

**Student name:** Anvarjon Radjapov

B.Sc(IT) section -3

**Enrolment No**.: A85204919022

Lab experiments:

1. Write a java program to add two integers and two float numbers. When no arguments are supplied, give a default value to calculate the sum. Use function overloading.

import java.util.Scanner;

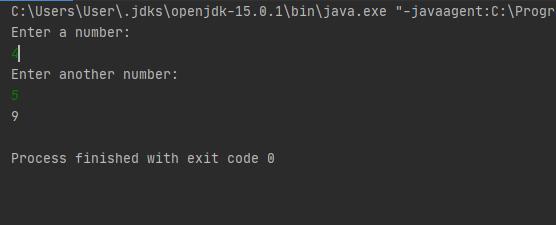
public class adding {

public static void main(String[] args){ Scanner scan= new Scanner(System.*in*); System.*out*.println("Enter a number:"); int a = scan.nextInt();

System.*out*.println("Enter another number:"); int b = scan.nextInt(); System.*out*.println(a+b);4

}

}



1. Write a program to implement constructor overloading by passing different number of parameters of different types.

public class Constructor {

float a,b;

Constructor()

{

a=0;

b=0;

System.*out*.println("\na="+a+"b="+b);

}

Constructor(int x)

{

a=x;

b=x;

System.*out*.println("\na="+a+"b="+b);

}

Constructor(int p,int q)

{

a=p;

b=q;

System.*out*.println("\na="+a+"b="+b);

}

Constructor(Constructor c)

{

a=c.a;

b=c.b;

System.*out*.println("\na="+a+"b="+b);

}

}

class ConsOver

{

public static void main(String[] args)

{

Constructor c1= new Constructor();

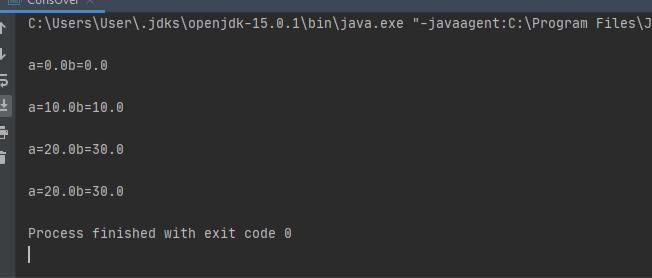
Constructor c2= new Constructor(10);

Constructor c3= new Constructor(20,30);

Constructor c4= new Constructor(c3);

}

}



1. Write a program to perform mathematical operations. Create a class called AddSub with methods to add and subtract. Create another class called MulDiv that extends from AddSub class to use the member data of the super class. MulDiv should have methods to multiply and divide A main function should access the methods and perform the mathematical operations.

import java.util.Scanner;

public class operations {

public static void main(String[] args)

{

Scanner user\_input= new Scanner(System.*in*);

int num1, num2;

System.*out*.print("Enter two numbers: ");

num1=user\_input.nextInt();

num2=user\_input.nextInt();

System.*out*.print("\nAddition: "+(num1+num2));

System.*out*.print("\nSubtraction: "+(num1-num2));

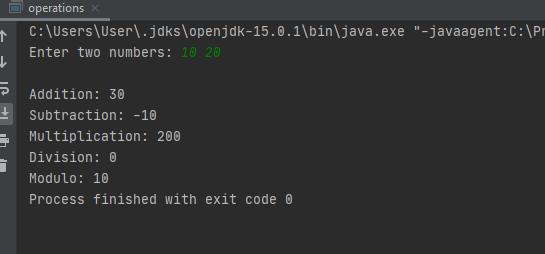
System.*out*.print("\nMultiplication: "+(num1\*num2));

System.*out*.print("\nDivision: "+(num1/num2));

System.*out*.print("\nModulo: "+(num1%num2));

}

}



1. Write a program with class variable that is available for all instances of a class .Use static variable declaration. Observe the changes that occur in the object’s member variable values.

package Program4;

public class Program4 {

static int *count*=0;

public void increment()

{

*count*++;

}

public static void main(String args[])

{

Program4 variable = new Program4();

Program4 variable1 = new Program4();

variable.increment();

variable1.increment();

System.*out*.println("First object: count is = " +

variable.*count*);

System.*out*.println("Second object: count is = " + variable1.*count*);

}

output:

First object: count is = 2

Second object: count is = 2

1. Write a java program to create a Student class with following attributes.:- Enrollment No:, Name, Mark of sub1, Mark of sub2, mark of sub3, Total Marks. Total of the three marks must be calculated only when the student passes in all three subjects. The pass mark for each subject is 50. If a candidate fails in any one of the subjects his total mark must be declared as zero. Using this condition write a constructor for this class. Write separate functions for accepting and displaying student details. In the main method create an array of three student objects and display the details.

import java.util.Scanner;

public class studentmark {

public static void main(String args[])

{

int marks[] = new int[6];

int i;

float total=0, avg;

Scanner scanner= new Scanner(System.*in*); for(i=0; i<6; i++) {

System.*out*.print("Enter Marks of Subject"+(i+1)+":"); marks[i] = scanner.nextInt(); total = total + marks[i];

}

scanner.close();

avg = total/6;

System.*out*.print("The student Grade is: ");

if(avg>=80)

{

System.*out*.print("A");

}

else if(avg>=60 && avg<80)

{

System.*out*.print("B");

}

else if(avg>=40 && avg<60)

{

System.*out*.print("C");

}

else

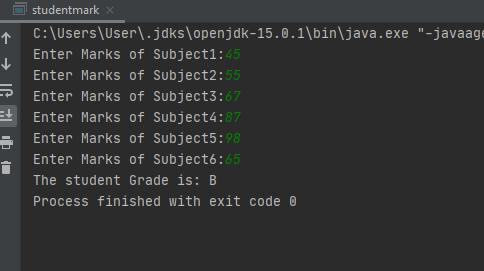
{

System.*out*.print("D");

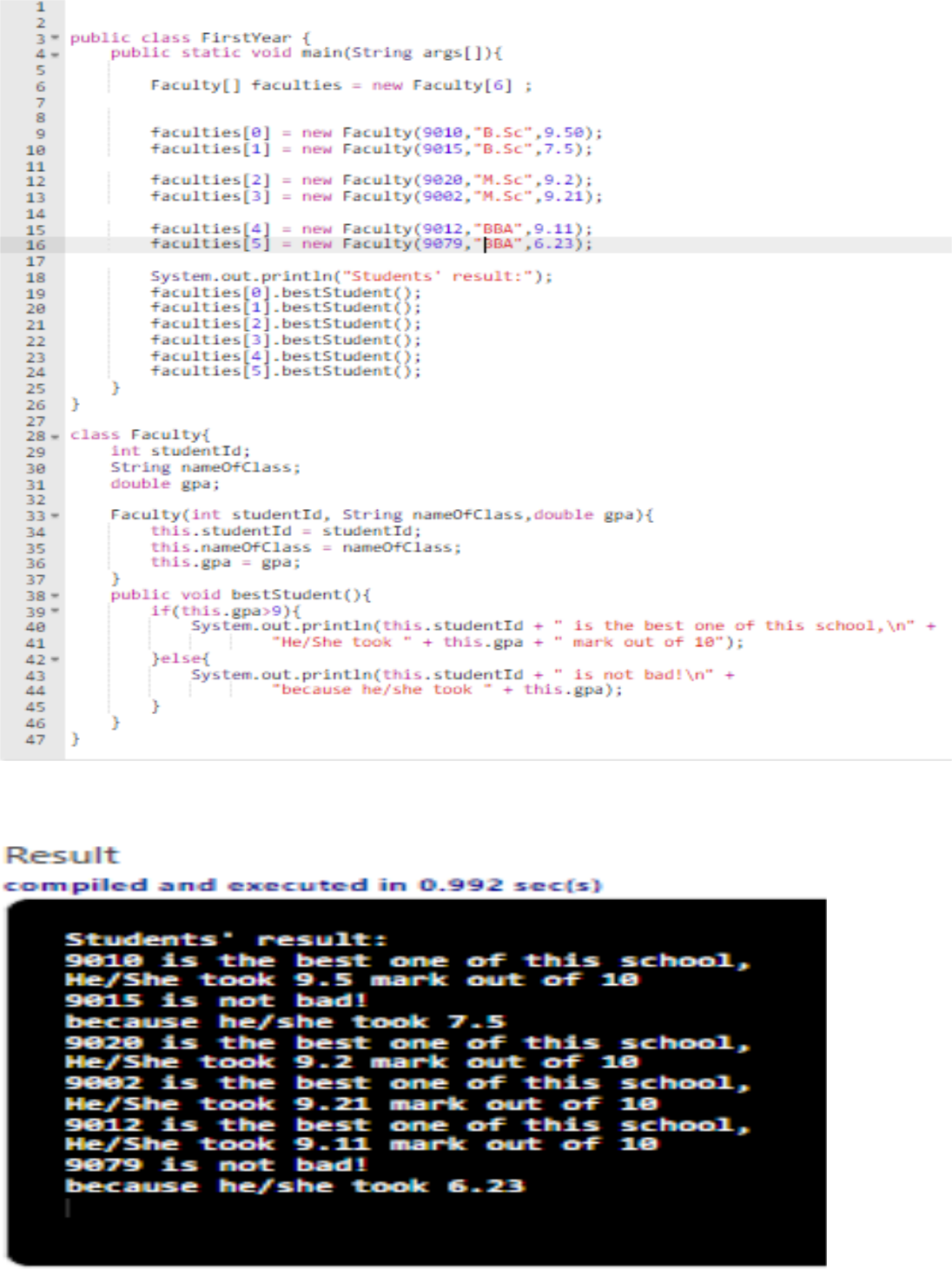
}

}

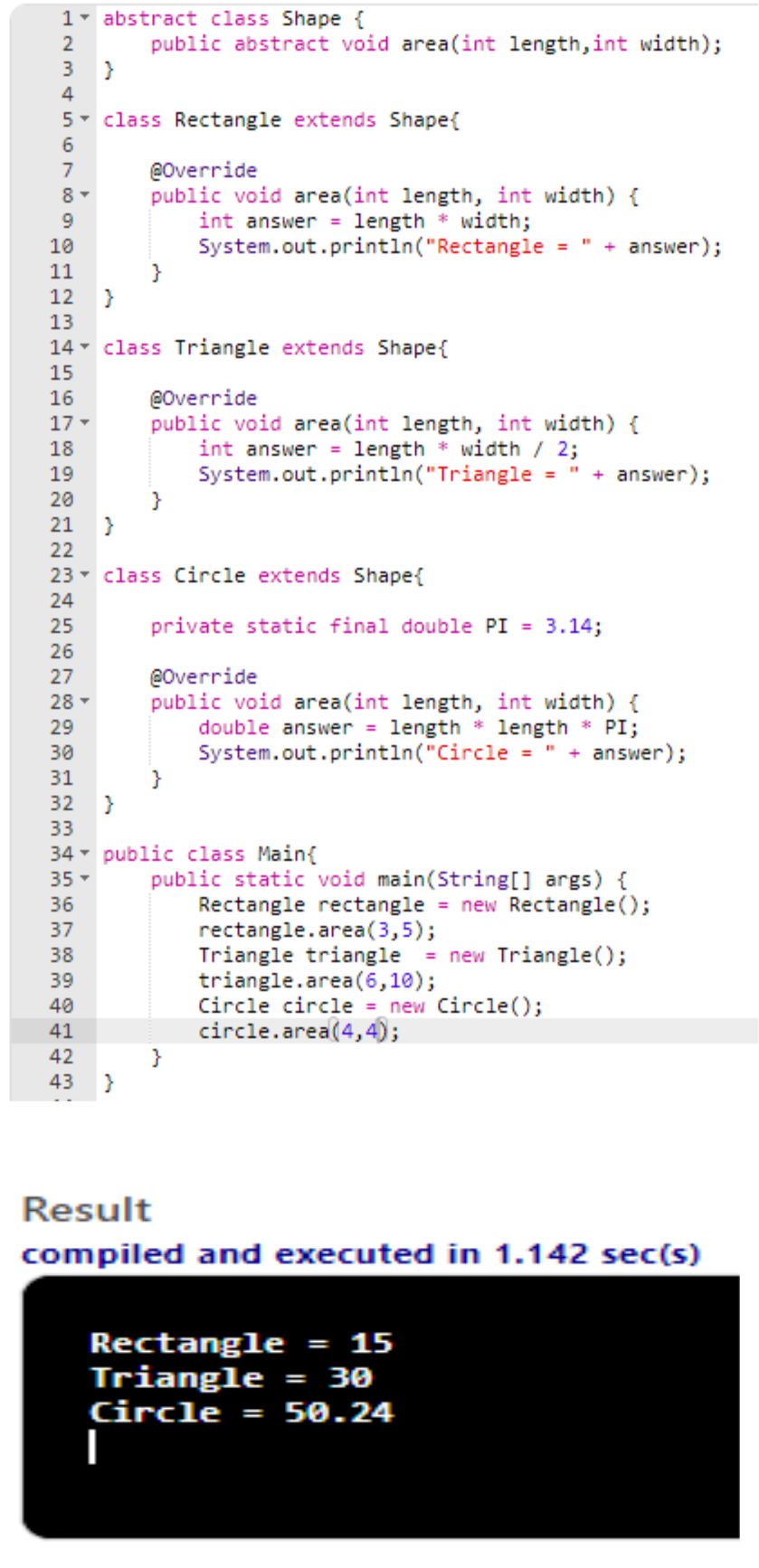
}



1. In a college first year class are having the following attributes Name of the class (B.Sc , BBA, M.Sc), Name of the staff No of the students in the class, Array of students in the class Define a class called first year with above attributes and define a suitable constructor. Also write a method called best student() which process a first year object and return the student with the highest total mark. In the main method define a first year object and find the best student of this class



1. Write a java program to create an abstract class named shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class shape. Each one of the class contains only the method print Area () that print the area of the given shape.



1. WAP that describes a class person. It should have instance variables to record name, age and salary. Create a person object. Set and display its

instance variables.

class person

{

String name;

int age;

float sal;

}

class prg15

{

public static void main(String args[])

{

person pr = new person();

pr.name = "Nodirbek";

pr.age = 21;

pr.sal = 89000;

System.out.println("Name of the Person is : "+pr.name);

System.out.println("Age of the Person is : "+pr.age);

System.out.println("Salary of the Person is : "+pr.sal);

}

}

**Output is:**

Name of the Person is : Nodirbek

Age of the Person is : 21

Salary of the Person is : 89000

1. Write a program to swap two values using object reference should have a swap function.

public class Swapping {

String model;

int price;

Swapping(String model, int price)

{

this.model = model;

this.price = price;

}

void print()

{

System.*out*.println("price = " + this.price + ", model

* " + this.model);

}

}

class CarWrapper

{

Swapping swapping;

CarWrapper(Swapping swapping){

this.swapping = swapping;

}

}

class Main

{

public static void swap(CarWrapper carWrapper, CarWrapper carWrapper1)

{

Swapping temp = carWrapper.swapping; carWrapper.swapping = carWrapper1.swapping; carWrapper1.swapping = temp;

}

public static void main(String[] args)

{

Swapping firstObject = new Swapping("Pigeon", 100000); Swapping secondObject = new Swapping("Jaguar",

120000);

CarWrapper carWrapper = new CarWrapper(firstObject); CarWrapper carWrapper1 = new CarWrapper(secondObject); *swap*(carWrapper, carWrapper1);carWrapper.swapping.print(); carWrapper1.swapping.print();

}

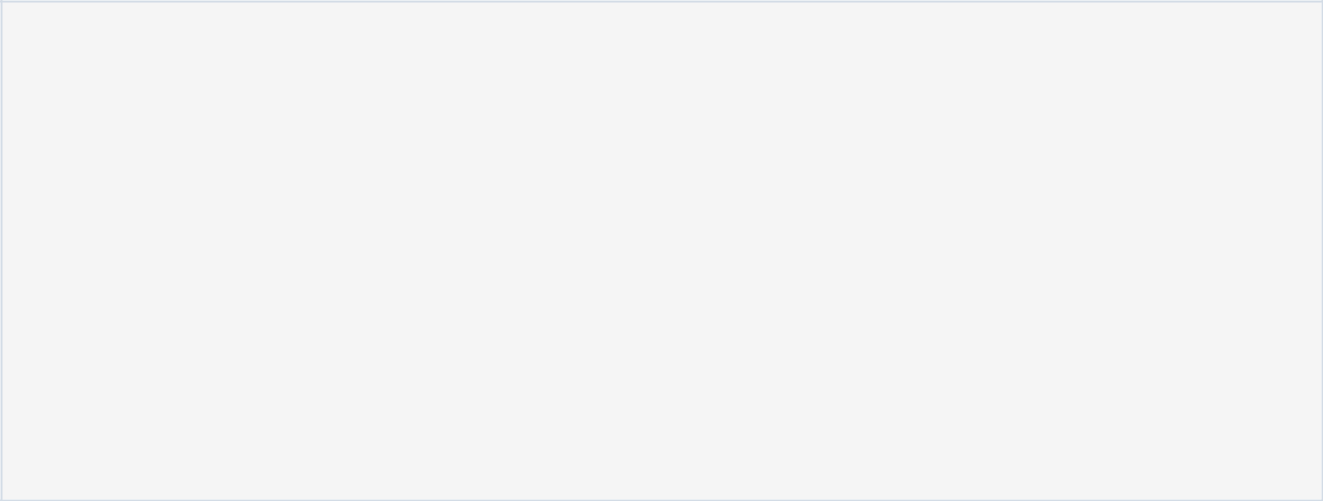
}

output is:

price = 12000, model = Chevrolet

price = 10000, model = opel

1. Write a program to explain the multithreading with the use of multiplication table. Three threads must be defined. Each one must create one multiplication table.



public class MultiplicationTable {

public static void main(String[] args) {

int num = 5;

for(int i = 1; i <= 10; ++i)

{

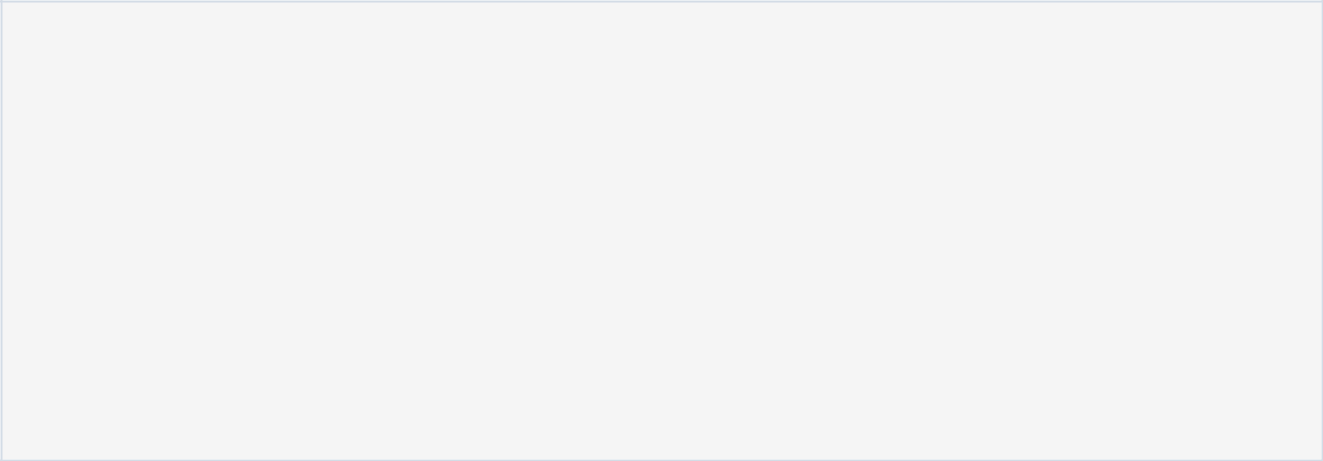
System.out.printf("%d \* %d = %d \n", num, i, num \* i);

}

}

}

Output:



5\*1=5

5\*2=10

5\*3=15

5\*4=20

5\*5=25

5\*6=30

5\*7=35

5\*8=40

5\*9=45

5\*10=50

11. Write a java program in which thread sleep for 5 sec and change the

name of thread.

public class ThreadChangeName extends Thread { public void run(){

for(int i=1;i<4;i++) {

try {

//call sleep method of thread

Thread.*sleep*(500);

}catch(InterruptedException exception){ System.*out*.println(exception);

}

//print current thread instance with loop variable System.*out*.println(Thread.*currentThread*().getName()

* " : " + i);

}

}

}

class Main{

public static void main(String args[])

{

ThreadChangeName firstthread = new ThreadChangeName();

ThreadChangeName secondthread = new ThreadChangeName();

//start threads one by one

firstthread.start();

secondthread.start();

}

}

output:

Thread-1 : 1

Thread-0 : 1

Thread-1 : 2

Thread-0 : 2

Thread-1 : 3

Thread-0 : 3

12. Write an application to illustrate Interface Inheritance.

import java.io.\*;

interface intfA

{

void m1();

}

interface intfB

{

void m2();

}

class sample implements intfA, intfB

{

public void m1()

{

System.out.println("Welcome: inside the method m1");

}

public void m2()

{

System.out.println("Welcome: inside the method m2");

}

}

class GFG

{

public static void main (String[] args)

{

sample ob1 = new sample();

* calling the method implemented
* within the class.

ob1.m1();

ob1.m2();

}

}

13. Write an application that shows how to create a user-defined exception.

class MyException extends Exception{

String str1;

MyException(String str2) {

str1=str2;

}

public String toString(){

return ("MyException Occurred: "+str1) ;

}

}

class Example1{

public static void main(String args[]){

try{

System.out.println("Starting of try block");

throw new MyException("This is My error Message");

}

catch(MyException exp){

System.out.println("Catch Block") ;

System.out.println(exp) ;

}

}

}

1. Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException. Display the exception in a message dialog box.



import java.awt.\*;

import java.awt.event.\*;

import java.applet.\*;

/\*<applet code="DivisionExample"width=230 height=250></applet>\*/

public class DivisionExample extends Applet implements ActionListener { String msg;

TextField num1, num2, res;

Label l1, l2, l3;

Button div;

public void init() {

l1 = new Label("Dividend");

l2 = new Label("Divisor");

l3 = new Label("Result");

num1 = new TextField(10);

num2 = new TextField(10);

res = new TextField(10);

div = new Button("Click");

div.addActionListener(this);

add(l1);

add(num1);

add(l2);

add(num2);

add(l3);

add(res);

add(div);

}

public void actionPerformed(ActionEvent ae) {

String arg = ae.getActionCommand();

int num1 = 0, num2 = 0;

if (arg.equals("Click")) {

if (this.num1.getText().isEmpty() && this.num2.getText().isEmpty()) { msg = "Enter the valid numbers!";

repaint();

} else {

try {

num1 = Integer.parseInt(this.num1.getText());

num2 = Integer.parseInt(this.num2.getText());

int num3 = num1 / num2;

res.setText(String.valueOf(num3));

msg = "Operation Succesfull!!!";

repaint();

} catch (NumberFormatException ex) { System.out.println(ex); res.setText("");

msg = "NumberFormatException - Non-numeric"; repaint();

} catch (ArithmeticException e) {

System.out.println("Can't be divided by Zero" + e);

res.setText("");

msg = "Can't be divided by Zero";

repaint();

}

}

}

}

public void paint(Graphics g) {

g.drawString(msg, 30, 70);

}

}

1. Write a small program to catch Negative Array Size Exception. This exception is caused when the array is initialized to negative values

public void abc() {

Scanner sc=new Scanner(System.in);

int s1=0;

System.out.println("enter number of elemtns ");

//Read the size of array from user and check if it is not negative

s1=sc.nextInt();

if(s1<0) throw new NegativeArraySizeException("Array size cannot be negative.");

int i[]=new int[s1];

int s2 = 0;

for(int i2=0;i2<i.length;i2++) {

System.out.println("Enter element");

s2=sc.nextInt();

//get input from user and dont allow negtive numbers in the array

if(s2<0) throw new IllegalArgumentException("Negative numbers not allowed"); ts.add(s2);

}

System.out.println("first,,,,"+s2);

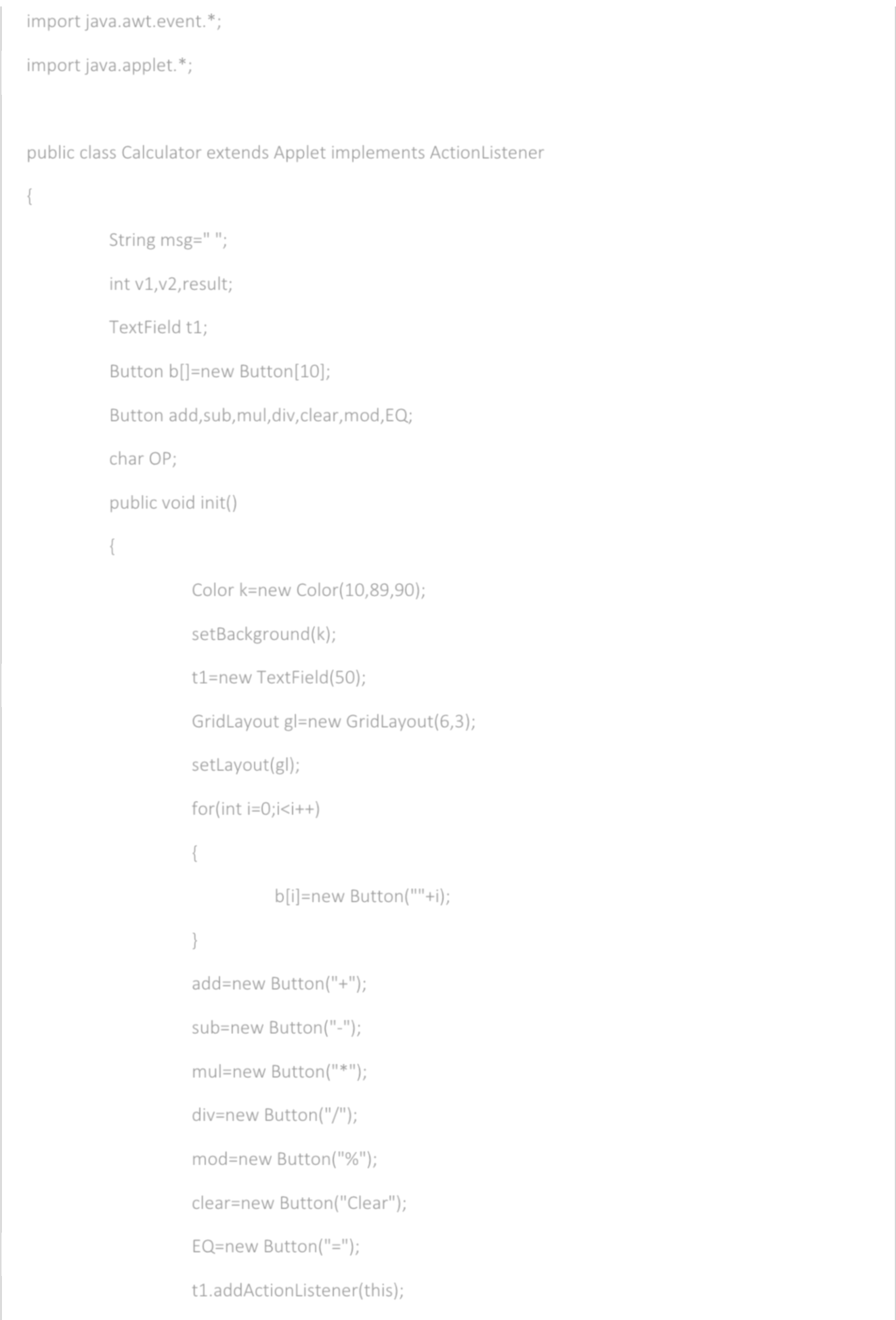
System.out.println("higest....."+ts.last());

}

1. Write a java program that works as a simple calculator. Use a Grid Layout to arrange Buttons for digits and for the + - \* % operations. Add a text field to display the result. Handle any possible exceptions like divide by zero.



import java.awt.\*;

import java.awt.event.\*;

import java.applet.\*;

public class Calculator extends Applet implements ActionListener

{

String msg=" ";

int v1,v2,result;

TextField t1;

Button b[]=new Button[10];

Button add,sub,mul,div,clear,mod,EQ;

char OP;

public void init()

{

Color k=new Color(10,89,90);

setBackground(k);

t1=new TextField(50);

GridLayout gl=new GridLayout(6,3);

setLayout(gl);

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

{

b[i]=new Button(""+i);

}

add=new Button("+");

sub=new Button("-");

mul=new Button("\*");

div=new Button("/");

mod=new Button("%");

clear=new Button("Clear");

EQ=new Button("=");

t1.addActionListener(this);

add(t1);

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

{

add(b[i]);

}

add(add);

add(sub);

add(mul);

add(div);

add(mod);

add(clear);

add(EQ);

for(int i=0;i<10;i++)

{

b[i].addActionListener(this);

}

add.addActionListener(this);

sub.addActionListener(this);

mul.addActionListener(this);

div.addActionListener(this);

mod.addActionListener(this);

clear.addActionListener(this);

EQ.addActionListener(this);

}

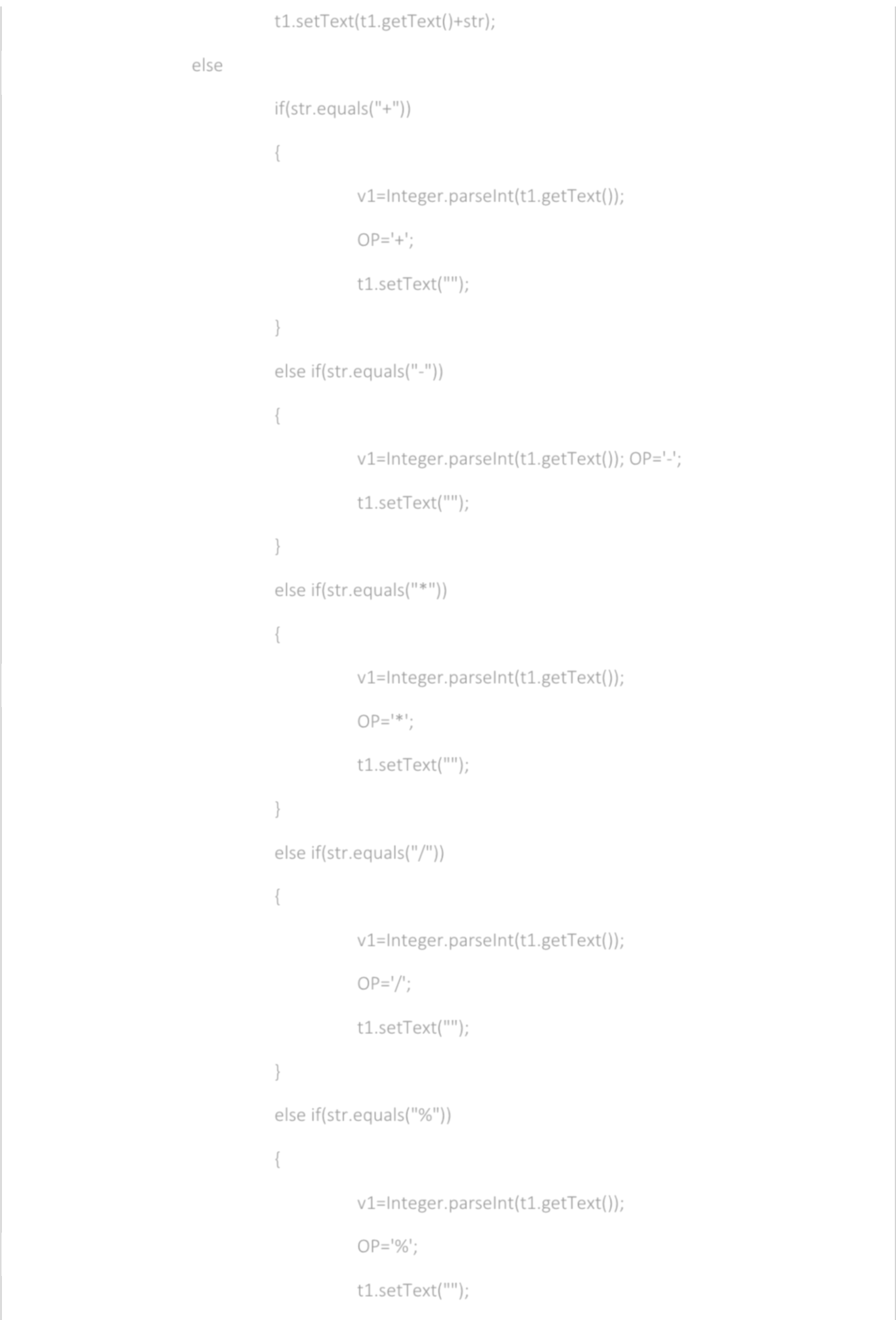
public void actionPerformed(ActionEvent ae)

{

String str=ae.getActionCommand();

char ch=str.charAt(0);

if ( Character.isDigit(ch))

t1.setText(t1.getText()+str);

else

if(str.equals("+"))

{

v1=Integer.parseInt(t1.getText());

OP='+';

t1.setText("");

}

else if(str.equals("-"))

{

v1=Integer.parseInt(t1.getText()); OP='-';

t1.setText("");

}

else if(str.equals("\*"))

{

v1=Integer.parseInt(t1.getText());

OP='\*';

t1.setText("");

}

else if(str.equals("/"))

{

v1=Integer.parseInt(t1.getText());

OP='/';

t1.setText("");

}

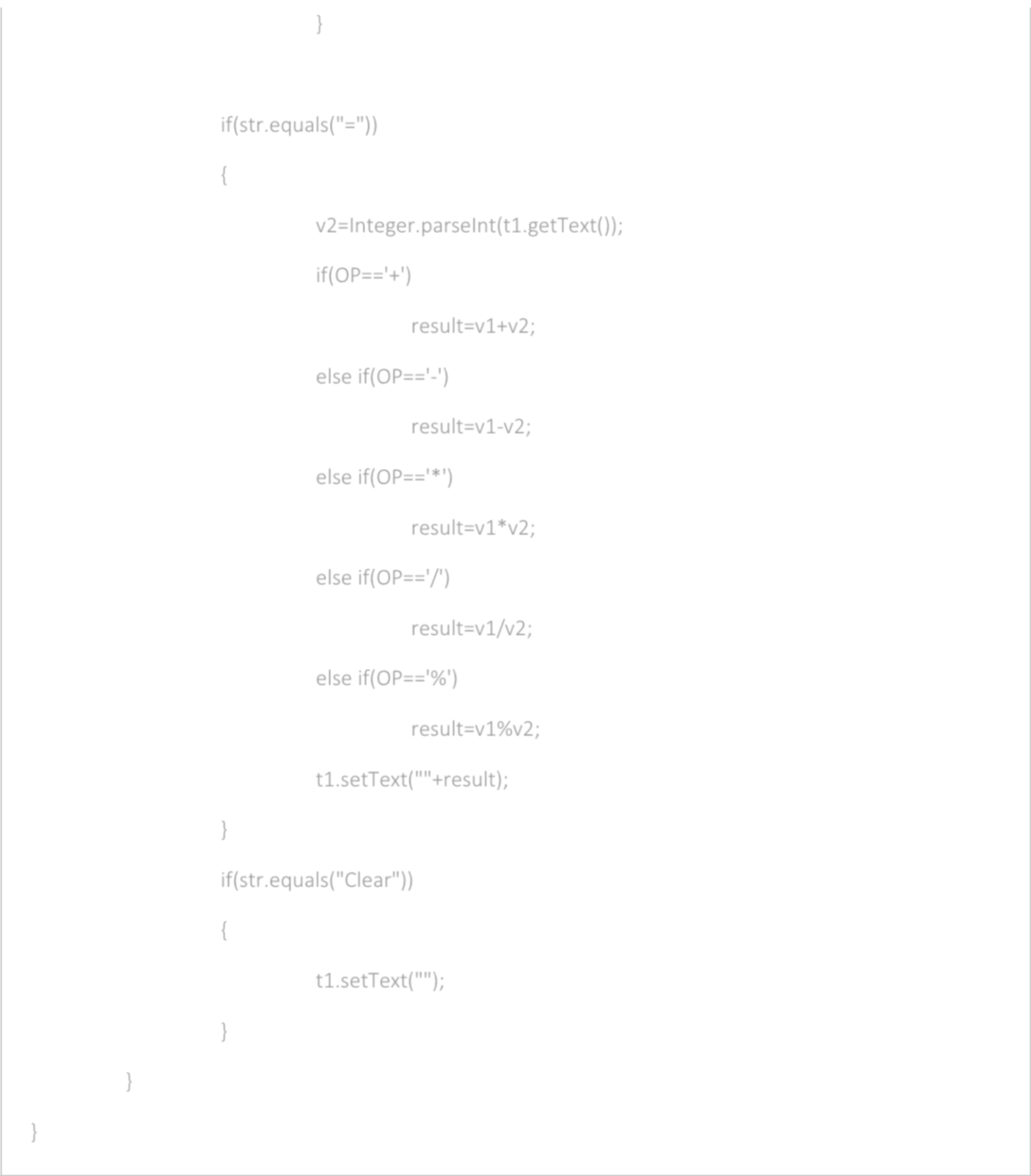
else if(str.equals("%"))

{

v1=Integer.parseInt(t1.getText());

OP='%';

t1.setText("");

}

if(str.equals("="))

{

v2=Integer.parseInt(t1.getText());

if(OP=='+')

result=v1+v2;

else if(OP=='-')

result=v1-v2;

else if(OP=='\*')

result=v1\*v2;

else if(OP=='/')

result=v1/v2;

else if(OP=='%')

result=v1%v2;

t1.setText(""+result);

}

if(str.equals("Clear"))

{

t1.setText("");

}

}

}

17. Write a program to implement all string operations.

import java.util.\*;

class StringOperation

{

public static void main(String[] args)

{

String first="",second="";

Scanner sc=new Scanner(System.in);

System.out.println("String Operation");

System.out.println();

System.out.print("Enter the first Sting: ");

first=sc.nextLine();

System.out.print("Enter the second Sting: ");

second=sc.nextLine();

System.out.println("The strings are: "+first+" , "+second); System.out.println("The length of the first string is :"+first.length());

System.out.println("The length of the second string is :"+second.length());

System.out.println("The concatenation of first and second string is :"+first.concat(" "+second));

System.out.println("The first character of " +first+" is: "+first.charAt(0));

System.out.println("The uppercase of " +first+" is: "+first.toUpperCase());

System.out.println("The lowercase of " +first+" is: "+first.toLowerCase());

System.out.print("Enter the occurance of a character in "+first+" : ");

String str=sc.next();

char c=str.charAt(0);

System.out.println("The "+c+" occurs at position " + first.indexOf(c)+ " in " + first); System.out.println("The substring of "+first+" starting from index 3 and ending at 6 is: " +

first.substring(3,7));

System.out.println("Replacing 'a' with 'o' in "+first+" is: "+first.replace('a','o'));

boolean check=first.equals(second);

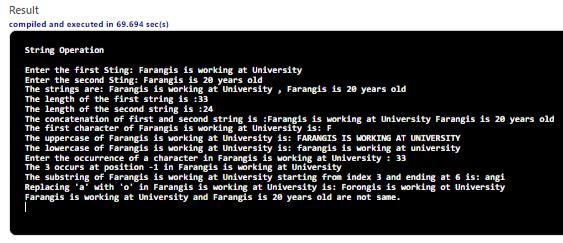
if(!check)

System.out.println(first + " and " + second + " are not same."); else

System.out.println(first + " and " + second + " are same.");

}

}



18. Write an applet program that displays a simple message

package Program18;

import java.applet.Applet;

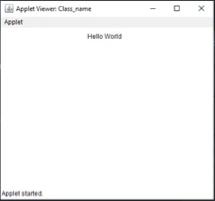
import java.awt.\*;

public class Class\_name extends Applet { public void init() {

Label labelName = new Label("Hello World"); add(labelName);

}

}



19. Write a program to create student report using applet, read the input using

text boxes and display the o/p using buttons.

import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

public class KeyBoard extends Applet implements ActionListener {

TextField name, addr, roll;

Label n,a,r,s;

Button b;

public void init(){

n= new Label("Student name: \n");

name= new TextField(20);

a= new Label("Address: \n");

addr= new TextField(20);

r= new Label("Enrollment No: \n"); roll = new TextField(20);

s= new Label("Registration Form\n");

add(s);

add(n);

add(name);

add(a);

add(addr);

add(r);

add(roll);

b= new Button("Register");

b.addActionListener(this);

add(b);

}

public void actionPerformed(ActionEvent ae){ if(ae.getSource()==b) {

System.*out*.println("Registered Successfully!");

}

else{

System.*out*.println("Failed registration");

}

}

public void paint(Graphics g){ g.drawString("Welcome to Registration Form!\n",

50,50);

}

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>KeyBoard Events</title>

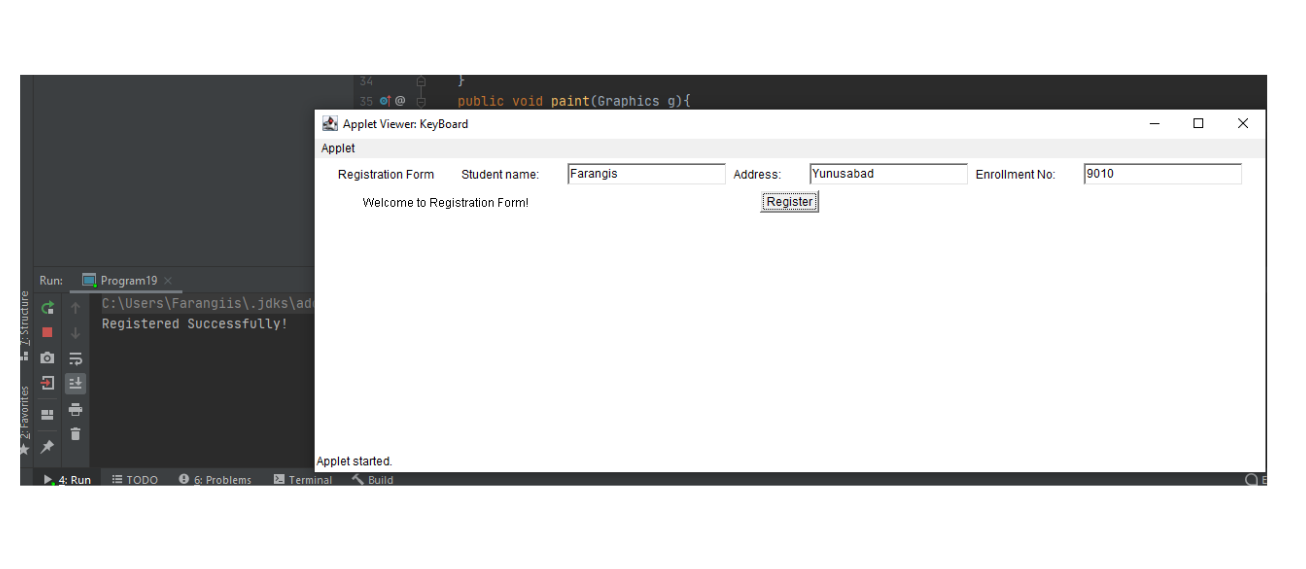
</head>

<body>

<Applet code=”KeyBoard.class” height=100 width=700></applet>

</body>

</html>



1. Write an Applet displaying line, rectangle, rounded rectangle, filled rectangle, filled rounded rectangle, circle, ellipse, arc, filled arc and polygon, all in different colors.

package Program20;

import java.awt.\*;

import javax.swing.\*;

public class Figures extends JApplet { public void init()

{

setSize(400, 400);

repaint();

}

public void paint(Graphics g) { g.setColor(Color.*red*); g.drawLine(100, 100, 150, 100);

g.setColor(Color.*blue*);

g.drawRect(70, 100, 30, 30);

g.setColor(Color.*yellow*);

g.fillRect(170, 100, 30, 30);

g.setColor(Color.*black*);

g.drawOval(70, 200, 30, 30);

g.setColor(Color.*pink*); g.fillOval(170, 200, 30, 30); g.setColor(Color.*CYAN*); g.drawArc(90, 150, 30, 30, 30, 270); g.setColor(Color.*GREEN*); g.fillArc(270, 150, 30, 30, 0, 180);

int xa[] = {120, 125, 150, 150, 200, 200}; int ya[] = {200, 100, 100, 175, 175, 200}; g.setColor(Color.*magenta*); g.drawPolygon(xa, ya, 6);

int cx, cy;

* center of th ellipse cx = 150;

cy = 175;

* major and minor axis

double A = 75, B = 50, px = 0, py = 0;

* set color g.setColor(Color.*ORANGE*);
* draw the ellipse

for (int i = 0; i <= 360; i++) { double x, y;

1. = A \* Math.*sin*(Math.*toRadians*(i));
2. = B \* Math.*cos*(Math.*toRadians*(i));

if (i != 0) {

// draw a line joining previous and new

point .

g.drawLine((int) px + cx, (int) py + cy, (int) x + cx, (int) y + cy);

}

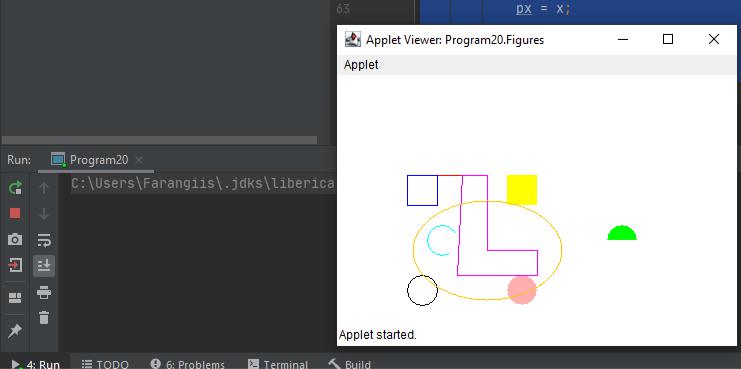
* store the previous points px = x;

py = y;

}

}

}



**Program: 21**

import java.awt.\*;

import java.awt.event.\*;

import java.applet.\*;

public class Movingball extends Applet implements ActionListener, Runnable {

int d=100;

Thread t;

Button btnStart,btnStop,btnPause,btnResume;

int i,m,j;

TextField txtSpeed;

Label lblSpeed;

public void init()

{

lblSpeed=new Label("Enter speed in ms");

txtSpeed=new TextField(5);

btnStart=new Button("Start");

btnStop=new Button("Stop");

btnPause=new Button("Pause");

btnResume=new Button("Resume");

btnStart.addActionListener(this);

btnStop.addActionListener(this);

btnPause.addActionListener(this);

btnResume.addActionListener(this);

add(lblSpeed);

add(txtSpeed);

add(btnStart);

add(btnStop);

add(btnPause);

add(btnResume);

}

public void actionPerformed(ActionEvent ae)

{

if((ae.getSource()==btnStart)&&(t==null))

{

t = new Thread(this);

t.start();

}

else if((ae.getSource()==btnStop)&&(t != null))

{

t.stop();

t=null;

}

else if((ae.getSource()==btnPause)&&(t != null))

{

t.suspend();

}

else if((ae.getSource()==btnResume)&&(t != null))

{

t.resume();

}

}

public void run()

{

m=((int)(Math.*random*()\*1000))%700;

for(j=500;;j--)

{

if(j<-1000)

{

m=((int)(Math.*random*()\*1000))%700; j=500;

}

repaint();

try

{

Thread.*sleep*(Integer.*parseInt*(txtSpeed.getText()));

}

catch(Exception e){}

}

}

public void paint(Graphics g)

{

for(int k=0,i=d;i>0;i--,k++)

{

g.setColor(new Color(100+k,100,100+k)); g.drawOval((m+200)%700+k/2,k/2+j,i,i);

}

for(int k=0,i=d;i>0;i--,k++)

{

g.setColor(new Color(100+k,200,80+k)); g.drawOval(k/2+(m%700),k/2+j,i,i);

}

for(int k=0,i=d;i>0;i--,k++)

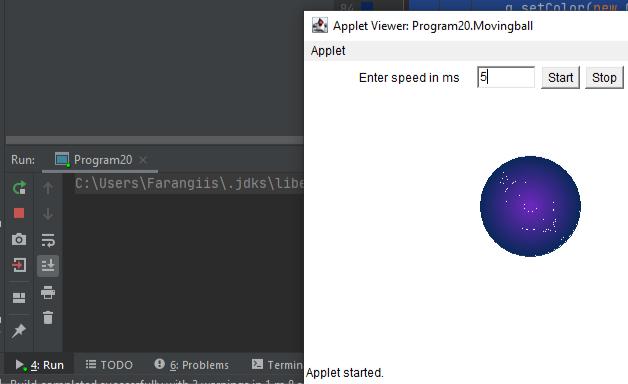
{

g.setColor(new Color(10+k,40,90+k)); g.drawOval(k/2+(m+500)%700,k/2+j,i,i);

}

}

}



**Program : 22**

1. Write an Applet that displays a counter in the middle of applet.

import java.applet.Applet;

import java.awt.Dimension;

import java.awt.Font;

import java.awt.FontMetrics;

import java.awt.Graphics;

/\*

<applet code = "Count" width = 500 height = 300>

</applet>

\*/

public class Count extends Applet implements Runnable { int counter;

Thread t;

public void init(){

counter = 0;

t = new Thread(this);

t.start();

}

public void run(){

try{

while(true){

repaint();

Thread.*sleep*(1000);

++counter;

}

}

catch(Exception e){

}

}

public void paint(Graphics g){

g.setFont(new Font("Serif",Font.*BOLD*,30));

FontMetrics fm = g.getFontMetrics();

String s = "" + counter;

Dimension d = getSize();

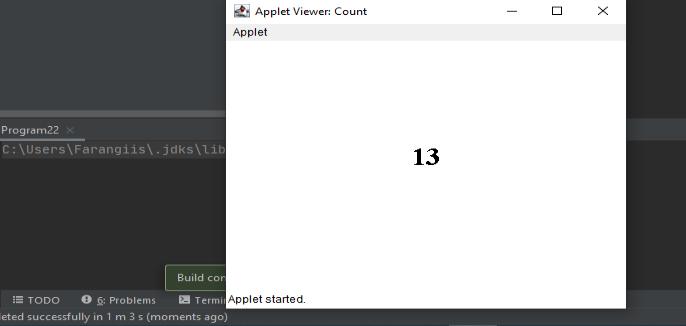
int x = d.width/2 - fm.stringWidth(s)/2;

int y = d.height/2;

g.drawString(s,x,y);

}

}



1. Write a java program that handles all mouse events and shows the event name at the center of the window when mouse event is fired(Use Adapter classes).

import java.awt.\*;

import java.awt.event.\*;

import java.applet.\*;

public class Mouse extends Applet implements MouseListener{ private int x,y;

private String str = " ";

public void init()

{

this.addMouseListener (this);

}

public void paint(Graphics g)

{

g.drawString(str,x,y);

}

public void update(Graphics g)

{

paint(g);

}

public void mouseClicked(MouseEvent m)

{

x = m.getX();

y = m.getY();

str = "x =" +x+",y = "+y;

repaint();

}

public void mouseEntered(MouseEvent m)

{

}

public void mouseExited(MouseEvent m)

{

}

public void mousePressed(MouseEvent m)

{

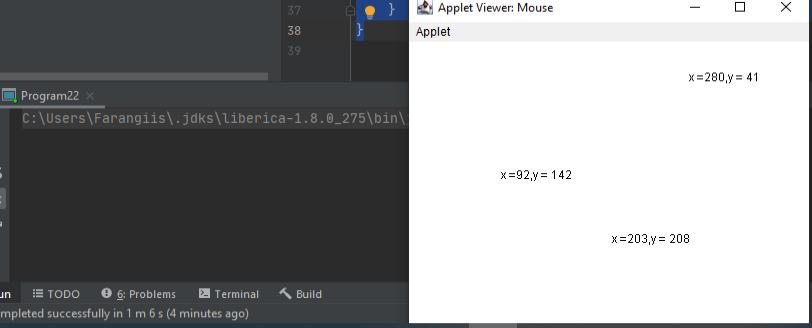
}

public void mouseReleased(MouseEvent m)

{

}

}



1. Write a program to implement keyboard events.

import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

public class KeyBoard extends Applet implements ActionListener {

TextField name,pass;

Label n,p,w;

Button s,c;

public void init() {

n= new Label(" UserName");

name=new TextField(20);

name.setEchoChar('\*');

p= new Label("Password");

pass= new TextField(20);

pass.setEchoChar('$');

name.setBackground(Color.*yellow*);

pass.setBackground(Color.*yellow*);

setBackground(Color.*blue*);

add(n);

add(name);

add(p);

add(pass);

w= new Label ("Please Login !!!");

add(w);

s= new Button("SUBMIT");

add(s);

s.addActionListener(this);

c=new Button("Cancel");

add(c);

}

public void actionPerformed(ActionEvent ae)

{ if(ae.getSource() == s)

{ if (name.getText().equals("Farangis") && pass.getText().equals("qwerty"))

{ w.setText("Made by Farangis");}

else { w.setText("Failed");

}

}

}

public void paint(Graphics g)

{

g.drawString(" welcome to first Applet",50,50);

}

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>KeyBoard Events</title>

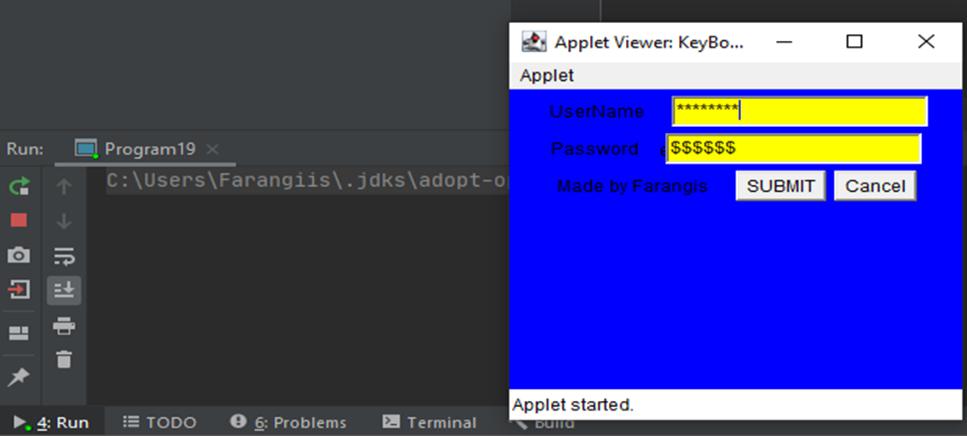
</head>

<body>

<Applet code=”KeyBoard.class” height=100 width=700></applet>

</body>

</html>



1. Write a program to create menu bar and pull down menus.

import java.awt.\*;

import java.awt.event.\*;

public class MenuEx extends Frame implements ActionListener {

Menu states, cities;

public MenuEx()

{

MenuBar mb = new MenuBar();

// begin

with creating menu bar

setMenuBar(mb);

// add menu bar to

frame

states = new Menu("Uzbekistan cities"); //

create menus

cities = new Menu("Uzbekistan Cities");

mb.add(states); // add menus to menu

bar

mb.add(cities);

states.addActionListener(this); // link with ActionListener for event handling

cities.addActionListener(this); states.add(new MenuItem("Tashkent")); states.add(new MenuItem("Bukhara")); states.add(new MenuItem("Samarkand")); states.addSeparator(); states.add(new MenuItem("Navoi")); states.add(new MenuItem("Xiva")); states.add(new MenuItem("Jizzakh")); cities.add(new MenuItem("Zarafshan")); cities.add(new MenuItem("Namangan")); cities.add(new MenuItem("Kokand")); cities.addSeparator();

cities.add(new MenuItem("Andijan"));

cities.add(new MenuItem("Sirdarya"));

cities.add(new MenuItem("Karshi"));

setTitle("Simple Menu Program");

creation methods

setSize(300, 300);

setVisible(true);

// frame

}

public void actionPerformed(ActionEvent e)

{

String str = e.getActionCommand(); menu item selected by the user

// know the

System.*out*.println("You selected " + str);

}

public static void main(String args[])

{

new MenuEx();

}

}

