Creazione e uso dei contenitori standard

La classe **JFrame** ha i seguenti metodi costruttori:

- JFrame() Crea una finestra senza titolo
- JFrame("*Stringa*") Crea una finestra senza titolo specificato in *Stringa*

Altri metodi che si applicano ad un oggetto della classe JFrame:

- setSize(larghezza, altezza) Dimensiona la finestra
- setLocation(x, y) Posiziona la finestra (coordinate del punto in alto a sinistra)
- pack() Ottimizza le dimensioni della finestra a seconda del contenuto
- setVisible(boolean) Per rendere visibile la finestra se *true* (deprecato il metodo show())
- setResizable(boolean) Per rendere ridimensionabile con mouse se *true*
- dispose() Per chiudere la finestra, deallocando le risorse

La classe **JPanel** ha il seguente metodo costruttore:

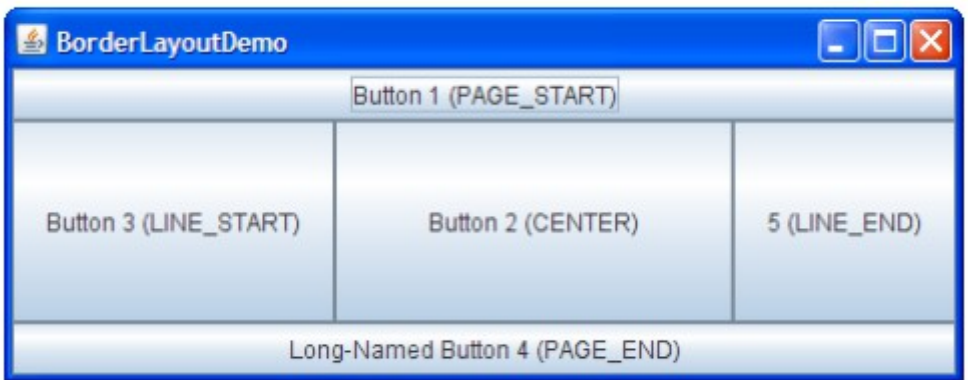
- JPanel () Crea un pannello per inserire componenti GUI

L'inserimento degli oggetti all'interno dei contenitori si realizza con il **metodo add()** con la sintassi:

oggettoContenitore.**add** (oggettoDaAggiungere)

Tale metodo è *polimorfo* cioè opera su argomenti di diverse classi.

nb: è possibile aggiungere un componente e contemporaneamente sceglierne la posizione nel contenitore (con possibile retrocompatibilità: West, North, East, South ed equivalenti costanti in swing)

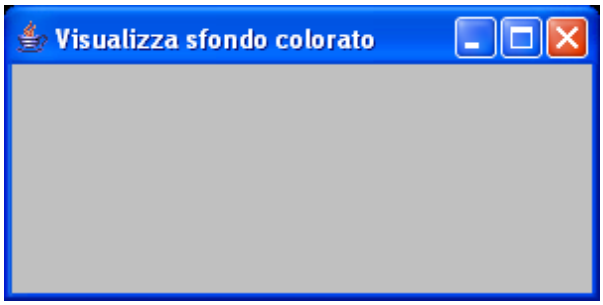


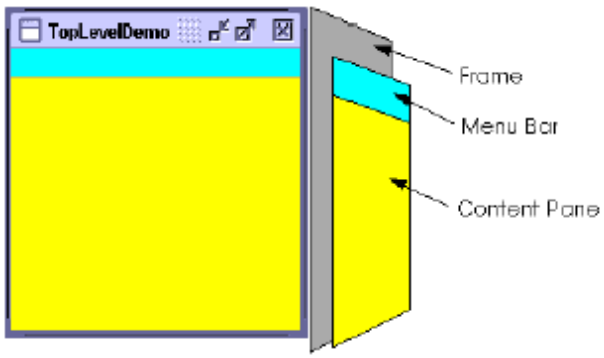
oggettoContenitore.add (oggettoDaAggiungere, *posizione*)

es: nomeContenitore.add (nomeComponente, BorderLayout.CENTER);
nomeContenitore.add (nomeComponente. BorderLayout.PAGE_END);

// esempio di **Applicazione** con uso di **oggetti GUI** in ambiente JCreator[1]

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

public class Finestra {

        private  JFrame f ;
        private  Container c;
        private  JPanel p;

        public Finestra ()    {    // costruttore

             f = new JFrame("Visualizza sfondo colorato");                    // crea frame o cornice invisibile
             c = f.getContentPane();                          // recupera il "muro grezzo"
                                                              // cioè  il riquadro dei contenuti senza barra dei menù

                                                  // per impostare le dimensioni e la posizione:
             f.setSize(300,150);                  // misure in pixel: larghezza, altezza
             f.setLocation(200,100);              // (0,0) angolo sup. sin
             p = new JPanel();
             p.setBackground(Color. lightGray);       // sfondo del pannello colorato

             c.add(p);                // aggiunge il pannello

             f.setVisible(true);    // mostra il frame (dimensioni 300x150)   -  deprecato show()

             f.setDefaultCloseOperation (JFrame.EXIT_ON_CLOSE);     // rende attiva l'icona di chiusura
                                                                    // nella barra del titolo
         }

        public static void main(String [] args) {
             Finestra o = new Finestra();       // creo oggetto che "ha - un"  oggetto JFrame ed un oggetto JPanel
        }
}
```





nb:
setDefaultCloseOperation(JFrame.FLAG);
questo metodo permette di modificare l'azione di default di Java cioè continuare ad eseguire l'applicazione che ha generato il frame anche dopo aver clickato sull'apposito bottone di chiusura del frame; segue un elenco del significato del campo **flag** :

EXIT_ON_CLOSE : terminazione del programma quando il frame viene chiuso

DISPOSE_ON_CLOSE : chiude e libera il solo oggetto frame, il processo continua

DO_NOTHING_ON_CLOSE : mantiene aperto il frame e continua il processo (ignoriamo l'evento)

HIDE_ON_CLOSE : chiude il frame e continua l'esecuzione

Per nascondere i frames: **setVisible(false)** - deprecato **hide()**

---

1 Introduzione all'uso e mirror per download anche nel sito potendo usare JavaDoc