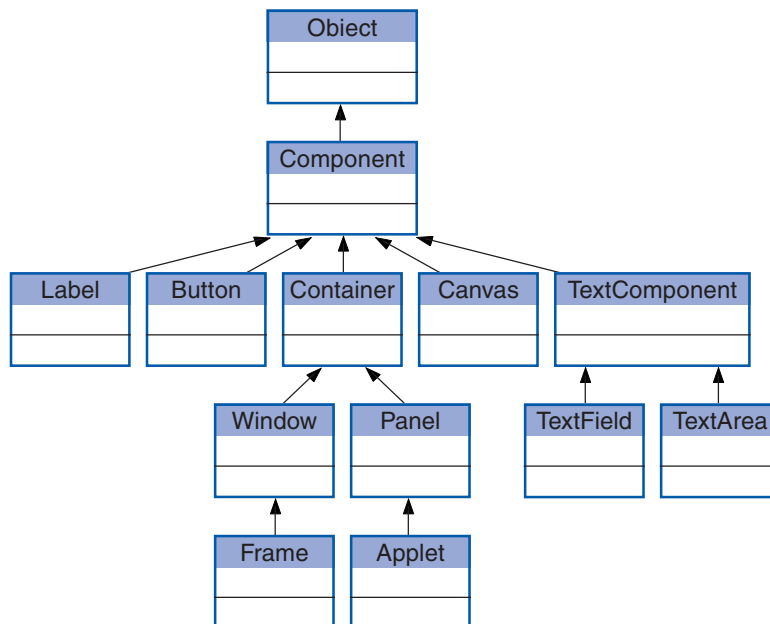


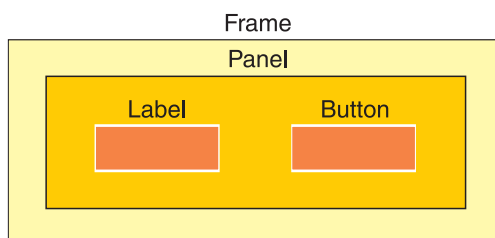
## I contenitori in AWT

Le classi di Java che realizzano le componenti e i contenitori, sono organizzate in una **gerarchia delle componenti** che ha come padre la classe **Component**.



I quattro contenitori in AWT sono rappresentati dalle sottoclassi della classe **Container**: *Window* e *Frame*, *Panel* e *Applet*.

Lo schema seguente rappresenta l'organizzazione delle componenti AWT per una generica finestra (*Frame*) contenente un pannello (*Panel*), al cui interno sono collocati un'etichetta e un pulsante.



### PROGETTO

#### Creare una finestra grafica composta da un'etichetta e un pulsante.

L'interfaccia grafica da creare è quella rappresentata con lo schema precedente, e cioè un'etichetta e un pulsante all'interno di un pannello, inserito all'interno di una finestra.

Per utilizzare le classi grafiche AWT bisogna importare il relativo package nel seguente modo:

```
import java.awt.*;
```

Per ogni elemento grafico, viene creato un oggetto come istanza della relativa classe. In particolare i contenitori vengono costruiti con le seguenti istruzioni:

```
Frame f = new Frame();  
Panel p = new Panel();
```

Le componenti invece sono costruite istanziando la classe e indicando come parametro il testo da visualizzare nella componente, come mostrano le seguenti istruzioni:

```
Label l = new Label("Etichetta");  
Button b = new Button("Bottone");
```

Per aggiungere una componente al contenitore, si usa il metodo **add**. In sequenza, le componenti atomiche (*l* e *b*) vengono prima aggiunte al contenitore intermedio *p*, che a sua volta è aggiunto al contenitore principale *f*.

Il codice completo per realizzare la finestra con le componenti AWT è riportata di seguito.

#### PROGRAMMA JAVA (FinestraAwt.java)

```
import java.awt.*;  
  
class FinestraAwt  
{  
    public static void main(String argv[])  
    {  
        // Contenitori  
        Frame f = new Frame();  
        Panel p = new Panel();  
  
        // Componenti  
        Label l = new Label("Etichetta");  
        Button b = new Button("Bottone");  
  
        // Aggiunta delle componenti al contenitore intermedio  
        p.add(l);  
        p.add(b);  
  
        // Aggiunta del contenitore intermedio al contenitore principale  
        f.add(p);  
  
        f.setSize(300,200);  
        f.setVisible(true);  
    }  
}
```

La classe *Window*, e quindi anche la sottoclasse *Frame*, dispone dei seguenti metodi per la gestione della finestra.

Metodo	Descrizione
<code>isActive()</code>	Restituisce il valore <i>true</i> se la finestra è attiva, altrimenti <i>false</i> .
<code>isFocused()</code>	Restituisce il valore <i>true</i> se la finestra ha il focus, altrimenti <i>false</i> .
<code>isShowing()</code>	Restituisce il valore <i>true</i> se la finestra è visibile sullo schermo, altrimenti <i>false</i> .
<code>setAlwaysOnTop(valore)</code>	Se valore è <i>true</i> , la finestra resta sempre visibile sopra tutte le altre finestre, se è <i>false</i> segue il normale comportamento.
<code>setCursor(cursore)</code>	Imposta l'immagine del puntatore del mouse.
<code>setSize(larg, alt)</code>	Modifica le dimensioni della finestra, impostando una larghezza <i>larg</i> e un'altezza <i>alt</i> .
<code>setVisible(valore)</code>	Se valore è <i>true</i> , rende visibile la finestra, se è <i>false</i> la nasconde.
<code>toBack()</code>	Se la finestra è visibile, la sposta in secondo piano e la disattiva togliendole il focus.
<code>toFront()</code>	Se la finestra non è visibile, la porta in primo piano e le assegna il focus.

Il metodo `setCursor` può ricevere come parametro uno dei seguenti valori, attributi statici della classe **Cursor**:

- `Cursor.CROSSHAIR_CURSOR`
- `Cursor.CUSTOM_CURSOR`
- `Cursor.DEFAULT_CURSOR`
- `Cursor.E_RESIZE_CURSOR`
- `Cursor.HAND_CURSOR`
- `Cursor.MOVE_CURSOR`
- `Cursor.N_RESIZE_CURSOR`
- `Cursor.NE_RESIZE_CURSOR`
- `Cursor.NW_RESIZE_CURSOR`
- `Cursor.S_RESIZE_CURSOR`
- `Cursor.SE_RESIZE_CURSOR`
- `Cursor.SW_RESIZE_CURSOR`
- `Cursor.TEXT_CURSOR`
- `Cursor.W_RESIZE_CURSOR`
- `Cursor.WAIT_CURSOR`.

Per esempio, le seguenti istruzioni mantengono la finestra in primo piano e impostano il cursore con l'immagine di attesa.

```
f.setAlwaysOnTop(true);
f.setCursor(Cursor.WAIT_CURSOR);
```

La classe `Window` rappresenta una generica area rettangolare, mentre la sottoclasse `Frame` descrive una finestra grafica più completa in cui è presente il titolo e i bordi, oltre ai pulsanti di ridimensionamento e chiusura in alto a destra.

La classe *Frame*, oltre ai metodi elencati precedentemente, dispone anche di altri due metodi

- **setTitle**(titolo), imposta il titolo visualizzato in alto nella finestra;
- **setResizable**(valore), stabilisce se i bordi della finestra sono ridimensionabili (con *valore* uguale a *true*) oppure no.

La classe *Panel* è il contenitore a cui vengono aggiunte tutte le componenti grafiche per disegnare l'interfaccia grafica dell'applicazione.

L'aggiunta delle componenti al contenitore viene eseguita richiamando il metodo *add* e indicando come parametro la componente da aggiungere. Tra gli altri metodi che consentono la manipolazione delle componenti del contenitore ci sono i seguenti.

Metodo	Descrizione
<code>remove(componente)</code>	Toglie la componente dal contenitore.
<code>removeAll()</code>	Toglie tutte le componenti dal contenitore.
<code>getComponentCount()</code>	Restituisce il numero di componenti.
<code>getComponentAt(x, y)</code>	Restituisce la componente sul pixel di coordinate (x,y).