Федеральное государственное автономное образовательное учреждение

высшего образования «Национальный исследовательский университет

ИТМО»

*Факультет программной инженерии и компьютерной техники*

*Направление подготовки: 09.03.01 - Информатика и вычислительная техника, Компьютерные системы и технологии*

*Дисциплина «Программирование»*

**Отчет**

**По лабораторной работе №3**

**Вариант №** **6431**

Выполнил:

Муравенко Григорий Павлович

Группа: Р3132

Преподаватель:

Абузов Ярослав Александрович

Г. Санкт-Петербург, 2024 г.

**Оглавление**

[Задание 3](#_Toc160256920)

[UML-диаграмма классов: 5](#_Toc160256921)

[Исходный код программы 6](#_Toc160256922)

[Вывод 7](#_Toc160256923)

[Список литературы. 8](#_Toc160256924)

# Задание

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса Flat, 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.TreeSet
* 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 json
* Чтение данных из файла необходимо реализовать с помощью класса java.util.Scanner
* Запись данных в файл необходимо реализовать с помощью класса java.io.BufferedWriter
* 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 13 команд (без их аргументов)
* max\_by\_area : вывести любой объект из коллекции, значение поля area которого является максимальным
* filter\_less\_than\_house house : вывести элементы, значение поля house которых меньше заданного
* print\_field\_descending\_house : вывести значения поля house 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 Flat {

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 double area; //Максимальное значение поля: 570, Значение поля должно быть больше 0

private Integer numberOfRooms; //Значение поля должно быть больше 0

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

private Furnish furnish; //Поле может быть null

private Transport transport; //Поле не может быть null

private House house; //Поле может быть null

}

public class Coordinates {

private Integer x; //Максимальное значение поля: 591, Поле не может быть null

private long y;

}

public class House {

private String name; //Поле не может быть null

private Long year; //Значение поля должно быть больше 0

private int numberOfFlatsOnFloor; //Значение поля должно быть больше 0

}

public enum Furnish {

DESIGNER,

FINE,

BAD;

}

public enum Transport {

NONE,

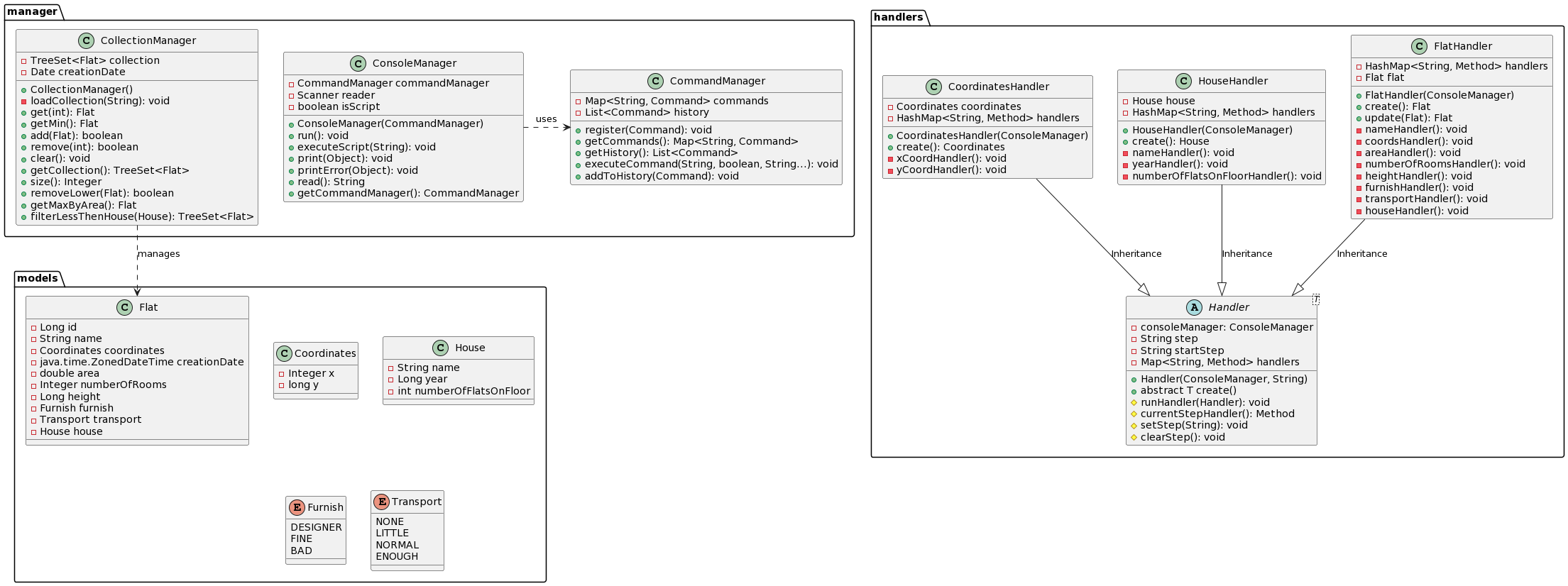
LITTLE,

NORMAL,

ENOUGH;

}

# UML-диаграмма классов:



# Исходный код программы

https://github.com/Deformater/itmo\_java\_lab\_5

# 

# Вывод

В ходе работы над лабораторной работой я научился работать со структурами в Java и файлами, попробовал проектировать архитектуру проекта. Углубил свои знания в ООП