



Лабораторная работа №4
По дисциплине
«Программирование»

Вариант №31184

Выполнили студенты группы Р3118:
Богданова Мария Михайловна
Рамеев Тимур Ильгизович

Преподаватель:
Кулинич Ярослав Вадимович

Оглавление

Текст задания.	3
Диаграмма классов	4
Исходный код	4
Вывод программы	5
Вывод	5

Текст задания.

Внимание! У разных вариантов разный текст задания!

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса `Person`, 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: **переменная окружения**.
- 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)
- `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_starts_with_name name` : вывести элементы, значение поля name которых начинается с заданной подстроки
- `print_ascending` : вывести элементы коллекции в порядке возрастания
- `print_descending` : вывести элементы коллекции 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; введенное 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 Person {
    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 double height; //Значение поля должно быть больше 0
    private java.time.ZonedDateTime birthday; //Поле может быть null
    private Color hairColor; //Поле может быть null
    private Country nationality; //Поле может быть null
    private Location location; //Поле может быть null
}

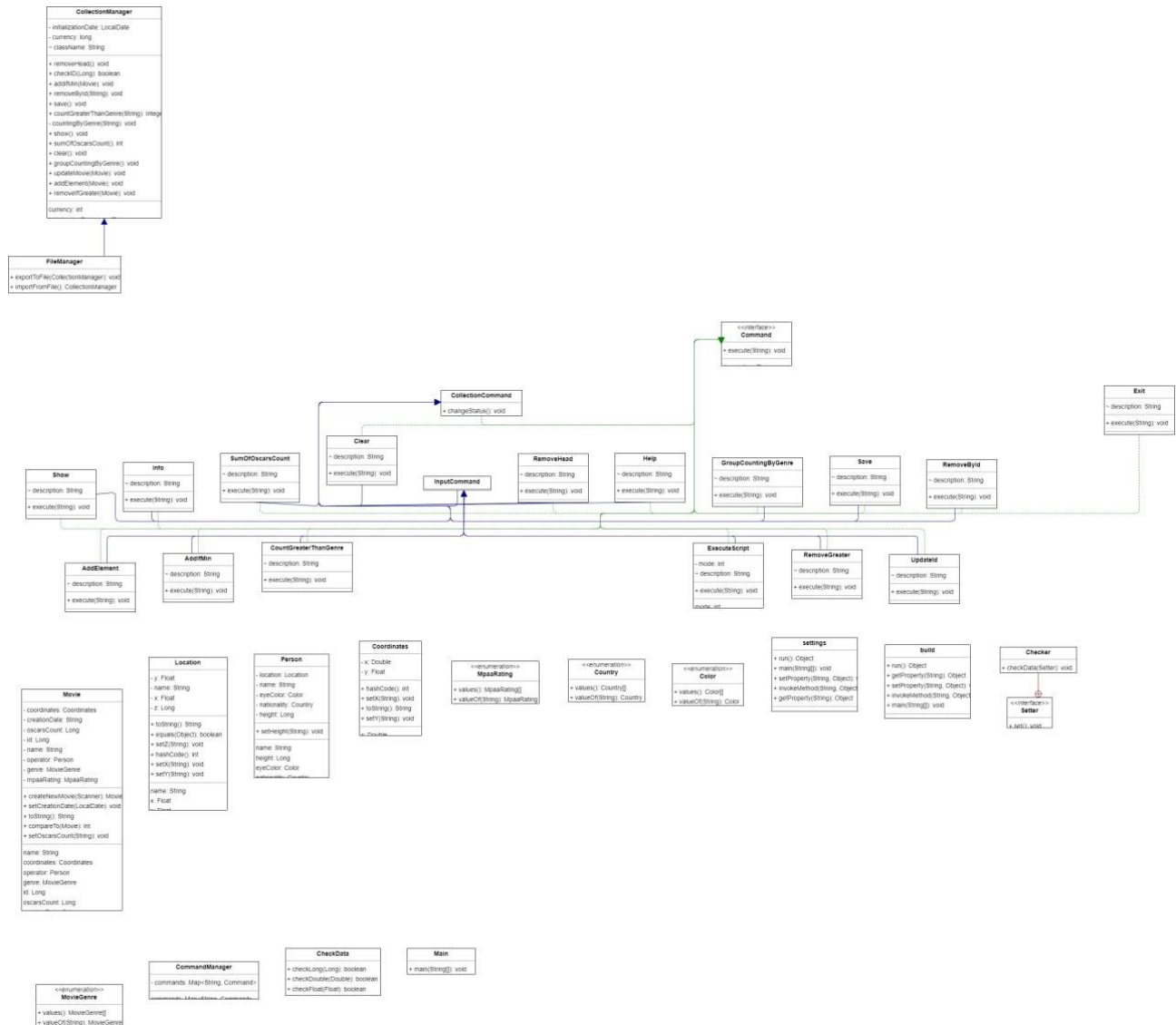
public class Coordinates {
    private Integer x; //Поле не может быть null
    private Float y; //Максимальное значение поля: 663, Поле не может быть null
}

public class Location {
    private Float x; //Поле не может быть null
    private long y;
    private Long z; //Поле не может быть null
}

public enum Color {
    GREEN,
    YELLOW,
    BROWN;
}

public enum Country {
    RUSSIA,
    UNITED_KINGDOM,
    FRANCE,
    SOUTH_KOREA;
}
```

Диаграмма классов



Исходный код

<https://github.com/Stt1xX/ForMyFutureJob/tree/main/Second%20sem/Programming/Lab5/Work/progalab5/src/>

Вывод программы

```
Фильм успешно добавлен.
Фильмов в коллекции:1

Введите команду: show
  Название: Test | id: 1
  Дата создания: 20/05/2023
  Количество оскаров: 1
  Жанр: DRAMA
  Координаты: x = 1.0, y = 2.0
  Рейтинг МРАА: null
  имя режиссера: 3
  Цвет глаз режиссера : RED
  Национальность режиссера: VATICAN
  Рост режиссера: 155
  Координаты оператора: x = 1.0, y = 2.0, z = 3
  Название локации: Location
  -----

Коллекция содержит 1 элементов.

Введите команду: info
Справка:
Дата инициализации коллекции: 2023-05-20
Количество элементов: 1
Тип коллекции: java.util.ArrayDeque
```

Вывод

В ходе выполнения данной лабораторной работы мы исследовали работу с коллекциями, изучили обобщенные и параметризованные типы, потоки ввода-вывода, а также ознакомились с пакетами `java.io/java.nio` и с утилитой `Javadoc`.