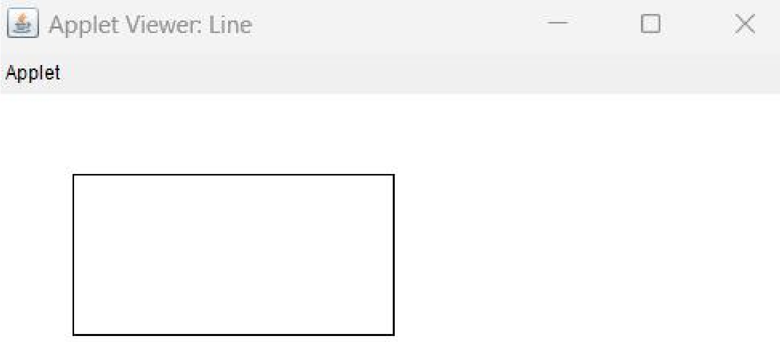
Q1. WAP Java draw a rectangle using line.

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

public class Line extends Applet {

   public void paint(Graphics g) {
      // Set the color of the Line to red
      g.setColor(Color.BLACK);

      // Draw the Line using lines
      g.drawLine(50, 50, 250, 50);
      g.drawLine(50, 50, 50, 150);
      g.drawLine(50, 150, 250, 150);
      g.drawLine(250, 50, 250, 150);
   }

   public static void main(String[] args) {
      Line applet = new Line();
      Frame frame = new Frame("Line Applet");
      frame.add(applet);
      frame.setSize(300, 200);
      frame.setVisible(true);
   }
}
/*
<html>
<applet code=Line height=500 width=500>
</applet>
</html>
 */
```

OUTPUT:

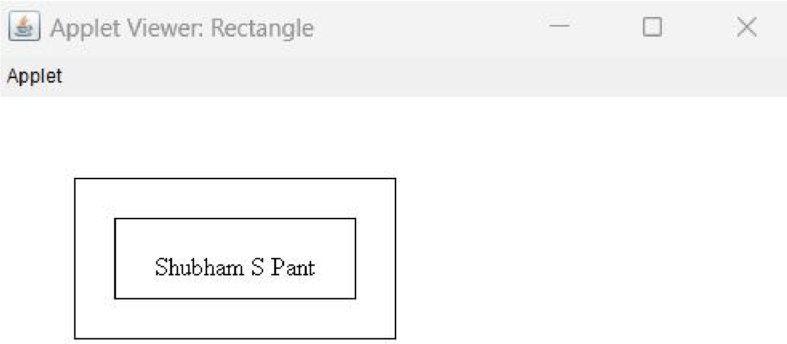


Q2. WAP Java draw a rectangle using line. Draw another rectangle inside the 1st rectangle and also display your name mid position of inside rectangle.

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

public class Rectangle extends Applet {
  public void paint(Graphics g) {
      int outerX = 50;
      int outerY = 50;
      int outerWidth = 200;
      int outerHeight = 100;
      g.setColor(Color.BLACK);
      g.drawRect(outerX, outerY, outerWidth, outerHeight);
      int innerX = outerX + 25;
      int innerY = outerY + 25;
      int innerWidth = outerWidth - 50;
      int innerHeight = outerHeight - 50;
      g.drawRect(innerX, innerY, innerWidth, innerHeight);
      g.setFont(new Font("TimesRoman", Font.PLAIN, 16));
      g.setColor(Color.BLUE);
      String text = "Aaryak Prasad";
      int textWidth = g.getFontMetrics().stringWidth(text);
      int textHeight = g.getFontMetrics().getHeight();
      int textX = innerX + (innerWidth - textWidth) / 2;
      int textY = innerY + (innerHeight + textHeight) / 2;
      g.drawString(text, textX, textY);
   }
   public static void main(String[] args) {
      Rectangle applet = new Rectangle();
      Frame frame = new Frame("Rectangle Drawing Applet");
      frame.add(applet);
      frame.setSize(300, 200);
      frame.setVisible(true);
      applet.init();
      applet.start();
```

```
        frame.addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent event) {
                applet.stop();
                applet.destroy();
                System.exit(0);
            }
        });
    }
}

/*
<html>
<applet code=Rectangle height=500 width=500>
</applet>
</html>
 */
```

OUTPUT:



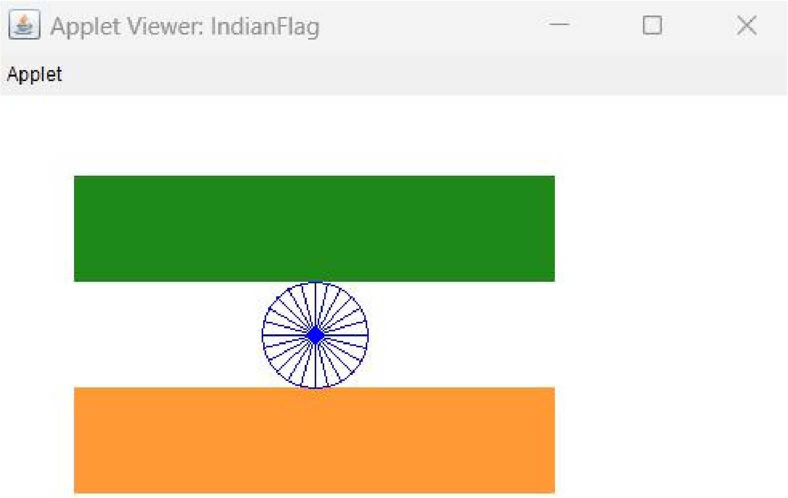
Shubham S Pant

Q3. WAP Java draw Indian Flag.

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

public class IndianFlag extends Applet {
   public void paint(Graphics g) {
      setBackground(Color.WHITE);
      g.setColor(new Color(30, 136, 24));
      g.fillRect(50, 50, 300, 66);
      g.setColor(Color.WHITE);
      g.fillRect(50, 116, 300, 66);
      g.setColor(new Color(255, 153, 51));
      g.fillRect(50, 182, 300, 66);
      g.setColor(Color.BLUE);
      int xCenter = 200;
      int yCenter = 149;
      int radius = 33;
      for (int i = 0; i < 24; i++) {
         double angle = Math.PI / 12 * i;
         int x = (int) (xCenter + radius * Math.cos(angle));
         int y = (int) (yCenter + radius * Math.sin(angle));
         g.drawLine(xCenter, yCenter, x, y);
      }
      g.setColor(Color.BLUE);
      int xCircle = 200 - radius;
      int yCircle = 149 - radius;
      int diameter = radius * 2;
      g.drawOval(xCircle, yCircle, diameter, diameter);
   }

   public static void main(String[] args) {
      IndianFlag applet = new IndianFlag();
      Frame frame = new Frame("Indian Flag Applet");
      frame.add(applet);
      frame.setSize(400, 400);
      frame.setVisible(true);
      applet.init();
```

```
        applet.start();
        frame.addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent event) {
                applet.stop();
                applet.destroy();
                System.exit(0);
            }
        });
    }
}

/*
<html>
<applet code=IndianFlag height=500 width=500>
</applet>
</html>
 */
```

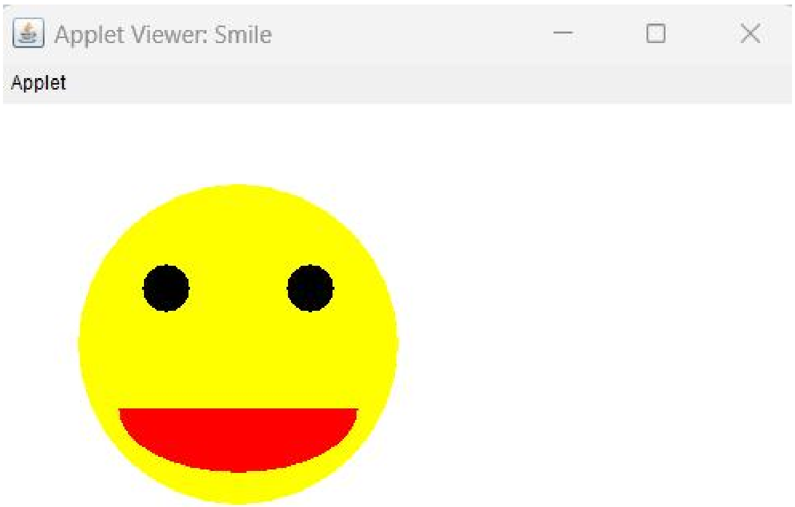
OUTPUT:



Q4. WAP Java draw a smiling face.

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

public class Smile extends Applet {
   public void paint(Graphics g) {
      setBackground(Color.WHITE);
      g.setColor(Color.YELLOW);
      g.fillOval(50, 50, 200, 200);
      g.setColor(Color.BLACK);
      g.fillOval(90, 100, 30, 30);
      g.fillOval(180, 100, 30, 30);
      g.setColor(Color.RED);
      g.fillArc(75, 150, 150, 80, 180, 180);
   }
   public static void main(String[] args) {
      Smile applet = new Smile();
      Frame frame = new Frame("Smile Face Applet");
      frame.add(applet);
      frame.setSize(300, 300);
      frame.setVisible(true);
      applet.init();
      applet.start();
      frame.addWindowListener(new WindowAdapter() {
         public void windowClosing(WindowEvent event) {
            applet.stop();
            applet.destroy();
            System.exit(0);
         }
      });
   }
}
/*
<html>
<applet code=Smile height=500 width=500>
</applet>
</html>
 */
```

OUTPUT:

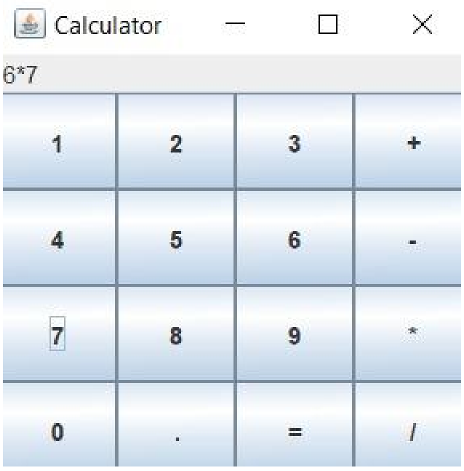


Q5. WAP Java to make a calculator.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Calculator extends JFrame implements ActionListener {

   private JTextField displayField;
   private JButton[] buttons;
   private String[] buttonLabels = { "1", "2", "3", "+", "4", "5", "6", "-",
"7", "8", "9", "*", "0", ".", "=", "/" };
   private double currentTotal = 0.0;
   private String currentOperator = "";

   public Calculator() {
      super("Calculator");
      setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      setSize(250, 250);
      setLayout(new BorderLayout());
      displayField = new JTextField();
      displayField.setEditable(false);
      add(displayField, BorderLayout.NORTH);
      JPanel buttonPanel = new JPanel();
      buttonPanel.setLayout(new GridLayout(4, 4));
      buttons = new JButton[buttonLabels.length];
      for (int i = 0; i < buttonLabels.length; i++) {
         buttons[i] = new JButton(buttonLabels[i]);
         buttons[i].addActionListener(this);
         buttonPanel.add(buttons[i]);
      }
      add(buttonPanel, BorderLayout.CENTER);
      setVisible(true);
   }
   public void actionPerformed(ActionEvent e) {
      String buttonText = ((JButton) e.getSource()).getText();
      if (buttonText.equals("+")) {
         currentTotal = Double.parseDouble(displayField.getText());
         currentOperator = "+";
         displayField.setText("");
      } else if (buttonText.equals("-")) {
         currentTotal = Double.parseDouble(displayField.getText());
         currentOperator = "-";
```

```
            displayField.setText("");
        } else if (buttonText.equals("*")) {
            currentTotal = Double.parseDouble(displayField.getText());
            currentOperator = "*";
            displayField.setText("");
        } else if (buttonText.equals("/")) {
            currentTotal = Double.parseDouble(displayField.getText());
            currentOperator = "/";
            displayField.setText("");
        } else if (buttonText.equals("=")) {
            if (currentOperator.equals("+")) {
                currentTotal += Double.parseDouble(displayField.getText());
            } else if (currentOperator.equals("-")) {
                currentTotal -= Double.parseDouble(displayField.getText());
            } else if (currentOperator.equals("*")) {
                currentTotal *= Double.parseDouble(displayField.getText());
            } else if (currentOperator.equals("/")) {
                currentTotal /= Double.parseDouble(displayField.getText());
            }
            displayField.setText("" + currentTotal);
        } else {
            displayField.setText(displayField.getText() + buttonText);
        }
    }

    public static void main(String[] args) {
        new Calculator();
    }
}
```

OUTPUT:



Calculator
6*7
1
2
3
+
4
5
6
-
7
8
9
*
0
.
=
/