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

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) {

		for (int i = 0; i < 75; i++) {

			int x = (int) (Math.random() * width);
			int y = (int) (Math.random() * height);

			int size = 15 + (int) (Math.random() * 60);

			int[] xPoint = new int[10];
			int[] yPoint = new int[10];

			for (int j = 0; j < 10; j++) {

				double angle = (j * 36) * Math.PI / 180;

				double radius;

				if (j % 2 == 0) {
					radius = size;
				} else {
					radius = size / 2;
				}

				xPoint[j] = x + (int) (radius * Math.cos(angle));
				yPoint[j] = y + (int) (radius * Math.sin(angle));
			}

			g.setColor(
					new Color((int) (Math.random() * 256), (int) (Math.random()
* 256), (int) (Math.random() * 256)));
```

```
                    g.fillPolygon(xPoint, yPoint, 10);

                }

            }

            /**
             * 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);
            }

}
```