

```

import java.awt.Dimension;
import java.awt.Graphics;
import java.awt.Color;
import javax.swing.JPanel;
import javax.swing.JFrame;
import java.util.Random;

public class Stars extends JPanel {

    // Unique version ID for this class to ensure saved objects
    can be loaded safely
    private static final long serialVersionUID = 1L;

    // Initial width of height of the window
    private static int width = 1000;
    private static int height = 650;

    // main method to launch the program as a standalone
    application - no need to modify
    public static void main(String[] args) {
        Stars panel = new Stars();
        panel.setPreferredSize(new Dimension(width,
height)); // content size window dimensions

        JFrame frame = new JFrame("Stars"); // Title of frame
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.add(panel);
        frame.pack();
        frame.setVisible(true);
    }

    /**
     * Draws 10 stars (or more) of random sizes in random
    locations. The stars do not have to be fully contained within the
    panel.
     * @param g the Graphics object used for drawing shapes, text,
    and images
     */
    public void drawStars(Graphics g) {
        Random randy = new Random();
        int x = 0;
        int y = 0;
        int s = 6;
        int red = 100;
        int green = 0;
        int blue = 50;
        double prop = 0.382;
        int n = 5;
        Color color = new Color(red,green,blue);
    }
}

```

```

        for(int i = 0; i < 100; i++) {
            x = randy.nextInt(width);
            y = randy.nextInt(height);
            s = randy.nextInt(100) + 10;
            red = randy.nextInt(256);
            green = randy.nextInt(256);
            blue = randy.nextInt(256);
            color = new Color(red,green,blue);
            prop = (double)(randy.nextInt(8000))/10000;
            n = randy.nextInt(20) + 2;
            singleStar(g,x,y,s,color,prop,n);
        }
    }

    public void singleStar(Graphics g, int x, int y, int size,
        Color C, double prop, int n) {
        g.setColor(C);
        int i = 0;
        int [] xp = new int [2 * n];
        int [] yp = new int [2 * n];
        while (i < 2 * n) {
            if (i % 2 == 0) {
                xp[i] = (int)Math.round(x + size *
Math.sin(i * (Math.PI / n)));
                yp[i] = (int)Math.round(y - size *
Math.cos(i * (Math.PI / n)));
            }
            else {
                xp[i] = (int)Math.round(x + prop *
size * Math.sin(i * (Math.PI / n)));
                yp[i] = (int)Math.round(y - prop *
size * Math.cos(i * (Math.PI / n)));
            }
            i++;
        }
        g.fillPolygon(xp, yp, 2 * n);
    }

    /**
     * Overrides JPanel's paintComponent method to perform custom
drawing.
     * @param g the Graphics object used for drawing shapes, text,
and images
     */
    @Override
    protected void paintComponent(Graphics g) {
        super.paintComponent(g); // Clears the panel before
drawing
        drawStars(g);
    }

```

}