



Power Java

제20장 스윙 컴포넌트 1



© 2009 인피니티박스 All rights reserved



이번 장에서 학습할 내용

- 텍스트 컴포넌트
- 텍스트 필드
- 텍스트 영역
- 스크롤 페인
- 체크 박스
- 라디오 버튼

스윙에서
제공하는
기초적인
컴포넌트들을
살펴봅시다.



© 2009 인피니티박스 All rights reserved



스윙 텍스트 컴포넌트들

종류	텍스트 컴포넌트	그림
텍스트 필드	JTextField JPasswordField JFormattedTextField	
일반 텍스트 영역	JTextArea	
스타일 텍스트 영역	JEditorPane JTextPane	

© 2009 인피니티박스 All rights reserved



스윙 텍스트 컴포넌트들

JTextField: 김철수

JPasswordField: ****

JFormattedTextField: 2009. 3. 7

여러 줄의 편집 가능한 텍스트를 나타낸다. 텍스트 영역에서는 동시에 하나의 폰트만 사용할 수 있다. 폰트는 변경이 가능하다. 텍스트 영역은 사용자로 하여금 비교적 장문의 포맷이 없는 텍스트를 입력하는데 사용된다.

이것은 편집이 불가능한 JEditorPane으로서, 이것은 HTML 문서로 초기화된다. 또는 URL로부터 초기화도

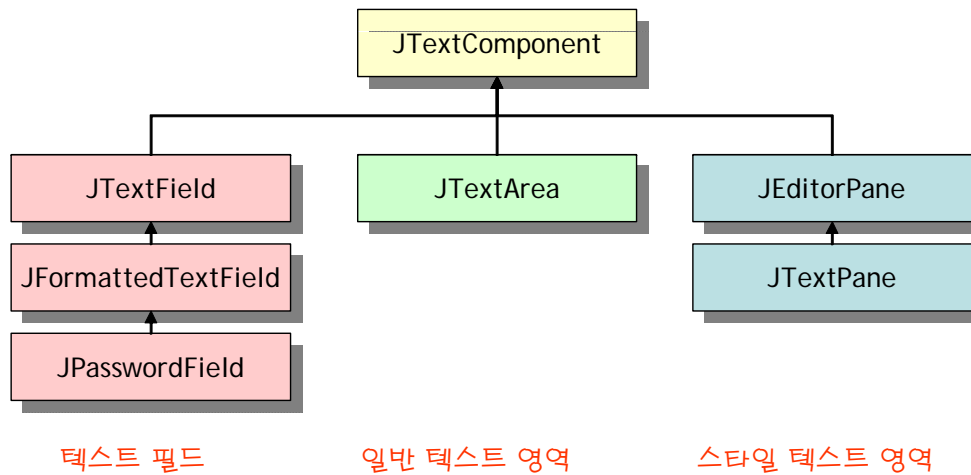
이것은 편집 가능한 JTextPane이다. JTextPane은 다양한 스타일이 가능한 텍스트 컴포넌트로서, 다양한 내장 컴포넌트와

아이콘들을 가지고 있다.

© 2009 인피니티박스 All rights reserved



상속 계층 구조



© 2009 인피니티박스 All rights reserved



중간 점검 문제

1. 폰트를 하나만 사용할 수 있는 메모장을 생성하려면 어떤 클래스를 사용하여야 하는가?
2. 사용자가 이미지나 사운드 파일을 포함하고 있는 텍스트를 생성하게 하려면 어떤 클래스를 사용하여야 하는가?

© 2009 인피니티박스 All rights reserved



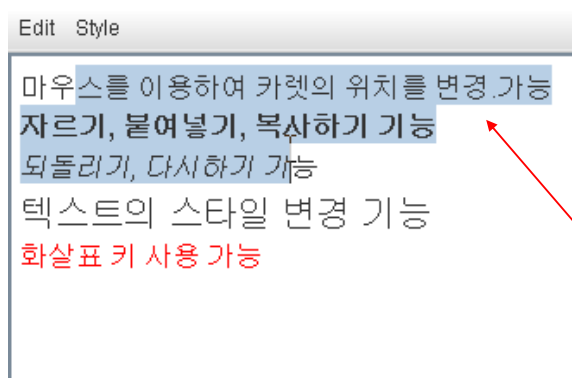
JTextComponent 클래스

- 컴포넌트가 가진 데이터는 document라고 하는 이름의 인스턴스에 저장된다. document는 MVC에서 모델(Model)에 해당한다.
- “오려두기”, “복사하기”, “붙이기”, “삽입하기”와 같은 기본적인 텍스트 편집 기능을 제공한다. 이들 기능들은 메뉴와 버튼들과 연결될 수 있다.
- 단축키 기능을 제공한다.
- 무제한의 되돌림 기능(undo)과 다시하기 기능(redo)을 제공한다.
- 교체가능한 카렛과 카렛 변경 리스터와 내비게이션 필터를 제공한다.
- 문서 필터를 제공한다.

© 2009 인피니티박스 All rights reserved



JTextComponent 클래스



이러한
기능들을
제공하는
클래스입니다.



© 2009 인피니티박스 All rights reserved



JTextComponent 메소드

■ 속성 설정 기능

메소드	설명
<code>void setEditable(boolean)</code> <code>boolean isEditable()</code>	편집 가능 여부를 설정한다.
<code>void setDragEnabled(boolean)</code> <code>boolean getDragEnabled()</code>	드래그 가능 여부를 설정한다.

■ 속성 설정 기능

메소드	설명
<code>String getSelectedText()</code>	현재 선택된 텍스트를 반환한다.
<code>void selectAll()</code> <code>void select(int start, int end)</code>	모든 텍스트를 선택하거나 start 와 end 사이의 텍스트를 선택한다.
<code>void setSelectionStart(int start)</code> <code>void setSelectionEnd(int end)</code> <code>int getSelectionStart()</code> <code>int getSelectionEnd()</code>	현재 선택된 위치를 설정하거나 반환한다.

© 2009 인피니티박스 All rights reserved

■ 텍스트 편집 기능

메소드	설명
<code>void cut()</code> <code>void copy()</code> <code>void paste()</code>	시스템의 클립보드에 텍스트를 자르거나 복사하거나 붙여넣는다.
<code>void replaceSelection(String s)</code>	선택된 텍스트를 대체한다.

■ 카렛 조작 기능

카렛(caret)이란 입력의 위치를 표시하는 커서이다.

메소드	설명
<code>void setCaretPosition(int p)</code> <code>void moveCaretPosition(int p)</code> <code>int getCaretPosition()</code>	문서내의 카렛의 위치를 설정하거나 반환한다.

■ 텍스트의 입출력

메소드	설명
<code>void read(Reader r, Object obj)</code> <code>void write(Writer w)</code>	텍스트를 읽거나 쓴다.
<code>String getText()</code>	컴포넌트에 입력된 Text 를 반환한다.
<code>setText(String str)</code>	Text 를 str 으로 설정한다.



텍스트 필드

JTextField:	김철수
JPasswordField:	****
JFormattedTextField:	2009. 3. 7

■ 생성자

생성자	설 명
<code>JTextField()</code>	<code>TextField</code> 를 생성한다.
<code>JTextField(int columns)</code>	지정된 칸수를 가지고 있는 <code>TextField</code> 를 생성한다.
<code>JTextField(String text)</code>	지정된 문자열로 초기화된 <code>TextField</code> 를 생성한다.
<code>JTextField(String text, int columns)</code>	지정된 칸수를 가지고 있고, 지정된 문자열로 초기화된 <code>TextField</code> 를 생성한다.

텍스트 필드를 간단히 생성하려면 다음과 같은 한다.

```
JTextField textfield = new JTextField(30); // 30은 칸의 수
```

© 2009 인피니티박스 All rights reserved



예제

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

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

        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);
    }
}
```



예제

```
button = new JButton("OK");
button.addActionListener(listener);
panel.add(button);
add(panel);
setVisible(true);
}
// 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 class TextFieldTest extends JFrame {
    public static void main(String[] args) {
        new MyFrame();
    }
}
```



예제



제곱 계산하기

숫자 입력: 9

제곱한 값: 81

OK



중간 점검 문제

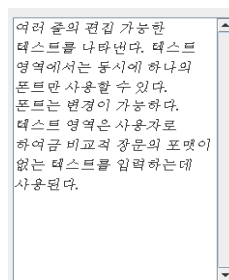
1. 텍스트 필드에서 엔터키나 리턴키를 치면 발생하는 이벤트는 무엇인가?
2. 패스워드를 입력받을 때 사용하는 클래스는?
3. 텍스트 필드에서 `getText()`로 받은 텍스트를 정수로 변환하려면 어떤 방법을 써야 하는가?

© 2009 인피니티박스 All rights reserved



텍스트 영역

- 텍스트 영역(TextArea): 여러 줄의 텍스트가 들어 갈 수 있는 컴포넌트



생성자	설 명
<code>JTextArea()</code>	비어있는 새로운 <code>TextArea</code> 를 생성한다.
<code>JTextArea(int rows, int columns)</code>	지정된 줄수와 칸수의 비어있는 <code>TextArea</code> 를 생성한다.
<code>JTextArea(String text)</code>	지정된 <code>Text</code> 가 입력된 <code>TextArea</code> 를 생성한다.

© 2009 인피니티박스 All rights reserved



예 제

```

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

class MyFrame extends JFrame implements ActionListener {
    protected JTextField textField;
    protected JTextArea textArea;

    public MyFrame() {
        setTitle("Text Area Test");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        textField = new JTextField(30);
        textField.addActionListener(this);

        textArea = new JTextArea(10, 30);
        textArea.setEditable(false);

        add(textField, BorderLayout.NORTH);
        add(textArea, BorderLayout.CENTER);

        pack();
        setVisible(true);
    }
}

```



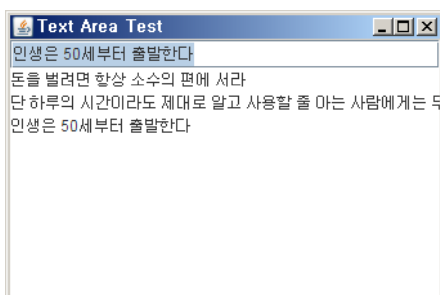
예 제

```

public void actionPerformed(ActionEvent evt) {
    String text = textField.getText();
    textArea.append(text + "\n");
    textField.selectAll();
    textArea.setCaretPosition(textArea.getDocument().getLength());
}

public class TextAreaTest extends JFrame {
    public static void main(String[] args) {
        new MyFrame();
    }
}

```





중간 점검 문제

1. 버튼을 누르면 "I've clicked"라는 문자열을 텍스트 영역에 추가하는 프로그램을 설계하고 작성하여 보라.
2. 텍스트 영역에다 텍스트를 추가하는 메소드는?

© 2009 인피니티박스 All rights reserved



스크롤 페인

- 텍스트 영역에 스크롤바를 만들려면 스크롤 페인에 텍스트 영역을 넣어야 한다.

```
textArea = new JTextArea(10, 30); // 텍스트 영역을 생성한다.  
JScrollPane scrollPane = new JScrollPane(textArea); // ①
```

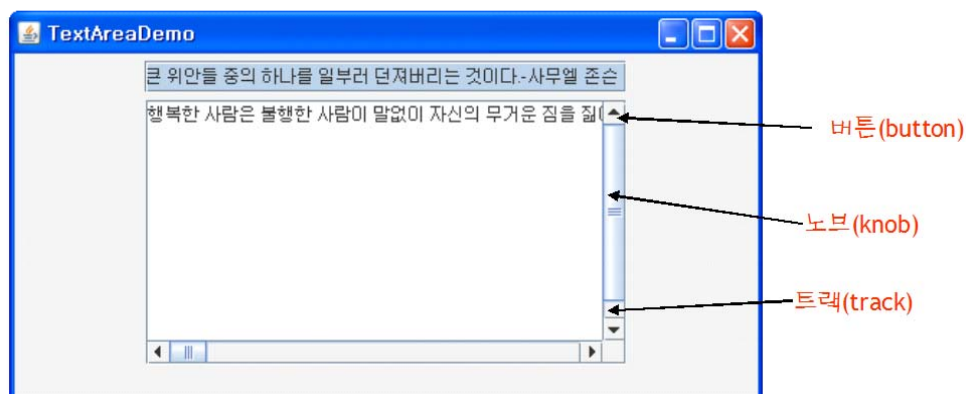


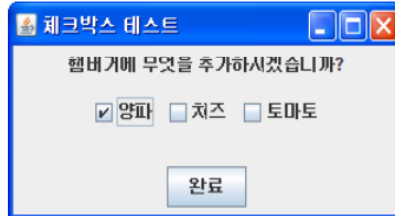
그림 20.4 텍스트 영역을 스크롤 페인에 넣은 결과

© 2009 인피니티박스 All rights reserved



체크 박스

체크 박스(**check box**)란 사용자가 클릭하여서 체크된 상태와 체크되지 않은 상태 중의 하나로 만들 수 있는 컨트롤이다. 체크 박스는 흔히 사용자로 하여금 **YES**와 **NO** 중에서 하나를 선택하게 하는데 사용된다. 그림 21.3은 3개의 체크 박스를 가진 프레임을 보여준다.



© 2009 인피니티북스 All rights reserved



JCheckBox의 메소드

■ 생성자와 메소드

	이름	설 명
생 성 자	<code>JCheckBox()</code>	레이블이 없는 Checkbox 를 생성한다.
	<code>JCheckbox(String label)</code>	지정된 레이블의 Checkbox 를 생성한다.
	<code>JCheckbox(String label, boolean selected)</code>	지정된 상태와 레이블을 가지는 Checkbox 을 생성한다.
메 소 드	<code>String getText()</code>	체크 박스에 표시되는 텍스트를 가져온다.
	<code>Boolean isSelected()</code>	만약 체크 박스가 선택되었으면 true 를 반환한다.
	<code>void setSelected(boolean value)</code>	매개 변수가 true 이면 체크 박스를 체크 상태로 만든다.
	<code>void setText(String text)</code>	체크 박스 텍스트를 설정한다.
	<code>void setToolTipText(String text)</code>	사용자가 체크 박스 위에 마우스 커서를 멈추면 표시되는 툴팁 을 설정한다.

© 2009 인피니티북스 All rights reserved



예제

```
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.applet.Applet;

import javax.swing.*;

class MyFrame extends JFrame implements ActionListener {

    private JButton buttonOK;
    private JCheckBox onion, cheese, tomato;

    public MyFrame() {
        setTitle("체크박스 테스트");
        setSize(300, 130);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        JLabel label = new JLabel("햄버거에 무엇을 추가하시겠습니까?");
        JPanel topPanel = new JPanel();
        topPanel.add(label);
        add(topPanel, BorderLayout.NORTH);
    }
}
```

© 2009 인피니티박스 All rights reserved



예제

```
// 체크 박스 생성 및 추가
JPanel panel = new JPanel();
onion = new JCheckBox("양파");
panel.add(onion);
cheese = new JCheckBox("치즈");
panel.add(cheese);
tomato = new JCheckBox("토마토");
panel.add(tomato);
add(panel, BorderLayout.CENTER);

// 버튼 생성 및 추가
buttonOK = new JButton("OK");
JPanel bottomPanel = new JPanel();
bottomPanel.add(buttonOK);
add(bottomPanel, BorderLayout.SOUTH);
buttonOK.addActionListener(this);

setVisible(true);
}
```

© 2009 인피니티박스 All rights reserved



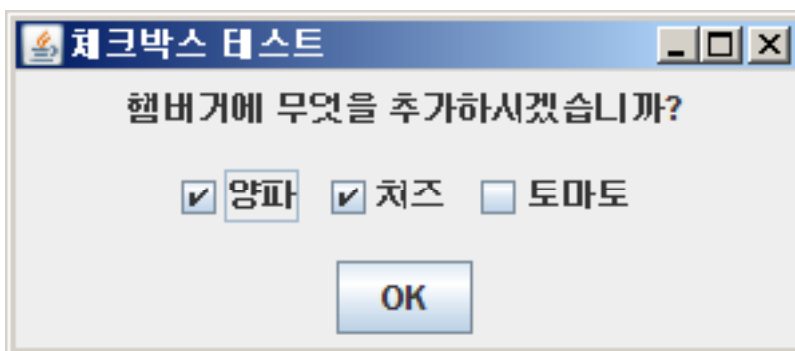
예제

```
public void actionPerformed(ActionEvent e) {
    if (e.getSource() == buttonOK) {
        String msg = "";
        if (onion.isSelected())
            msg += "양파\n";
        if (cheese.isSelected())
            msg += "치즈";
        if (tomato.isSelected())
            msg += "토마토";
        msg = "선택한 옵션은 다음과 같습니다.\n" + msg;
        System.out.println(msg);
        onion.setSelected(false);
        cheese.setSelected(false);
        tomato.setSelected(false);
    }
}

public class MyFrameTest extends JFrame {
    public static void main(String[] args) {
        new MyFrame();
    }
}
```



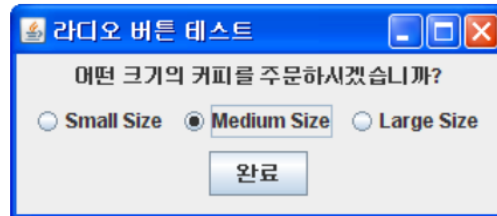
예제





라디오 버튼

라디오 버튼은 체크 박스와 비슷하지만 하나의 그룹 안에서는 한 개의 버튼만 선택할 수 있다는 점이 다르다. 만약 하나의 라디오 버튼을 클릭하면 다른 버튼은 자동적으로 선택이 해제된다.



© 2009 인피니티박스 All rights reserved



생성자와 메소드

	이름	설 명
생	<code>JRadioButton(String text)</code>	라디오 버튼을 생성한다.
성 자	<code>JRadioButton(String text, boolean selected)</code>	초기 상태를 가지는 라디오 버튼을 생성한다.

- `JRadioButton radio1 = new JRadioButton("Small Size");`
- `JRadioButton radio1 = new JRadioButton("Small Size", true);`

© 2009 인피니티박스 All rights reserved



그룹핑 하기

```
// 라디오 버튼을 생성한다.  
JRadioButton radio1 = new JRadioButton("선택 1");  
JRadioButton radio2 = new JRadioButton("선택 2");  
JRadioButton radio3 = new JRadioButton("선택 3");  
  
// ButtonGroup 객체를 생성한다.  
ButtonGroup group = new ButtonGroup();  
  
// 라디오 버튼들을 ButtonGroup 객체에 추가한다.  
group.add(radio1);  
group.add(radio2);  
group.add(radio3);
```

© 2009 인피니티박스 All rights reserved



예제

```
import javax.swing.*;  
import javax.swing.border.Border;  
  
import java.awt.event.*;  
import java.awt.*;  
  
class MyFrame extends JFrame implements ActionListener {  
  
    private JRadioButton small, medium, large;  
    private JLabel text;  
    private JPanel topPanel, sizePanel, resultPanel;  
  
    public MyFrame() {  
        setTitle("라디오 버튼 테스트");  
        setSize(300, 150);  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
  
        topPanel = new JPanel();  
        JLabel label = new JLabel("어떤 크기의 커피를 주문하시겠습니까?");  
        topPanel.add(label);  
        add(topPanel, BorderLayout.NORTH);  
    }  
}
```

© 2009 인피니티박스 All rights reserved



예 제

```
sizePanel = new JPanel();
small = new JRadioButton("Small Size");
medium = new JRadioButton("Medium Size");
large = new JRadioButton("Large Size");

ButtonGroup size = new ButtonGroup();
size.add(small);
size.add(medium);
size.add(large);
small.addActionListener(this);
medium.addActionListener(this);
large.addActionListener(this);
sizePanel.add(small);
sizePanel.add(medium);
sizePanel.add(large);
add(sizePanel, BorderLayout.CENTER);

resultPanel = new JPanel();
text = new JLabel("크기가 선택되지 않았습니다.");
text.setForeground(Color.red);
resultPanel.add(text);
add(resultPanel, BorderLayout.SOUTH);
setVisible(true);
}
```



예 제

```
public void actionPerformed(ActionEvent e) {
    if (e.getSource() == small) {
        text.setText("Small 크기가 선택되었습니다.");
    }
    if (e.getSource() == medium) {
        text.setText("Medium 크기가 선택되었습니다.");
    }
    if (e.getSource() == large) {
        text.setText("Large 크기가 선택되었습니다.");
    }
}

public class RadioButtonTest extends JFrame {
    public static void main(String[] args) {
        new MyFrame();
    }
}
```




예제



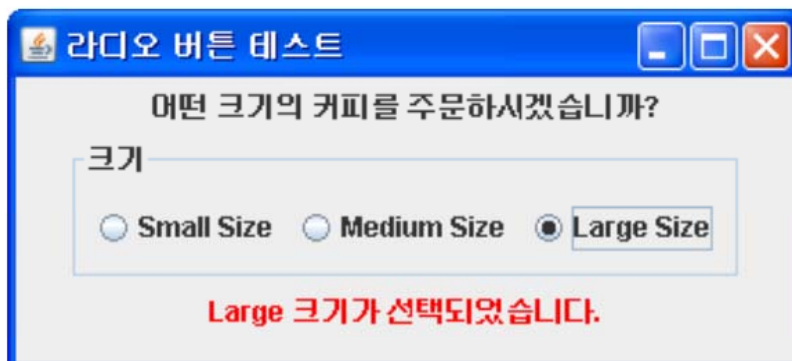
© 2009 인피니티박스 All rights reserved



경계 만들기

앞의 라디오 버튼 예제에 경계를 생성하려면 `Border` 객체를 생성하고 `sizePanel`에 경계를 설정하여야 한다.

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



© 2009 인피니티박스 All rights reserved



Q & A

