

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное  
образовательное учреждение высшего образования  
"Санкт-Петербургский национальный исследовательский университет ИТМО"

**ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ**

**Лабораторная работа №5**  
по дисциплине  
**"ПРОГРАММИРОВАНИЕ"**

Вариант №2225368

*Выполнил:*  
Студент группы Р3118  
Шульга Артём Игоревич  
*Преподаватель:*  
Сорокин Роман Борисович



Санкт-Петербург, 2022

## Текст задания

Введите вариант:

### Внимание! У разных вариантов разный текст задания!

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса `City`, 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.HashMap`
- 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.Scanner`
- 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:**

- `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)
- `history` : вывести последние 13 команд (без их аргументов)
- `replace_if_greater null {element}` : заменить значение по ключу, если новое значение больше старого
- `remove_lower_key null` : удалить из коллекции все элементы, ключ которых меньше, чем заданный
- `filter_greater_than_climate climate` : вывести элементы, значение поля `climate` которых больше заданного
- `print_ascending` : вывести элементы коллекции в порядке возрастания
- `print_field_ascending_government` : вывести значения поля `government` 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.

**Описание хранимых в коллекции классов:**

```
public class City {
    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 area; //Значение поля должно быть больше 0
    private int population; //Значение поля должно быть больше 0
    private long metersAboveSeaLevel;
    private java.time.LocalDate establishmentDate;
    private Climate climate; //Поле может быть null
    private Government government; //Поле может быть null
    private Human governor; //Поле не может быть null
}

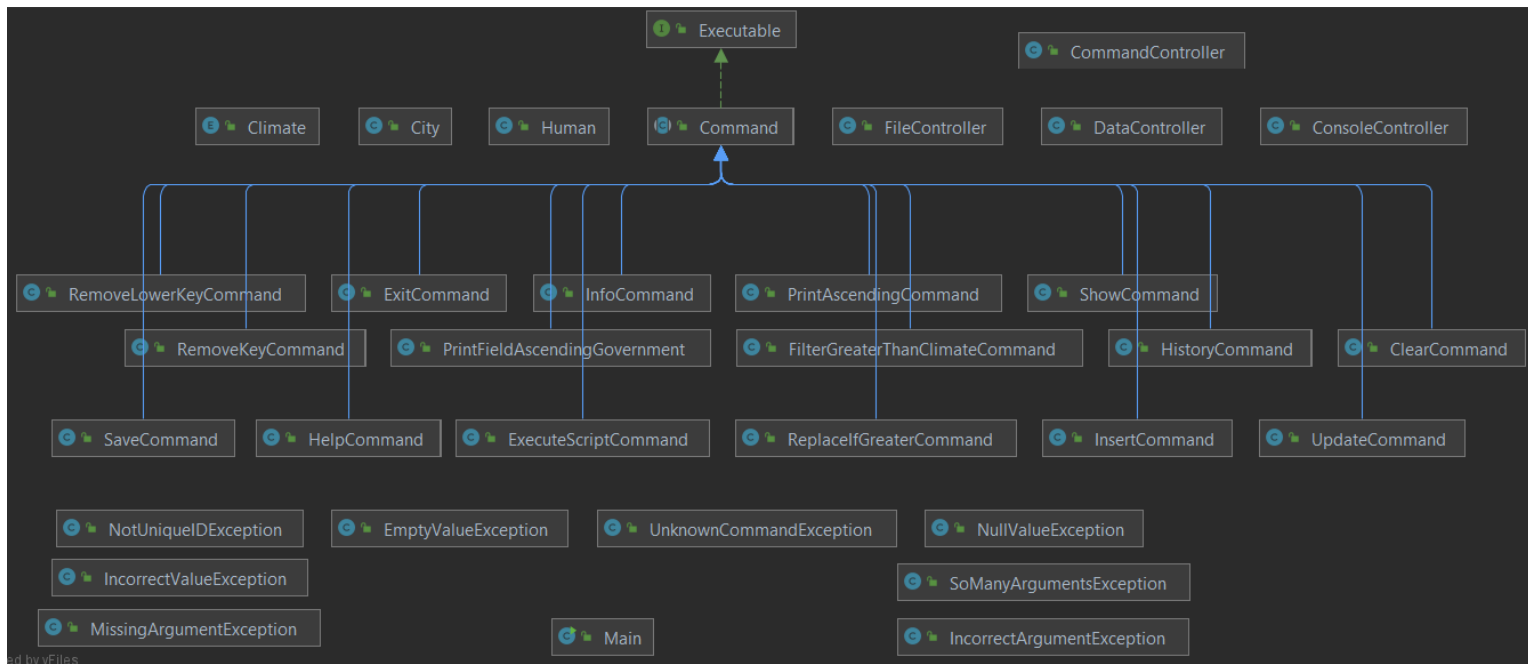
public class Coordinates {
    private float x; //Значение поля должно быть больше -407
    private Integer y; //Поле не может быть null
}

public class Human {
    private long age; //Значение поля должно быть больше 0
    private java.time.LocalDate birthday;
}

public enum Climate {
    RAIN_FOREST,
    HUMIDSUBTROPICAL,
    SUBARCTIC;
}

public enum Government {
    KRITARCHY,
    MATRIARCHY,
    OLIGARCHY,
    PLUTOCRACY,
    REPUBLIC;
}
```

## Диаграмма классов



## **Исходный код программы**

Репозиторий на github

<https://github.com/korlir/itmo-2sem/tree/main/prog/lab5>

## **Вывод по лабораторной работе**

В ходе выполнения данной лабораторной работы я научился работать с коллекциями в Java, работать с потоками ввода и вывода, а также познакомился с утилитой `javadoc`. Знания, полученные мною в ходе выполнения данной работы, пригодятся мне при дальнейшей разработке программных продуктов.