

Summary 14 - An extended example from JavaEyes

Let's see how the StopButton works.

```
public class JavaEyesFrame extends JFrame {
    ...
    public JavaEyesFrame() {
        ...
        AnimatedPanel animationPanel = new AnimatedPanel(
            JavaEyesFrame.DEFAULT_FRAME_SIZE.width);
        ...
        StopButton stopButton = new StopButton(
            "Stop the Square", animationPanel);
        ...
        this.add(stopButton);
        ...
        this.add(animationPanel);
    }
    ...
}
```

```
public class StopButton extends JButton
    implements ActionListener {
    private static final Color DEFAULT_COLOR =
        Color.MAGENTA;
    private AnimatedPanel squarePanel;

    public StopButton(String title, AnimatedPanel panel) {
        super(title);

        this.squarePanel = panel;
        this.setBackground(StopButton.DEFAULT_COLOR);
        this.addActionListener(this);
    }

    public void actionPerformed(ActionEvent event) {
        this.squarePanel.stop();
    }
}
```

```
public class AnimatedPanel extends JPanel
    implements Runnable {
    ...

    public void run() {
        ...
        this.squareIsMoving = true;
        while (this.squareIsMoving) {
            ...
            this.angle += 2 * Math.PI *
                AnimatedPanel.ANGLE_DELTA;
            this.repaint();
        }
    }

    @Override
    public void paintComponent(Graphics graphics) {
        super.paintComponent(graphics);

        int x = ...;
        int y = ...;

        graphics.setColor(this.squareColor);
        graphics.fillRect(x, y,
            this.squareSize, this.squareSize);
    }

    public void stop() {
        this.squareIsMoving = false;
    }
}
```

1. The JavaEyesFrame constructs and adds an AnimationPanel and a StopButton. Do you see where the *add* method is defined? How the StopButton has-a AnimatedPanel?
2. Pressing the StartButton (not shown here) causes the *run* method of the AnimatedPanel to run. Do you see what controls how long it runs?
3. Pressing the StopButton causes its *actionPerformed* to run. Do you see how that makes the square stop moving?