Министерство образования Калининградской области

государственное бюджетное учреждение Калининградской области

профессиональная образовательная организация

«Колледж информационных технологий и строительства»

(ГБУ КО ПОО «КИТиС»)

**Отчет по учебной практике**

УП.01 Разработка модулей программного обеспечения для компьютерных систем

по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Специальность 09.02.07 «Информационные системы и программирование»

Сроки прохождения практики:

с «17» ноября 2022 г. по «07» декабря 2022 г.

Место практики ГБУ КО ПОО «КИТиС»

|  |  |
| --- | --- |
| Выполнил: | студентка 4 курса,  группы ИСп 19-1  Лебедева Екатерина Георгиевна  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |
| Проверила: | Большакова-Стрекалова Анна Викторовна  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (оценка)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) |

Калининград, 2022

Содержание

|  |  |
| --- | --- |
| Введение | 3 |
| 1 Решение задач | 4 |
| 1.1 Задача 1 | 4 |
| 1.2 Задача 2 | 6 |
| 1.3 Задача 3 | 11 |
| 1.4 Задача 4 | 13 |
| 1.5 Задача 5 | 17 |
| 1.6 Задача 6 | 18 |
| 1.7 Задача 7 | 20 |
| Заключение | 26 |
| Список использованных источников | 27 |

1. Введение

Учебная практика по модулю ПМ 01 «Разработка модулей программного обеспечения для компьютерных систем» проходит на базе колледжа.

Данная практика являет способам показать и закрепить полученные знания в областях разработки автоматизированных информационных систем, разработки мобильных приложений, развитие профессиональных навыков в области проектирования баз данных и проектирования программных интерфейсов.

Задачами практики являются разработка приложений по работе с матрицами, классами и вложенными классами, работе с циклами, разработки визуального интерфейса, проектирование базы данных и мобильного приложения. Также в ходе работы требуется составить техническое задание и разработать диаграммы.

1. Решение задач
   1. Задача 1



Код:

public class Main {

static void createMatrix(int[][] a) {

for (int i = 0; i < a.length; i++) {

for (int j = 0; j < a.length; j++) {

System.out.print(a[i][j] + " ");

}

System.out.println();

}

System.out.println();

}

static void rotateM270(int[][] a) {

int[][] b = new int[a.length][a.length];

for (int i = 0; i < b.length; i++) {

for (int j = 0; j < b.length; j++) {

b[i][j] = a[j][a.length - i - 1];

}

}

System.out.println("Rotating by 270 degree:");

createMatrix(b);

}

static void rotateM90(int[][] a) {

int[][] b = new int[a.length][a.length];

for (int i = 0; i < b.length; i++) {

for (int j = 0; j < b.length; j++) {

b[j][i] = a[a.length - i - 1][j];

}

}

System.out.println("This matrix is rotated by 90 degree:");

createMatrix(b);

}

static void rotateM180(int[][] a) {

int[][] b = new int[a.length][a.length];

for (int i = 0; i < b.length; i++) {

for (int j = 0; j < b.length; j++) {

b[j][i] = a[a.length - j - 1][a.length - i - 1];

}

}

System.out.println("This matrix is rotated by 180 degree:");

createMatrix(b);

}

public static void main(String[] args) {

int[][] a = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

System.out.println("Original matrix:");

createMatrix(a);

rotateM270(a);

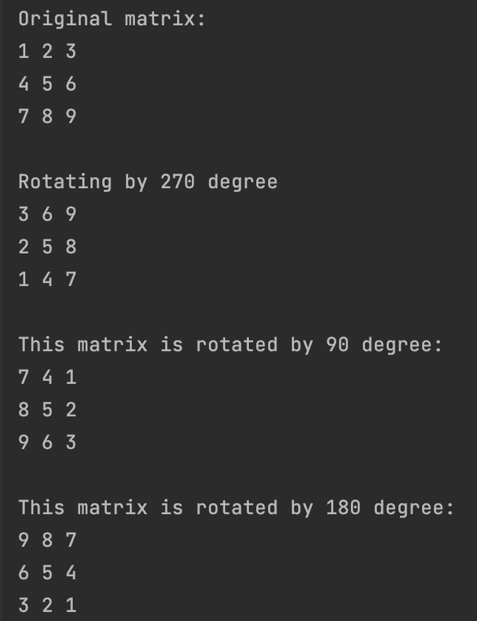
rotateM90(a);

rotateM180(a);

}

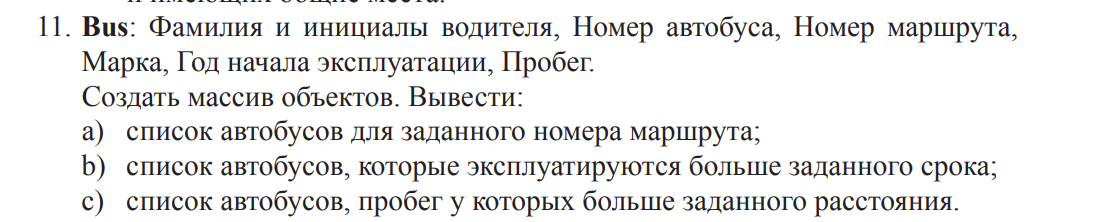
}

Вывод:



* 1. Задача 2

Задание:



Код:

Main

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Bus[] buses\_list = busese\_list(); ///// ручное заполнение

Bus[] buses\_list = bus(); ////// автозаполнение

System.out.println("----------a)Вывести список автобусов для заданного номера маршрута.");

System.out.print("Введите номер маршрута: ");

bus\_check\_route(buses\_list, scanner.nextInt());

System.out.println("----------b)Вывести список автобусов, которые эксплуатируются больше заданного срока.");

System.out.print("Введите срок эксплуатации: ");

bus\_check\_year(buses\_list, scanner.nextInt());

System.out.println("----------c)Вывести список автобусов, пробег у которых больше заданного расстояния.");

System.out.print("Введите пробег: ");

bus\_check\_mileage(buses\_list, scanner.nextInt());

}

static Bus[] busese\_list(){

Scanner scanner1 = new Scanner(System.in);

System.out.print("Введите количество автобусов: ");

Bus[] buses = new Bus[scanner1.nextInt()];

for (int i = 0; i < buses.length; i++) {

buses[i] = new Bus();

System.out.println();

System.out.print("Введите фамилию водителя: ");

buses[i].setDriver(scanner1.next());

System.out.print("Введите номер автобуса: ");

buses[i].setNumber\_bus(scanner1.nextInt());

System.out.print("Введите номер маршрута: ");

buses[i].setRoute\_number(scanner1.nextInt());

System.out.print("Введите марку автобуса: ");

buses[i].setMark(scanner1.next());

System.out.print("Введите год начала эксплуатации: ");

buses[i].setYear\_operation(scanner1.nextInt());

System.out.print("Введите пробег атобуса: ");

buses[i].setMileage(scanner1.nextInt());

}

return buses;

}

static Bus[] bus(){

Bus[] buses = new Bus[3];

buses[0] = new Bus("Петров А.С.", 651, 45, "Mercedes", 1996, 216023);

buses[1] = new Bus("Семенов П.С.", 651, 12, "Mercedes", 1995, 226540);

buses[2] = new Bus("Иванов Н.В.", 651, 45, "Mazda", 2003, 186234);

return buses;

}

static void bus\_check\_route(Bus[] buses, int route){

for (Bus bus : buses) {

if(bus.getRoute\_number()==route)

System.out.println(bus);

}

System.out.println();

}

static void bus\_check\_year(Bus[] buses,int n){

for (Bus bus : buses) {

if((2016-bus.getYear\_operation())>n)

System.out.println(bus);

}

System.out.println();

}

static void bus\_check\_mileage(Bus[] buses, int mileage){

for (Bus bus : buses) {

if(bus.getMileage()>mileage)

System.out.println(bus);

}

}

}

Bus

public class Bus {

private String driver;

private int number\_bus;

private int route\_number;

private String mark;

private int year\_operation;

private int mileage;

public Bus(){

this.driver = "";

this.number\_bus = 0;

this.route\_number = 0;

this.mark = "";

this.year\_operation = 0;

this.mileage = 0;

}

public Bus(String driver, int number\_bus, int route\_number, String mark, int year\_operation, int mileage) {

this.driver = driver;

this.number\_bus = number\_bus;

this.route\_number = route\_number;

this.mark = mark;

this.year\_operation = year\_operation;

this.mileage = mileage;

}

public String getDriver() {

return driver;

}

public void setDriver(String surname) {

this.driver = surname;

}

public int getNumber\_bus() {

return number\_bus;

}

public void setNumber\_bus(int number\_bus) {

this.number\_bus = number\_bus;

}

public int getRoute\_number() {

return route\_number;

}

public void setRoute\_number(int route\_number) {

if(route\_number>0) {

this.route\_number = route\_number;

} else System.out.println("Номер маршрута не может быть отрицательным или нулевым!");

}

public String getMark() {

return mark;

}

public void setMark(String mark) {

this.mark = mark;

}

public int getYear\_operation() {

return year\_operation;

}

public void setYear\_operation(int year\_operation) {

if(year\_operation>0) {

this.year\_operation = year\_operation;

} else System.out.println("Год начала эксплуатации не может быть отрицательным или нулевым!");

}

public int getMileage() {

return mileage;

}

public void setMileage(int mileage) {

if(mileage>=0) {

this.mileage = mileage;

} else System.out.println("Пробег не может быть отрицательным!");

}

@Override

public String toString() {

return "Автобус --- " +

"Водитель=" + driver +

", Номер автобуса=" + number\_bus +

", Номер маршрута=" + route\_number +

", Марка=" + mark +

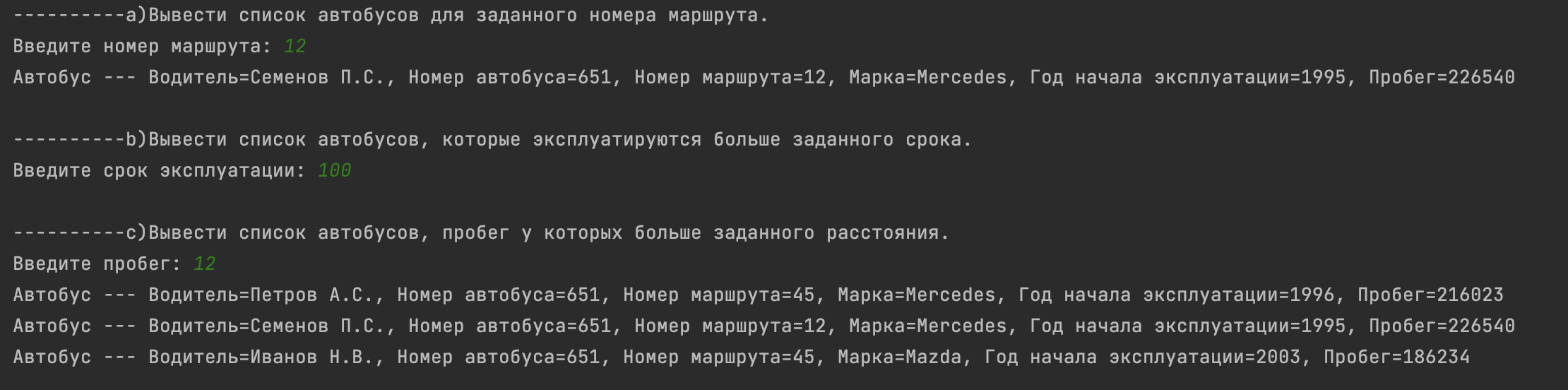
", Год начала эксплуатации=" + year\_operation +

", Пробег=" + mileage;

}

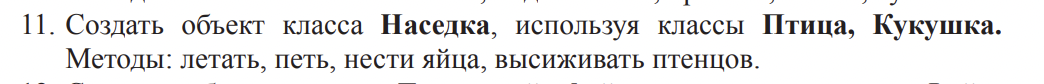
}

Вывод:



* 1. Задание 3

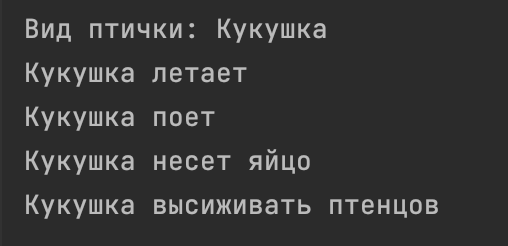
Задание:



Код:

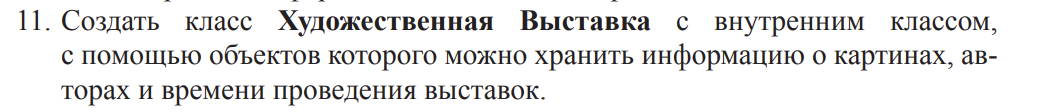
public class Main {  
 public static void main(String[] args) {  
 Сuckoo cuckoo = new Сuckoo("Кукушка");  
  
 cuckoo.showName();  
 cuckoo.fly();  
 cuckoo.sing();  
 cuckoo.laysAnEgg();  
 cuckoo.hatchingChicks();  
 }  
}  
class Bird{  
 String kind;  
 public Bird(String kind){  
 this.kind = kind;  
 }  
 public void showName (){  
 System.out.println("Имя животного: " + kind );  
 }  
}  
class Сuckoo extends Bird {  
 public Сuckoo(String name) {  
 super(name);  
 }  
 public void showName(){  
 System.out.println("Вид птички: " + kind);  
 }  
 public void fly(){  
 System.out.println(kind + " летает");  
 }  
 public void sing(){  
 System.out.println(kind + " поет");  
 }  
 public void laysAnEgg(){  
 System.out.println(kind + " несет яйцо ");  
 }  
 public void hatchingChicks(){  
 System.out.println(kind + " высиживают птенцов");  
 }  
}

Вывод:



* 1. Задание 4

Код:



import java.util.ArrayList;

public class Main {

public static void main(String[] args) {

ArrayList<ArtEObject> Art = new ArrayList<>();

Art.add(new picture("Мишки в сосновом бору"));

Art.add(new picture("Подсолнухи"));

Art.add(new records("---"));

Art.add(new Autor("Шишкин"));

Art.add(new Autor("Ван Гог"));

print(Art);

}

public static void print(ArrayList<ArtEObject> city) {

System.out.println("На выставке:");

for (ArtEObject object: city) {

System.out.println(object.toString());

}

}

}

class ArtEObject {

String name;

String info = "";

public void setName(String name) {

this.name = name;

}

public String getName() {

return name;

}

public String getInfo() {

return info;

}

public void setInfo(String info) {

this.info = info;

}

@Override

public String toString() {

return "CityObject{" +

"name='" + name + '\'' +

'}';

}

public void printInfo() {

System.out.println(info);

}

public ArtEObject(String name) {

this.name = name;

}

}

class picture extends ArtEObject {

public picture(String name) {

super(name);

}

public picture(String name, String info) {

super(name);

this.info = info;

}

@Override

public String toString() {

if (info.isEmpty()) {

return "Картина - " + name;

} else {

return "Картина - " + name + " - " + info;

}

}

}

class Autor extends ArtEObject {

public Autor(String name) {

super(name);

}

public Autor(String name, String info) {

super(name);

this.info = info;

}

@Override

public String toString() {

if (info.isEmpty()) {

return "Автор - " + name;

} else {

return "Автор - " + name + " - " + info;

}

}

}

class records extends ArtEObject {

public records(String name) {

super(name);

}

public records(String name, String info) {

super(name);

this.info = info;

}

@Override

public String toString() {

if (info.isEmpty()) {

return "Запись - " + name;

} else {

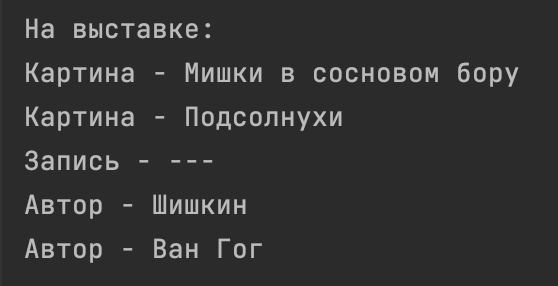
return "Запись - " + name + " - " + info;

}

}

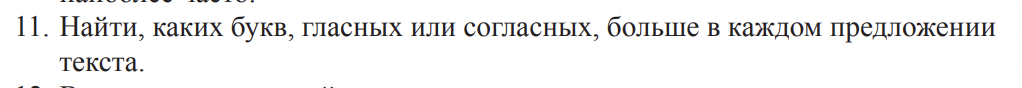
}

Вывод:

1

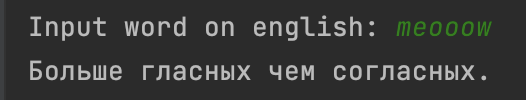
* 1. Задание 5

Задание:



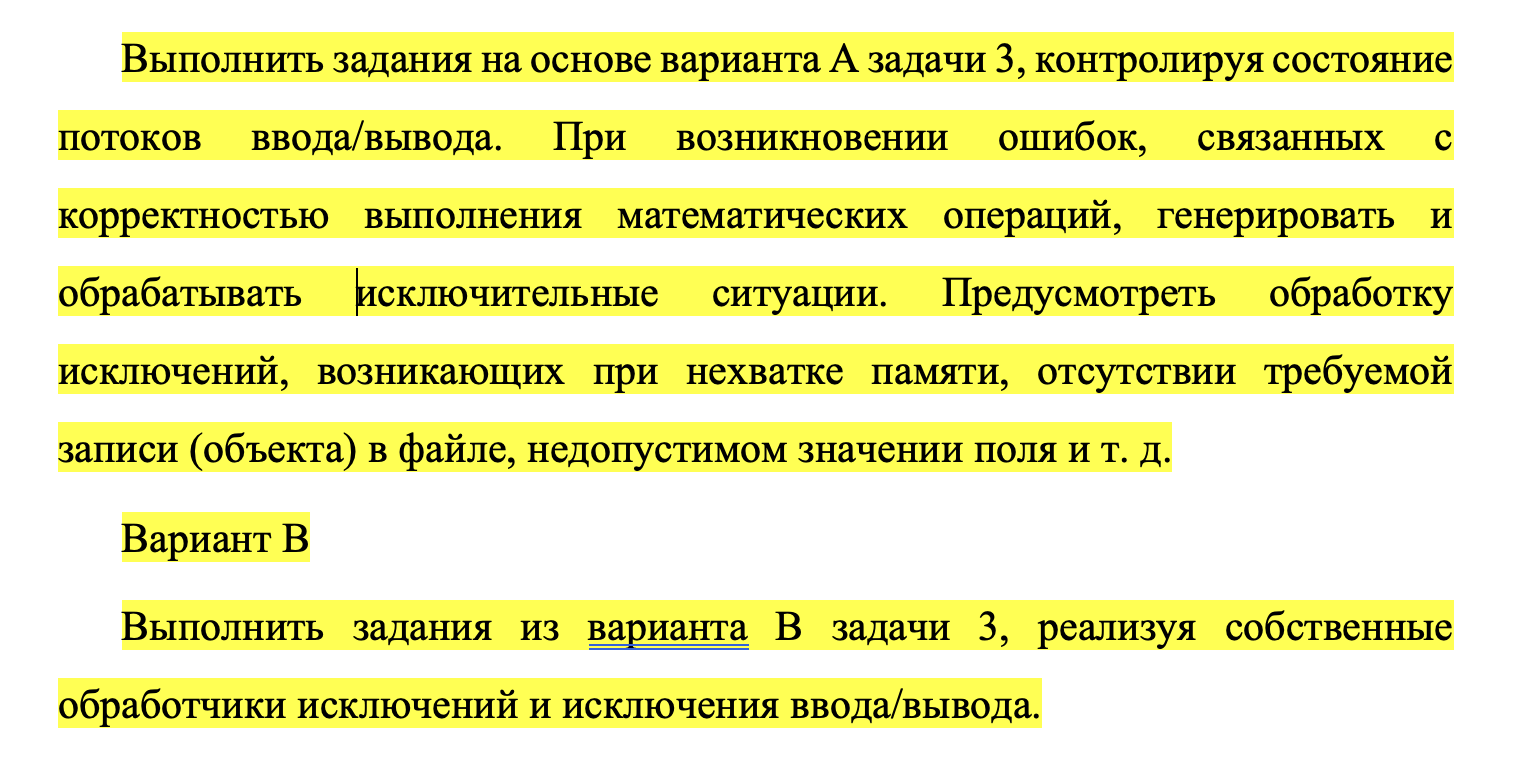
Код:

import java.util.Arrays;  
import java.util.HashSet;  
import java.util.Scanner;  
import java.util.Set;  
public class Main {  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.in);  
 System.out.print("Input word on english: ");  
 String str = in.nextLine();  
 Set<Character> vowels =  
 new HashSet<>(  
 Arrays.asList('a', 'e', 'o', 'u', 'i'));  
 Set<Character> consonants =  
 new HashSet<>(  
 Arrays.asList('b', 'c', 'd', 'p', 'l', 'v', 'j'));  
 int numberOfVowels = 0;  
 int numberOfConsonants = 0;  
 for (int i = 0; i < str.length(); i++) {  
 char ch = Character.toLowerCase(str.charAt(i));  
 if (vowels.contains(ch)) {  
 numberOfVowels++;  
 } else if (consonants.contains(ch)) {  
 numberOfConsonants++;  
 }  
 }  
 if (numberOfVowels > numberOfConsonants) {  
 System.out.println("Больше гласных чем согласных.");  
 } else {  
 System.out.println("Больше согласных чем гласных");  
 }  
 }  
}



* 1. Задание 6

Задание:



Код:

import java.io.IOException;

public class Main {

public static void main(String[] args) {

Сuckoo cuckoo = new Сuckoo("Кукушка");

cuckoo.showName();

cuckoo.fly();

cuckoo.sing();

cuckoo.laysAnEgg();

cuckoo.hatchingChicks();

}

}

class Bird{

String kind;

public Bird(String kind){

this.kind = kind;

}

public void showName (){

System.out.println("Имя животного: " + kind );

}

}

class Сuckoo extends Bird {

public Сuckoo(String name) {

super(name);

}

public void showName(){

System.out.println("Вид птички: " + kind);

}

public void fly(){

System.out.println(kind + " летает");

}

public void sing(){

System.out.println(kind + " поет");

}

public void laysAnEgg(){

System.out.println(kind + " несет яйцо ");

}

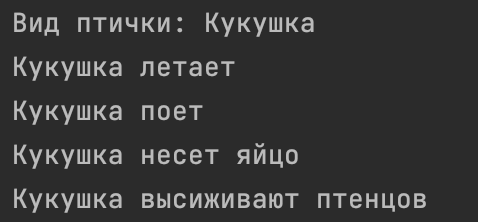
public void hatchingChicks(){

System.out.println(kind + " высиживают птенцов");

try {} catch (Exception e) {}

}

}



* 1. Задание 7

Задание:

Нарисовать звездочки и кругляшочки

Код:

Main

import java.awt.Color;

import java.awt.Dimension;

import java.awt.Graphics;

import java.util.ArrayList;

import java.util.List;

import java.util.Random;

import javax.swing.JFrame;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

public class Main extends JPanel {

private List shapes = new ArrayList<>();

private Random random = new Random();

public Main(int i, String shape) {

setBackground(Color.BLACK);

setPreferredSize(new Dimension(400, 400));

switch (shape) {

case "Circles":

for (int j = 0; j < i; j++) {

addCircle(390, 390);

}

break;

case "Stars":

for (int j = 0; j < i; j++) {

addStar(380, 380);

}

break;

case "Both":

int mid = i / 2;

for (int j = 0; j < mid; j++) {

addCircle(390, 390);

}

for (int j = mid; j < i; j++) {

addStar(380, 380);

}

break;

}

}

@Override

protected void paintComponent(Graphics g) {

super.paintComponent(g);

for (Object s : shapes) {

if (s instanceof Circle) {

((Circle) s).draw(g);

} else if (s instanceof Star) {

((Star) s).draw(g);

}

}

}

public void addCircle(int maxX, int maxY) {

shapes.add(new Circle(random.nextInt(maxX), random.nextInt(maxY)));

repaint();

}

public void addStar(int maxX, int maxY) {

shapes.add(new Star(random.nextInt(maxX), random.nextInt(maxY)));

repaint();

}

public static void main(String[] args) {

String shapeAmount = JOptionPane.showInputDialog(null,

"How many shapes?", "Random Shapes...", JOptionPane.PLAIN\_MESSAGE);

int amount = Integer.parseInt(shapeAmount);

String shapes[] = {"Stars", "Circles", "Both"};

String shape = (String) JOptionPane.showInputDialog(null,

"Pick the shape you want", "Random Shapes...",

JOptionPane.PLAIN\_MESSAGE, null, shapes, shapes[0]);

JFrame frame = new JFrame();

frame.add(new Main(amount, shape));

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.pack();

frame.setLocationRelativeTo(null);

frame.setVisible(true);

}

}

Star

import java.awt.Color;

import java.awt.Graphics;

import java.awt.Graphics2D;

import java.awt.geom.GeneralPath;

public class Star {

int x, y, width, height;

public Star(int x, int y) {

this.x = x;

this.y = y;

}

public void draw(Graphics g) {

int xPoints[] = {9, 15, 0, 18, 3};

int yPoints[] = {0, 18, 6, 6, 18};

Graphics2D g2d = (Graphics2D) g;

GeneralPath star = new GeneralPath();

star.moveTo(xPoints[0] + x, yPoints[0] + y);

for (int i = 1; i < xPoints.length; i++) {

star.lineTo(xPoints[i] + x, yPoints[i] + y);

}

star.closePath();

g2d.setColor(Color.YELLOW);

g2d.fill(star);

}

}

Circle

import java.awt.Color;

import java.awt.Graphics;

import java.awt.Graphics2D;

import java.awt.geom.Ellipse2D;

public class Circle {

int x, y, width, height;

public Circle(int x, int y) {

this.x = x;

this.y = y;

}

public void draw(Graphics g) {

Graphics2D g2d = (Graphics2D) g;

Ellipse2D.Double circle = new Ellipse2D.Double(x, y, 10, 10);

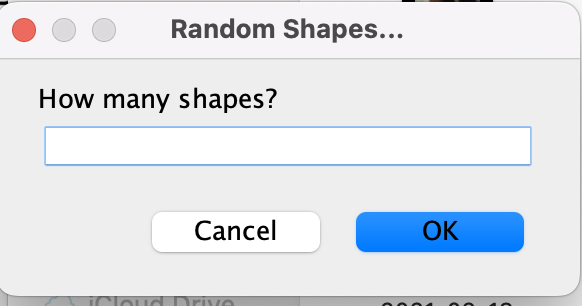
g2d.setColor(Color.GRAY);

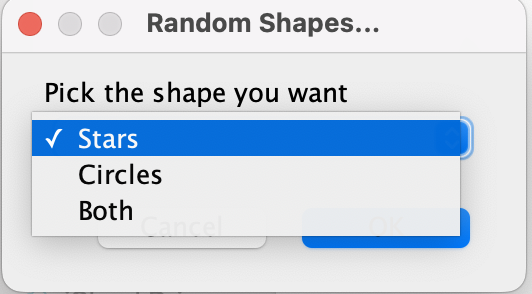
g2d.fill(circle);

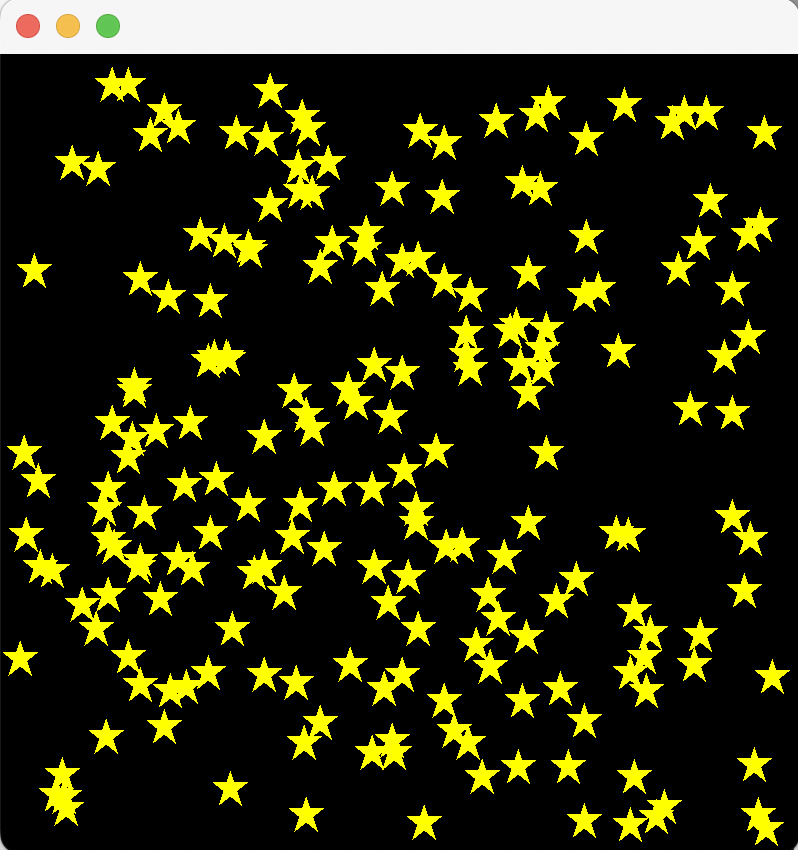
}

}

Результат:







Заключение

В ходе прохождения практики были освоены навыки по разработке автоматических информационных систем, закреплены знания в области проектирования программных интерфейсов и баз данных.

Цели и задачи практики достигнуты. Разработаны алгоритмы решения задач с матрицами и циклами.

Список использованных источников

1. Google Sites – <https://sites.google.com/>
2. Онлайн-редактор блок-схем – <https://programforyou.ru/>
3. учебник «Java – Методы  программирования»
4. Семакин И. Г. – «Основы алгоритмизации и программирования»
5. Руководство по использованию - <https://habr.com/ru/post/125799/>