

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ»

КАФЕДРА «ИИТ»

Лабораторная работа №3

Выполнил:

Студент 4 курса

Группа АС-50

Годухин Р.А

Проверил:

Крощенко.А.А

Брест 2020

## Вариант 2

**Задание 1:** Равносторонний треугольник, заданный длинами сторон. Предусмотреть возможность определения площади и периметра, а так же логический метод, определяющий существует ли такой треугольник. Конструктор должен позволить создавать объекты с начальной инициализацией. Реализовать метод equals, выполняющий сравнение объектов данного типа.

```
package com.company;
class Triangle
{
    double rebro;
    public Triangle()
    {
        this.rebro = 5;
    }
    public Triangle(double rebro)
    {
        this.rebro = rebro;
    }
    public void setRebro(double rebro)
    {
        this.rebro = rebro;
    }
    public double getRebro()
    {
        return this.rebro;
    }
    public double perimeter()
    {
        return rebro*3;
    }
    public double square()
    {
        return Math.pow(rebro, 2)*Math.sqrt(3) / 4;
    }
    public boolean checkExists()
    {
        if(rebro>0)
            return true;
        else
            return false;
    }
}
```

```

public boolean equals(Triangle triangle)
{
    return rebro == triangle.rebro;
}
public String toString()
{
    return "Triangle, rebro = "+rebro;
}
}
public class Main {
    public static void main(String[] args) {
        Triangle triangle = new Triangle();
        System.out.println(triangle.toString());
        Triangle triangle2 = new Triangle(7);
        System.out.println("perimeter "+triangle2.perimeter());
        System.out.println("square "+triangle2.square());
        System.out.println("Vtoroy treugolnik suchestvuet "+ triangle2.checkExists());
        System.out.println("vtoroy treugolnik raven pervomy "+triangle2.equals(triangle));
        Triangle triangle3 = new Triangle();
        triangle3.setRebro(7);
        System.out.println("vtoroy treugolnik raven tret'emu "+triangle2.equals(triangle3));
    }
}

```

```

Triangle, rebro = 5.0
perimeter 21.0
square 21.217622392718745
Vtoroy treugolnik suchestvuet true
vtoroy treugolnik raven pervomy false
vtoroy treugolnik raven tret'emu true

Process finished with exit code 0
|

```

## **Задание 2:** Автоматизированная система проката автомобилей

Составить программу, которая хранит и обрабатывает информацию о прокате автомобилей. О

каждом автомобиле (Car) содержится следующая информация:

- id;
- Марка;
- Модель;
- Год выпуска;
- Цвет;
- Цена;
- Регистрационный номер;
- Номер машины.
- ФИО лица, взявшего на прокат (при наличии);
- Номер паспорта лица-арендатора (при наличии).

Программа должна обеспечить вывод списков:

- автомобилей;
- автомобилей заданной марки;
- автомобилей заданной модели, которые эксплуатируются больше n лет;
- автомобилей заданного года выпуска, цена которых больше указанной;
- автомобилей, взятых на прокат;

- автомобилей, взятых на прокат с выводом личной информации об арендаторах.

```
cars.txt x
1 0 audi a6 2010 красный 250000 5467TM-1 45tyfd4
2 1 volkswagen polo 2018 серебристый 120000 3211AB-3 4eruvxcR Коровьев_Семен_Валерьевич AB2341243
3 2 lada vesta 2018 синий 80000 5622GV-2 2vi4fasd222 Гулькова_Анастасия_Дмитриевна AB4629489
4 3 volkswagen passat 2004 серый 56000 4612AM-4 23njdfs3 Авдосюк_Алексей_Алексеевич AB2956122
5 4 renault laguna 1998 черный 4500 3451FE-1 i2fedfn2360
```

```
package com.company;
```

```
import java.io.File;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.text.ParseException;
import java.util.ArrayList;
import java.util.InputMismatchException;
import java.util.Scanner;
```

```
class Car
{
    private int id;
    private String brand;
    private String model;
    private int year;
    private String color;
    private double cost;
    private String regNumber;
    private String carNumber;
    private String fio = null;
    private String passportNumber = null;

    public void print(boolean getHolderInfo){
        System.out.print("id: "+id+", brand: "+brand+", model: "+model+", year: "
            +year+", color: "+color+ ", cost: "+cost+", regNumber: "
            +regNumber+", carNumber: "+carNumber);
        if(getHolderInfo)
            System.out.println(", fio: "+fio+", passportNumber: "+passportNumber);
        else
            System.out.println();
    }
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getBrand() {
        return brand;
    }
    public void setBrand(String brand) {
        this.brand = brand;
    }
}
```

```
public String getModel() {  
    return model;  
}  
  
public void setModel(String model) {  
    this.model = model;  
}  
  
public int getYear() {  
    return year;  
}  
  
public void setYear(int year) {  
    this.year = year;  
}  
  
public String getColor() {  
    return color;  
}  
  
public void setColor(String color) {  
    this.color = color;  
}  
  
public double getCost() {  
    return cost;  
}  
  
public void setCost(double cost) {  
    this.cost = cost;  
}  
  
public String getRegNumber() {  
    return regNumber;  
}  
  
public void setRegNumber(String regNumber) {  
    this.regNumber = regNumber;  
}  
  
public String getCarNumber() {  
    return carNumber;  
}  
  
public void setCarNumber(String carNumber) {  
    this.carNumber = carNumber;  
}  
  
public String getFio() {  
    return fio;  
}  
  
public void setFio(String fio) {  
    this.fio = fio;  
}  
  
public String getPassportNumber() {  
    return passportNumber;  
}
```

```

    }
    public void setPassportNumber(String passportNumber) {
        this.passportNumber = passportNumber;
    }
}

```

```

class Controller

```

```

{
    private ArrayList<Car> cars = new ArrayList<Car>();

    public boolean loadFromFile(String path)
    {
        ArrayList<Car> tempList = new ArrayList<Car>();
        File file = new File(path);

        Scanner scanner = null;

        try {
            scanner = new Scanner(file, "utf-8");
        }
        catch(FileNotFoundException e){
            return false;
        }

        while(scanner.hasNextLine()) {
            String[] values = scanner.nextLine().split(" ");

            if (values.length < 8 || values.length > 10)
                return false;

            Car car = new Car();

            try {
                car.setId(Integer.parseInt(values[0]));
                car.setBrand(values[1]);
                car.setModel(values[2]);
                car.setYear(Integer.parseInt(values[3]));
                car.setColor(values[4]);
                car.setCost(Double.parseDouble(values[5]));
                car.setRegNumber(values[6]);
                car.setCarNumber(values[7]);

                if (values.length > 8)
                    car.setFio(values[8]);
                if (values.length > 9)
                    car.setPassportNumber(values[9]);

                tempList.add(car);
            }
            catch(Exception e) {
                return false;
            }
        }

        scanner.close();
    }
}

```

```

        cars = tempList;
        return true;
    }
    public void printCars()
    {
        for(int i=0; i<cars.size(); i++)
            cars.get(i).print(false);
    }
    public void printCarsByBrand(String brand)
    {
        for(int i=0; i<cars.size(); i++)
            if(cars.get(i).getBrand().equals(brand))
                cars.get(i).print(false);
    }
    public void printCarsByBrandWhereUsesGreaterThenNyears(String brand, int n)
    {
        for(int i=0; i<cars.size(); i++)
            if(cars.get(i).getBrand().equals(brand) && 2020-cars.get(i).getYear()>n)
                cars.get(i).print(false);
    }
    public void printCarsByYearWhereCostGreaterThen(int year, double cost)
    {
        for(int i=0; i<cars.size(); i++)
            if(cars.get(i).getYear() == year && cars.get(i).getCost() > cost)
                cars.get(i).print(false);
    }
    public void printCarsWhereHaveHolderWithoutHolderInfo()
    {
        for(int i=0; i<cars.size(); i++)
            if(cars.get(i).getFio() != null)
                cars.get(i).print(false);
    }
    public void printCarsWhereHaveHolder()
    {
        for(int i=0; i<cars.size(); i++)
            if(cars.get(i).getFio() != null)
                cars.get(i).print(true);
    }
}

```

```

public class Main {

    public static void main(String[] args) {
        Controller controller = new Controller();
        if(!controller.loadFromFile("cars.txt"))
        {
            System.out.println("Ошибка чтения данных из файла.");
            return;
        }
        boolean menuKey = true;
        int code;
        Scanner scanner = new Scanner(System.in);
        String brand;
        int n, year;
        double cost;
        while(menuKey)
        {

```



```

        for(int i=0; i<10; i++)
            System.out.println();
        System.out.println("1 - вывести все автомобили");
        System.out.println("2 - вывести все автомобили заданной марки");
        System.out.println("3 - вывести все автомобили заданной марки, которые эксплуатируются больше n лет");
        System.out.println("4 - вывести все автомобили заданного года выпуска, цена которых больше указанной");
        System.out.println("5 - вывести все автомобили, которые взяты на прокат");
        System.out.println("6 - вывести все автомобили, которые взяты на прокат, с выводом личной информации об арендаторах");
        System.out.println("7 - завершить работу");
        code = scanner.nextInt();
        scanner.nextLine();
        if(code==1)
            controller.printCars();
        else if(code==2)
        {
            System.out.print("Введите бренд:");
            brand = scanner.nextLine();
            controller.printCarsByBrand(brand);
        }
        else if(code==3)
        {
            System.out.print("Введите бренд:");
            brand = scanner.nextLine();
            System.out.print("Введите n:");
            n = scanner.nextInt();
            scanner.nextLine();
            controller.printCarsByBrandWhereUsesGreaterThenNyears(brand, n);
        }
        else if(code==4)
        {
            System.out.print("Введите год выпуска:");
            year = scanner.nextInt();
            scanner.nextLine();
            System.out.print("Введите цену:");
            cost = scanner.nextDouble();
            scanner.nextLine();
            controller.printCarsByYearWhereCostGreaterThen(year, cost);
        }
        else if(code == 5)
            controller.printCarsWhereHaveHolderWithoutHolderInfo();
        else if(code == 6)
            controller.printCarsWhereHaveHolder();
        else if(code == 7)
            menuKey= false;
        else
            System.out.println("Неверный код");

        System.out.println("Нажмите Enter для продолжения...");
        scanner.nextLine();
    }

    scanner.close();

}

```

```
}
```

- 1 - вывести все автомобили
- 2 - вывести все автомобили заданной марки
- 3 - вывести все автомобили заданной марки, которые эксплуатируются больше n лет
- 4 - вывести все автомобили заданного года выпуска, цена которых больше указанной
- 5 - вывести все автомобили, которые взяты на прокат
- 6 - вывести все автомобили, которые взяты на прокат, с выводом личной информации об арендаторах
- 7 - завершить работу

1

```
1
id: 0, brand: audi, model: a6, year: 2010, color: красный, cost: 250000.0, regNumber: 5467TM-1, carNumber: 45tyfd4
id: 1, brand: volkswagen, model: polo, year: 2018, color: серебристый, cost: 120000.0, regNumber: 3211AB-3, carNumber: 4eruvxcR
id: 2, brand: lada, model: vesta, year: 2018, color: синий, cost: 80000.0, regNumber: 5622GV-2, carNumber: 2vi4fasd222
id: 3, brand: volkswagen, model: passat, year: 2004, color: серый, cost: 56000.0, regNumber: 4612AM-4, carNumber: 23njdfs3
id: 4, brand: renault, model: laguna, year: 1998, color: черный, cost: 4500.0, regNumber: 3451FE-1, carNumber: i2fedfn2360
```

2

```
2
Введите бренд:audi
id: 0, brand: audi, model: a6, year: 2010, color: красный, cost: 250000.0, regNumber: 5467TM-1, carNumber: 45tyfd4
id: 1, brand: volkswagen, model: polo, year: 2018, color: серебристый, cost: 120000.0, regNumber: 3211AB-3, carNumber: 4eruvxcR
```

3

```
Введите бренд:audi
Введите n:5
id: 0, brand: audi, model: a6, year: 2010, color: красный, cost: 250000.0, regNumber: 5467TM-1, carNumber: 45tyfd4
id: 1, brand: volkswagen, model: polo, year: 2018, color: серебристый, cost: 120000.0, regNumber: 3211AB-3, carNumber: 4eruvxcR
```

4

```
4
Введите год выпуска:2018
Введите цену:90000
id: 1, brand: volkswagen, model: polo, year: 2018, color: серебристый, cost: 120000.0, regNumber: 3211AB-3, carNumber: 4eruvxcR
id: 2, brand: lada, model: vesta, year: 2018, color: синий, cost: 80000.0, regNumber: 5622GV-2, carNumber: 2vi4fasd222
```

5

```
5
id: 1, brand: volkswagen, model: polo, year: 2018, color: серебристый, cost: 120000.0, regNumber: 3211AB-3, carNumber: 4eruvxcR
id: 2, brand: lada, model: vesta, year: 2018, color: синий, cost: 80000.0, regNumber: 5622GV-2, carNumber: 2vi4fasd222
id: 3, brand: volkswagen, model: passat, year: 2004, color: серый, cost: 56000.0, regNumber: 4612AM-4, carNumber: 23njdfs3
```

6

```
6
id: 1, brand: volkswagen, model: polo, year: 2018, color: серебристый, cost: 120000.0, regNumber: 3211AB-3, carNumber: 4eruvxcR, fio: Коровьев_Семен_Валерьевич, passportNumber: AB2341243
id: 2, brand: lada, model: vesta, year: 2018, color: синий, cost: 80000.0, regNumber: 5622GV-2, carNumber: 2vi4fasd222, fio: Гулькова_Анастасия_Дмитриевна, passportNumber: AB4629489
id: 3, brand: volkswagen, model: passat, year: 2004, color: серый, cost: 56000.0, regNumber: 4612AM-4, carNumber: 23njdfs3, fio: Авдоскич_Алексей_Алексеевич, passportNumber: AB2956122
```