Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

*Факультет Программной инженерии и компьютерной техники*

**Лабораторная работа №5**

Вариант № 313204

Группа: P3132

Выполнил: Волков Г. А.

Проверил:

к.т.н. преподаватель Наумова Н. А.

г.Санкт-Петербург

2023г.

Оглавление

[Оглавление 2](#_Toc136518954)

[Текст задания 3](#_Toc136518955)

[Диаграмма классов 6](#_Toc136518956)

[Исходный код программы 7](#_Toc136518957)

[Вывод 8](#_Toc136518958)

Текст задания

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса Route, 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.LinkedHashMap
* 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.FileReader
* Запись данных в файл необходимо реализовать с помощью класса java.io.OutputStreamWriter
* 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
* insert null {element} : добавить новый элемент с заданным ключом
* update id {element} : обновить значение элемента коллекции, id которого равен заданному
* remove\_key null : удалить элемент из коллекции по его ключу
* 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)
* history : вывести последние 13 команд (без их аргументов)
* replace\_if\_greater null {element} : заменить значение по ключу, если новое значение больше старого
* remove\_lower\_key null : удалить из коллекции все элементы, ключ которых меньше, чем заданный
* group\_counting\_by\_name : сгруппировать элементы коллекции по значению поля name, вывести количество элементов в каждой группе
* count\_by\_distance distance : вывести количество элементов, значение поля distance которых равно заданному
* print\_field\_descending\_distance : вывести значения поля distance 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'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 Route {

private Integer 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.util.Date creationDate; //Поле не может быть null, Значение этого поля должно генерироваться автоматически

private Location from; //Поле может быть null

private Location to; //Поле не может быть null

private Double distance; //Поле не может быть null, Значение поля должно быть больше 1

}

public class Coordinates {

private long x; //Значение поля должно быть больше -866

private Long y; //Максимальное значение поля: 107, Поле не может быть null

}

public class Location {

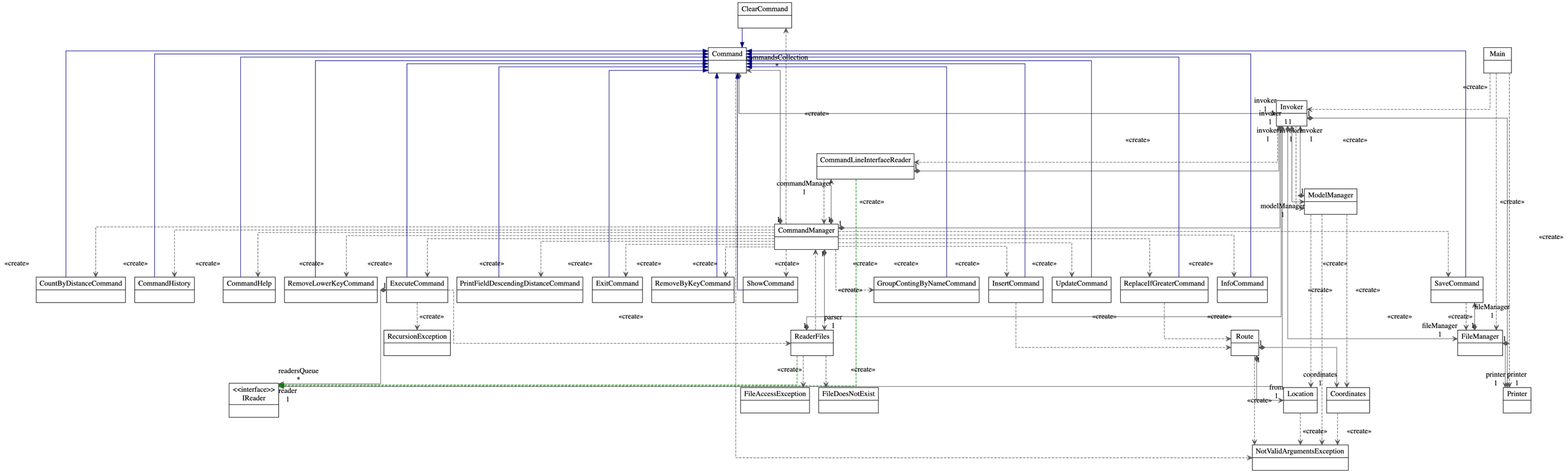
private long x;

private long y;

private Long z; //Поле не может быть null

private String name; //Поле не может быть null

}

Диаграмма классов

Исходный код программы

Ссылка на GitHub: <https://github.com/GrigoryWolf/Lab5>

[](https://github.com/GrigoryWolf/Lab5)

Вывод

В ходе выполнения данной лабораторной работы я научился работать с потоками ввода/вывода, различными коллекциями, файлами, познакомился с инструментом автоматического документирования кода, параметризованными типами.