Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет Программной инженерии и компьютерной техники

Дисциплина “Программирование”

Отчёт по лабораторной работе №5

Вариант №4003

Выполнил:

Ануфриев Андрей Сергеевич, Р3119

Проверил:

Шибаев Семен Сергеевич

г. Санкт-Петербург

2025 год

**Оглавление**

[Задание 3](#_Toc185945594)

[Диаграмма классов 5](#_Toc185945595)

[Исходный код программы 5](#_Toc185945596)

[Результат работы программы. 5](#_Toc185945597)

[Вывод 7](#_Toc185945598)

[Ответы на вопросы 8](#_Toc185945599)

# Задание

Реализовать консольное приложение, которое реализует управление коллекцией объектов в интерактивном режиме. В коллекции необходимо хранить объекты класса Ticket, 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.LinkedList
* 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 xml
* Чтение данных из файла необходимо реализовать с помощью класса java.io.FileReader
* Запись данных в файл необходимо реализовать с помощью класса java.io.FileWriter
* 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\_first : удалить первый элемент из коллекции
* add\_if\_max {element} : добавить новый элемент в коллекцию, если его значение превышает значение наибольшего элемента этой коллекции
* remove\_lower {element} : удалить из коллекции все элементы, меньшие, чем заданный
* remove\_all\_by\_event event : удалить из коллекции все элементы, значение поля event которого эквивалентно заданному
* remove\_any\_by\_event event : удалить из коллекции один элемент, значение поля event которого эквивалентно заданному
* print\_unique\_price : вывести уникальные значения поля price 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 Ticket {

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

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

private String comment; //Поле может быть null

private TicketType type; //Поле может быть null

private Event event; //Поле может быть null

}

public class Coordinates {

private Double x; //Поле не может быть null

private float y; //Значение поля должно быть больше -121

}

public class Event {

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 java.util.Date date; //Поле не может быть null

private EventType eventType; //Поле может быть null

}

public enum TicketType {

VIP,

USUAL,

BUDGETARY,

CHEAP;

}

public enum EventType {

E\_SPORTS,

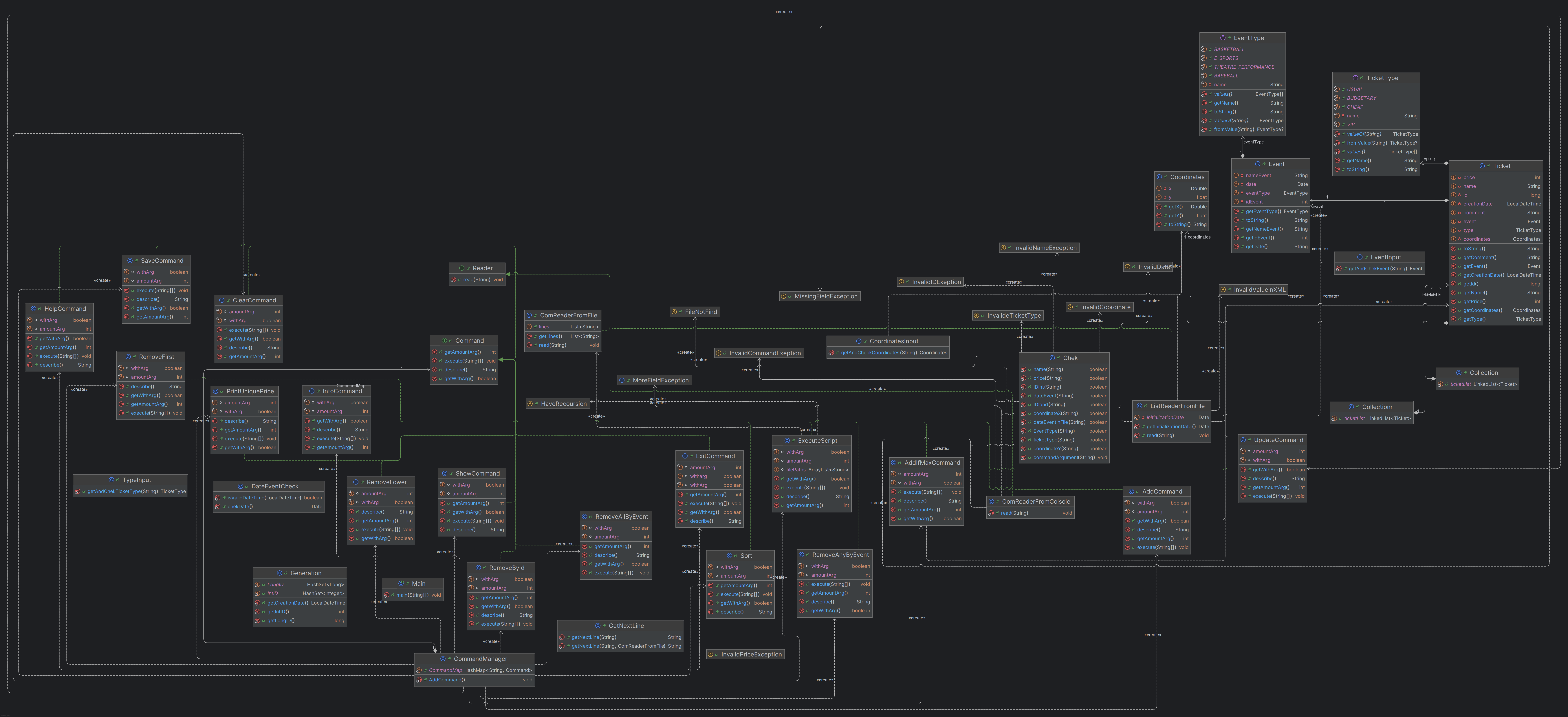
BASEBALL,

BASKETBALL,

THEATRE\_PERFORMANCE;

}

# Диаграмма классов



# Исходный код программы

# 

# Вывод

Я научился проектировать и реализовывать классы. Узнал и применил основные принципы ООП в Java.