МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Факультет информационных технологий и робототехники

Кафедра программного обеспечения информационных систем и технологий

**Отчет**

**по лабораторной работе № 1**

**«**Объектно-Ориентированное Программирование. Абстракция**»**

по дисциплине: «КПО»

Выполнили: студенты гр. 10701121

Журович К.С.

Гвоздь В.И.

Асанов Н.К.

Принял: преподаватель Станкевич С.

Минск 2023

**Цель работы:**

Научиться грамотно анализировать предметную область и с помощью абстракции выделять существенные детали, на базе которых в дальнейшем проектируются классы и объекты будущей программной системы согласно методологии ООП, а также практически закрепить данные навыки при решении соответствующих задач (бизнес проблем).

**Постановка задачи**

Необходимо решить задачу с использованием методологии ООП. Для чего необходимо подобрать самостоятельно соответствующую проблемную (предметную/доменную) область, которая базируется на объектах и событиях реального мира (примеры соответствующих предметных областей приведены ниже). Спроектировать классы (собственные пользовательские типы данных) в языке Java для программного представления данных объектов и основной логики будущей программной системы. Система должна решать, как минимум, два полезных действия и иметь дополнительно следующие опции: − не менее 3 разнообразных классов предметной области; − не менее 5 атрибутов (состояния) и методов (поведения) в классе-сущности; − не менее 3 методов, которые реализуют бизнес-логику программы, в соответствующих функциональных классах; − хранить глобальные характеристики системы или характеристики уровня отдельных классов. На базе спроектированной программной системы реализовать программу и продемонстрировать её работоспособность

**Задание**

**Вариант 10**

Игровая комната (Game room). Есть игровая комната с соответствующими игрушками. Необходимо провести подсчёт стоимости всех игрушек, их общий вес и найти самую дорогую игрушку

**1. Совместная работа с использованием службы размещения репозиториев GitHub**

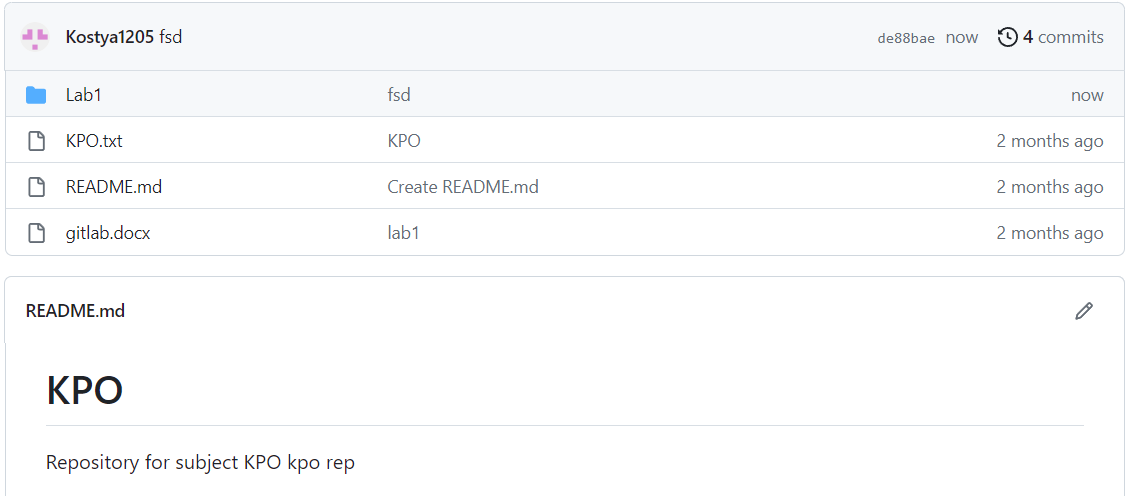


Рис. 1 – Файлы проекты размещены в репозитории GitHub

**2.1. Разработка предметной области**

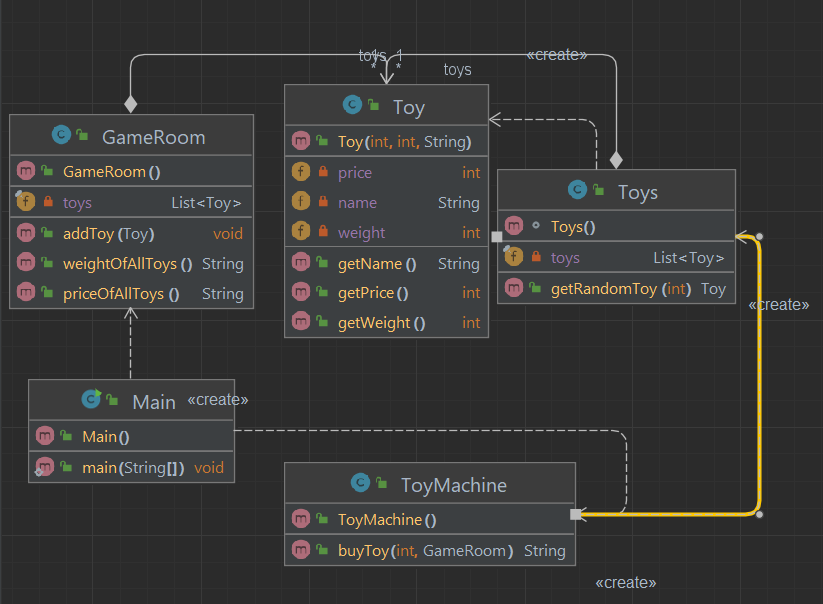


Рис. 2 – Диаграмма классов, созданная в среде Visual Studio

**2.2. Листинг программы**

Класс GameRoom

public class GameRoom {  
 private final List<Toy> toys;  
  
 public GameRoom(){  
 toys = new ArrayList<>();  
 }  
  
 public void addToy(Toy toy){  
 toys.add(toy);  
 }  
 public String priceOfAllToys(){  
 int sum = 0;  
 for (Toy toy:toys) {  
 sum+=toy.getPrice();  
 }  
 return "Цена всех игрушек в комнате равна "+sum;  
 }  
  
 public String weightOfAllToys(){  
 int sum = 0;  
 for (Toy toy:toys) {  
 sum+=toy.getWeight();  
 }  
 return "Вес всех игрушек в комнате равен "+sum;  
 }  
}

Класс Toy

public class Toy {  
 private int price;  
 private int weight;  
 private String name;  
  
 public Toy(int price, int weight,String name) {  
 this.price = price;  
 this.weight = weight;  
 this.name = name;  
 }  
  
 public int getPrice() {  
 return price;  
 }  
  
 public int getWeight() {  
 return weight;  
 }  
  
 public String getName() {  
 return name;  
 }  
}

Класс ToyMachin

public class ToyMachine {  
 public String buyToy(int price,GameRoom gameRoom){  
 Toys toys = new Toys();  
 Toy toy = toys.getRandomToy(price);  
 if(toy!=null) {  
 gameRoom.addToy(toy);  
 return "Игрушка "+toy.getName()+" добалена и игровую комнату";  
 }else return "Нет игрушки с такой ценой ("+price+")";  
 }  
}

Класс Toys

public class Toys {  
 private final List<Toy> toys = new ArrayList<>();  
 Toys(){  
 toys.add(new Toy(2,200,"Стикер"));  
 toys.add(new Toy(1,100,"Брелок"));  
 toys.add(new Toy(2,300,"Матрёшка"));  
 toys.add(new Toy(1,400,"Автомобиль"));  
 toys.add(new Toy(2,400,"Поезд"));  
 toys.add(new Toy(1,200,"Солдатик"));  
 }  
 public Toy getRandomToy(int price){  
 for (int i = 0;i<toys.size()\*2;i++) {  
 Random random = new Random();  
 Toy toy =toys.get(random.nextInt(toys.size()));  
 if(toy.getPrice()==price)  
 return toy;  
 }  
 return null;  
 }  
}

Класс Main

public class Main {  
 public static void main(String[] args) {  
 GameRoom gameRoom = new GameRoom();  
 ToyMachine toyMachine = new ToyMachine();  
 System.*out*.println(toyMachine.buyToy(1,gameRoom));  
 System.*out*.println(toyMachine.buyToy(2,gameRoom));  
 System.*out*.println(toyMachine.buyToy(1,gameRoom));  
 System.*out*.println(toyMachine.buyToy(2,gameRoom));  
  
 System.*out*.println(gameRoom.priceOfAllToys());  
 System.*out*.println(gameRoom.weightOfAllToys());  
 }  
}

**3. Выполнение программы**

В программе реализован интерфейс IWagon с двумя дочерними классами PassengerWagon и CargoWagon, интерфейс Ilocomotive с дочерним классом DieselLocomotive, класс Train, где реализована бизнес логика проекта. Также реализован класс Program, созданный для демонстрации функционала проекта. При запуске программы создается поезд, вычисляется общая длина состава, выводится информация о самом длинном и самом коротком вагонах, а так же общий вес состава и рассчитывается его максимальная скорость.

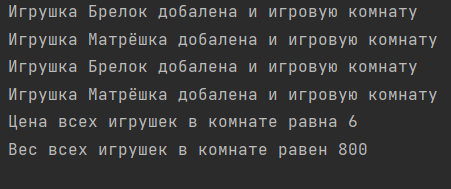


Рис. 3 – Результат выполнения метода Main

**4. Выводы**

Мы научились анализировать предметную область и с помощью абстракции выделять существенные детали, на базе которых в дальнейшем проектируются классы и объекты будущей программной системы согласно методологии ООП, а также практически закрепить данные навыки при решении соответствующих задач (бизнес проблем). В данной лабораторной работе предметной областью выступал Игровая Комната, а бизнес функции заключались в поиске, расчёте общей стоимости игрушек и веса всех игрушек,а также довавление типичных игрушек в комнату.