

Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет ПИ и КТ

Лабораторная работа №5

по дисциплине: «Программирование»

Вариант 3111002

Выполнил:

**Болорболд Аригуун**,

группа P3111

Преподаватель:

**Письмак Алексей Евгеньевич**

Санкт-Петербург

2023



1. **Задание:**

Внимание! У разных вариантов разный текст задания!

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса Organization, 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.Hashtable
* 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 json
* Чтение данных из файла необходимо реализовать с помощью класса java.io.InputStreamReader
* 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
* insert null {element} : добавить новый элемент с заданным ключом
* update id {element} : обновить значение элемента коллекции, id которого равен заданному
* remove\_key null : удалить элемент из коллекции по его ключу
* 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\_lower {element} : удалить из коллекции все элементы, меньшие, чем заданный
* history : вывести последние 9 команд (без их аргументов)
* remove\_lower\_key null : удалить из коллекции все элементы, ключ которых меньше, чем заданный
* filter\_contains\_name name : вывести элементы, значение поля name которых содержит заданную подстроку
* filter\_less\_than\_type type : вывести элементы, значение поля type которых меньше заданного
* print\_field\_descending\_annual\_turnover : вывести значения поля annualTurnover всех элементов 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'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 Organization {

private Integer 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.LocalDate creationDate; //Поле не может быть null, Значение этого поля должно генерироваться автоматически

private double annualTurnover; //Значение поля должно быть больше 0

private OrganizationType type; //Поле не может быть null

private Address officialAddress; //Поле может быть null

}

public class Coordinates {

private Double x; //Поле не может быть null

private int y; //Максимальное значение поля: 614

}

public class Address {

private String street; //Поле может быть null

private String zipCode; //Длина строки должна быть не меньше 9, Поле не может быть null

}

public enum OrganizationType {

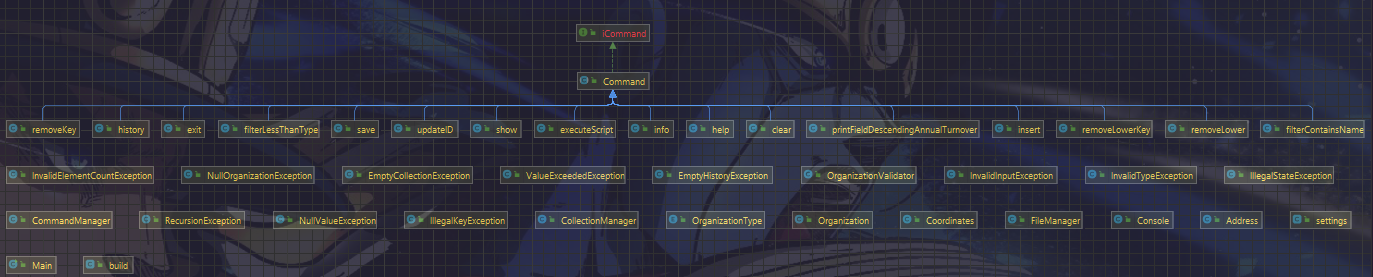
COMMERCIAL,

PUBLIC,

PRIVATE\_LIMITED\_COMPANY;

}

1. Диаграмма:



1. Исходный код:

<https://github.com/XVIIStarPlatinum/Programming/tree/main/II/Lab5>

Вывод:

Несмотря на некоторые спорные моменты с оформлением задания, всё обошлось почти идеально. Программа умеет улавливать исключения и обработать их, умеет улавливать рекурсии. Лично научился работать с новыми вещами, как коллекции, потоки (не те) и лямбда-выражениями. За 6 лабой!