

JTextArea

JTextArea es una parte del paquete java Swing. Representa un área de varias líneas que muestra texto. Se utiliza para editar el texto .

JTextArea hereda la clase JComponent. El texto en JTextArea se puede establecer en diferentes fuentes disponibles y se puede anexar al texto nuevo. Un área de texto se puede personalizar según las necesidades del usuario.

Los constructores de JTextArea son:

1. **JTextArea()** : construye una nueva área de texto en blanco.
2. **JTextArea(String s)**: construye una nueva área de texto con un texto inicial dado.
3. **JTextArea(int row, int column)** : construye una nueva área de texto con un número determinado de filas y columnas.
4. **JTextArea(String s, int row, int column)** : construye una nueva área de texto con un número determinado de filas y columnas y un texto inicial dado.

Métodos comúnmente utilizados:

1. **append(String s)** : anexa la cadena dada al texto del área de texto.
2. **getLineCount()** : obtener el número de líneas en el texto del área de texto.
3. **setFont(Font f)** : establece la fuente del área de texto en la fuente dada.
4. **setColumns(int c)** : establece el número de columnas del área de texto en un entero dado.
5. **setRows(int r)** : establece el número de filas del área de texto en un entero dado.
6. **getColumns()** : obtener el número de columnas del área de texto.
7. **getRows()** : obtener el número de filas del área de texto.

1. Programa para crear un JTextArea simple

```
// Java Program to create a simple JTextArea
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
class text extends JFrame implements ActionListener {

    // JFrame
    static JFrame f;

    // JButton
    static JButton b;

    // label to display text
    static JLabel l;

    // text area
    static JTextArea jt;
```

```
// default constructor
text()
{
}

// main class
public static void main(String[] args)
{
    // create a new frame to store text field and button
    f = new JFrame("textfield");

    // create a label to display text
    l = new JLabel("nothing entered");

    // create a new button
    b = new JButton("submit");

    // create a object of the text class
    text te = new text();

    // addActionListener to button
    b.addActionListener(te);

    // create a text area, specifying the rows and columns
    jt = new JTextArea(10, 10);

    JPanel p = new JPanel();

    // add the text area and button to panel
    p.add(jt);
    p.add(b);
    p.add(l);

    f.add(p);
    // set the size of frame
    f.setSize(300, 300);

    f.show();
}

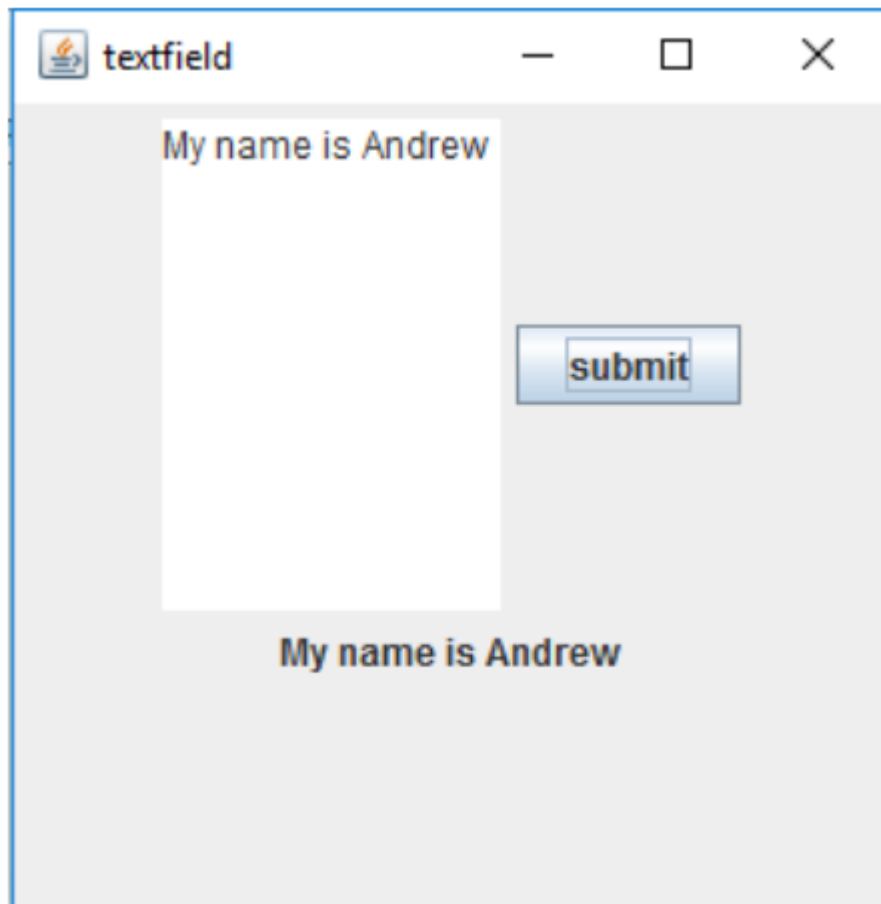
// if the button is pressed
public void actionPerformed(ActionEvent e)
{
    String s = e.getActionCommand();
    if (s.equals("submit")) {
        // set the text of the label to the text of the field
    }
}
```

```

        l.setText(jt.getText());
    }
}

```

Salida:



2. Programa para crear un JTextArea y establecer un texto inicial y agregar botones para cambiar la fuente del área de texto.

```

// Java Program to create a JTextArea and
// set a initial text and add buttons to change
// the font of text area.
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
class text11 extends JFrame implements ActionListener {

    // JFrame
    static JFrame f;

    // JButton
    static JButton b, b1, b2, b3;

```

```
// label to display text
static JLabel l, l1;

// text area
static JTextArea jt;

// default constructor
text11()
{
}

// main class
public static void main(String[] args)
{
    // create a new frame to store text field and button
    f = new JFrame("textfield");

    // create a label to display text
    l = new JLabel("nothing entered");
    l1 = new JLabel("0 lines");

    // create a new buttons
    b = new JButton("submit");
    b1 = new JButton("plain");
    b2 = new JButton("italic");
    b3 = new JButton("bold");

    // create a object of the text class
    text11 te = new text11();

    // addActionListener to button
    b.addActionListener(te);
    b1.addActionListener(te);
    b2.addActionListener(te);
    b3.addActionListener(te);

    // create a text area, specifying the rows and columns
    jt = new JTextArea("please write something ", 10, 10);

    JPanel p = new JPanel();

    // add the text area and button to panel
    p.add(jt);
    p.add(b);
    p.add(b1);
    p.add(b2);
```

```
p.add(b3);
p.add(l);
p.add(l1);


f.add(p);
// set the size of frame
f.setSize(300, 300);

f.show();
}

// if the button is pressed
public void actionPerformed(ActionEvent e)
{
    String s = e.getActionCommand();
    if (s.equals("submit")) {
        // set the text of the label to the text of the field
        l.setText(jt.getText() + ", ");
        l1.setText(jt.getLineCount() + " lines");
    }
    else if (s.equals("bold")) {

        // set bold font
        Font f = new Font("Serif", Font.BOLD, 15);
        jt.setFont(f);
    }
    else if (s.equals("italic")) {
        // set italic font
        Font f = new Font("Serif", Font.ITALIC, 15);
        jt.setFont(f);
    }
    else if (s.equals("plain")) {
        // set plain font
        Font f = new Font("Serif", Font.PLAIN, 15);
        jt.setFont(f);
    }
}
}
```

Salida:


 textfield

My name is Andrew
student of JU

submitplain

italicbold

My name is Andrew student of JU , 2 lines

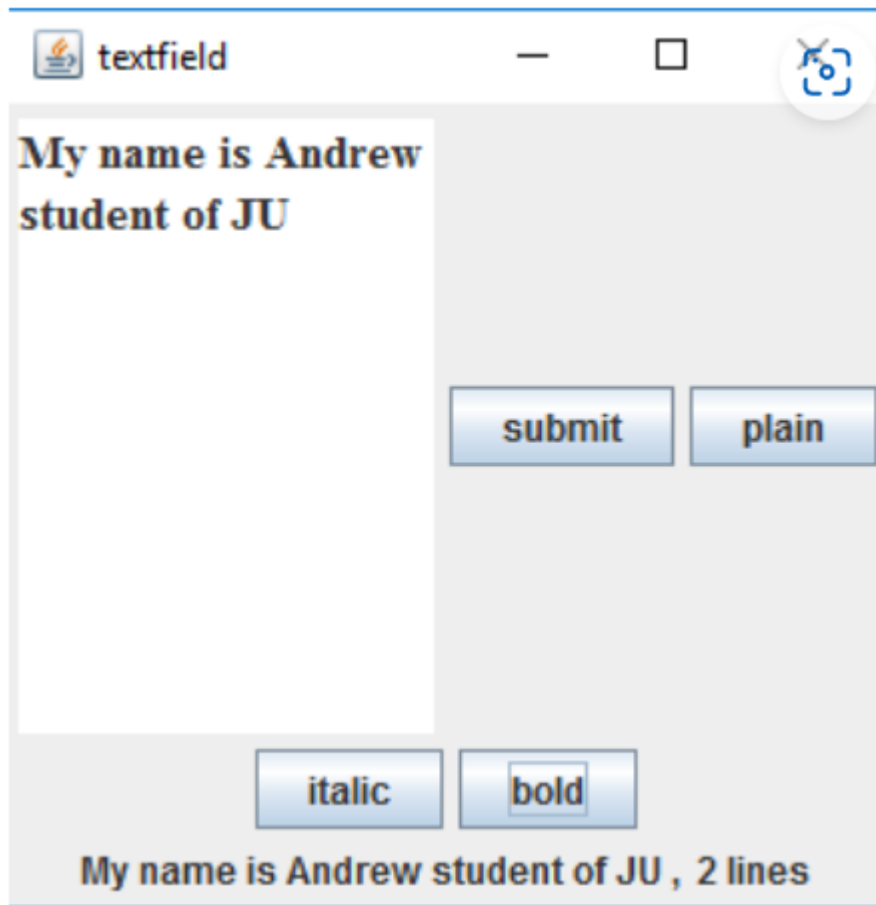
 textfield

*My name is Andrew
student of JU*

submitplain

italicbold

My name is Andrew student of JU , 2 lines



Bibliografía:

- Extraído de [Java Swing | JTextArea](#). Consultado el 28 de noviembre del 2022