

Федеральное государственное автономное образовательное учреждение высшего
образования Национальный исследовательский университет ИТМО

Факультет программной инженерии и компьютерной техники

Программирование

Лабораторная работа №5

Вариант № 40635

Выполнил: студент группы Р3108,
Васильев Н. А.

Преподаватель: Письмак Алексей
Евгеньевич

Санкт-Петербург 2024

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.LinkedHashSet**
- 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.util.Scanner**
- 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, 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_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
- **max_by_location** : вывести любой объект из коллекции, значение поля location которого является максимальным
- **count_greater_than_weight weight** : вывести количество элементов, значение поля weight которых больше заданного
- **filter_less_than_height height** : вывести элементы, значение поля height которых меньше заданного

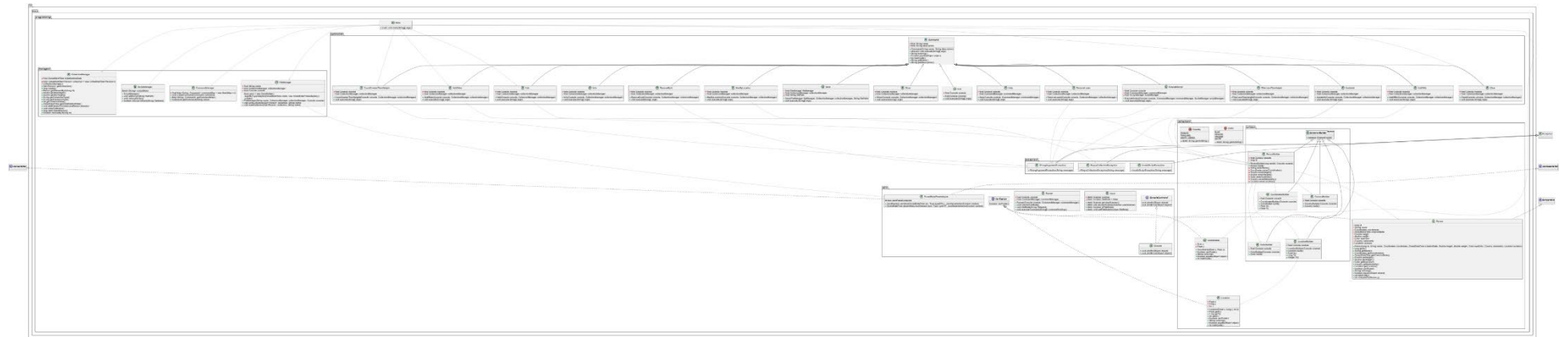
Формат ввода команд:

- 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 Person {
    private long 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.ZonedDateTime creationDate; //Поле не может быть null, Значение
    этого поля должно генерироваться автоматически
    private Double height; //Поле не может быть null, Значение поля должно быть
    больше 0
    private double weight; //Значение поля должно быть больше 0
    private Color eyeColor; //Поле не может быть null
    private Country nationality; //Поле может быть null
    private Location location; //Поле может быть null
}
public class Coordinates {
    private float x;
    private Float y; //Поле не может быть null
}
public class Location {
    private Float x; //Поле не может быть null
    private Long y; //Поле не может быть null
    private int z;
}
public enum Color {
    BLUE,
    YELLOW,
    ORANGE,
    WHITE;
}
public enum Country {
    FRANCE,
    THAILAND,
    SOUTH_KOREA;
```

Диаграмма классов реализованной объектной модели.



Исходный код программы

Ссылка на GitHub: <https://github.com/kihort-si/itmo/blob/main/proga/lab5>

Вывод

При выполнении данной лабораторной работы я познакомился с коллекциями, потоками ввода-вывода и работой с файлами в Java, утилитой javadoc.