



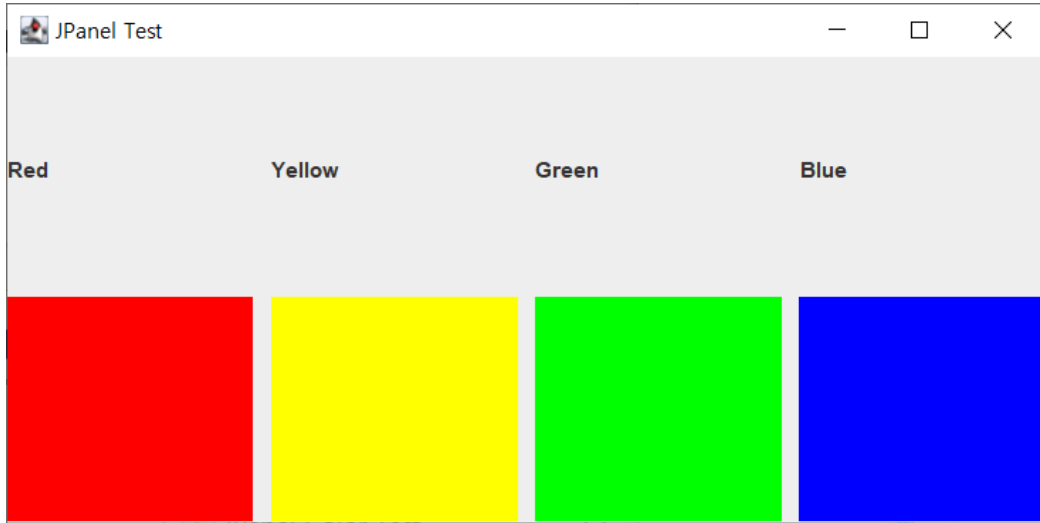
Lab 07. Chapter 9

실습 1> 레이아웃 구성 실습 (JPanelTest.java)

필요에 따라 추가 레이아웃 또는 설정 사용 가능,
결과 UI는 예제와 동일하지 않아도 됩니다.

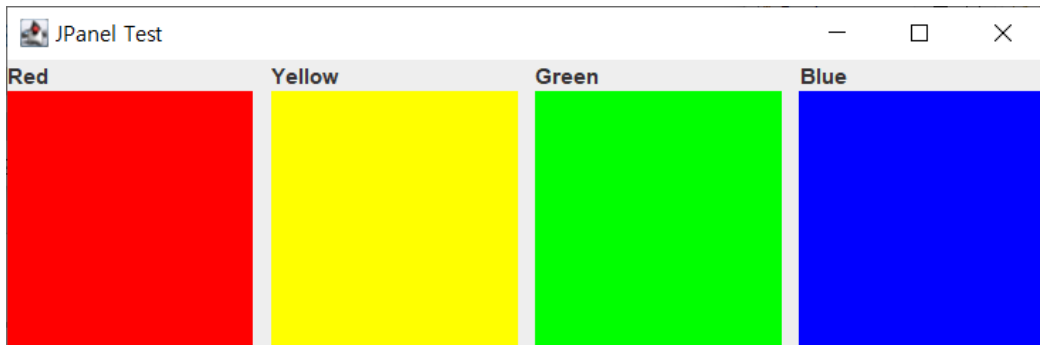
- 다음과 같은 프로그램의 화면만 만들어보자. 배경색이 다른 패널을 이용한다.

(1)



GridLayout으로 설정

(2)



BorderLayout, GridLayout 혼합으로 설정

```
import java.awt.*;  
import javax.swing.*;  
  
public class JPanelTest extends JFrame {  
    public JPanelTest() {  
        setSize(600, 200);  
        setTitle("JPanel Test");  
  
        // 라벨 생성 4개  
  
        // 배경색 설정을 위한 패널 생성 4개  
  
        // 패널의 배경색 설정  
  
        // 레이아웃 설정  
  
        // 라벨과 색상 패널을 추가  
  
        setVisible(true);  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    }  
    public static void main(String[] args) {  
        new JPanelTest();  
    }  
}
```

결과 UI는 하나만 작성해도 됨

실습 2> 레이아웃 구성 실습 2(MemberGUI.java)

- 회원의 정보를 받아서 데이터베이스에 저장하는 프로그램의 화면을 만들어보자.
- 여러 개의 패널과 배치관리자를 혼용하여 완성해 본다.
- 필요한 컴포넌트들 및 생성, 배치를 위한 패널들과 각 레이아웃 구성

회원 정보 등록 화면

회원 등록

이름 홍길동

패스워드

이메일 주소 abc@example.com

전화번호 010-1234-5678

등록하기 취소

참고 예)

회원 정보 등록 화면

회원 등록

이름 홍길동

패스워드

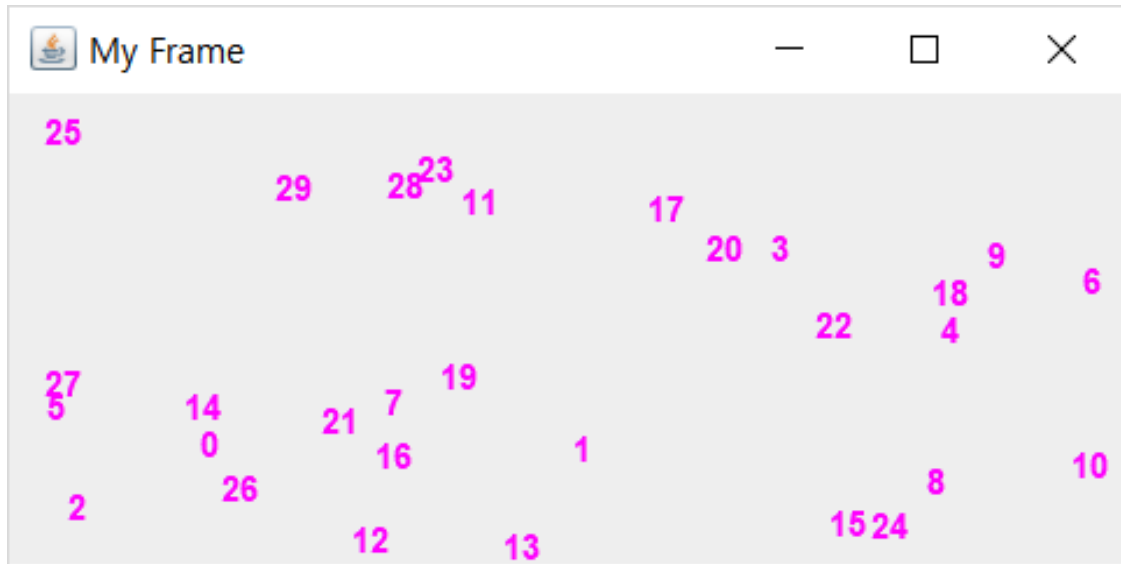
이메일 주소 abc@example.com

전화번호 010-1234-5678

등록하기 취소

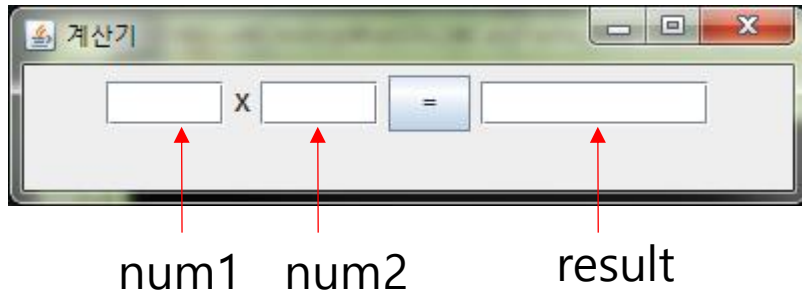
실습 3> (RandomNumberGUI.java)

- 다음과 같이 난수를 발생해서 30개의 레이블을 불규칙하게 배치하여 보자.
 - 라벨 배열 객체 생성, 반복문 활용해서 난수 생성 및 위치값 설정
 - 라벨의 크기가 너무 작으면 숫자값이 잘 안 보일 수 있음. 적절하게 조정 필요
 - 절대 위치로 배치하려면 `setLayout(null)`을 설정해야 한다.
 - `setSize()`, `setLocation()`을 함께 사용해서 위치, 크기 값을 설정
 - 또는 `setBounds()`를 사용해서 위치, 크기 값을 설정



실습 4> (CalcFrame.java)

- 다음 그림과 같은 형태의 UI를 만들고 "=" 버튼의 이벤트를 구현하시오.
(CalcFrame.java)



- 버튼을 눌렀을 때(ActionEvent) num1과 num2의 숫자의 곱셈 연산을 해서 계산 결과를 result에 보여주도록 한다.



실습 5> (MemberInfo.java)

- 이름과 학과, 연락처를 입력하고 등록 버튼을 누르면 텍스트 영역에 추가되는 프로그램을 작성하시오.

