

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 по программированию

Консольное приложение

Вариант № 3116011

Группа: Р3116

Выполнил:

Сиразетдинов А. Н.

Проверил:

Письмак А. Е.

г. Санкт-Петербург

2023 год

ОГЛАВЛЕНИЕ

Текст задания.....	3
Диаграмма классов разработанной программы.....	6
Исходный код программы	8
Вывод	9

ТЕКСТ ЗАДАНИЯ

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.ArrayDeque**
- При запуске приложения коллекция должна автоматически заполняться значениями из файла.
- Имя файла должно передаваться программе с помощью: аргумент командной строки.
- Данные должны храниться в файле в формате **xml**
- Чтение данных из файла необходимо реализовать с помощью класса **java.util.BufferedReader**
- Запись данных в файл необходимо реализовать с помощью класса **java.io.BufferedOutputStream**
- 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:

AddElement

Команда 'add' Добавляет новый элемент в коллекцию

AddIfMax

Команда 'add_if_max' Добавляет элемент в коллекцию если он больше максимального

Clear

Команда 'clear' Очищает коллекцию

Command

Абстрактный класс для всех команд

CountByAverageMark

Посчитать количество групп с требуемой средней оценкой

CountLessThanExpelledStudents

Команда 'count_less_than_expelled_students' Выводит количество элементов, значение поля expelled_students которых меньше заданному

Execute

Команда 'execute_script' Читает и исполняет скрипт из указанного файла.

Exit

Команда 'exit' завершить программу (без сохранения в файл)

Help

Команда 'help' вывести справку по доступным командам

History

Команда 'history' Выводит последние 5 команд (без их аргументов)

Info

Команда 'info' Выводит в стандартный поток вывода информацию о коллекции (тип, дата инициализации, количество элементов и т.д.)

RemoveAllByAverageMark

Команда 'remove_all_by_average_mark' Удаляет из коллекции все элементы, значение поля average_mark которого эквивалентно заданному

RemoveById

Команда 'remove_by_id' Удаляет элемент из коллекции по его id

RemoveGreater

Команда 'remove_greater' Удаляет из коллекции все элементы, превышающие заданный

Save

Команда 'save' Сохраняет коллекцию в файл

Show

Команда 'show' Выводит в стандартный поток вывода все элементы коллекции в строковом представлении

Update

Команда 'update' Обновляет значение элемента коллекции, id которого равен заданному

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:

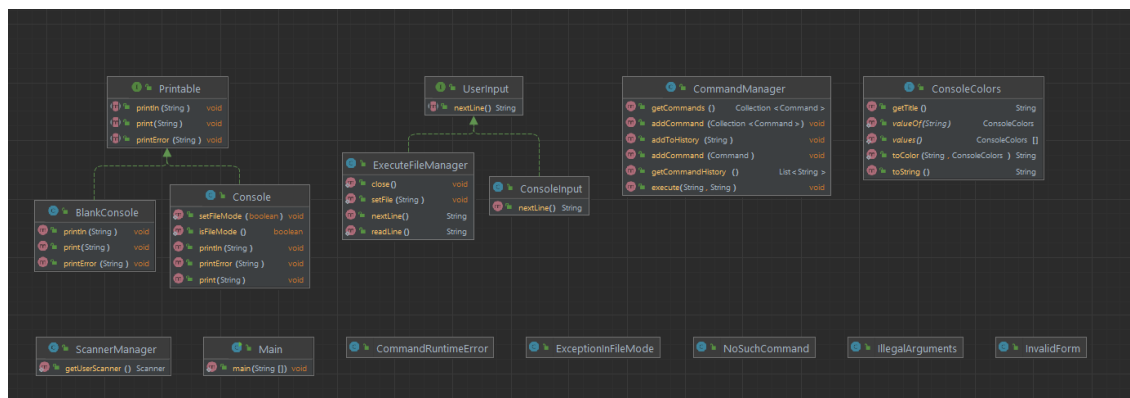
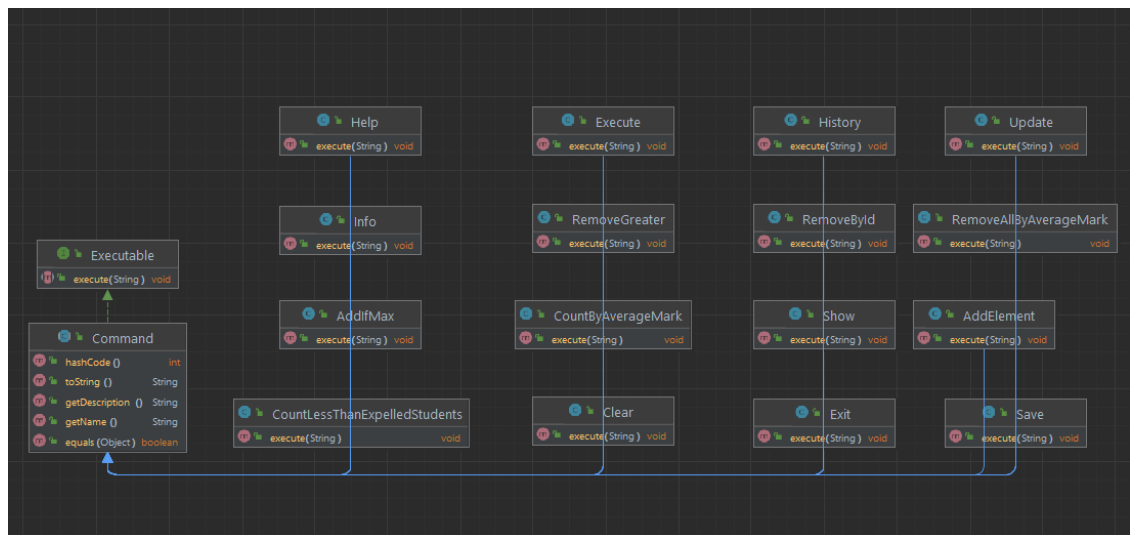
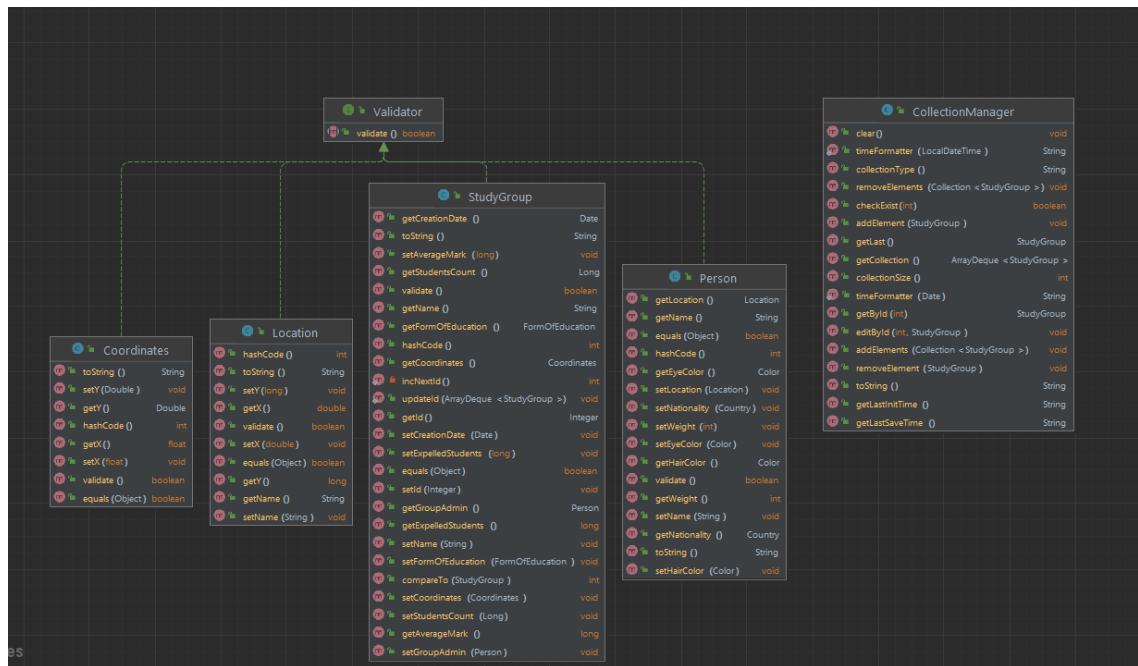
```
public class StudyGroup {
    private int id; //Значение поля должно быть больше 0, Значение этого поля должно быть
    уникальным, Значение этого поля должно генерироваться автоматически
    private String name; //Поле не может быть null, Строка не может быть пустой
    private Coordinates coordinates; //Поле не может быть null
    private java.time.LocalDate creationDate; //Поле не может быть null, Значение этого поля должно
    генерироваться автоматически
    private Integer studentsCount; //Значение поля должно быть больше 0, Поле не может быть null
    private Long expelledStudents; //Значение поля должно быть больше 0, Поле может быть null
    private float averageMark; //Значение поля должно быть больше 0
    private FormOfEducation formOfEducation; //Поле не может быть null
    private Person groupAdmin; //Поле не может быть null
}
public class Coordinates {
    private int x;
    private long y; //Максимальное значение поля: 463
}
```

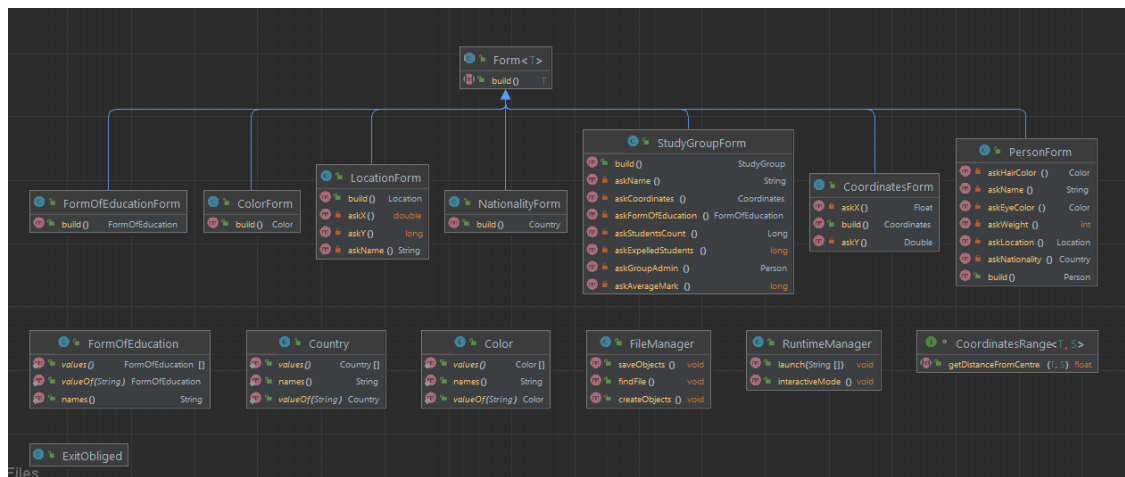
```

public class Person {
    private String name; //Поле не может быть null, Строка не может быть пустой
    private Integer weight; //Поле не может быть null, Значение поля должно быть больше 0
    private Color eyeColor; //Поле может быть null
    private Color hairColor; //Поле может быть null
    private Country nationality; //Поле не может быть null
    private Location location; //Поле может быть null
}
public class Location {
    private Long x; //Поле не может быть null
    private long y;
    private String name; //Строка не может быть пустой, Поле может быть null
}
public enum FormOfEducation {
    DISTANCE_EDUCATION,
    FULL_TIME_EDUCATION,
    EVENING_CLASSES;
}
public enum Color {
    BLUE,
    ORANGE,
    WHITE;
}
public enum Color {
    GREEN,
    RED,
    YELLOW,
    WHITE,
    BROWN;
}
public enum Country {
    USA,
    CHINA,
    JAPAN;
}

```

ДИАГРАММА КЛАССОВ РАЗРАБОТАННОЙ ПРОГРАММЫ.





ИСХОДНЫЙ КОД ПРОГРАММЫ

[github](#)

[javadocs](#)

ВЫВОД

В процессе выполнения лабораторной работы я познакомился с организацией коллекций, их сортировки, работы с файлами и работы с консолью. Узнал про различные потоки ввода-вывода, утилиту Javadoc, классы оболочки и дженерики