

```
1 import java.awt.Dimension;
2
3
4
5
6
7
8
9 public class Stars extends JPanel {
10
11     // Unique version ID for this class to ensure saved objects can be loaded safely
12     private static final long serialVersionUID = 1L;
13
14     // Initial width of height of the window
15     private static int width = 1000;
16     private static int height = 650;
17
18     // main method to launch the program as a standalone application - no need to modify
19     public static void main(String[] args) {
20         Stars panel = new Stars();
21         panel.setPreferredSize(new Dimension(width, height)); // content size window dimensions
22
23         JFrame frame = new JFrame("Stars"); // Title of frame
24         frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
25         frame.add(panel);
26         frame.pack();
27         frame.setVisible(true);
28     }
29
30     /**
31      * Draws 10 stars (or more) of random sizes in random locations. The stars do not have to
32      * be fully contained within the panel.
33      * @param g the Graphics object used for drawing shapes, text, and images
34      */
35     public void drawStars(Graphics g) {
36         Random rand = new Random();
37         for(int d=0; d<10; d++) {
38             int centerX = rand.nextInt(getWidth());
39             int centerY = rand.nextInt(getHeight());
40             int outerRadius = rand.nextInt(130) + 20;
41             int innerRadius = outerRadius/2;
42             Polygon starShape = new Polygon();
43
44             for (int i=0; i<10; i++) {
45                 int useRadius = 0;
46                 double angle = Math.PI/2 + i*Math.PI/5;
47                 if (i%2 == 0) { //if even i, then its an outer point
48                     useRadius = outerRadius;
49                 }
50                 else {
51                     useRadius = innerRadius;
52                 }
53                 int x = (int)(centerX + useRadius * Math.cos(angle));
54                 int y = (int)(centerY - useRadius * Math.sin(angle));
55                 starShape.addPoint(x, y);
56             }
57             g.setColor(new Color(rand.nextInt(256), rand.nextInt(256), rand.nextInt(256)));
58             g.fillPolygon(starShape);
59         }
60     }
61
62     /**
63      * Overrides JPanel's paintComponent method to perform custom drawing.
```

```
63     * @param g the Graphics object used for drawing shapes, text, and images
64     */
65     @Override
66     protected void paintComponent(Graphics g) {
67         super.paintComponent(g); // Clears the panel before drawing
68         drawStars(g);
69     }
70
71 }
72
```