

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 по программированию №5

Вариант 3215

Группа: Р3132

Выполнил: Овчаренко Александр Андреевич

Проверил: Усков Иван Владимирович

г. Санкт-Петербург

2022 г.

Оглавление

Задание	3
Выполнение работы	6
Пример выполнения программы	8
Итог	10

Задание

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса `SpaceMarine`, 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 `json`
- Чтение данных из файла необходимо реализовать с помощью класса `java.io.BufferedReader`
- 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_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
- `group_counting_by_name` : сгруппировать элементы коллекции по значению поля name, вывести количество элементов в каждой группе
- `count_by_loyal loyal` : вывести количество элементов, значение поля loyal которых равно заданному
- `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.

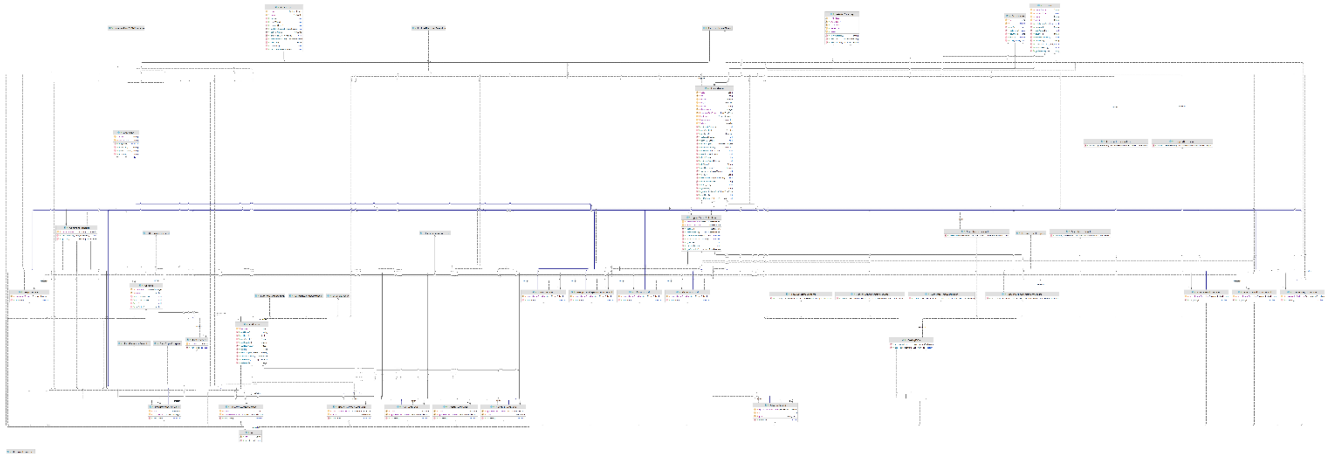
Описание хранимых в коллекции классов:

```
public class SpaceMarine {
    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.LocalDateTime creationDate; //Поле не может быть null, Значение
этого поля должно генерироваться автоматически
    private Integer health; //Поле не может быть null, Значение поля должно быть
больше 0
    private Integer heartCount; //Поле не может быть null, Значение поля должно быть
больше 0, Максимальное значение поля: 3
    private Boolean loyal; //Поле может быть null
    private AstartesCategory category; //Поле не может быть null
    private Chapter chapter; //Поле может быть null
}
public class Coordinates {
    private double x;
    private Long y; //Поле не может быть null
}
public class Chapter {
    private String name; //Поле не может быть null, Строка не может быть пустой
    private String parentLegion;
    private long marinesCount; //Значение поля должно быть больше 0, Максимальное
значение поля: 1000
    private String world; //Поле не может быть null
}
public enum AstartesCategory {
    AGGRESSOR,
    INCEPTOR,
    TACTICAL,
    CHAPLAIN,
    HELIX;
}
```

Выполнение работы

С uml-диаграммой можно ознакомиться по ссылке:

[lab_5_prog/lab_5.png at master · sasaovch/lab_5_prog \(github.com\)](https://github.com/sasaovch/lab_5_prog/blob/master/lab_5_prog.png)



Ссылка на код программы:

[sasaovch/lab_5_prog \(github.com\)](https://github.com/sasaovch/lab_5_prog)

Ссылка на Javadoc:

[lab_5_prog/index.html at master · sasaovch/lab_5_prog \(github.com\)](https://github.com/sasaovch/lab_5_prog/blob/master/lab_5_prog/index.html)

Пример выполнения программы

```
Enter file path
$ file.json
Collection is created
$ 234
Enter world of chapter
$ Mars
Aleksy has been successfully added
$ show
-----
Name: Olya
Id: 2
Health: 2
HeartCount: 2
Loyal: false
Initialization time: 2022-02-14 16:31:45
Chapter:
    Name: Aleks
    ParentLegion: LOL
    MarinesCount: 1
    World: Earth
Coordinates: X - 2.0| Y - 2
INCEPTOR
-----
Name: Danil
Id: 3
Health: 3
HeartCount: 3
$ 234
Enter world of chapter
$ Mars
Aleksy has been successfully added
$ show
-----
Name: Olya
Id: 2
Health: 2
HeartCount: 2
Loyal: false
Initialization time: 2022-02-14 16:31:45
Chapter:
    Name: Aleks
    ParentLegion: LOL
    MarinesCount: 1
    World: Earth
Coordinates: X - 2.0| Y - 2
INCEPTOR
-----
Name: Danil
Id: 3
Health: 3
HeartCount: 3
```


Loyal: null
Initialization time: 2022-02-14 16:32:22
Chapter:
 Name: df
 ParentLegion: Moscow
 MarinesCount: 3
 World: Berga
Coordinates: X - 3.0| Y - 3

TACTICAL

Name: Aleks
Id: 1
Health: 234
HeartCount: 2
Loyal: null
Initialization time: 2022-02-15 08:53:02
Chapter:
null
Coordinates: X - 1.0| Y - 1

HELIX

Name: Aleksey
Id: 4
Health: 324

```
$ remove_id 4
Error! Unknow command. Print 'help' for information
$ help
add {element}          add new element in collection
group_counting_by_name group the elements of the collection by the value of the name field, print the number of elements in each group
add_if_min {element}   add element if its value is less than minimal value in collection (value is health)
execute_script file_name read and execute the script from the specified file
show                   print all elements of collection
clear                  clear the collection
save                   save collection in file
update_id {element}    update the value of a collection item whose id is equal to the specified one
remove_by_id id        remove element with such id
remove_greater {element} remove all items from collection that exceed the specified
remove_lower {element} remove all items smaller than the specified one from the collection
help                   print info about all commands
exit                  completes programm
print_descending       print all the elements of the collection in descending order
count_by_loyal loyal   print the number of elements whose value of the loyal field is equal to the specified
info                   print info about collection: type, initialization date, number of elements
$ remove_by_id 4
Aleksey has been successfully removed
$ exit
Good Buy!
\(?_?)/
```

Итог

В результате выполнения лабораторной работы был изучен способ организации работы с потоками ввода\вывода, получилось сделать консольное приложение. Я понял, что использование системы контроля версии Git очень помогает при работе с большим проектом. Мне понравилось выполнять данную лабораторную работу, я научился делать приложение, взаимодействующее с пользователем.