Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа№5

Вариант 3117

Выполнил:

Бармичев Григорий Андреевич

Группа P3110

Проверил:

Мустафаева Айнур Вугар Кызы

Преподаватель практик

**Содержание**

[Задание 3](#_Toc194396838)

[Диаграмма классов 5](#_Toc194396839)

[Исходный код программы 5](#_Toc194396840)

[Вывод 5](#_Toc194396841)

## Задание

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса LabWork, 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.Vector
* 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 csv
* Чтение данных из файла необходимо реализовать с помощью класса java.util.Scanner
* 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
* 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)
* shuffle : перемешать элементы коллекции в случайном порядке
* reorder : отсортировать коллекцию в порядке, обратном нынешнему
* min\_by\_coordinates : вывести любой объект из коллекции, значение поля coordinates которого является минимальным
* filter\_less\_than\_minimal\_point minimalPoint : вывести элементы, значение поля minimalPoint которых меньше заданного
* print\_ascending : вывести элементы коллекции 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 LabWork {

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 Double minimalPoint; //Поле может быть null, Значение поля должно быть больше 0

private String description; //Длина строки не должна быть больше 5287, Поле не может быть null

private int tunedInWorks;

private Difficulty difficulty; //Поле не может быть null

private Discipline discipline; //Поле не может быть null

}

public class Coordinates {

private Integer x; //Поле не может быть null

private float y; //Максимальное значение поля: 654

}

public class Discipline {

private String name; //Поле не может быть null, Строка не может быть пустой

private Long practiceHours; //Поле не может быть null

}

public enum Difficulty {

NORMAL,

VERY\_HARD,

INSANE,

HOPELESS;

}

## Диаграмма классов

Изображение выглядит как снимок экрана, текст, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

## Исходный код программы

Ссылка на репозиторий: <https://github.com/barmichevg/prog_lab5/tree/main/src>

## Вывод

Во время выполнения данной лабораторной работы я научился работать с различными структурами данных в Java и файлами, а также углубил свои знания о ООП в Јаѵа, изучил параметризованные типы, wildcard-параметры и утилиту javadoc.