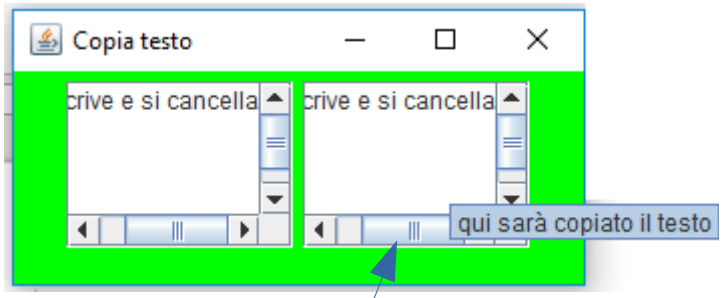
Aree di testo ed eventi associati al *documento*




```
/**
 * GuiText.java
 * applicazione con uso di aree di testo
 * ed eventi swing
 * @author  classi 4^
 * @version 1.00 2018/2/19
 */
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.event.*;  // non esiste DocumentAdapter

public class GuiText extends JFrame implements DocumentListener {

        private Container c;
        private JPanel p;
        private JTextArea tA, tB;
        private JScrollPane tabA, tabB;   // se si desiderano barre a scorrimento

        public GuiText(){ super();
        }
        public GuiText(String s){ super(s);
        }
                                                  // si usano due metodi di DocumentListener
        public void insertUpdate (DocumentEvent ev){   // si vuole copiare il testo
              String s = tA.getText();
              tB.setText(s);     // non si accoda: si riscrive
        }
        public void removeUpdate (DocumentEvent e){  // ed anche le cancellazioni
               String s = tA.getText();
               tB.setText(s);
         }
         public void changedUpdate (DocumentEvent e) {    // mai chiamato
         }
         public void ini(){
                  c= getContentPane();
                  setSize(300, 145); // misure in pixel: larghezza, altezza
                  setLocation(200,100); // (0,0) angolo sup. sin
                  p = new JPanel();
                  p.setBackground(Color.green); // sfondo del pannello colorato
                 tA = new JTextArea (5,30);      // righe, colonne
                 tA.getDocument().addDocumentListener(this); // notifica evento all'ascoltatore
                 tB =new JTextArea (5,30);
                 tB.setEditable(false);
                 tB.setToolTipText("qui sarà copiato il testo");    // visualizza al passaggio del mouse
                tabA = new JScrollPane(tA);  // volendo visualizzare eventuali barre a scorrimento
                tabB = new JScrollPane(tB);
                p.add(tabA);
                p.add(tabB);
                c.add(p); // aggiunge il pannello
                setVisible(true); // mostra il frame
                setDefaultCloseOperation (JFrame.EXIT_ON_CLOSE);
        }
        public static void main(String[] args) {    GuiText gui= new GuiText("Copia testo");
                                                     gui.ini();
        }
}
```

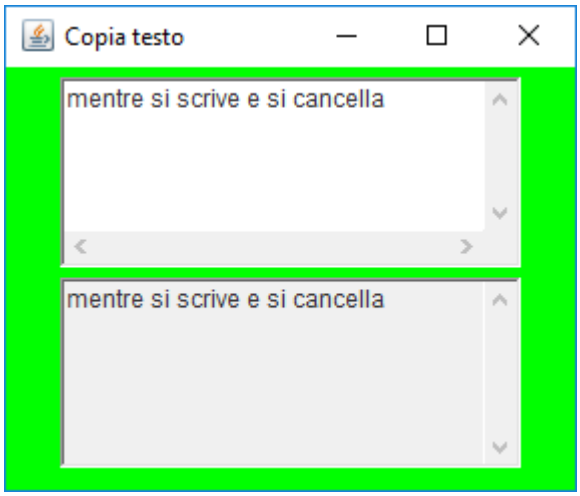
Aree di testo ed eventi associati al *testo*

```
/**
 * GuiArea.java
 *
 * applicazione con aree di testo (componenti awt) ed eventi
 *
 * @author classi 4^
 * @version 1.00 2018/2/19
 */
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class GuiArea extends JFrame implements TextListener {

        private Container c;
        private JPanel p;
        private TextArea tA, tB;  // componenti awt  con barre a scorrimento
        public GuiArea(){ super();
        }
        public GuiArea(String s){ super(s);
        }
        public void textValueChanged (TextEvent ev){
            String s = tA.getText();
            tB.setText(s);          // non si accoda: si riscrive
        }
        public void ini(){
                c= getContentPane();
                setSize(300,250); // misure in pixel: larghezza, altezza
                 setLocation(200,100); // (0,0) angolo sup. sin
                 p = new JPanel();
                 p.setBackground(Color.green); // sfondo del pannello colorato
                tA = new TextArea (5,30);  // righe, colonne
                tA.addTextListener(this); // notifica evento all'ascoltatore
                tB =new TextArea (5,30);
                tB.setEditable(false);
                 p.add(tA);
                 p.add(tB);
                 c.add(p); // aggiunge il pannello
                 setVisible(true); // mostra il frame
                 setDefaultCloseOperation (JFrame.EXIT_ON_CLOSE);
        }
        public static void main(String[] args) {
              GuiArea gui= new GuiArea("Copia testo");
              gui.ini();

        }
}
```



II parte:

*in seguito si **archivi su file di testo** il contenuto*
di una area di testo
e viceversa si visualizzi nell'area di testo il contenuto
letto da file di testo

Aree di testo, accesso a file di testo;
eventi associati al *documento* o al *focus (selezione componente)* o al *mouse*