

GRAFICA BI-DIMENSIONALE IN JAVA

<http://www.disi.unige.it/person/MagilloP/INTERF03/LEZGRAF/grajava.html>

Esempio

Pannello con grafica personalizzato: [ExDraw1.java](#).

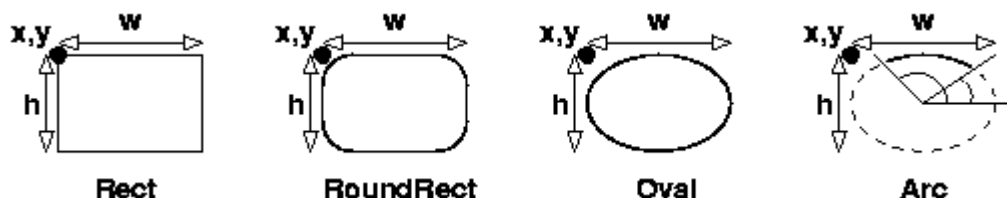
La funzione **paintComponent** riempie un rettangolo sfumato, vi disegna dentro un fumetto con scritto grande "Ciao!", traccia con tratto spesso tre linee alla base del fumetto.



Disegno diretto

Metodi per disegnare direttamente una figura. I parametri sono interi esprimenti numero di pixel, cioè le primitive vanno fornite direttamente in *device space*.

- `clearRect(x,y, width,height)`: pulisce il rettangolo indicato riempiendolo col colore di sfondo.
- `void draw/fillRect(x,y, width,height)`: Disegna **rettangolo** vuoto o pieno.
- `void draw/fillRoundRect(x,y, width,height, arcWidth,arcHeight)`: Disegna **rettangolo con angoli smussati**, vuoto o pieno.
- `void draw/fill3DRect(x,y, width,height, arcWidth,arcHeight)`: Disegna **rettangolo con effetto di rilievo 3D**, vuoto o pieno.
- `draw/fillOval(x,y, width,height)`: Disegna **ellisse**, vuoto o pieno.
- `draw/fillArc(x,y, width,height, startAngle,arcAngle)`: Disegna **arco** circolare o ellittico, vuoto o pieno. Gli angoli sono in gradi, angolo positivo significa rotazione in senso antiorario, negativo in senso orario. Centro dell'arco e' il centro del rettangolo di origine (x,y) e dimensioni width, height.



- `drawLine(x1,y1, x2,y2)`: Disegna **segmento di retta** tra (x1,y1) e (x2,y2).
- `draw/fillPolygon(arrayX, arrayY, n)`: Disegna **poligono** con n vertici definiti dai due array di x e y.
- `drawPolyline(arrayX, arrayY, n)`: Disegna **spezzata poligonale** con n vertici definiti dai due array di x e y.
- `drawString(stringa,x,y)`: Disegna **stringa** iniziando dalle coordinate (x,y). La stringa si estende a destra di x e sopra y, cioè per ascisse maggiori di x e ordinate MINORI di y.
- `drawImage(Image, x,y, ImageObserver)`: Disegna **immagine** alla posizione (x,y). Come image observer si passa il componente stesso (this).
- `drawImage(Image, x,y, width,height, ImageObserver)`: disegna **immagine** alla posizione (x,y), **scalata** alla larghezza ed altezza indicate.

Esempio per disegnare immagine che occupa tutta la componente:

```
Dimension d = getSize();
g.drawImage (image, 0,0, d.width,d.height, this);
```

Tutorial Sun

<http://download.oracle.com/javase/tutorial/2d/TOC.html>