



개인별 과제

과 목	자바실무
담당 교수	이 건 익
학 번	201720970
학 과	소프트웨어·미디어·산업공학부
이 름	권 대 한

개인별 과제물(퍼즐 만들기) - 10점

※ 클래스이름을 "**Puzzle_자신의학번**"으로 하시오.

※ 설정이 안된 부분은 알아서 하시면 됩니다. 프레임의 크기(500×300)도 알아서 하세요.

1. 실행 초기화면

- ① 4 × 4 퍼즐을 만든다.
- ② 1부터 16까지의 숫자가 나타나게 한다.
- ③ 숫자는 랜덤하게 나타나게 한다.
- ④ 마지막에 발생한 숫자 16을 안보이게 한다.
- ⑤ 프레임 타이틀은 처음에는 "**4 × 4 Puzzle(학번_이름)**"로 하고 조건에 따라 변경한다.



4	15		8
2	13	12	10
9	6	3	7
5	14	11	1

2. 게임화면

- ⑥ 게임 중 프레임 타이틀에 **이동횟수 : 몇회** 표시하기



1	2	3	4
6		12	5
14	11	9	10
15	13	7	8

⑦ 게임중 이동횟수 300회 이상일 경우에는

프레임 타이틀에 **이동횟수 : 몇회 - 실패**로 표시한다.

모든 컴포넌트를 비활성화시킨다.

이동횟수 : 300회 - 실패(9122126_홍길동)			
1	2	3	4
5	6	7	8
9	10		14
13	15	11	12

⑧ 게임중 성공할 경우에는

프레임 타이틀에 **이동횟수 : 몇회 - 성공**으로 표시한다.

이동횟수 : 114회 - 성공(9122126_홍길동)			
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

⑨ 아래에 소스 코드를 반드시 텍스트로 삽입하시오.

[소스 코드 삽입]

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

import java.util.Random;
import java.util.Vector;

public class Puzzle_201720970 extends JFrame {
    int count = 1;
    JButton[] Puzzles = new JButton[16];
    Vector<String> num = new Vector<String>();

    public Puzzle_201720970() {
        String Number = "";
        Random random = new Random();

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setTitle("4 X 4 Puzzle (201720970_권대한)");
        setLayout(new GridLayout(4, 4));

        while (num.size() != 16) {
            Number = Integer.toString(random.nextInt(16) + 1);
            while (!num.isEmpty() && num.contains(Number)) {
                Number = Integer.toString(random.nextInt(16) + 1);
            }
            num.add(Number);
        }
        for (int i = 0; i < Puzzles.length; i++) {
            Puzzles[i] = new JButton();
            add(Puzzles[i]);
            Puzzles[i].setText(num.get(i));
            if (num.get(i).equals("16")) {
                Puzzles[i].setEnabled(false);
                Puzzles[i].setText("");
            }
            Puzzles[i].addActionListener(new PressedListener());
        }
        setSize(500, 300);
        setVisible(true);
    }

    private int[] Check = new int[4];

    private void NumberCheck(int index) {
        Check[0] = index - 4; // u
        Check[1] = index + 4; // d
        Check[2] = index - 1; // l
        Check[3] = index + 1; // r
        if (Check[1] > 16) {
            Check[1] = -1;
        }
    }
}
```

```

        if (Check[2] < 0 || Check[2] % 4 == 3) {
            Check[2] = -1;
        }
        if (Check[3] % 4 == 0) {
            Check[3] = -1;
        }
    }

    private class PressedListener implements ActionListener {
        public void actionPerformed(ActionEvent e) {
            int index = 0, i1 = 0;

            JButton b1 = (JButton) e.getSource();
            JFrame f1 = (JFrame) b1.getTopLevelAncestor();

            while (!b1.equals(Puzzles[index])) {
                index++;
            }
            NumberCheck(index);
            for (int i = 0; i < Check.length; i++) {
                if (Check[i] > -1) {
                    //Array Size Exception!
                    if (Check[i] == 16) {
                        continue;
                    } else if (!Puzzles[Check[i]].isEnabled()) {
                        if (count > 299) {
                            f1.setTitle("이동 횟수 : " + count +
                                "회" + " - 실패 (201720970_권대한)");

                            for (JButton b2 : Puzzles) {
                                b2.setEnabled(false);
                            }
                            break;
                        }
                        f1.setTitle("이동 횟수 : " + count + "회" + "
(201720970_권대한)");

                        count++;
                        while
(Puzzles[i1].getText().equals(Integer.toString(i1 + 1))) {
                            // boolean?
                            ++i1;
                        }
                        // if 14, Is successes
                        if (i1 > 13) {
                            f1.setTitle("이동 횟수 : " + count +
                                "회" + " - 성공 (201720970_권대한)");
                        }
                        // b1.getText() isPressed

                        Puzzles[Check[i]].setText(Puzzles[index].getText());
                        Puzzles[index].setText("");
                        Puzzles[Check[i]].setEnabled(true);
                        Puzzles[index].setEnabled(false);
                        break;
                    }
                }
            }
        }
    }
}

```

```
        }  
    }  
  
    public static void main(String[] args) {  
        new Puzzle_201720970();  
    }  
}
```

⑩ 아래에 3개의 실행 결과 화면을 삽입하시오.

[실행 결과 화면]

㉠ 실행 초기 화면

4 X 4 Puzzle (201720970_권대한)			
10	8	4	15
14	5	6	9
	3	7	11
13	2	1	12

이동 횟수 : 87회 (201720970_권대한)			
6	3	10	5
7	4	2	12
	8	11	9
13	14	15	1

㉡ 퍼즐실패시 화면

이동 횟수 : 300 - 실패 (201720970_권대한)			
3	5		7
4	2	8	6
9	10	11	12
13	14	15	1

㉔ 퍼즐성공시 화면

이동 횟수 : 278회 - 성공 (201720970_권대한)			
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	