

La dimensione del testo nelle aree grafiche

La classe **FontMetrics** contiene le informazioni sulle caratteristiche dei font usati nelle aree grafiche, tra cui la dimensione, intesa come la larghezza e l'altezza di una stringa di testo. Per ottenere un'istanza di *FontMetrics* si deve richiamare il metodo **getFontMetrics** sull'oggetto di classe *Graphics*, su cui, precedentemente, è stato impostato un font con il metodo *setFont*.

Con l'oggetto *FontMetrics* si possono utilizzare due metodi per ottenere le informazioni sulla dimensione delle stringhe:

- **getHeight**, restituisce il numero di pixel corrispondente all'altezza della stringa in base al font impostato;
- **stringWidth**, restituisce la larghezza in pixel di una stringa, passata come parametro, in base al font impostato.

Le stringhe di testo sono visualizzate nelle aree grafiche con il metodo *drawString* che, nei suoi parametri, consente di indicare la posizione di partenza della stringa. La conoscenza delle dimensioni del testo nelle aree grafiche permette di posizionare e allineare correttamente le stringhe. L'altezza di una stringa, calcolata in base al font impostato, permette di decidere l'interlinea con cui visualizzare più linee di testo senza sovrapporle. Invece la larghezza di una stringa permette di allineare il testo, al centro o a destra, calcolando la giusta posizione da passare al metodo *drawString*.

Il seguente esempio mostra come posizionare, in un'area grafica, due righe di testo con interlinea singola, allineandole al centro.

Implementazione della classe (*AreaGrafica.java*)

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

class AreaGrafica extends JPanel
{
    public void paintComponent(Graphics g)
    {
        super.paintComponent(g);

        String riga1 = "Il linguaggio di programmazione";
        String riga2 = "JAVA";

        // Imposta il font
        Font font = new Font("SansSerif", Font.BOLD, 16);
        g.setFont(font);

        // Calcola altezza e larghezza del testo
        FontMetrics metrica = g.getFontMetrics(font);
        int altezza = metrica.getHeight();
        int larghezza1 = metrica.stringWidth(riga1);
        int larghezza2 = metrica.stringWidth(riga2);

        int w = getSize().width;
        g.drawString(riga1, (w / 2) - (larghezza1 / 2), 50);
        g.drawString(riga2, (w / 2) - (larghezza2 / 2), 50+altezza);
    }
}
```

la posizione della stringa è calcolata in base alla larghezza dell'area grafica (w) e alla larghezza delle stringhe (larghezza1, larghezza2)

Programma Java (*DimensioneTesto.java*)

```
import javax.swing.*;

class DimensioneTesto extends JFrame
{
    public DimensioneTesto()
    {
        AreaGrafica area = new AreaGrafica();

        this.getContentPane().add(area);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public static void main(String args[])
    {
        DimensioneTesto f = new DimensioneTesto();
        f.setTitle("Testo");
        f.setSize(300,200);
        f.setVisible(true);
    }
}
```

