

# 자바2

(GUI-3 : 스윙 컴포넌트)

# Chapter 08

## 스윙 컴포넌트

### Chapter08의 학습목표

- 스윙 컴포넌트의 기능들을 세부적으로 알아본다

### Swing

## 스윙 컴포넌트에 이미지 표시하기

- 거의 모든 스윙 컴포넌트에는 텍스트 옆에 이미지를 추가로 표시할 수 있음
- ImageIcon 인스턴스를 생성하여 사용
- JPEG, GIF, PNG 이미지 파일을 읽을 수 있음

```
ImageIcon image = new ImageIcon("image.gif"); //이미지 생성  
JLabel label = new JLabel("이미지 레이블"); //라벨 생성  
label.setIcon(image); //이미지를 라벨의 이미지로 설정
```

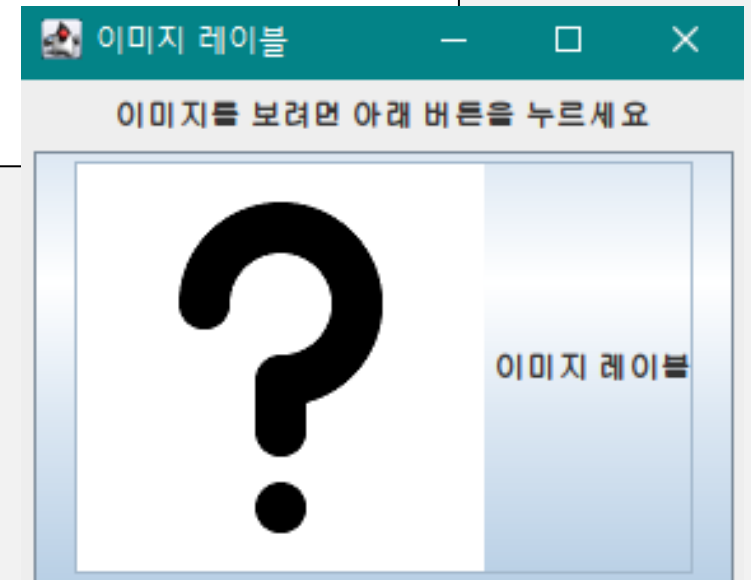


### Swing

#### 이미지 표시 예제 : 레이블에 이미지 표시하기

이미지 버튼을 표시하고 사용자가 버튼을 누르면 레이블의 텍스트를 이미지로 바꾸어서 표시하는 프로그램을 작성

```
class MyFrame extends JFrame implements ActionListener {  
    private JPanel panel;  
    private JLabel label;  
    private JButton button;
```



### Swing

## 이미지 표시 예제 : 레이블에 이미지 표시하기

```
public MyFrame() {  
    setTitle("이미지 레이블");  
    setSize(300,250);  
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
  
    panel = new JPanel();  
    label = new JLabel("이미지를 보려면 아래 버튼을 누르세요");  
  
    button = new JButton("이미지 레이블");  
    ImageIcon icon = new ImageIcon("ques.png");  
    button.setIcon(icon);  
    button.addActionListener(this);  
    panel.add(label);  
    panel.add(button);  
  
    add(panel);  
    setVisible(true);  
}
```

### Swing

#### 이미지 표시 예제 : 레이블에 이미지 표시하기

```
@Override
public void actionPerformed(ActionEvent e) {
    ImageIcon peng = new ImageIcon("peng.jpg");
    label.setIcon(peng);
    label.setText(null);
}

public static void main(String[] args) {
    MyFrame frame = new MyFrame();
}
}
```

Swing

체크 박스 (Check box)

- 사용자가 클릭하여서 체크된 상태와 체크되지 않은 상태 중의 하나로 만들 수 있는 컨트롤
- 사용자로 하여금 YES / NO 중에서 하나를 선택하게 하는데 사용됨
- 체크박스 클릭 시 ItemEvent가 발생 -> ItemListener를 구현

```
onion = new JCheckBos("양파"); //체크되어있지 않은 체크박스
onion = new JCheckBos("양파", true); //체크되어있는 체크박스
```



	이름	설명
생성자	JCheckBox()	레이블이 없는 Checkbox를 생성한다.
	JCheckbox(String label)	지정된 레이블의 Checkbox를 생성한다.
	JCheckbox(String label, boolean selected)	지정된 상태와 레이블을 가지는 Checkbox을 생성한다.
메소드	String getText()	체크 박스에 표시되는 텍스트를 가져온다.
	Boolean isSelected()	만약 체크 박스가 선택되었으면 true를 반환한다.
	void setSelected(boolean value)	매개 변수가 true이면 체크 박스를 체크 상태로 만든다.
	void setText(String text)	체크 박스 텍스트를 설정한다.

### Swing

## 체크 박스를 이용한 이미지 표시 예제

체크 박스를 이용해서 그림이 나오게 하거나 나오지 않도록 하는 프로그램을 작성

```
class CheckBoxPanel extends JPanel implements ItemListener {
    JCheckBox[] buttons = new JCheckBox[3];
    String[] tong = { "두통", "복통", "치통" };
    JLabel[] pictureLabel = new JLabel[3];
    ImageIcon[] icon = new ImageIcon[3];

    public CheckBoxPanel() {
        super(new GridLayout(0, 4));

        // 체크 박스 생성
        for (int i = 0; i < 3; i++) {
            buttons[i] = new JCheckBox(tong[i]);
            buttons[i].addItemListener(this);
            pictureLabel[i] = new JLabel(tong[i] + ".jpg");
            icon[i] = new ImageIcon(tong[i] + ".jpg");
        }
    }
}
```



### Swing

#### 체크 박스를 이용한 이미지 표시 예제

```
JPanel checkPanel = new JPanel(new GridLayout(0, 1));

checkPanel.add(buttons[0], BorderLayout.WEST);
checkPanel.add(buttons[1], BorderLayout.CENTER);
checkPanel.add(buttons[2], BorderLayout.EAST);

add(checkPanel);
add(pictureLabel[0]);
add(pictureLabel[1]);
add(pictureLabel[2]);
}
```

### Swing

## 체크 박스를 이용한 이미지 표시 예제

```
/** 체크 박스의 아이템 이벤트를 처리한다. */
public void itemStateChanged(ItemEvent e) {
    ImageIcon image = null;
    Object source = e.getItemSelectable();
    for (int i = 0; i < 3; i++) {
        if (source == buttons[i]) {
            if (e.getStateChange() == ItemEvent.DESELECTED)
                pictureLabel[i].setIcon(null);
            else
                pictureLabel[i].setIcon(icon[i]);
        }
    }
}
```

### Swing

#### 체크 박스를 이용한 이미지 표시 예제

```
class MyFrame extends JFrame {  
    public MyFrame() {  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        CheckBoxPanel panel = new CheckBoxPanel();  
        panel.setOpaque(true);  
        add(panel);  
        setSize(500, 200);  
        setVisible(true);  
    }  
  
    public static void main(String[] args) {  
        MyFrame frame = new MyFrame();  
    }  
}
```

### Swing

## 라디오 버튼 (Radio button)

- 체크 박스와 비슷하지만 하나의 그룹 안에서는 한개의 버튼만 선택이 가능
- 하나의 라디오 버튼을 클릭하면 다른 버튼은 자동적으로 선택이 해제됨
- JRadioButton : 라디오 버튼을 생성하는 클래스
- ButtonGroup : 버튼들을 그룹핑하는데 사용되는 클래스

```
JRadioButton radio1 = new JRadioButton("선택 1"); //라디오 버튼 생성
ButtonGroup group = new ButtonGroup(); // 그룹핑 객체를 생성
group.add(radio1); // 라디오 버튼을 그룹에 추가
```



### Swing

## 경계 (Border)

- 시각적으로 컴포넌트들을 그룹핑 할 때 사용하는 장식적인 요소
- 체크 박스나 라디오 버튼을 그룹핑 할 때 함께 사용

```
Border border = BorderFactory.createTitledBorder("크기");  
sizePanel.setBorder(border);
```

Swing

텍스트 필드 (Text Field)

- 입력이 가능한 한 줄의 텍스트 필드를 만드는데 사용
- 텍스트 필드 안에서 문자열 선택 및 복사 붙여넣기 가능
- JTextField의 자식 클래스 : JPasswordField (입력하는 문자가 보이지 않음), JFormattedTextField (입력할 수 있는 문자를 제한)

JTextField: 김철수

JPasswordField: \*\*\*\*

JFormattedTextField: 2009. 3. 7

```
JTextField textfield = new JTextField(30); // 30자 크기의 텍스트 필드를 생성
JTextField textfield = new JTextField("Initial String");
```

메소드 명	설명
getText()	텍스트 필드에 들어 있는 텍스트를 받아옴
setText(문자열)	텍스트 필드 안에 텍스트를 삽입
selectAll()	텍스트 필드에 들어있는 텍스트를 모두 선택
requestFocus()	텍스트 필드가 사용자로부터 입력을 받을 수 있도록 키보드 포커스를 호출

### Swing

#### 텍스트 필드 예제 : 사용자로부터 받은 정수의 제곱 계산

사용자로부터 정수를 입력 받은 후에 정수의 제곱을 구하는 프로그램

```
class MyFrame extends JFrame {  
    private JButton button;  
    private JTextField text, result;  
  
    public MyFrame() {  
        setSize(300, 130);  
        setTitle("제곱 계산하기");  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
  
        ButtonListener listener = new ButtonListener(); // 리스너 객체 생성
```

### Swing

#### 텍스트 필드 예제 : 사용자로부터 받은 정수의 제곱 계산

```
JPanel panel = new JPanel();
panel.add(new JLabel("숫자 입력: "));           // 레이블 생성
text = new JTextField(15);                     // 컬럼수가 15인 텍스트 필드 생성
text.addActionListener(listener);             // 텍스트 필드에 리스너 연결
panel.add(text);

panel.add(new JLabel("제곱한 값: "));
result = new JTextField(15);                   // 결과를 나타낼 텍스트 필드
result.setEditable(false);                    // 편집 불가 설정
panel.add(result);
button = new JButton("OK");
button.addActionListener(listener);
panel.add(button);
add(panel);
setVisible(true);
}
```



### Swing

#### 텍스트 필드 예제 : 사용자로부터 받은 정수의 제곱 계산

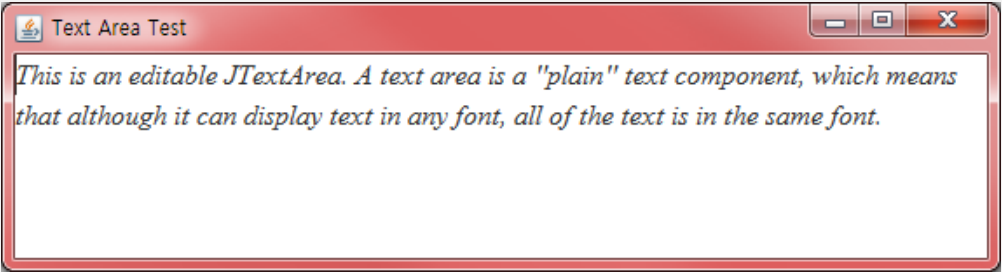
```
// 텍스트 필드와 버튼의 액션 이벤트 처리
private class ButtonListener implements ActionListener {
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == button || e.getSource() == text) {
            String name = text.getText();
            int value = Integer.parseInt(name);
            result.setText(" " + value * value);
            text.requestFocus();
        }
    }
}

public static void main(String[] args) {
    MyFrame frame = new MyFrame();
}
```

Swing

텍스트 영역 (Text Area)

- 한 줄이 아닌 여러 줄의 텍스트가 들어 갈 수 있는 컴포넌트



생성자	설명
JTextArea()	TextArea 생성
JTextArea(int rows, int columns)	줄 수와 칸수를 지정하여 TextArea를 생성
JTextArea(String text)	지정된 Text가 입력된 TextArea를 생성

메소드 명	설명
getText()	텍스트 영역에 들어 있는 텍스트를 받아옴
setText(문자열)	텍스트 영역 안에 텍스트를 넣음
setEditable(boolean)	텍스트 영역을 편집 가능/불가능으로 설정
append()	텍스트 영역에 텍스트를 추가

### Swing

## 콤보 박스 (Combo box)

- 여러 항목 중에서 하나를 선택할 때 사용
- 텍스트 필드와 리스트의 결합

### 방법 1

//콤보박스 생성

```
JComboBox combo = new JComboBox();
```

//콤보박스에 항목을 추가

```
combo.addItem("dog");
```

```
combo.addItem("lion");
```

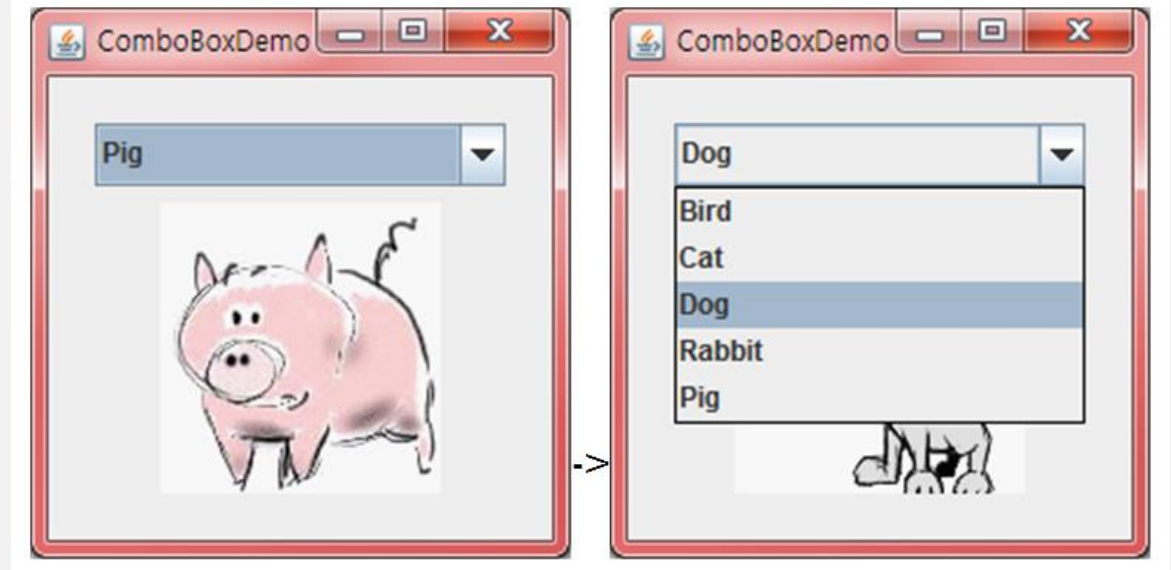
```
combo.addItem("tiger");
```

### 방법 2

//배열 생성 후 생성자에 전달

```
String[] names = {"dog", "lion", "tiger"};
```

```
JComboBox combo = new JComboBox(names);
```



### Swing

## 콤보박스를 이용한 이미지 표시 예제

```
class MyFrame extends JFrame implements ActionListener {
    JLabel label;

    public MyFrame() {
        setTitle("콤보 박스");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 300);

        String[] animals = { "A", "B", "C" };
        JComboBox animallist = new JComboBox(animals);
        animallist.setSelectedIndex(0);
        animallist.addActionListener(this);

        label = new JLabel();
        label.setHorizontalAlignment(JLabel.CENTER);
        changePicture(animals[animallist.getSelectedIndex()]);
        add(animallist, BorderLayout.PAGE_START);
        add(label, BorderLayout.PAGE_END);
        setVisible(true);
    }
}
```

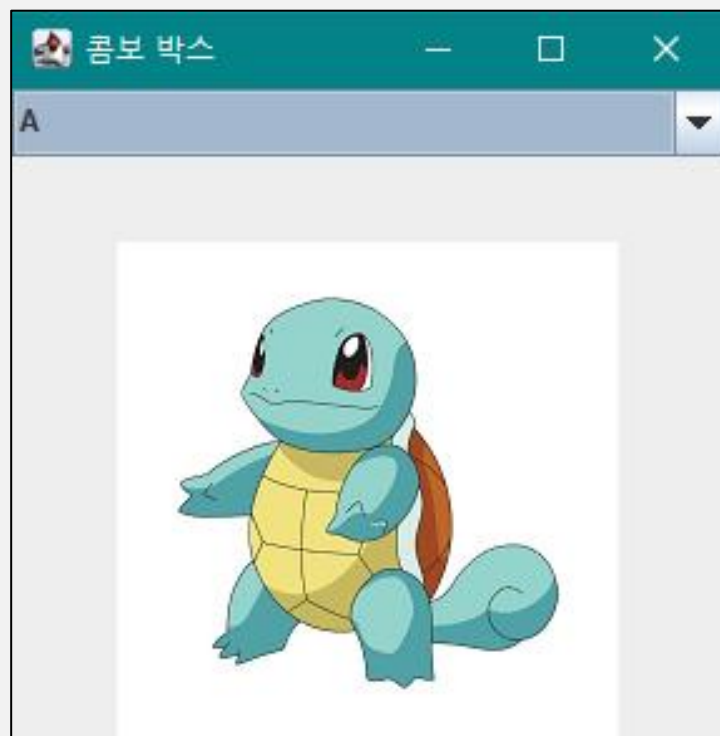
### Swing

## 콤보박스를 이용한 이미지 표시 예제

```
public void actionPerformed(ActionEvent e) {
    JComboBox cb = (JComboBox) e.getSource();
    String name = (String) cb.getSelectedItem();
    changePicture(name);
}

protected void changePicture(String name) {
    ImageIcon icon = new ImageIcon(name + ".png");
    label.setIcon(icon);
    if (icon != null) {
        label.setText(null);
    } else {
        label.setText("이미지가 발견되지 않았습니다.");
    }
}

public static void main(String[] args) {
    MyFrame frame = new MyFrame();
}
```

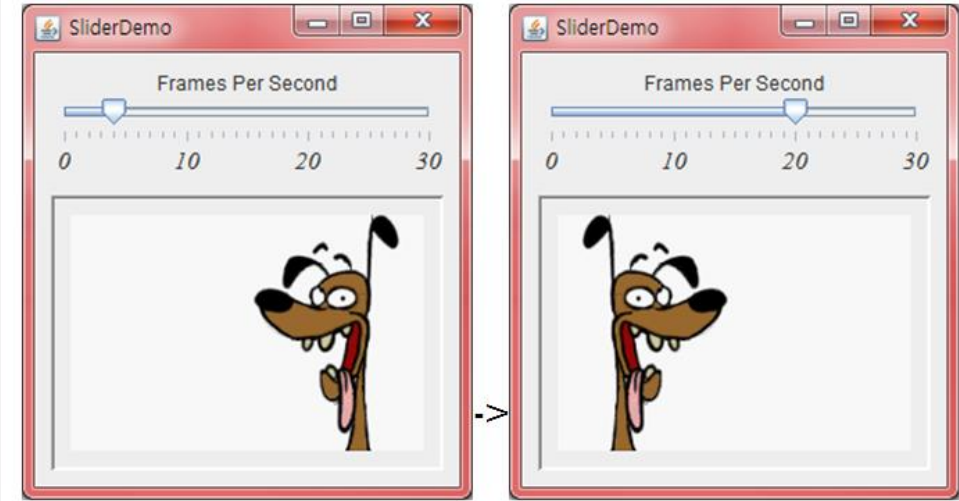


Swing

슬라이더 (Slider)

- 사용자가 특정한 범위 안에서 하나의 값을 선택 가능한 컴포넌트
- 사용자로부터 일정 범위 안의 수치 값을 입력 받는 경우 사용
- JSlider 클래스의 생성자를 호출

`silder = new JSlider(눈금시작값, 마지막값, 초기값)`



메소드 명	설명
setMajorTickSpacint(int)	큰 눈금 간격을 설정
setMinorTickSpacint(int)	작은 눈금 간격을 설정
setPaintTicks(boolean)	눈금을 표시할지 여부를 설정
setPaintLabels(boolean)	값을 레이블로 표시할지 여부를 설정

### Swing

#### 슬라이더 예제 : 이미지 크기 변경

```
class MyFrame extends JFrame implements ChangeListener {  
    static final int INIT_VALUE = 15;  
    private JButton buttonOK;  
    private JSlider slider;  
    private JButton button;  
  
    public MyFrame() {  
        JPanel panel;  
  
        setTitle("슬라이더 테스트");  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    }  
}
```

### Swing

#### 슬라이더 예제 : 이미지 크기 변경

```
panel = new JPanel();
JLabel label = new JLabel("슬라이더를 움직여보세요", JLabel.CENTER);
label.setAlignmentX(Component.CENTER_ALIGNMENT);
panel.add(label);
slider = new JSlider(0, 30, INIT_VALUE);
slider.setMajorTickSpacing(10);      // 큰 눈금 간격
slider.setMinorTickSpacing(1);       // 작은 눈금 간격
slider.setPaintTicks(true);           // 눈금을 표시한다.
slider.setPaintLabels(true);          // 값을 레이블로 표시한다.
slider.addChangeListener(this);      // 이벤트 리스너를 붙인다.
panel.add(slider);
```



### Swing

#### 슬라이더 예제 : 이미지 크기 변경

```
button = new JButton("");
ImageIcon icon = new ImageIcon("test.png");
button.setIcon(icon);
button.setSize(INIT_VALUE * 10, INIT_VALUE * 10);
panel.add(button);
add(panel);

setSize(300, 300);
setVisible(true);
}
```

### Swing

#### 슬라이더 예제 : 이미지 크기 변경

```
public void stateChanged(ChangeEvent e) {
    JSlider source = (JSlider) e.getSource();
    if (!source.getValueIsAdjusting()) {
        int value = (int) source.getValue();
        button.setSize(value * 10, value * 10);
    }
}

public static void main(String[] args) {
    MyFrame frame = new MyFrame();
}
```

### Swing

## 파일 선택기 (File Chooser)

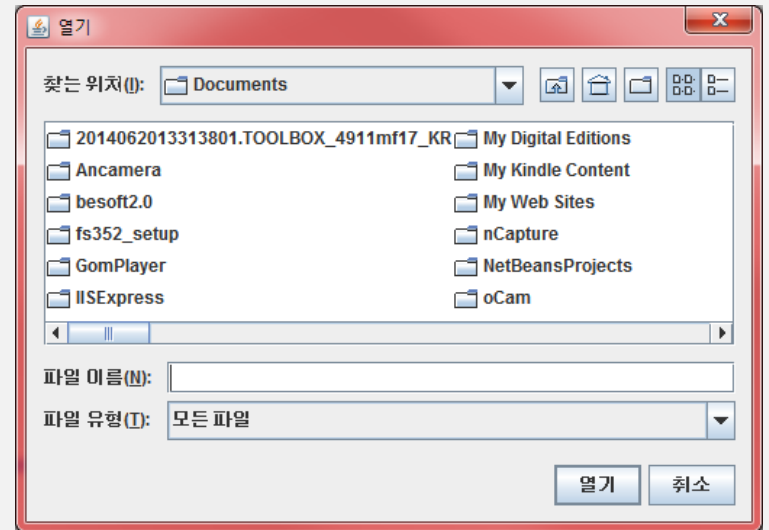
- 파일 시스템을 탐색하여 파일이나 디렉토리를 선택하는 GUI를 제공
- JFileChooser 클래스를 사용
- 단지 파일을 선택하기 위한 GUI만을 제공하기 때문에 열거나 쓰는 것은 아니다

//파일 선택기 생성

```
final JFileChooser fc = new JFileChooser();
```

//파일 선택기 대화 상자를 엽니다

```
int returnVal = fc.showOpenDialog(컴포넌트)
```



### Swing

#### 파일 선택기 예제 : 버튼 클릭시 파일 선택기 창이 뜨는 프로그램

```
class MyFrame extends JFrame implements ActionListener {
    JButton openButton, saveButton;
    JFileChooser fc;

    public MyFrame() {
        setTitle("파일 선택기 테스트");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300, 200);

        fc = new JFileChooser();
    }
}
```

### Swing

## 파일 선택기 예제 : 버튼 클릭시 파일 선택기 창이 뜨는 프로그램

```
JLabel label = new JLabel("파일 선택기 컴포넌트 테스트입니다.");
openButton = new JButton("파일 오픈");
openButton.addActionListener(this);

saveButton = new JButton("파일 저장");
saveButton.addActionListener(this);

JPanel panel = new JPanel();
panel.add(label);
panel.add(openButton);
panel.add(saveButton);
add(panel);
setVisible(true);

}
```

### Swing

#### 파일 선택기 예제 : 버튼 클릭시 파일 선택기 창이 뜨는 프로그램

```
public void actionPerformed(ActionEvent e) {  
    // “파일 오픈”버튼에 대한 액션 이벤트 처리  
    if (e.getSource() == openButton) {  
        int returnVal = fc.showOpenDialog(this);  
        if (returnVal == JFileChooser.APPROVE_OPTION) {  
            File file = fc.getSelectedFile();  
            // 실제 파일을 오픈한다.  
        } else {  
            // 사용자 취소  
        }  
    }  
    // “파일 저장”버튼에 대한 액션 이벤트 처리  
}
```

### Swing

#### 파일 선택기 예제 : 버튼 클릭시 파일 선택기 창이 뜨는 프로그램

```
        else if (e.getSource() == saveButton) {
            int returnVal = fc.showSaveDialog(this);
            if (returnVal == JFileChooser.APPROVE_OPTION) {
                File file = fc.getSelectedFile();
                // 실제 파일에 저장한다.
            } else {
                // 사용자 취소
            }
        }
    }

    public static void main(String[] args) {
        MyFrame frame = new MyFrame();
    }
}
```

수고하셨습니다.