



객체 지향 언어 및 실습



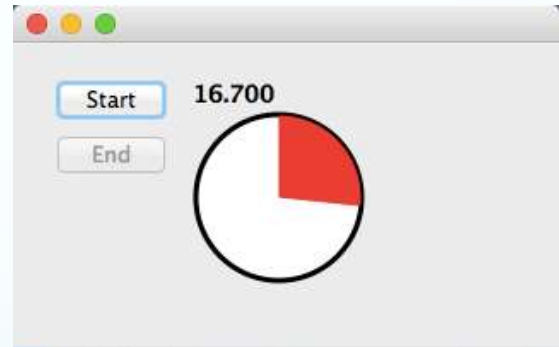
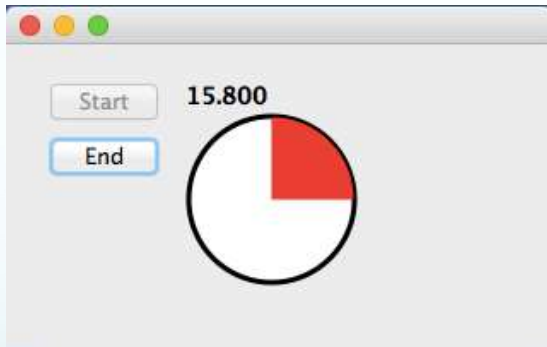
학번 : 2016110056

학과 : 불교학부

이름 : 박승원

날짜 : 2017년 6월 16일

- "Start" 버튼을 누르면 스레드가 동작하여서 카운터를 증가시키고 "Stop" 버튼을 클릭하면 카운터 증가가 중지되는 애플리케이션을 작성하여 보자. 스레드를 사용한다 .



- 코드와 실행 캡처화면 두개 이상을 제출하시오.
- (단 타이틀을 본인의 학번 및 성명으로 지정하시오.)
- 제출일: 6/20 21:00PM

// 2016110056 박승원

```
import java.util.*;
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
import java.awt.Graphics;
import java.awt.Color;

class clock extends JPanel implements ActionListener{
    JButton start , end;
    JLabel count;
    int sec=0;
    time_pass t;

    class time_pass extends Thread {
        public void run() {
            while(true) {
                sec++;
                count.setText( Integer . toString ( sec ));
                repaint () ;
            }
        }
    }
}
```

```

        try {
            Thread.sleep (1000);
        } catch (Exception e) {
            System.out.println (e.getMessage());
        }
    }
}

```

```

public void paintComponent(Graphics g) {
    g.setColor (Color.RED);
    g.fillArc (100, 100, 100, 100, 90, -sec%60 * 6);
}

```

```

public void actionPerformed(ActionEvent e) {
    if (e.getSource () == start ) {
        t = new time_pass();
        t.start ();
    } else try {
        t.stop ();
    } catch (Exception e2) {}
}

```

```

public clock () {
    start = new JButton(" start ");
    end = new JButton("end");
    count = new JLabel("0");
    start.addActionListener (this);
    end.addActionListener (this);
    add( start );
    add(end);
    add(count);
    setVisible (true);
}
}

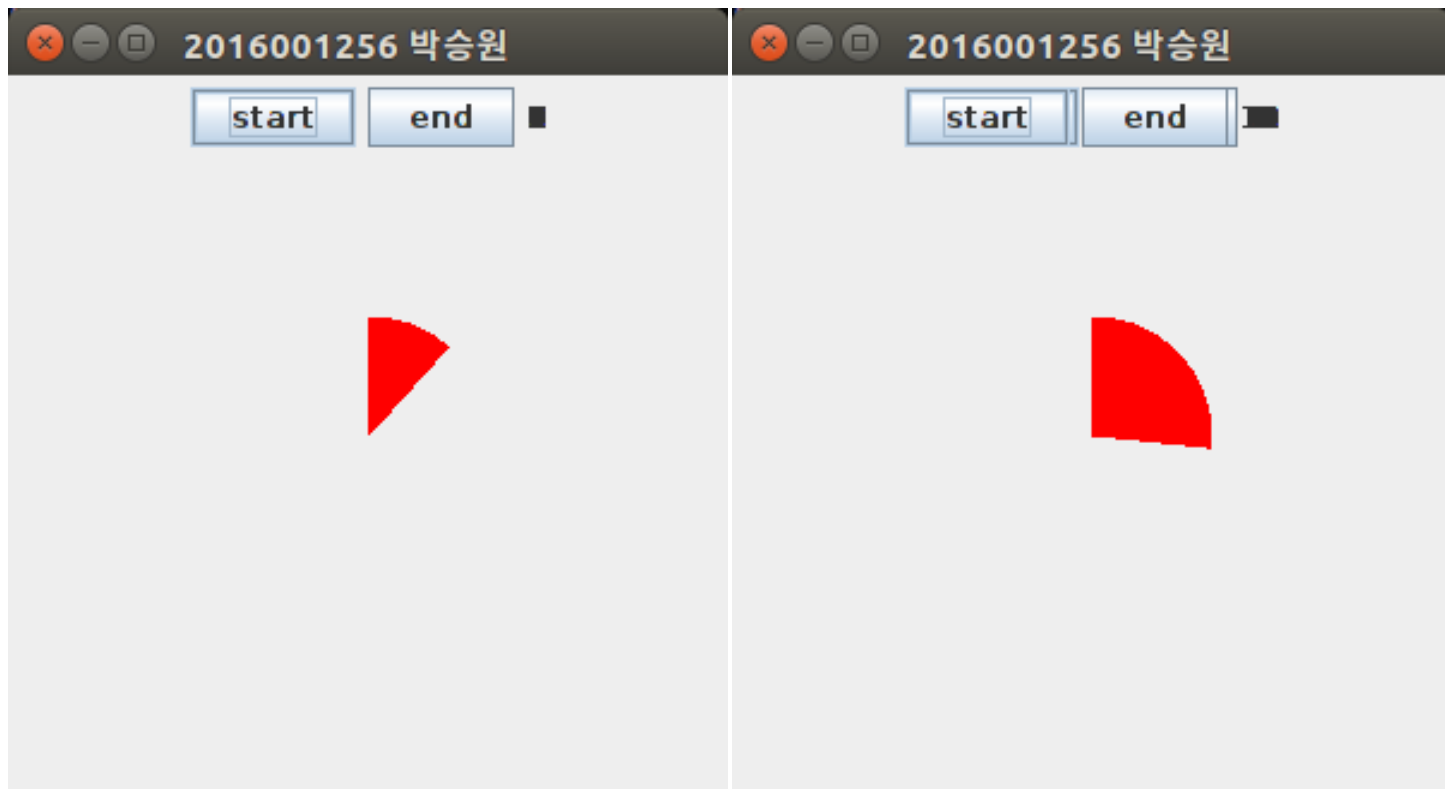
```

```

public class problem1 {
    public static void main(String[] args) {
        JFrame jf = new JFrame("2016001256 박승원 ");
        jf.add(new clock());
        jf.setSize (300, 300);
    }
}

```

```
    jf . setDefaultCloseOperation (JFrame.EXIT_ON_CLOSE);  
    jf . setVisible (true);  
}  
}
```



JLabel이 텍스트가 업데이트가 되지 않고 위에 겹쳐 써지는 버그가 있다.