**Termwork - 2**

**Problem definition:**

**Create an appropriate GUI which allows the user to select an item from the menu. When draw menu item is selected, draws the selected shape(Allowed shapes are: Circle,Rectangle &amp; Triangle) in drawing area by getting appropriate dimensions of the selected shape from the user through keyboard entry, using the concept of ABSTRACT CLASS, INHERITANCE and DYNAMIC DISPATCH features of JAVA Programming Language. The code must be robust for all possible erroneous input conditions, displaying appropriate error messages in message window specially designed for them.**

**CODE:**

**Tw2\_check.java**

import java.applet.\*;

import java.awt.\*;

import java.awt.event.\*;

public class tw2\_check extends Applet implements ItemListener,ActionListener {

Checkbox circle,square,rectangle,triangle;

Panel mainPanel,cb,tb,tb1;

CardLayout cardL0;

TextField a,b,c,d;

String msg,a1,b1,c1,d1;

TextField p,q,r,s,t,u;

String p1,q1,r1,s1,t1,u1;

int x1,x2,x3,y1,y2,y3;

int a2,b2,c2,d2;

boolean circ,squr,rect,tri;

public void init() {

// TODO start asynchronous download of heavy resources

cardL0=new CardLayout();

mainPanel=new Panel();

mainPanel.setLayout(cardL0);

circle=new Checkbox("Circle",null,false);

square=new Checkbox("Square",null,false);

rectangle=new Checkbox("Rectangle",null,false);

triangle=new Checkbox("Triangle",null,false);

cb=new Panel();

cb.add(circle);

cb.add(square);

cb.add(rectangle);

cb.add(triangle);

a=new TextField(10);

b=new TextField(10);

c=new TextField(10);

d=new TextField(10);

p=new TextField(10);

q=new TextField(10);

r=new TextField(10);

s=new TextField(10);

t=new TextField(10);

u=new TextField(10);

tb=new Panel();

tb1=new Panel();

tb.add(a);

tb.add(b);

tb.add(c);

tb.add(d);

tb1.add(p);

tb1.add(q);

tb1.add(r);

tb1.add(s);

tb1.add(t);

tb1.add(u);

mainPanel.add(cb,"checkbox");

mainPanel.add(tb,"TextField");

mainPanel.add(tb1,"TextField");

add(mainPanel);

circle.addItemListener(this);

square.addItemListener(this);

rectangle.addItemListener(this);

triangle.addItemListener(this);

a.addActionListener(this);

b.addActionListener(this);

c.addActionListener(this);

d.addActionListener(this);

p.addActionListener(this);

q.addActionListener(this);

r.addActionListener(this);

s.addActionListener(this);

t.addActionListener(this);

u.addActionListener(this);

}

public void itemStateChanged(ItemEvent ie){

circ=circle.getState();

squr=square.getState();

rect=rectangle.getState();

tri=triangle.getState();

if(circ==true||squr==true||rect==true){

cb.setVisible(false);

tb.setVisible(true);

}

if(tri==true){

cb.setVisible(false);

tb1.setVisible(true);

}

repaint();

}

public void actionPerformed(ActionEvent ae){

if(circ==true || squr==true || rect==true){

a1=a.getText();

b1=b.getText();

c1=c.getText();

d1=d.getText();

a2=Integer.parseInt(a1);

b2=Integer.parseInt(b1);

c2=Integer.parseInt(c1);

d2=Integer.parseInt(d1);

tb.setVisible(false);

}

else{

p1=p.getText();

q1=q.getText();

r1=r.getText();

s1=s.getText();

t1=t.getText();

u1=u.getText();

x1=Integer.parseInt(p1);

x2=Integer.parseInt(q1);

x3=Integer.parseInt(r1);

y1=Integer.parseInt(s1);

y2=Integer.parseInt(t1);

y3=Integer.parseInt(u1);

tb1.setVisible(false);

}

repaint();

}

public void paint(Graphics g){

//Panel draw=new Panel();

if(circ==true){

g.drawOval(a2, b2, c2, d2);

}

else if(squr==true){

g.drawRect(a2, b2, c2, d2);

}

else if(rect==true){

g.drawRect(a2, b2, c2, d2);

}

else if(tri==true){

int x[] = { x1, x2,x3};

int y[] = { y1,y2,y3};

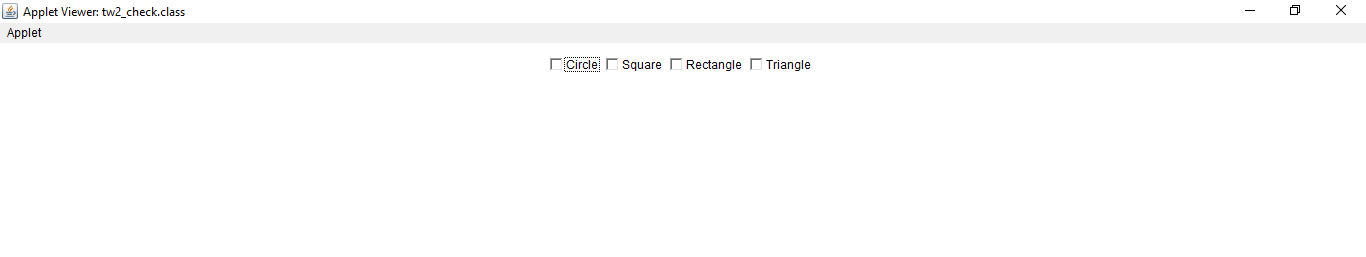
g.drawPolygon (x, y, 3);

}

}

}

**OUTPUT(SCREENSHOT) ::**

****

