

Министерство высшего образования и науки Российской Федерации
Национальный научно-исследовательский Университет ИТМО
Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №5
по дисциплине
«Программирование»

Работу выполнил:
Афанасьев Кирилл Александрович,
Студент группы Р3106
Преподаватель:
Байрамова Хумай

г. Санкт-Петербург, 2023

Задание:

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.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.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_greater {element}** : удалить из коллекции все элементы, превышающие заданный
- **min_by_creation_date** : вывести любой объект из коллекции, значение поля `creationDate` которого является минимальным
- **count_greater_than_distance distance** : вывести количество элементов, значение поля `distance` которых больше заданного
- **print_field_ascending_distance** : вывести значения поля `distance` всех элементов в порядке возрастания

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 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.Date creationDate; //Поле не может быть null, Значение этого поля должно
    генерироваться автоматически
    private Location from; //Поле может быть null
    private Location to; //Поле может быть null
    private int distance; // Значение поля должно быть больше 1
}

public class Coordinates {
    private double x; // Значение поля должно быть больше -107
    private Float y; //Значение поля должно быть больше -39. Поле не может быть null
}
```

```

}
public class Location {
    private float x;
    private Long y; // Поле не может быть null
    private Long z; // Поле не может быть null
    private String name; //Строка не может быть пустой, Поле может быть null
}

```

Исходный код:

https://github.com/Zerumi/no5_170123_600900

Диаграмма классов разработанной программы:



Выводы:

Во время выполнения данной лабораторной работы я:

- Ознакомился углубленно с понятием обобщенного программирования, и его примером, реализованным в Java API, а именно, *CollectionFramework*, с использованием материалов книги К. Хорстманна «Java. Библиотека профессионала, том 1» (главы 8-9).
- Разработал 46 классов, описал логику команд в соответствии с паттерном «Команда», ознакомился с его реализацией, практическим применением и смыслом.
- Ознакомился с принципом работы с потоками ввода-вывода данных в Java, библиотеками *java.io* и *java.nio*; а также ознакомился со *Stream API* для выполнения простейших задач.
- Применил знания из предыдущих лабораторных работ для их дополнительного практического применения.

Полученные мною знания пригодятся на практике, в том числе для дальнейшего изучения языка.