

src/Stopwatch.java

```
1  import java.awt.*;
2  import java.awt.event.ActionEvent;
3  import java.awt.event.ActionListener;
4  import javax.swing.*;
5
6  public class Stopwatch extends JFrame implements ActionListener {
7
8      private JLabel timeLabel;
9      private JButton startButton, stopButton, resetButton;
10     private Timer timer;
11
12     private int elapsedTime = 0; // in milliseconds
13     private boolean isRunning = false;
14
15     public Stopwatch() {
16         setTitle("Stopwatch");
17         setSize(300, 200);
18         setLayout(new FlowLayout());
19         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
20         setLocationRelativeTo(null); // center the frame
21
22         // Time Label
23         timeLabel = new JLabel(formatTime(0));
24         timeLabel.setFont(new Font("Verdana", Font.PLAIN, 30));
25         add(timeLabel);
26
27         // Buttons
28         startButton = new JButton("Start");
29         stopButton = new JButton("Stop");
30         resetButton = new JButton("Reset");
31
32         add(startButton);
33         add(stopButton);
34         add(resetButton);
35
36         // Add listeners
37         startButton.addActionListener(this);
38         stopButton.addActionListener(this);
39         resetButton.addActionListener(this);
40
41         // Timer - ticks every 1000ms (1 second)
42         timer = new Timer(1000, e -> {
43             elapsedTime += 1000;
44             timeLabel.setText(formatTime(elapsedTime));
45         });
46     }
```

```
47         setVisible(true);
48     }
49
50     private String formatTime(int milliseconds) {
51         int totalSeconds = milliseconds / 1000;
52         int hours = totalSeconds / 3600;
53         int minutes = (totalSeconds % 3600) / 60;
54         int seconds = totalSeconds % 60;
55
56         return String.format("%02d:%02d:%02d", hours, minutes, seconds);
57     }
58
59     @Override
60     public void actionPerformed(ActionEvent e) {
61         if (e.getSource() == startButton && !isRunning) {
62             timer.start();
63             isRunning = true;
64         } else if (e.getSource() == stopButton && isRunning) {
65             timer.stop();
66             isRunning = false;
67         } else if (e.getSource() == resetButton) {
68             timer.stop();
69             isRunning = false;
70             elapsedTime = 0;
71             timeLabel.setText(formatTime(elapsedTime));
72         }
73     }
74
75     public static void main(String[] args) {
76         new Stopwatch();
77     }
78 }
79
```