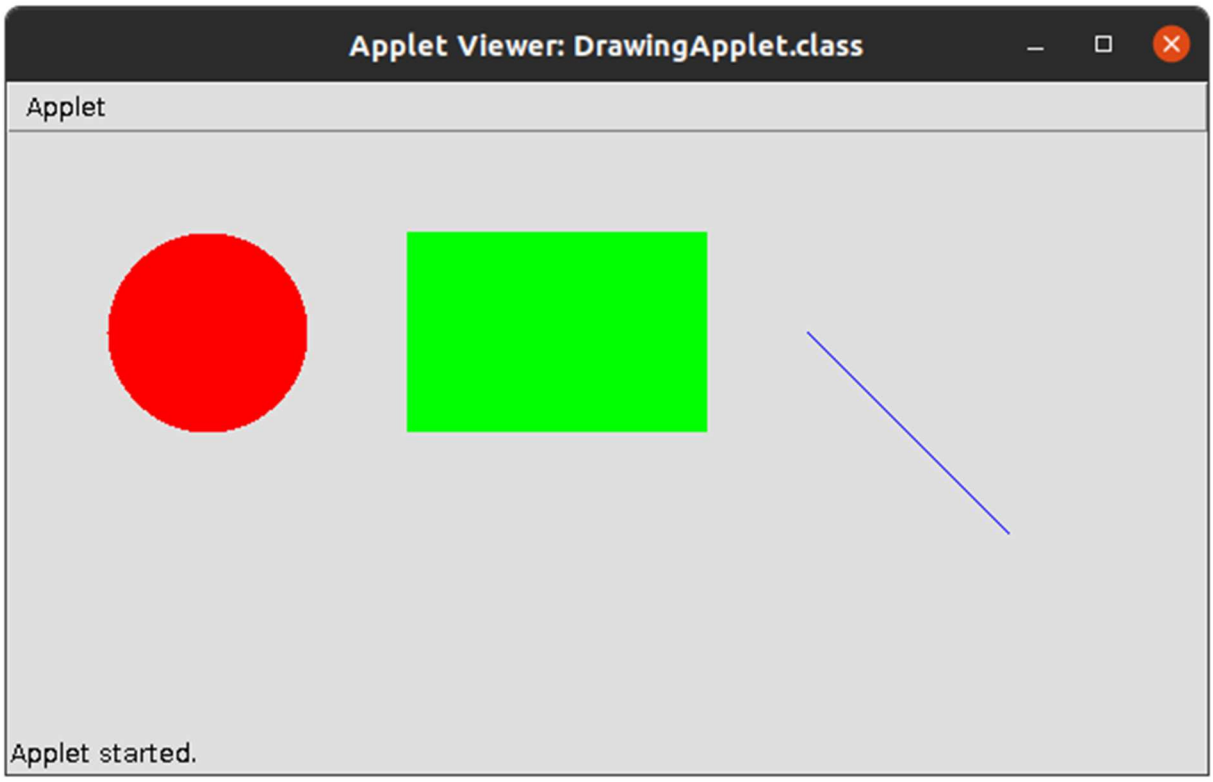
CYCLE- 5

1. Program to draw Circle, Rectangle, Line in Applet.

CODE:

```
import java.applet.Applet;
import java.awt.*;

public class DrawingApplet extends Applet {
    public void paint(Graphics g) {

        g.setColor(Color.red);
        g.fillOval(50, 50, 100, 100);



        g.setColor(Color.green);
        g.fillRect(200, 50, 150, 100);



        g.setColor(Color.blue);
        g.drawLine(400, 100, 500, 200);
    }
}
 /*<applet code="DrawingApplet.class" width="600" height="300"></applet>*/
```

OUTPUT:



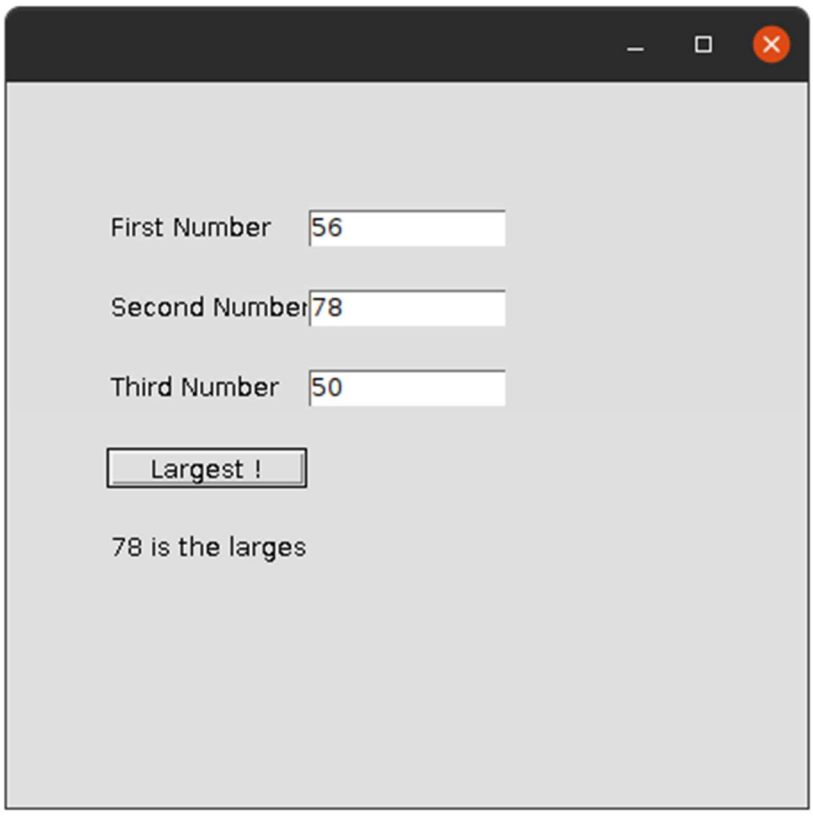
2. Program to find maximum of three numbers using AWT.

CODE:

```
import java.awt.*;
import java.awt.event.*;
public class Largenum implements ActionListener{
   Frame f=new Frame();
   Label l1=new Label("First Number");
   Label l2=new Label("Second Number");
   Label l3=new Label("Third Number");
   Label res=new Label("Result");
   TextField t1=new TextField();
   TextField t2=new TextField();
   TextField t3=new TextField();
   Button b1=new Button("Largest !");
   Largenum(){
   l1.setBounds(50,100,100,20);
   l2.setBounds(50,140,100,20);
   l3.setBounds(50,180,100,20);
   t1.setBounds(150,100,100,20);
   t2.setBounds(150,140,100,20);
   t3.setBounds(150,180,100,20);
   b1.setBounds(50,220,100,20);
   res.setBounds(50,260,100,20);
   f.add(l1);
   f.add(l2);
   f.add(l3);
   f.add(t1);
   f.add(t2);
   f.add(t3);
   f.add(res);
```

```
    f.add(b1);
    b1.addActionListener(this);
    f.setLayout(null);
    f.setVisible(true);
    f.setSize(400,400);
}
public static void main(String[] args){
     new Largenum();
}
public void actionPerformed(ActionEvent e){
if(e.getSource()==b1){
int n1=Integer.parseInt(t1.getText());
int n2=Integer.parseInt(t2.getText());
int n3=Integer.parseInt(t3.getText());
int largeres= (n1 > n2) ? (n1 > n3 ? n1 : n3) : (n2 > n3 ? n2 : n3);
res.setText(String.valueOf(largeres)+" is the largest");
}
}
}
```

OUTPUT:


First Number
56
Second Number
78
Third Number
50
Largest !
78 is the larges

3. Find the percentage of marks obtained by a student in 5 subjects. Display a happy face if he secures above 50% or a sad face if otherwise.

CODE:

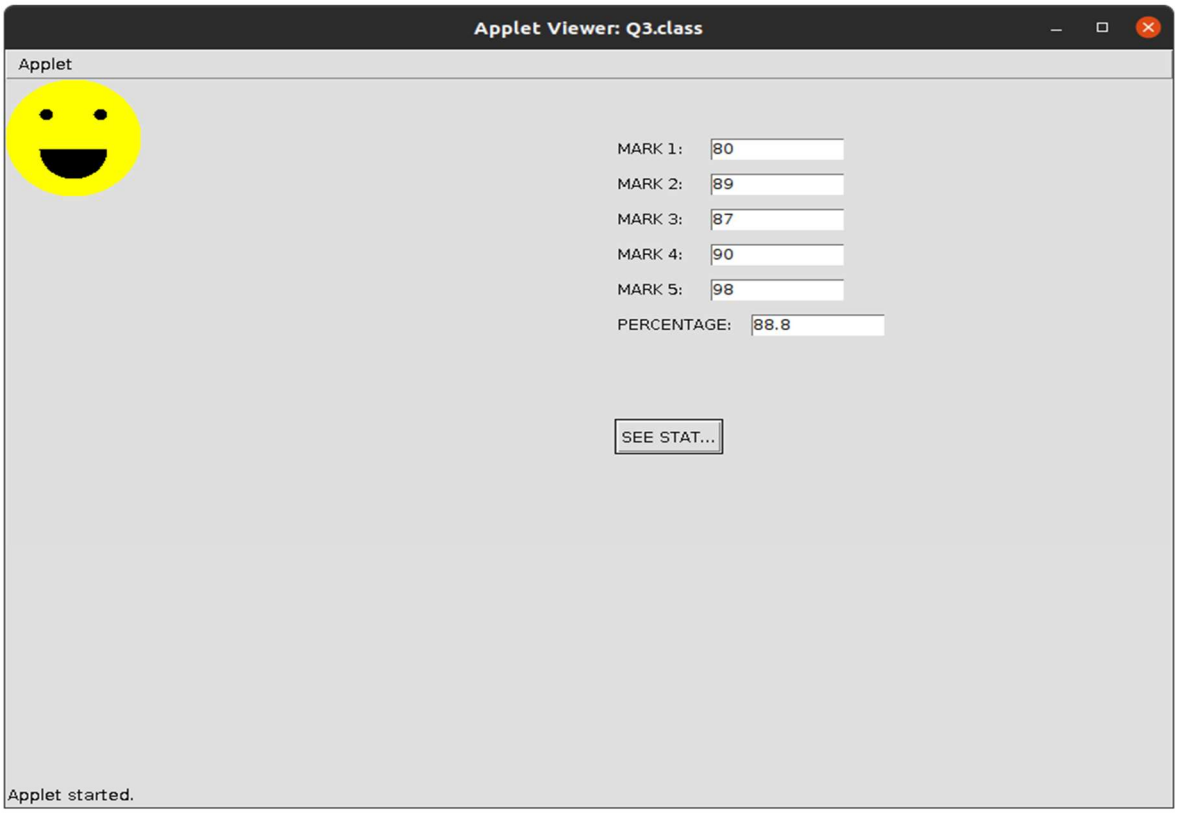
```
import java.applet.*;
import java.awt.*;
import java.awt.Graphics;
import java.awt.event.*;

public class Q3 extends Applet implements ActionListener {
  Label l1,l2,l3,l4,l5,l6;
   TextField t1,t2,t3,t4,t5,t6;
  Button b;
  public void init(){
     l1 = new Label("MARK 1:");
      t1 = new TextField();
      l2 = new Label("MARK 2:");
      t2 = new TextField();
     l3 = new Label("MARK 3:");
      t3 = new TextField();
      l4 = new Label("MARK 4:");
      t4 = new TextField();
      l5 = new Label("MARK 5:");
      t5 = new TextField();
      l6 = new Label("PERCENTAGE:");
      t6 = new TextField();


      b = new Button("SEE STATUS");


      setLayout(null);
```

```
        l1.setBounds(450,50,70,20);
        t1.setBounds(520,50,100,20);
        l2.setBounds(450,80,70,20);
        t2.setBounds(520,80,100,20);
        l3.setBounds(450,110,70,20);
        t3.setBounds(520,110,100,20);
        l4.setBounds(450,140,70,20);
        t4.setBounds(520,140,100,20);
        l5.setBounds(450,170,70,20);
        t5.setBounds(520,170,100,20);
        l6.setBounds(450,200,100,20);
        t6.setBounds(550,200,100,20);

        b.setBounds(450,290,80,30);

       add(l1);
       add(l2);
       add(l3);
       add(l4);
       add(l5);
       add(l6);
       add(t1);
       add(t2);
       add(t3);
       add(t4);
       add(t5);
       add(t6);
       add(b);
       b.addActionListener(this);

   }
```

```
 public void actionPerformed(ActionEvent e){
     float m1, m2,m3, m4,m5,percent;

    m1= Float.parseFloat(t1.getText());
    m2= Float.parseFloat(t2.getText());
     m3= Float.parseFloat(t3.getText());
     m4= Float.parseFloat(t4.getText());
     m5= Float.parseFloat(t5.getText());

     percent=((m1+m2+m3+m4+m5)*100)/500;

    t6.setText(String.valueOf(percent));
     repaint();
 }


public void paint(Graphics g){

     float p;
    p= Float.parseFloat(t6.getText());

     if(p> 50.0) {
         g.setColor(Color.YELLOW);
        g.fillOval(0,0,100,100);
         g.setColor(Color.black);
        g.fillOval(25,25,10,10);
       g.fillOval(65,25,10,10);
         g.setColor(Color.black);
        g.fillArc (25,35,50,50,0,-180);
     }
     else {
         g.setColor(Color.YELLOW);
```

```
            g.fillOval(0,0,100,100);
            g.setColor(Color.black);
            g.fillOval(25,25,10,10);
            g.fillOval(75,25,10,10);
           g.setColor(Color.black);
            g.drawArc(25,35,50,50,0,180);
        }
    }
}
/*<applet code="Q3.class" width="600" height="600">
</applet> */
```

OUTPUT:



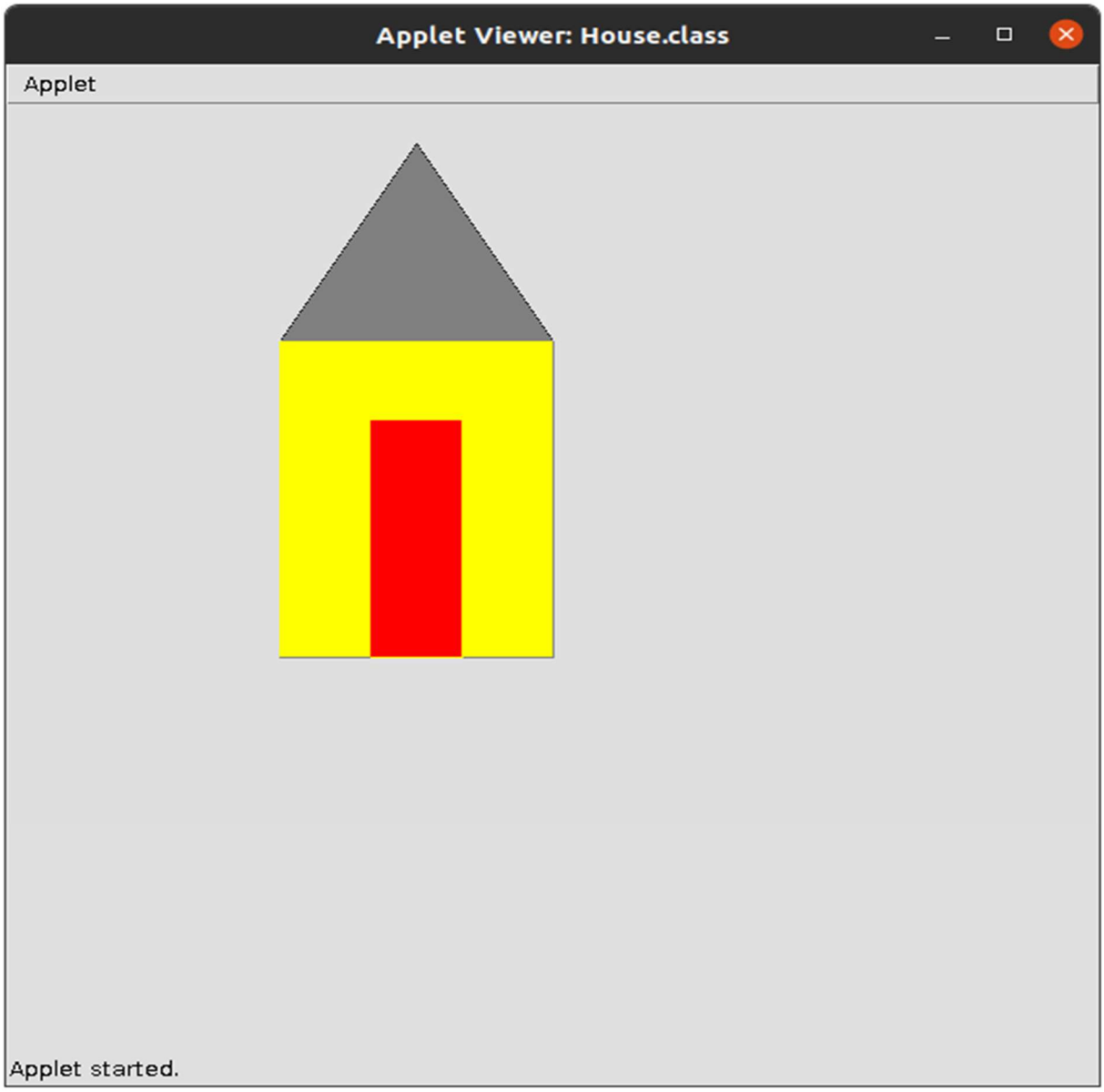
4. Using 2D graphics commands in an Applet, construct a house. On mouse click event, change the color of the door from blue to red.

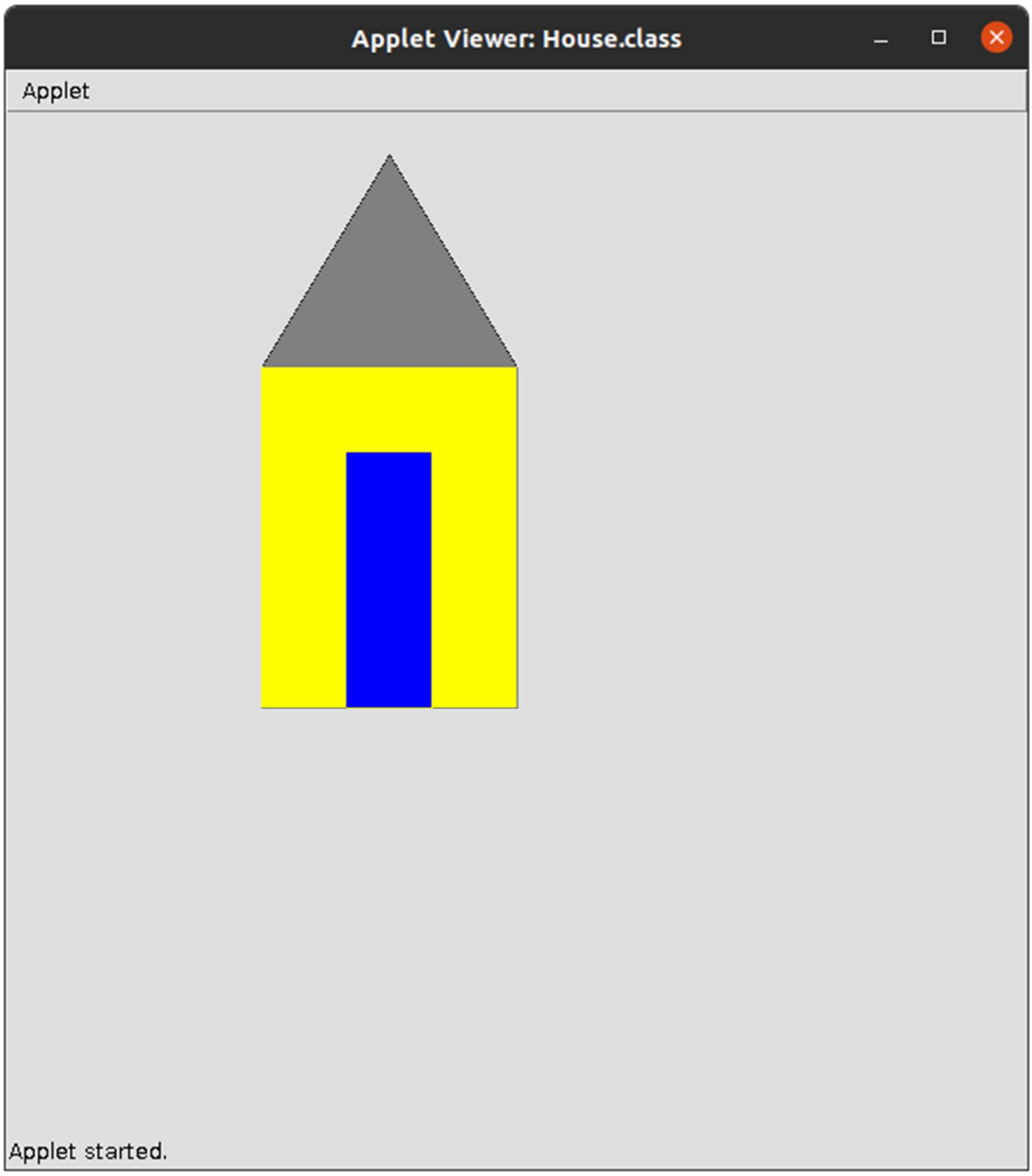
CODE:

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
public class House extends Applet implements MouseListener
{
  int a,b;
   public void init()
   {
      addMouseListener( this);
   }
   public void paint(Graphics g)
   {
      int x[]={150,300,225};
      int y[]={150,150,25};
      g.drawPolygon(x,y,3);
      g.setColor(Color.GRAY);
      g.fillPolygon(x,y,3);

      g.drawRect(150,150,150,200);//House
      g.setColor(Color.YELLOW);
      g.fillRect(150,150,150,200);

      g.drawRect(200,200,50,150);//Door
      g.setColor(Color.blue);
      g.fillRect(200,200,50,150);

      if(a>200 && a<300 && b>200 && b<300)
```

```
        {
            g.setColor(Color.red);
            g.fillRect(200, 200, 50, 150);
        }
    }
    public void mouseClicked(MouseEvent e)
    {

    }
    public void mouseEntered(MouseEvent e)
    {

    }

    @Override
    public void mouseExited(MouseEvent e) {

    }

   public void mousePressed(MouseEvent e)
    {
        a=e.getX();
        b=e.getY();
        repaint();

    }
   public void mouseReleased(MouseEvent e)
    {
 }
}
/*<applet code="House.class" width="600" height="600"></applet>*/
```

OUTPUT:




Applet Viewer: House.class
Applet
Applet started.

5. Implement a simple calculator using AWT components.

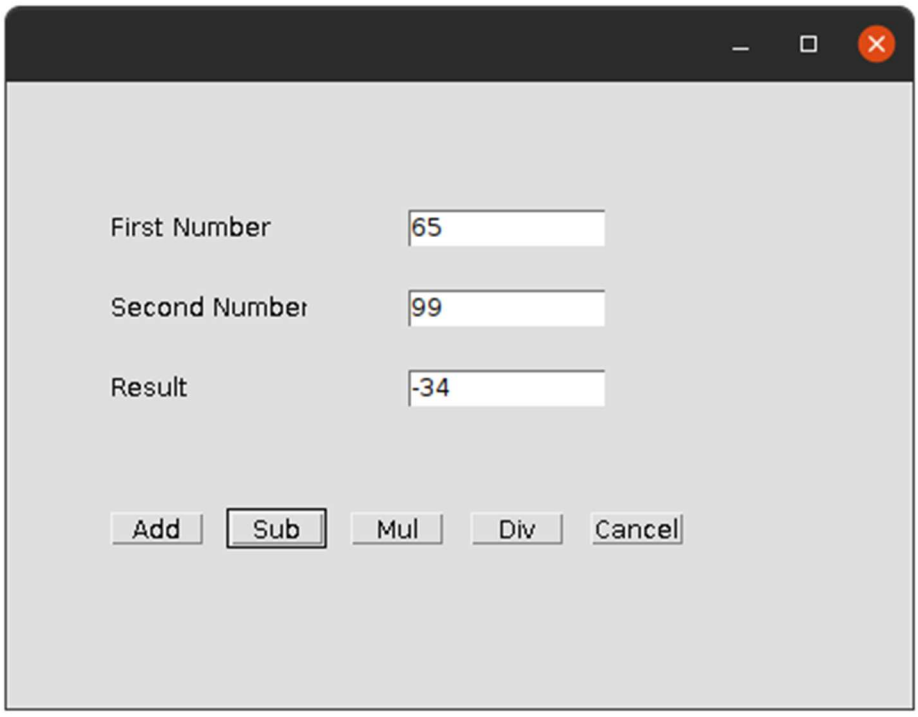
CODE:

```
import java.awt.*;
import java.awt.event.*;
public class Calculator implements ActionListener
{
    Frame f=new Frame();
    Label l1=new Label("First Number");
    Label l2=new Label("Second Number");
    Label l3=new Label("Result");
    TextField t1=new TextField();
    TextField t2=new TextField();
    TextField t3=new TextField();
    Button b1=new Button("Add");
    Button b2=new Button("Sub");
Button b3=new Button("Mul");
Button b4=new Button("Div");
Button b5=new Button("Cancel");
Calculator()
{
l1.setBounds(50,100,100,20);
l2.setBounds(50,140,100,20);
l3.setBounds(50,180,100,20);
t1.setBounds(200,100,100,20);
t2.setBounds(200,140,100,20);
t3.setBounds(200,180,100,20);
b1.setBounds(50,250,50,20);
b2.setBounds(110,250,50,20);
b3.setBounds(170,250,50,20);
b4.setBounds(230,250,50,20);
b5.setBounds(290,250,50,20);
```

```
f.add(l1);
f.add(l2);
f.add(l3);
f.add(t1);
f.add(t2);
f.add(t3);
f.add(b1);
f.add(b2);
f.add(b3);
f.add(b4);
f.add(b5);
b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
b5.addActionListener(this);
f.setLayout(null);
f.setVisible(true);
f.setSize(400,350);
}
public void actionPerformed(ActionEvent e)
{
int n1=Integer.parseInt(t1.getText());
int n2=Integer.parseInt(t2.getText());
if(e.getSource()==b1)
{
t3.setText(String.valueOf(n1+n2));
}
if(e.getSource()==b2)
{
t3.setText(String.valueOf(n1-n2));
```

```
}
if(e.getSource()==b3)
{
t3.setText(String.valueOf(n1*n2));
}
if(e.getSource()==b4)
{
t3.setText(String.valueOf(n1/n2));
}
if(e.getSource()==b5)
{
System.exit(0);
}
}
public static void main(String...s)
{
new Calculator();
}
}
/*<applet code="Calculator.class" width="600" height="600">
</applet>*/
```

OUTPUT:


First Number
65
Second Number
99
Result
-34
Add
Sub
Mul
Div
Cancel

6. Develop a program that has a Choice component which contains the names of shapes such as rectangle, triangle, square and circle. Draw the corresponding shapes for given parameters as per user's choice.

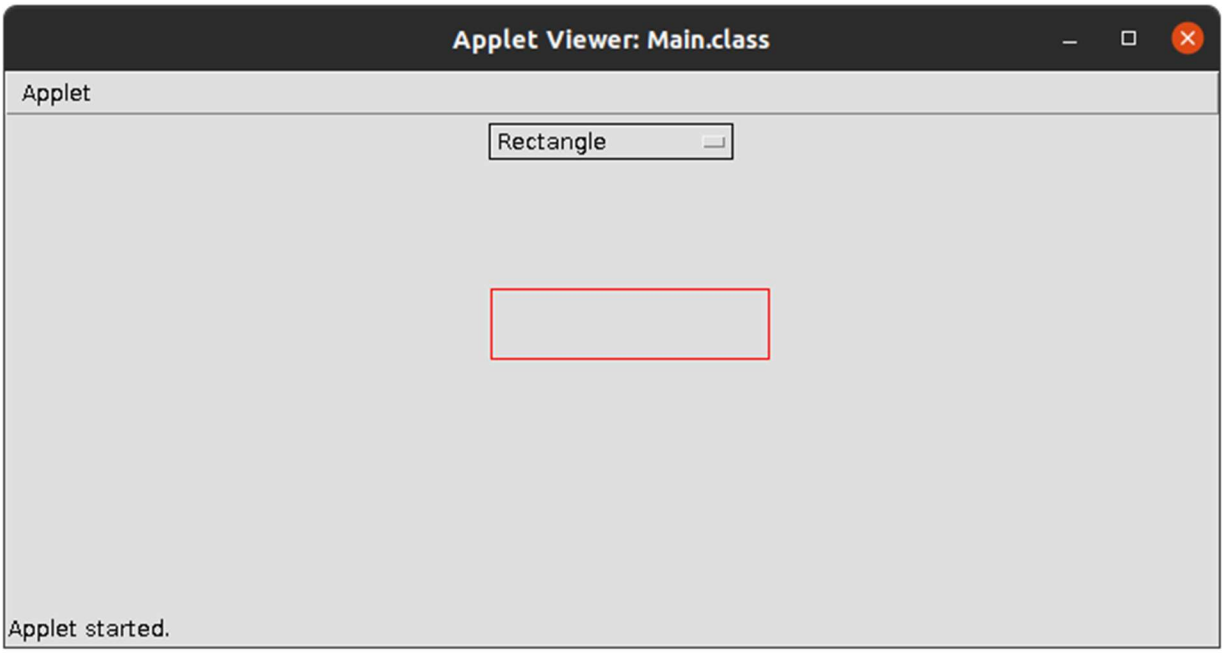
CODE:

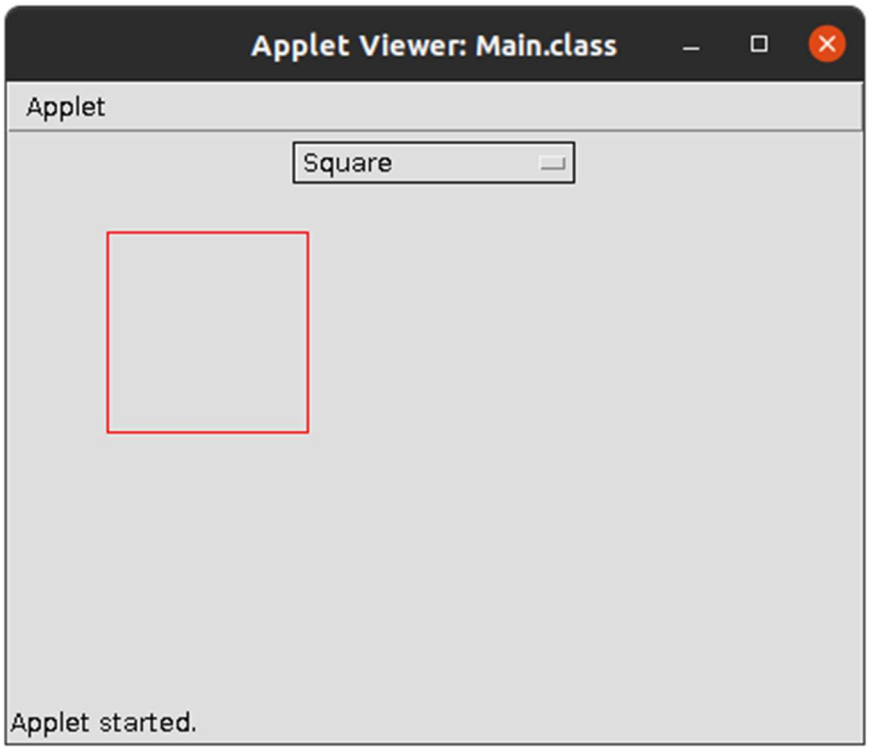
```
import java.applet.*;
import java.awt.*;
import java.awt.Graphics;
import java.awt.event.*;
public class Main extends Applet implements ItemListener
  {
    Choice figure = new Choice();
    int Select;
    public void init()
    {

     figure.addItem("Select your choice");
     figure.addItem("Rectangle");
      figure.addItem("Square");
      figure.addItem("Circle");
      figure.addItem("Triangle");
      add(figure);
      figure.addItemListener(this);



    }
   public void itemStateChanged (ItemEvent e)
  {

     Select = figure.getSelectedIndex();
     repaint();
```

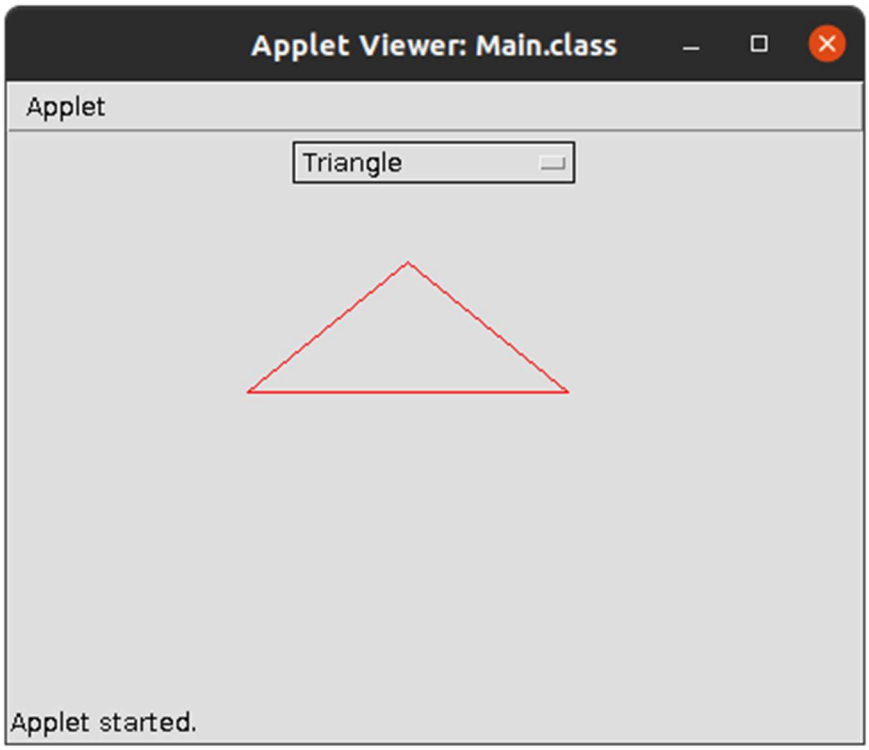
```
    }

 public void paint(Graphics g)
         {

g.setColor(Color.red);
        super.paint(g);

        if (Select == 1)
        {

           g.drawRect(280, 100, 160,40);
        }
        if (Select == 2)
        {
          g.drawRect(50,50,100,100);
        }
       if (Select == 3)
        {

            g.drawOval(150,150,100,100);
       }
        if (Select ==4)
        {
            g.drawLine(120, 130, 280, 130);
            g.drawLine(120, 130, 200, 65);
            g.drawLine(200, 65, 280, 130);
        }
         }
}
```

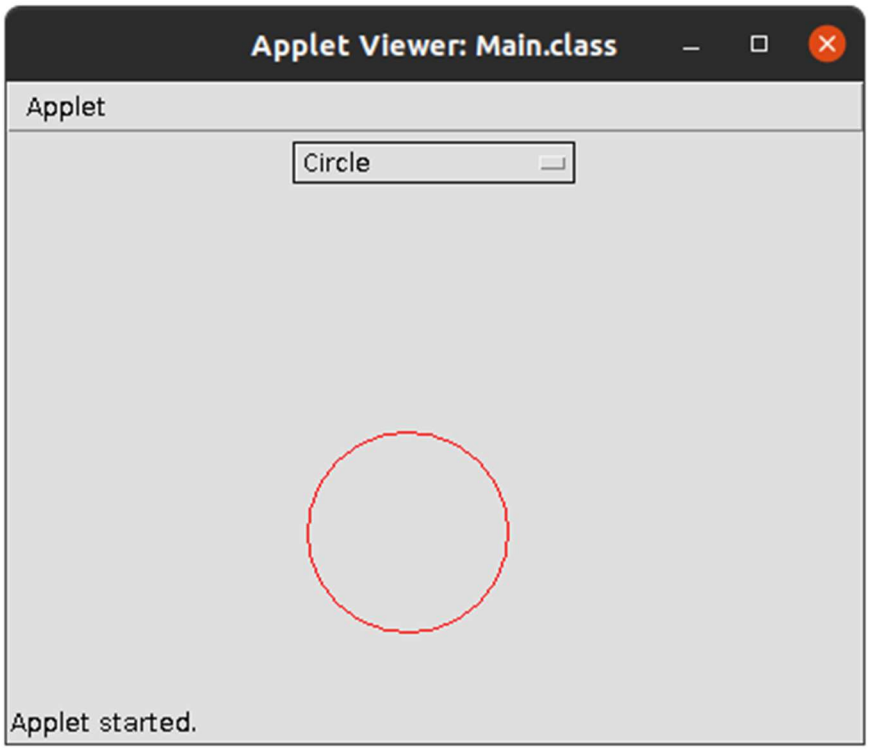
```
/*<applet code="Main.class" width="600" height="600">
</applet>*/
```

OUTPUT:




Applet Viewer: Main.class
Applet
Square
Applet started.


Applet Viewer: Main.class
Applet
Triangle
Applet started.


Applet Viewer: Main.class
Applet
Circle
Applet started.

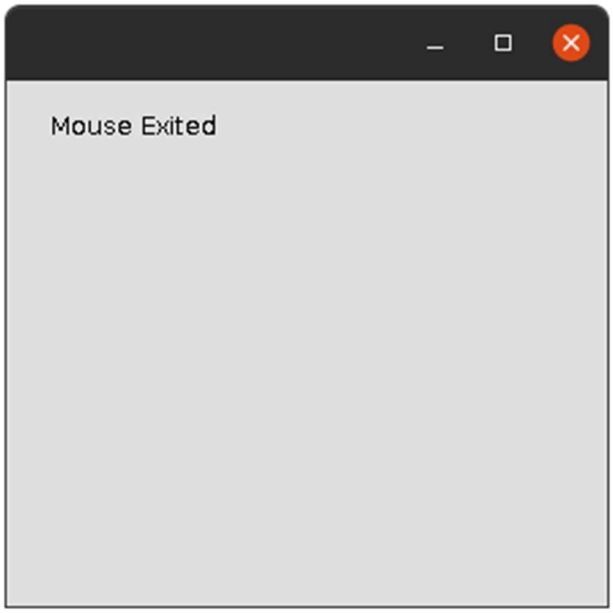
7. Develop a program to handle all mouse events and window events.

CODE:

```
import java.awt.*;
import java.awt.event.*;
public class Mouseevents extends Frame implements MouseListener{
Label l;
Mouseevents(){
addMouseListener(this);
l=new Label();
l.setBounds(20,50,100,20);
add(l);
setSize(300,300);
setLayout(null);
setVisible(true);
}
public void mouseClicked(MouseEvent e) {
l.setText("Mouse Clicked");
}
public void mouseEntered(MouseEvent e) {
l.setText("Mouse Entered");
}
public void mouseExited(MouseEvent e) {
l.setText("Mouse Exited");
}
public void mousePressed(MouseEvent e) {
l.setText("Mouse Pressed");
}
public void mouseReleased(MouseEvent e) {
l.setText("Mouse Released");
}
public static void main(String[] args) {
```

```
new Mouseevents();
}
}
```

OUTPUT:



8. Develop a program to handle Key events.

CODE:

```
import java.awt.FlowLayout;
import java.awt.Frame;
import java.awt.Label;
import java.awt.TextField;
import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;
public class KE implements KeyListener
{
Label lb1, lbl2, lb;
TextField tf1;
Frame fr;
String s;
KE()
{
fr = new Frame("KeyEventListener Example");
lb1= new Label(" Key Events will be displayed based on the actions",
Label.CENTER);
lbl2= new Label();
lb= new Label();
tf1 = new TextField(20);
fr.setLayout(new FlowLayout());
fr.add(lb1);
fr.add(tf1);
fr.add(lbl2);
tf1.addKeyListener(this);
fr.setSize(460,250);
fr.setVisible(true);
}
public void keyPressed(KeyEvent ev)
```

```
{

lbl2.setText(" Key pressed");
}
public void keyReleased(KeyEvent ev)
{
lbl2.setText("Released");
}
public void keyTyped(KeyEvent ev)
{
lbl2.setText("Key is typed");
fr.setVisible(true);
}
public static void main(String[] args)
{
new KE();
}
}
```

OUTPUT: