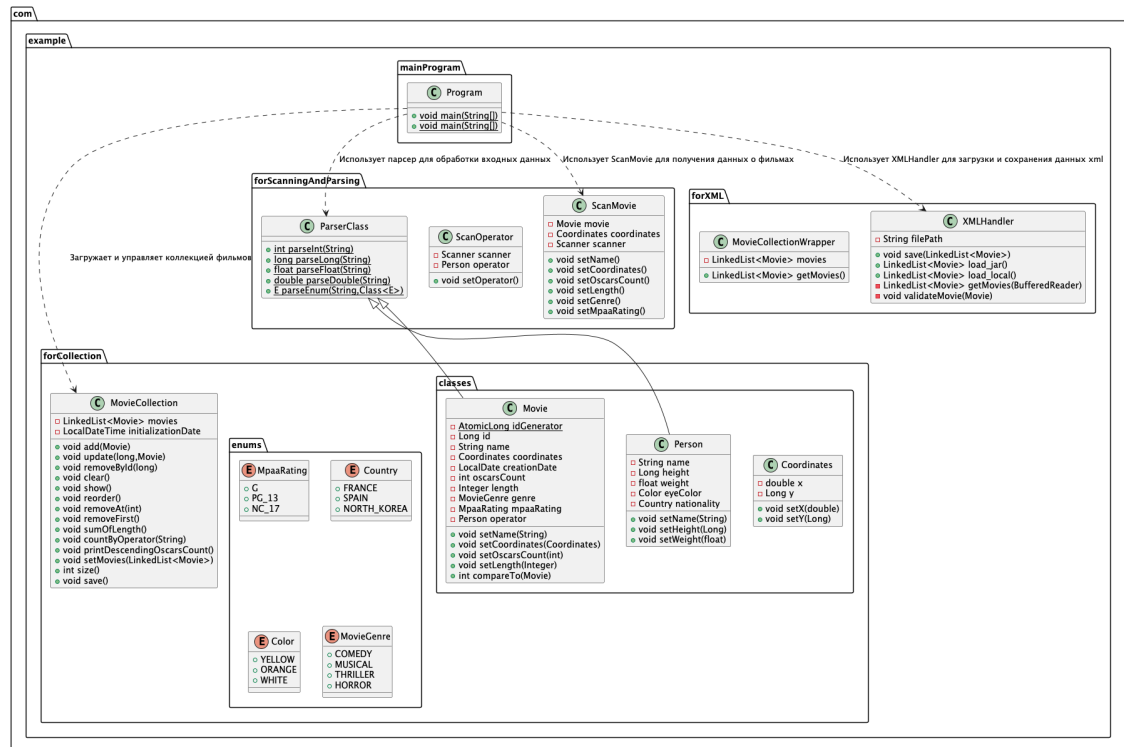
public enum MovieGenre {
    COMEDY,
    MUSICAL,
    THRILLER,
    HORROR;
}

public enum Mpaarating {
    G,
    PG_13,
    NC_17;
}

public enum Color {
    YELLOW,
    ORANGE,
    WHITE;
}

public enum Country {
    FRANCE,
    SPAIN,
    NORTH_KOREA;
}
```

## Диаграмма классов разработанной программы



Также данная диаграмма доступна по [ссылке](#)

## Исходный код программы

Исходный код программы доступен по [ссылке](#)

## Выводы по работе

В ходе выполнения лабораторной работы я повторил основные принципы ООП, повторил работу с классами, объектами, методами и конструкторами. Изучил работу с файлами и потоками ввода-вывода в Java.