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**

Работу выполнил:

Гурьянов Кирилл Алексеевич

группа: Р3131

Преподаватель:

Осипов Святослав Владимирович

Санкт-Петербург

2022

**Номер варианта – 62938. Задание 1.**

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса MusicBand, 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.HashSet
* 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.BufferedInputStream
* 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, 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)
* add\_if\_min {element} : добавить новый элемент в коллекцию, если его значение меньше, чем у наименьшего элемента этой коллекции
* remove\_lower {element} : удалить из коллекции все элементы, меньшие, чем заданный
* history : вывести последние 11 команд (без их аргументов)
* filter\_starts\_with\_description description : вывести элементы, значение поля description которых начинается с заданной подстроки
* filter\_less\_than\_number\_of\_participants numberOfParticipants : вывести элементы, значение поля numberOfParticipants которых меньше заданного
* print\_unique\_genre : вывести уникальные значения поля genre всех элементов в коллекции

**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 MusicBand {

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

private Long numberOfParticipants; //Поле может быть null, Значение поля должно быть больше 0

private long albumsCount; //Значение поля должно быть больше 0

private String description; //Поле не может быть null

private MusicGenre genre; //Поле может быть null

private Person frontMan; //Поле может быть null

}

public class Coordinates {

private Float x; //Максимальное значение поля: 146, Поле не может быть null

private int y;

}

public class Person {

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

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

private Color eyeColor; //Поле может быть null

private Color hairColor; //Поле не может быть null

private Country nationality; //Поле не может быть null

private Location location; //Поле может быть null

}

public class Location {

private Float x; //Поле не может быть null

private int y;

private Long z; //Поле не может быть null

}

public enum MusicGenre {

JAZZ,

POST\_ROCK,

PUNK\_ROCK;

}

public enum Color {

RED,

WHITE,

BROWN;

}

public enum Color {

RED,

BLACK,

BLUE,

ORANGE,

WHITE;

}

public enum Country {

GERMANY,

INDIA,

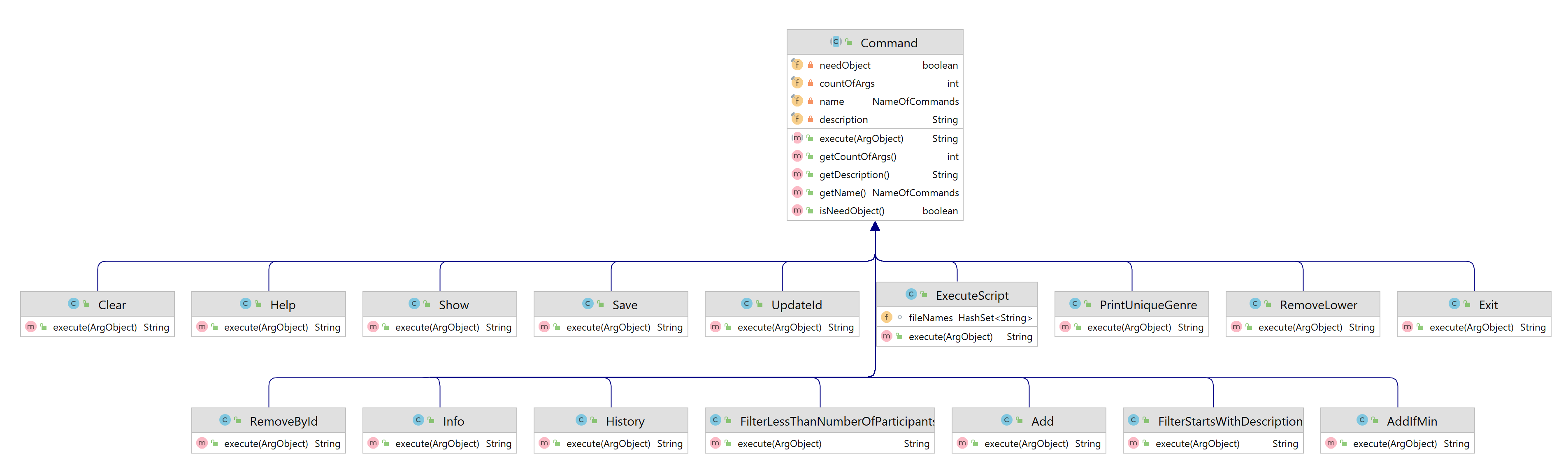
VATICAN;

}

**Отчёт по работе должен содержать:**

1. Текст задания.
2. Диаграмма классов разработанной программы.
3. Исходный код программы.
4. Выводы по работе.

**Задание 2: Диаграмма классов реализованной объектной модели**



Изображение выглядит как стол

Автоматически созданное описание

**Задание 3. Исходный код программы:**

<https://gitlab.com/itmo-programming/p3131/student-8>

**Задание 4. Вывод:**

В ходе выполнения лабораторной работы было реализовано консольное приложение, которое реализует управление коллекцией в интерактивном режиме. Мною были изучены коллекции и сортировка элементов коллекций в Java. Также изучены потоки вводы-вывода, байтовые и символьные потоки, работа в файлами, пакет java.nio, а также создание автоматический документации Javadoc.