



УНИВЕРСИТЕТ ИТМО

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»
ФАКУЛЬТЕТ СИСТЕМ УПРАВЛЕНИЯ И РОБОТОТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №5

по дисциплине
“Программирование”
Вариант: 786421.

выполнил:

Студент группы R3137:

Ракин Илья Николаевич

Преподаватель:

Исаев Александр Сергеевич

г. Санкт-Петербург
2022

1. Задание

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса `StudyGroup`, 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.Stack`
- 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 `csv`
- Чтение данных из файла необходимо реализовать с помощью класса `java.io.BufferedReader`
- Запись данных в файл необходимо реализовать с помощью класса `java.io.FileOutputStream`
- 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)
- `remove_at index`: удалить элемент, находящийся в заданной позиции коллекции (index)
- `remove_last`: удалить последний элемент из коллекции
- `remove_greater {element}`: удалить из коллекции все элементы, превышающие заданный
- `sum_of_expelled_students`: вывести сумму значений поля `expelledStudents` для всех элементов коллекции
- `filter_starts_with_name name`: вывести элементы, значение поля `name` которых начинается с заданной подстроки
- `print_field_ascending_students_count`: вывести значения поля `studentsCount` 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; 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.

2. Описание хранимых в коллекции классов:

```
public class StudyGroup {
    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.ZonedDateTime creationDate; //Поле не может быть null, Значение этого поля должно генерироваться
    автоматически
    private long studentsCount; //Значение поля должно быть больше 0
    private Long expelledStudents; //Значение поля должно быть больше 0, Поле может быть null
    private FormOfEducation formOfEducation; //Поле может быть null
    private Semester semesterEnum; //Поле не может быть null
    private Person groupAdmin; //Поле не может быть null
}

public class Coordinates {
    private Long x; //Поле не может быть null
    private double y; //Значение поля должно быть больше -352
}

public class Person {
    private String name; //Поле не может быть null, Строка не может быть пустой
    private String passportID; //Поле не может быть null
    private Color eyeColor; //Поле не может быть null
    private Country nationality; //Поле может быть null
}

public enum FormOfEducation {
    DISTANCE_EDUCATION,
    FULL_TIME_EDUCATION,
    EVENING_CLASSES;
}

public enum Semester {
    FIRST,
    SECOND,
}
```

```

        THIRD,
        FOURTH,
        FIFTH;
    }
    public enum Color {
        BLUE,
        YELLOW,
        WHITE;
    }
    public enum Country {
        UNITED_KINGDOM,
        FRANCE,
        SPAIN,
        INDIA,
        SOUTH_KOREA;
    }
}

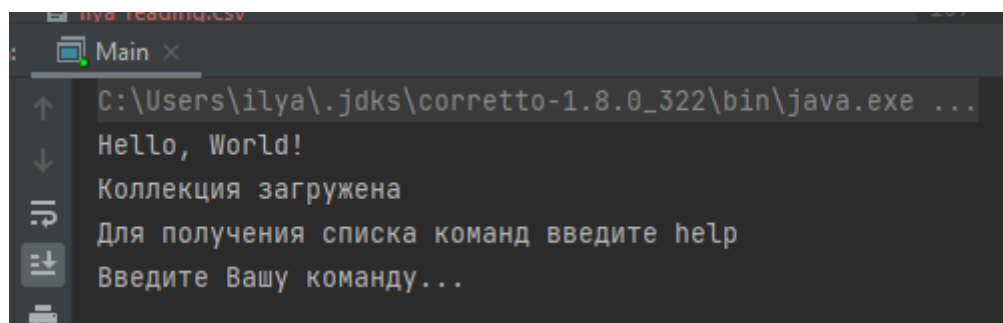
```

3. Исходный код

<https://github.com/xhackerino/Lab5>



4. Результат работы



5. Вывод:

У меня получилось!