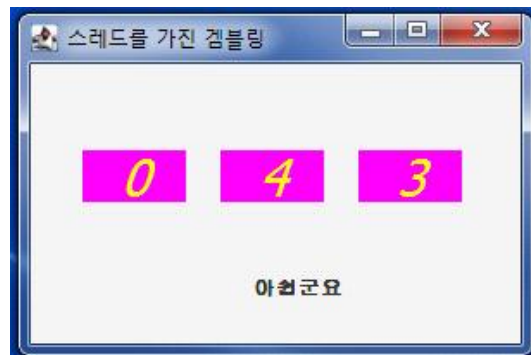


## Report\_#9(12장)

스윙으로 간단한 갬블링 게임을 만들어 보자. 그림과 같이 3개의 컴포넌트가 있다. 이 컴포넌트에는 각각 0~4까지의 한 자릿수가 출력된다. <Enter> 키를 입력할 때마다 3개의 수를 랜덤하게 발생시키고 이 수를 각 레이블에 출력한다. 그리고 나서 모두 동일한 수인지 판단하여 모두 동일한 수이면 “축하합니다!!”를 아니면 “아쉽군요”를 출력한다. 본 예제에서는 스레드를 사용해야 한다.



### Hint:

- 이번 레포트는 자바 스레드 실습이므로 학생들의 편의를 위해 GUI에 관해서는 아래에 주어짐. TO DO 부분을 작성하면 됨
- 게임을 하면 `gambling = true`, 안하면 `gambling = false` 로 셋팅하면 됨.
  1. 게임을 하지 않으면 기다림: `if(!gambling)`이면 `wait()`.
  2. 마우스 클릭으로 게임을 시작: `gambling = true; notify();`
  3. 게임을 진행:
    - 무한루핑을 돌림
    - `waitForGambling();` (마우스 클릭에 의해 게임 진행 지시를 기다림)
    - 0~4까지의 랜덤 수를 label에 넣음(첫번째, 두번째, 세번째 동일)
    - 게임이 성공했는지 판별: `if(첫번째수 == 두번째수 && 두번째수 == 세번째수)`
    - `gambling = false;` (스톱하여 다음 게임이 가능하도록 설정)

```
import java.awt.*;  
import java.awt.event.*;  
import javax.swing.*;
```

```
public class Report12 extends JFrame {  
    Report12() {  
        super("쓰레드를 가진 갬블링");  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        setContentPane(new GamePanel()); //GamePanel을 컨텐트팬으로 등록  
        setSize(300, 200);  
    }  
}
```

```

        setVisible(true);
    }
    class GamePanel extends JPanel { //화면 디자인
        JLabel[] label = new JLabel[3]; //3개의 레이블
        JLabel result = new JLabel("마우스를 클릭할 때마다 게임합니다."); //결과 출력
        /* TO DO 스레드(thread) 생성 */
        public GamePanel() {
            setLayout(null); //절대 위치에 컴포넌트 배치
            for(int i=0; i<label.length; i++) {
                label[i] = new JLabel("0"); //초기 레이블 생성
                label[i].setSize(60, 30); //레이블 크기
                label[i].setLocation(30+80*i, 50); //레이블 위치
                label[i].setHorizontalAlignment(JLabel.CENTER);
                label[i].setOpaque(true); //레이블에 배경색 설정이 가능하도록 설정
                label[i].setBackground(Color.MAGENTA);
                label[i].setForeground(Color.YELLOW);
                label[i].setFont(new Font("Tahoma", Font.ITALIC, 30));
                add(label[i]);
            }
            result.setHorizontalAlignment(JLabel.CENTER); //결과를 출력할 레이블 생성
            result.setSize(250, 20);
            result.setLocation(30, 120);
            add(result);
            /* TO DO 스레드 시작 */
            addMouseListener(new MouseAdapter() { //마우스 리스너 구현
                public void mousePressed(MouseEvent e) {
                    if((thread).isReady()) thread.startGambling();
                }
            });
        }
    }
    class GamblingThread extends Thread {
        JLabel[] label; //게임 숫자를 출력
        JLabel result; //결과를 출력
        long delay = 200; //지연시간(sleep) 값
        boolean gambling = false; //게임을 할 것인지 결정
        public GamblingThread(JLabel[] label, JLabel result) {
            //TO DO
        }
        boolean isReady() {
            return !gambling; //게임 중이면 준비되지 않았음
        }
        synchronized public void waitForGambling() { //게임하지 않으면 기다림
            //TO DO
        }
        synchronized public void startGambling() { //마우스 클릭으로 게임을 진행
            //TO DO
        }
        public void run() { //게임을 진행
            //TO DO
            //첫번째, 두번째, 세번째 수 조정
            //게임이 성공하였는지 판별(축하합니다!! 또는 아쉽군요)
        }
    }
    public static void main(String[] arg) {
        new Report12();
    }
}

```