

# Interfaccia Grafica

## Introduzione

L'interfaccia grafica può essere realizzata con più di una modalità operativa, noi ci baseremo sull'uso di una serie di classi che permettono di istanziare oggetti che supportano sia dal punto di vista informativo sia da quello grafico la realizzazione di componenti necessari per comporre interfacce grafiche. Queste classi fanno parte del package swing.

## Passo 1: istanza della finestra

### Il frame

Facciamo uso del package swing:

```
import javax.swing.*;
```

Definiamo per l'oggetto di vista un attributo che corrisponda alla cornice della finestra dell'interfaccia grafica: un oggetto JFrame.

Il costruttore dell'oggetto di vista si occupa di istanziare la finestra

```
package amico;
import javax.swing.*;

/**
 *
 * @author ragno
 */
public class VistaContattoGui {
    /** la finestra */
    JFrame finestra;

    public VistaContattoGui() {
        //costruzione della finestra
        //costruisco gli oggetti
        finestra=new JFrame();
    }
}
```

### Visibilità

La finestra esiste ma non si vede nulla, per cui occorre abilitarne la visibilità agendo su un attributo visible che assume valori booleani (true, false); per questo scopo utilizziamo il relativo metodo setter: setVisible passando parametro effettivo a valore true (vero)

```
public VistaContattoGui() {
    //costruzione della finestra
    //costruisco gli oggetti
    finestra=new JFrame();
    //rendi visibile la finestra
```

```
    finestra.setVisible(true);  
}
```

## I thread

Ora la finestra compare, ma è una piccola porzione di barra del titolo in cui compare una icona a sinistra (la classica tazzina java) e i tre bottoni 'standard' a destra. Notiamo che la finestra è comunque ridimensionabile secondo le modalità standard.

Si noti che il programma è in stato running, ma apparentemente non succede nulla.

In effetti il metodo main ha svolto i suoi passi: istanza dell'oggetto di vista, che a sua volta ha istanziato e reso visibile un JFrame, poi il main avendo compiuto tutto deve solo chiudere. Ma come vediamo questo non accade.

In realtà cioè che succede è che con la visualizzazione dell'oggetto JFrame si attiva un "processo parallelo", un "thread", che affianca il thread del metodo main (main thread).

La terminazione del metodo main non è più quindi l'unica condizione per chiudere l'intera applicazione.

Fino a che rimane attivo il thread associato alla finestra l'applicazione non viene chiusa.

Se si chiude la finestra con l'apposito bottone (x) in altro a destra si può osservare che l'applicazione non è ancora chiusa. Chiusura di finestra e termine del thread non sono la stessa cosa.

Si può però impostare le proprietà della finestra in modo che la chiusura della finestra comporti la terminazione (exit) del thread, per questo obiettivo utilizziamo il metodo setDefaultCloseOperation come nell'esempio segue:

```
public VistaContattoGui() {  
    //costruzione della finestra  
    //costruisco gli oggetti  
    finestra=new JFrame();  
    //aggancia l'evento di chiusura finestra per terminare  
    finestra.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    //rendi visibile la finestra  
    finestra.setVisible(true);  
}
```

Si può ora verificare che la chiusura della finestra comporta la completa terminazione dell'applicazione.

## Dimensione del frame

Abbiamo visto che l'operatore può ridimensionare la finestra secondo le modalità tipiche per le finestre standard, ma è preferibile che il frame sia reso visibile con una dimensione che ne dia una immediata percezione di finestra.

A questo scopo la classe JFrame offre il metodo setSize che prevede due parametri rispettivamente larghezza e altezza:

```
public VistaContattoGui() {  
    //costruzione della finestra  
    //costruisco gli oggetti  
    finestra=new JFrame();  
    //aggancia l'evento di chiusura finestra  
    finestra.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    //dimensiona la finestra
```

```

finestra.setSize(200,100);
//rendi visibile la finestra
finestra.setVisible(true);
}

```

## **Passo 2: oggetti nella finestra**

### **Jpanel**

Il frame che abbiamo predisposto è un contenitore vuoto, per poter visualizzare il testo previsto dalla nostra applicazione per prima cosa occorre “riempire” la finestra con un “pannello” in cui sarà poi possibile inserire gli elementi di visualizzazione veri e propri.

Per questo scopo è disponibile la classe JPanel, l'oggetto di vista quindi avrà un attributo di questa classe, il costruttore prevede ad istanziarlo e poi ad inserirlo nel frame con il metodo add:

```

public class VistaContattoGui {
    /** la finestra */
    JFrame finestra;
    /** il pannello */
    JPanel pannello;

    public VistaContattoG() {
        //costruzione della finestra
        //costruisco gli oggetti
        finestra=new JFrame();
        pannello=new JPanel();
        //inizializzo la finestra
        finestra.add(pannello);
        finestra.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        finestra.setSize(200,300);
        finestra.setVisible(true);
    }
}

```

### **Jlabel**

Per l'inserimento del testo faremo uso di un oggetto etichetta che si ottiene istanziando la classe JLabel, questo oggetto va inserito nel panel:

```

public class VistaContattoGui {
    /** la finestra */
    JFrame finestra;
    /** il pannello */
    JPanel pannello;
    /** l'etichetta */
    JLabel etichetta;

    public VistaContattoG(String titolo) {

```

```

//costruzione della finestra
//costruisco gli oggetti
finestra=new JFrame();
pannello=new JPanel();
etichetta = new JLabel();
//inizializzo la finestra
finestra.add(pannello);
pannello.add(etichetta);
finestra.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
finestra.setSize(200,300);
finestra.setVisible(true);
}
}

```

### **Passo 3 metodo visualizza**

L'effettiva produzione del messaggio va affidata al metodo visualizza, che riceve i dati dal parametro contatto e ne deriva una stringa che viene posta nell'etichetta con il metodo setText:

```

/**
 * visualizza i dati in linea
 * @param c il contatto da visualizzare
 */
public void visualizza(Contatto c){
    etichetta.setText(c.getNome() + " " + c.getCognome() + " " +
c.getTelefono());
}

```

cc

---

cc Quest'opera è stata rilasciata con licenza Creative Commons Attribuzione - Non commerciale - Condividi allo stesso modo 3.0 Italia. Per leggere una copia della licenza visita il sito web <http://creativecommons.org/licenses/by-nc-sa/3.0/it/> o spedisci una lettera a Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.  
Giovanni Ragno – ITIS Belluzzi Bologna 2011-12