

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ
КАФЕДРА ИИТ

Отчет по лабораторной работе №3

Выполнил:
Студент 4 курса
Группы АС-50
Куц Д.А.
Проверил:
Крощенко А.А.

Брест 2020

Цель работы: научиться создавать и использовать классы в программах на языке программирования Java.

Задание 1. Реализовать простой класс.

9) Множество вещественных чисел переменной мощности – Предусмотреть возможность пересечения двух множеств, вывода на печать элементов множества, а так же метод, определяющий, принадлежит ли указанное значение множеству. Класс должен содержать методы, позволяющие добавлять и удалять элемент в/из множества. Конструктор должен позволить создавать объекты с начальной инициализацией. Реализацию множества осуществить на базе структуры ArrayList. Реализовать метод equals, выполняющий сравнение объектов данного типа.

Код:

```
import java.util.ArrayList;

class Set {

    private final int power = 0;
    private final ArrayList<Double> numbers;

    public Set() { numbers = new ArrayList<>(); }

    public int getPower() { return power; }

    public boolean existCheck(double element) { return
numbers.contains(element); }

    public void add(double element) { numbers.add(element); }

    public void remove(double element) { numbers.remove(element); }

    public void removeAll() { numbers.removeAll(numbers); }

    public ArrayList intersection(Set set) {
        ArrayList<Double> crossedNumbers = new ArrayList<>();
        for (double element: numbers) if(set.numbers.contains(element))
crossedNumbers.add(element);
        return crossedNumbers;
    }

    public void print() {
        for (double element: numbers) System.out.print(element + "; ");
    }

    public String toString() {
        StringBuilder str = new StringBuilder();
        for (double element: numbers) {
            str.append(element);
            str.append("; ");
        }
        return str.toString();
    }

    public boolean equals(Set set) {
        if(set.getPower() != power) return false;
        for (double element: numbers) if (!set.existCheck(element)) return
false;
        return true;
    }
}
```

```

}

public class task1 {

    public static void main(String[] args) {
        Set set1 = new Set();
        for(int i = 0; i < 5 ; i++) {
            double random = Math.random()*100;
            set1.add(random);
        }
        Set set2 = new Set();
        for(int i = 0; i < 5 ; i++) {
            double random = Math.random()*100;
            set2.add(random);
        }

        System.out.println("Set 1 to string: " + set1.toString());
        System.out.println("Set 2 to string: " + set2.toString());

        set1.add(55.55);
        set2.add(55.55);
        System.out.println("\nElement 55.55 added to Set 1 and Set 2.");
        System.out.print("Set 1 after adding: ");
        set1.print();
        System.out.print("\nSet 2 after adding: ");
        set2.print();
        ArrayList<Double> crossedNumbers = set2.intersection(set1);
        System.out.print("\nIntersection check. Result of intersection: ");
        for (double element: crossedNumbers) System.out.print(element + "
");

        set1.remove(55.55);
        System.out.print("\n\nElement 55.55 removed.\nSet 1 after removal:
");

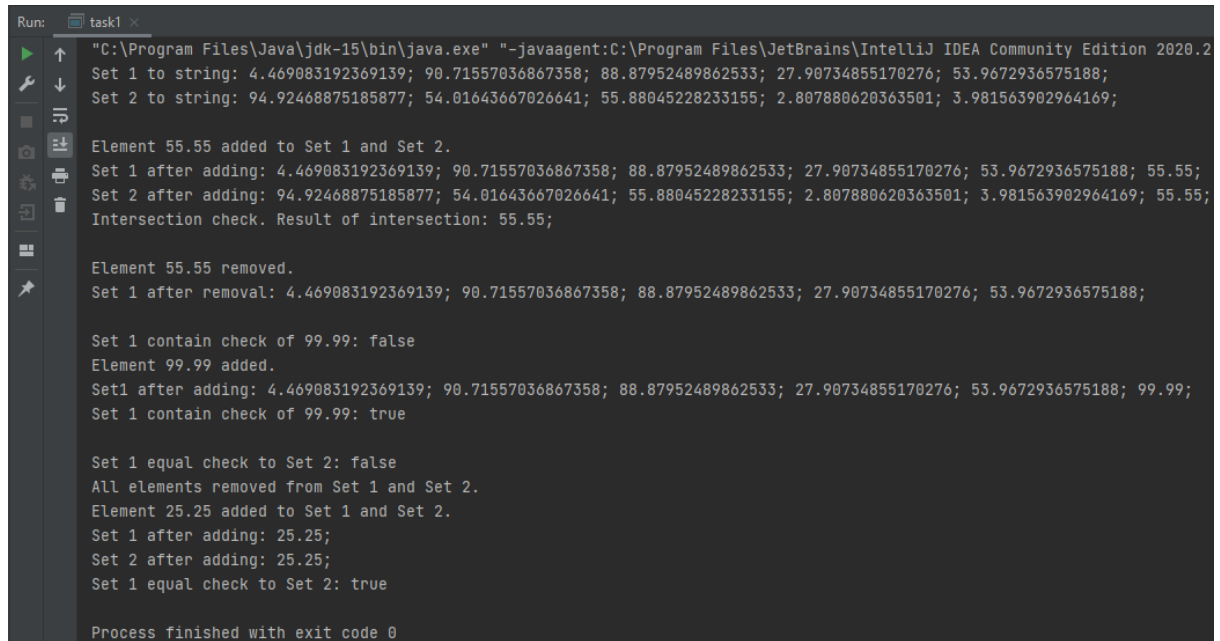
        set1.print();

        System.out.println("\n\nSet 1 contain check of 99.99: " +
set1.existCheck(99.99));
        set1.add(99.99);
        System.out.print("Element 99.99 added.\nSet1 after adding: ");
        set1.print();
        System.out.println("\nSet 1 contain check of 99.99: " +
set1.existCheck(99.99));

        System.out.println("\nSet 1 equal check to Set 2: " +
set1.equals(set2));
        set1.removeAll();
        set2.removeAll();
        System.out.println("All elements removed from Set 1 and Set 2.");
        set1.add(25.25);
        set2.add(25.25);
        System.out.println("Element 25.25 added to Set 1 and Set 2.");
        System.out.print("Set 1 after adding: ");
        set1.print();
        System.out.print("\nSet 2 after adding: ");
        set2.print();
        System.out.println("\nSet 1 equal check to Set 2: " +
set1.equals(set2));
    }
}

```

Результат:



```
Run: task1 x
"C:\Program Files\Java\jdk-15\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.2
Set 1 to string: 4.469083192369139; 90.71557036867358; 88.87952489862533; 27.90734855170276; 53.9672936575188;
Set 2 to string: 94.92468875185877; 54.01643667026641; 55.88045228233155; 2.807880620363501; 3.981563902964169;

Element 55.55 added to Set 1 and Set 2.
Set 1 after adding: 4.469083192369139; 90.71557036867358; 88.87952489862533; 27.90734855170276; 53.9672936575188; 55.55;
Set 2 after adding: 94.92468875185877; 54.01643667026641; 55.88045228233155; 2.807880620363501; 3.981563902964169; 55.55;
Intersection check. Result of intersection: 55.55;

Element 55.55 removed.
Set 1 after removal: 4.469083192369139; 90.71557036867358; 88.87952489862533; 27.90734855170276; 53.9672936575188;

Set 1 contain check of 99.99: false
Element 99.99 added.
Set1 after adding: 4.469083192369139; 90.71557036867358; 88.87952489862533; 27.90734855170276; 53.9672936575188; 99.99;
Set 1 contain check of 99.99: true

Set 1 equal check to Set 2: false
All elements removed from Set 1 and Set 2.
Element 25.25 added to Set 1 and Set 2.
Set 1 after adding: 25.25;
Set 2 after adding: 25.25;
Set 1 equal check to Set 2: true

Process finished with exit code 0
```

Задание 2. Разработать автоматизированную систему на основе некоторой структуры данных, манипулирующей объектами пользовательского класса. Реализовать требуемые функции обработки данных.

9) Автоматизированная система склада

Написать программу для моделирования автоматизированного склада. На складе хранится различная продукция (Product) Каждая продукция характеризуется следующей информацией:

- id;
- Наименование;
- UPC (штрих-код);
- Производитель;
- Цена;
- Срок хранения;
- Количество.

Программа должна иметь следующий функционал:

- Генерация списка продукции на складе;
- Предоставлять список товаров для заданного наименования;
- Предоставлять список товаров для заданного наименования, цена которых не превосходит заданную;
- Предоставлять список товаров, срок хранения которых истек.

Код:

```
import java.util.ArrayList;
import java.util.Scanner;

class Product {

    private String name, ups, manufacturer;
    private int id, price, shelfLife, amount;

    public int getId() { return id; }

    public void setId(int id) { this.id = id; }

    public String getName() { return name; }

    public void setName(String name) { this.name = name; }

    public String getUps() { return ups; }

    public void setUps(String ups) { this.ups = ups; }

    public String getManufacturer() { return manufacturer; }

    public void setManufacturer(String manufacturer) { this.manufacturer =
manufacturer; }

    public int getPrice() { return price; }

    public void setPrice(int price) { this.price = price; }

    public int getShelfLife() { return shelfLife; }

    public void setShelfLife(int shelfLife) { this.shelfLife = shelfLife; }

    public int getAmount() { return amount; }

    public void setAmount(int amount) { this.amount = amount; }

}

public class task2 {

    public static void main(String[] args) {

        ArrayList<String> nameList = new ArrayList<>();
        nameList.add("Note 9 Pro");
        nameList.add("V20 SE");
        nameList.add("Galaxy S10");
        nameList.add("30 Premium");
        nameList.add("iPhone 12 Pro Max");

        ArrayList<String> upsList = new ArrayList<>();
        upsList.add("1245443220981");
        upsList.add("1748642140582");
        upsList.add("1954551458684");
        upsList.add("1116863922921");
        upsList.add("1666148228101");

        ArrayList<String> manufacturerList = new ArrayList<>();
        manufacturerList.add("Redmi");
        manufacturerList.add("VIVO");
        manufacturerList.add("Samsung");
        manufacturerList.add("HONOR");
        manufacturerList.add("Apple");
```

```
ArrayList<Integer> priceList = new ArrayList<>();
priceList.add(749);
priceList.add(729);
priceList.add(2449);
priceList.add(1499);
priceList.add(5399);

ArrayList<Integer> shellLifeList = new ArrayList<>();
shellLifeList.add(6);
shellLifeList.add(26);
shellLifeList.add(10);
shellLifeList.add(15);
shellLifeList.add(3);

ArrayList<Integer> amountList = new ArrayList<>();
amountList.add(250);
amountList.add(189);
amountList.add(121);
amountList.add(154);
amountList.add(24);

Product product1 = new Product(),
product2 = new Product(),
product3 = new Product(),
product4 = new Product(),
product5 = new Product();

product1.setId(1);
product1.setName(nameList.get(0));
product1.setUps(upsList.get(0));
product1.setManufacturer(manufacturerList.get(0));
product1.setPrice(priceList.get(0));
product1.setShelfLife(shellLifeList.get(0));
product1.setAmount(amountList.get(0));

product2.setId(2);
product2.setName(nameList.get(1));
product2.setUps(upsList.get(1));
product2.setManufacturer(manufacturerList.get(1));
product2.setPrice(priceList.get(1));
product2.setShelfLife(shellLifeList.get(1));
product2.setAmount(amountList.get(1));

product3.setId(3);
product3.setName(nameList.get(2));
product3.setUps(upsList.get(2));
product3.setManufacturer(manufacturerList.get(2));
product3.setPrice(priceList.get(2));
product3.setShelfLife(shellLifeList.get(2));
product3.setAmount(amountList.get(2));

product4.setId(4);
product4.setName(nameList.get(3));
product4.setUps(upsList.get(3));
product4.setManufacturer(manufacturerList.get(3));
product4.setPrice(priceList.get(3));
product4.setShelfLife(shellLifeList.get(3));
product4.setAmount(amountList.get(3));

product5.setId(5);
product5.setName(nameList.get(4));
product5.setUps(upsList.get(4));
product5.setManufacturer(manufacturerList.get(4));
```

```

        product5.setPrice(priceList.get(4));
        product5.setShelfLife(shelfLifeList.get(4));
        product5.setAmount(amountList.get(4));

        System.out.println("\nСмартСклад v0.1");
        System.out.println("1 - список товаров");
        System.out.println("2 - список товаров для заданного наименования");
        System.out.println("3 - список товаров для заданного наименования с
        фильтром цены");
        System.out.println("4 - список товаров с истекшим сроком хранения");
        System.out.println("0 - выход из программы");

        Scanner numScanner = new Scanner(System.in);
        System.out.print("\nВведите, чтобы продолжить: ");
        int num = numScanner.nextInt();

        switch (num) {
            case 1: {
                System.out.println("\nID: " + product1.getId() + "\nНазвание: " + product1.getName() + "\nUPS: " + product1.getUps() + "\nПроизводитель: " + product1.getManufacturer() + "\nЦена: " + product1.getPrice() + " руб\nСрок хранения: " + product1.getShelfLife() + " месяцев(-а)\nКоличество: " + product1.getAmount() + " шт");
                System.out.println("\nID: " + product2.getId() + "\nНазвание: " + product2.getName() + "\nUPS: " + product2.getUps() + "\nПроизводитель: " + product2.getManufacturer() + "\nЦена: " + product2.getPrice() + " руб\nСрок хранения: " + product2.getShelfLife() + " месяцев(-а)\nКоличество: " + product2.getAmount() + " шт");
                System.out.println("\nID: " + product3.getId() + "\nНазвание: " + product3.getName() + "\nUPS: " + product3.getUps() + "\nПроизводитель: " + product3.getManufacturer() + "\nЦена: " + product3.getPrice() + " руб\nСрок хранения: " + product3.getShelfLife() + " месяцев(-а)\nКоличество: " + product3.getAmount() + " шт");
                System.out.println("\nID: " + product4.getId() + "\nНазвание: " + product4.getName() + "\nUPS: " + product4.getUps() + "\nПроизводитель: " + product4.getManufacturer() + "\nЦена: " + product4.getPrice() + " руб\nСрок хранения: " + product4.getShelfLife() + " месяцев(-а)\nКоличество: " + product4.getAmount() + " шт");
                System.out.println("\nID: " + product5.getId() + "\nНазвание: " + product5.getName() + "\nUPS: " + product5.getUps() + "\nПроизводитель: " + product5.getManufacturer() + "\nЦена: " + product5.getPrice() + " руб\nСрок хранения: " + product5.getShelfLife() + " месяцев(-а)\nКоличество: " + product5.getAmount() + " шт");
                break;
            }
            case 2: {
                Scanner nameScanner = new Scanner(System.in);
                System.out.print("Введите название модели: ");
                String nameSearch = nameScanner.nextLine();

                if (product1.getName().equals(nameSearch))
                    System.out.println("\nID: " + product1.getId() + "\nНазвание: " + product1.getName() + "\nUPS: " + product1.getUps() + "\nПроизводитель: " +

```

```

product1.getManufacturer() + "\nЦена: "
                                + product1.getPrice() + " руб\nСрок хранения: " +
product1.getShelfLife() + " месяцев(-а)\nКоличество: "
                                + product1.getAmount() + " шт");
        else if (product2.getName().equals(nameSearch))
            System.out.println("\nID: " + product2.getId() +
"\nНазвание: " + product2.getName() + "\nUPS: "
                                + product2.getUps() + "\nПроизводитель: " +
product2.getManufacturer() + "\nЦена: "
                                + product2.getPrice() + " руб\nСрок хранения: " +
product2.getShelfLife() + " месяцев(-а)\nКоличество: "
                                + product2.getAmount() + " шт");
        else if (product3.getName().equals(nameSearch))
            System.out.println("\nID: " + product3.getId() +
"\nНазвание: " + product3.getName() + "\nUPS: "
                                + product3.getUps() + "\nПроизводитель: " +
product3.getManufacturer() + "\nЦена: "
                                + product3.getPrice() + " руб\nСрок хранения: " +
product3.getShelfLife() + " месяцев(-а)\nКоличество: "
                                + product3.getAmount() + " шт");
        else if (product4.getName().equals(nameSearch))
            System.out.println("\nID: " + product4.getId() +
"\nНазвание: " + product4.getName() + "\nUPS: "
                                + product4.getUps() + "\nПроизводитель: " +
product4.getManufacturer() + " год(-а)\nЦена: "
                                + product4.getPrice() + " руб\nСрок хранения: " +
product4.getShelfLife() + " месяцев(-а)\nКоличество: "
                                + product4.getAmount() + " шт");
        else if (product5.getName().equals(nameSearch))
            System.out.println("\nID: " + product5.getId() +
"\nНазвание: " + product5.getName() + "\nUPS: "
                                + product5.getUps() + "\nПроизводитель: " +
product5.getManufacturer() + "\nЦена: "
                                + product5.getPrice() + " руб\nСрок хранения: " +
product5.getShelfLife() + " месяцев(-а)\nКоличество: "
                                + product5.getAmount() + " шт");
        else System.out.println("Товары не найдены");
        break;
    }
    case 3: {
        Scanner nameScanner = new Scanner(System.in);
        System.out.print("Введите название модели: ");
        String nameSearch = nameScanner.nextLine();
        Scanner priceScanner = new Scanner(System.in);
        System.out.print("Введите цену (верхний порог): ");
        int priceSearch = priceScanner.nextInt();

        if (product1.getName().equals(nameSearch) &&
product1.getPrice() <= priceSearch)
            System.out.println("\nID: " + product1.getId() +
"\nНазвание: " + product1.getName() + "\nUPS: "
                                + product1.getUps() + "\nПроизводитель: " +
product1.getManufacturer() + "\nЦена: "
                                + product1.getPrice() + "\nСрок хранения: " +
product1.getShelfLife() + " месяцев(-а)\nКоличество: "
                                + product1.getAmount() + " шт");
        else if (product2.getName().equals(nameSearch) &&
product2.getPrice() <= priceSearch)
            System.out.println("\nID: " + product2.getId() +
"\nНазвание: " + product2.getName() + "\nUPS: "
                                + product2.getUps() + "\nПроизводитель: " +
product2.getManufacturer() + "\nЦена: "
                                + product2.getPrice() + "\nСрок хранения: " +
product2.getShelfLife() + " месяцев(-а)\nКоличество: "

```



```

        + product2.getAmount() + " шт");
    else if (product3.getName().equals(nameSearch) &&
product3.getPrice() <= priceSearch)
        System.out.println("\nID: " + product3.getId() +
"\nНазвание: " + product3.getName() + "\nUPS: "
        + product3.getUps() + "\nПроизводитель: " +
product3.getManufacturer() + "\nЦена: "
        + product3.getPrice() + "\nСрок хранения: " +
product3.getShelfLife() + " месяцев (-a) \nКоличество: "
        + product3.getAmount() + " шт");
    else if (product4.getName().equals(nameSearch) &&
product4.getPrice() <= priceSearch)
        System.out.println("\nID: " + product4.getId() +
"\nНазвание: " + product4.getName() + "\nUPS: "
        + product4.getUps() + "\nПроизводитель: " +
product4.getManufacturer() + "\nЦена: "
        + product4.getPrice() + "\nСрок хранения: " +
product4.getShelfLife() + " месяцев (-a) \nКоличество: "
        + product4.getAmount() + " шт");
    else if (product5.getName().equals(nameSearch) &&
product5.getPrice() <= priceSearch)
        System.out.println("\nID: " + product5.getId() +
"\nНазвание: " + product5.getName() + "\nUPS: "
        + product5.getUps() + "\nПроизводитель: " +
product5.getManufacturer() + "\nЦена: "
        + product5.getPrice() + "\nСрок хранения: " +
product5.getShelfLife() + " месяцев (-a) \nКоличество: "
        + product5.getAmount() + " шт");
    else System.out.println("Товары не найдены");
    break;
}
case 4: {
    if (product1.getShelfLife() > 25)
        System.out.println("\nID: " + product1.getId() +
"\nНазвание: " + product1.getName() + "\nUPS: "
        + product1.getUps() + "\nПроизводитель: " +
product1.getManufacturer() + "\nЦена: "
        + product1.getPrice() + "\nСрок хранения: " +
product1.getShelfLife() + " месяцев (-a) \nКоличество: "
        + product1.getAmount() + " шт");
    if (product2.getShelfLife() > 25)
        System.out.println("\nID: " + product2.getId() +
"\nНазвание: " + product2.getName() + "\nUPS: "
        + product2.getUps() + "\nПроизводитель: " +
product2.getManufacturer() + "\nЦена: "
        + product2.getPrice() + "\nСрок хранения: " +
product2.getShelfLife() + " месяцев (-a) \nКоличество: "
        + product2.getAmount() + " шт");
    if (product3.getShelfLife() > 25)
        System.out.println("\nID: " + product3.getId() +
"\nНазвание: " + product3.getName() + "\nUPS: "
        + product3.getUps() + "\nПроизводитель: " +
product3.getManufacturer() + "\nЦена: "
        + product3.getPrice() + "\nСрок хранения: " +
product3.getShelfLife() + " месяцев (-a) \nКоличество: "
        + product3.getAmount() + " шт");
    if (product4.getShelfLife() > 25)
        System.out.println("\nID: " + product4.getId() +
"\nНазвание: " + product4.getName() + "\nUPS: "
        + product4.getUps() + "\nПроизводитель: " +
product4.getManufacturer() + "\nЦена: "
        + product4.getPrice() + "\nСрок хранения: " +
product4.getShelfLife() + " месяцев (-a) \nКоличество: "
        + product4.getAmount() + " шт");
}
}

```

```

        if (product5.getShelfLife() > 25)
            System.out.println("\nID: " + product5.getId() +
"\nНазвание: " + product5.getName() + "\nUPS: "
+ product5.getUps() + "\nПроизводитель: " +
product5.getManufacturer() + "\nЦена: "
+ product5.getPrice() + "\nСрок хранения: " +
product5.getShelfLife() + " месяцев(-а)\nКоличество: "
+ product5.getAmount() + " шт");

            break;
        }
        case 0: break;
        default: System.out.println("Ошибка. Неверное значение");
    }
}
}

```

Результат:

- Генерация списка продукции на складе:

```

СмартСклад v0.1
1 - список товаров
2 - список товаров для заданного наименования
3 - список товаров для заданного наименования с фильтром цены
4 - список товаров с истекшим сроком хранения
0 - выход из программы

Введите, чтобы продолжить: 1

ID: 1
Название: Note 9 Pro
UPS: 1245443220981
Производитель: Redmi
Цена: 749 руб
Срок хранения: 6 месяцев(-а)
Количество: 250 шт

ID: 2
Название: V20 SE
UPS: 1748642140582
Производитель: VIVO
Цена: 729 руб
Срок хранения: 26 месяцев(-а)
Количество: 189 шт

ID: 3
Название: Galaxy S10
UPS: 1954551458684
Производитель: Samsung
Цена: 2449 руб
Срок хранения: 10 месяцев(-а)
Количество: 121 шт

ID: 4
Название: 30 Premium
UPS: 1116863922921
Производитель: HONOR
Цена: 1499 руб
Срок хранения: 15 месяцев(-а)
Количество: 154 шт

ID: 5
Название: iPhone 12 Pro Max
UPS: 1666148228101
Производитель: Apple
Цена: 5399 руб
Срок хранения: 3 месяцев(-а)
Количество: 24 шт

Process finished with exit code 0

```

- Предоставление списка товаров для заданного наименования:

```

Введите, чтобы продолжить: 2
Введите название модели: Note 9 Pro

ID: 1
Название: Note 9 Pro
UPS: 1245443220981
Производитель: Redmi
Цена: 749 руб
Срок хранения: 6 месяцев(-а)
Количество: 250 шт

Process finished with exit code 0

```

- Предоставление списка товаров для заданного наименования, цена которых не превосходит заданную;

```
Введите, чтобы продолжить: 3
Введите название модели: iPhone 12 Pro Max
Введите цену (верхний порог): 5500

ID: 5
Название: iPhone 12 Pro Max
UPS: 1666148228101
Производитель: Apple
Цена: 5399
Срок хранения: 3 месяцев(-а)
Количество: 24 шт

Process finished with exit code 0
```

- Предоставлять список товаров, срок хранения которых истек.

```
Введите, чтобы продолжить: 4

ID: 2
Название: V20 SE
UPS: 1748642140582
Производитель: VIVO
Цена: 729
Срок хранения: 26 месяцев(-а)
Количество: 189 шт

Process finished with exit code 0
```

Вывод: в ходе лабораторной работы научился создавать и использовать классы в программах на языке программирования Java.