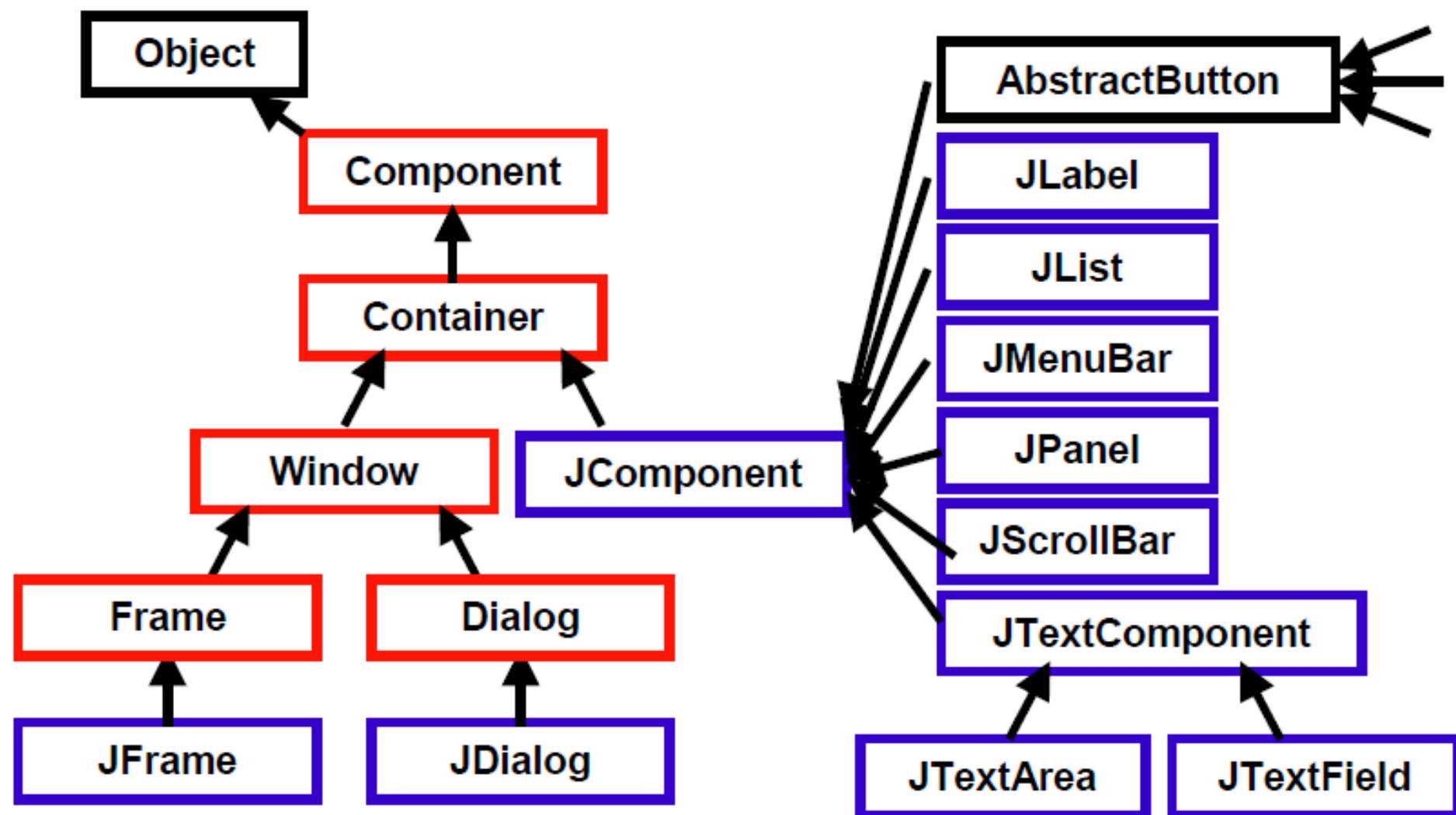


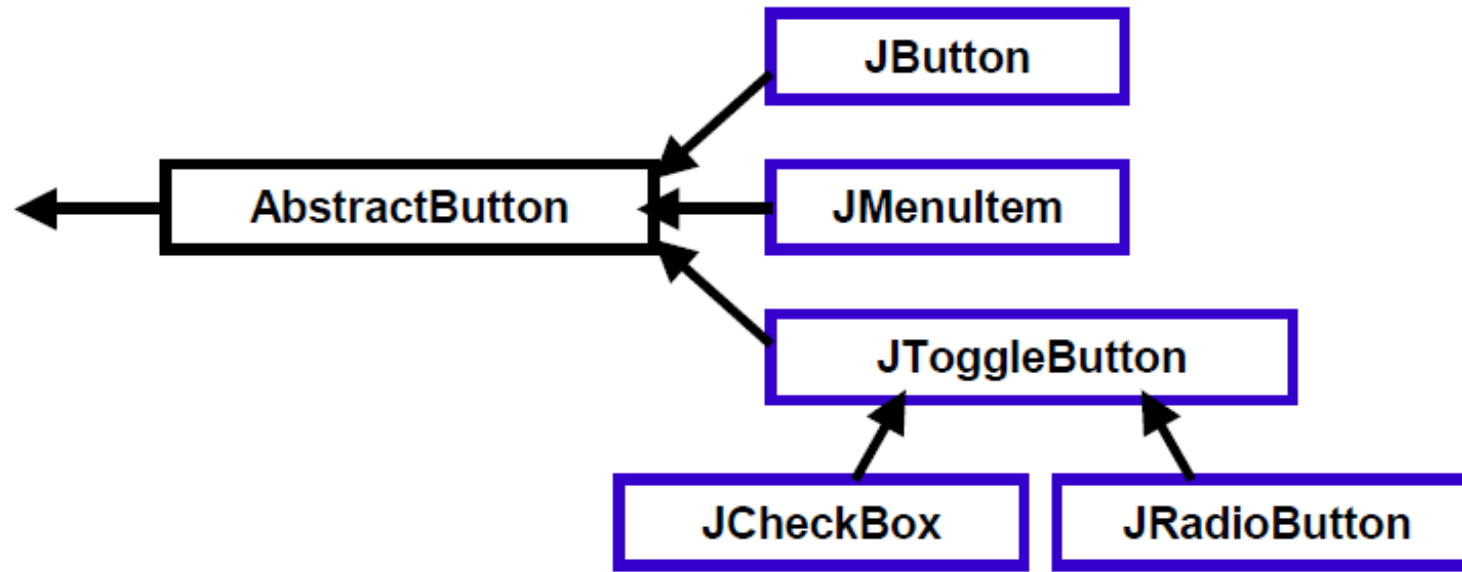
Disegnare su un Panel

JButton
in JAVA SWING

SWING: GERARCHIA DI CLASSI



SWING: GERARCHIA DI CLASSI



Container: tutti i componenti principali sono contenitori, destinati a contenere altri componenti

Window: le finestre sono casi particolari di contenitori e si distinguono in frame e finestre di dialogo

JFrame: componente finestra principale: ha un aspetto grafico, una cornice ridimensionabile e un titolo

JComponent: è il generico componente grafico

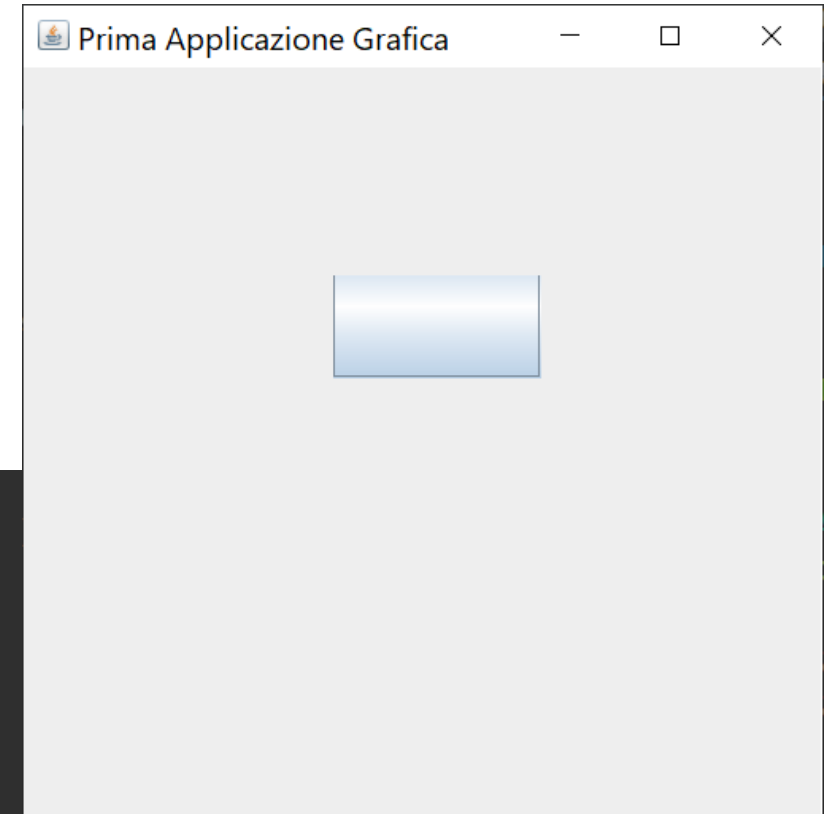
JPanel: il pannello, un componente destinato a contenere altri componenti grafici per organizzarli

IL COMPONENTE JButton

esempio_1

```
public class Main {  
    public static void main(String[] args) {  
        new MyFrame();  
    } //main  
} //class
```

```
public class MyFrame extends JFrame {  
  
    MyFrame(){  
        JButton button = new JButton();  
        button.setBounds(150, 100, 100, 50);  
  
        this.setTitle("Prima Applicazione Grafica");  
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        this.setLayout(null); //no layout Manager set  
        this.setSize(400,400);  
        this.setVisible(true);  
        this.add(button);  
    } //MyFrame  
} //class
```



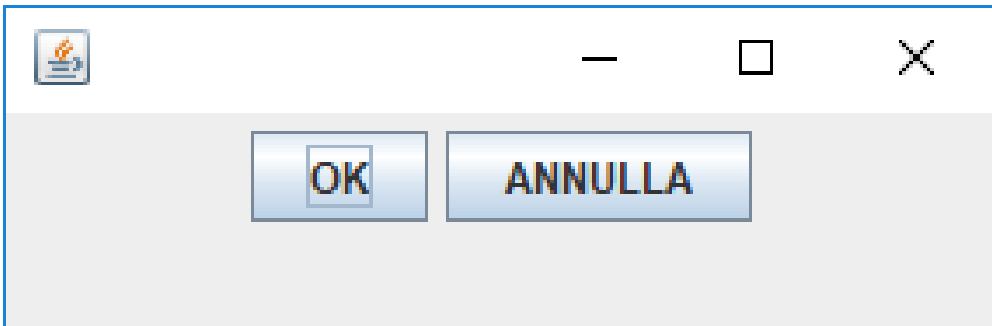
IL COMPONENTE JButton

Un altro esempio, usando un JPanel

```
JButton button_1 = new JButton("OK");  
JButton button_2 = new JButton("ANNULLA");
```

```
public class ImmaginePannello extends JPanel {  
    public ImmaginePannello() {  
        super();  
        JButton button_1 = new JButton("OK");  
        JButton button_2 = new JButton("ANNULLA");  
        add(button_1); add(button_2);  
    } //costruc  
} //class
```

```
public class Main {  
    public static void main(String[] args) {  
        JFrame frame = new JFrame();  
        Container c = f.getContentPane();  
        ImmaginePannello panel = new ImmaginePannello();  
        c.add(panel);  
        frame.setBounds(200,200,250,180);  
        frame.setVisible(true);  
    } //func  
} //class
```



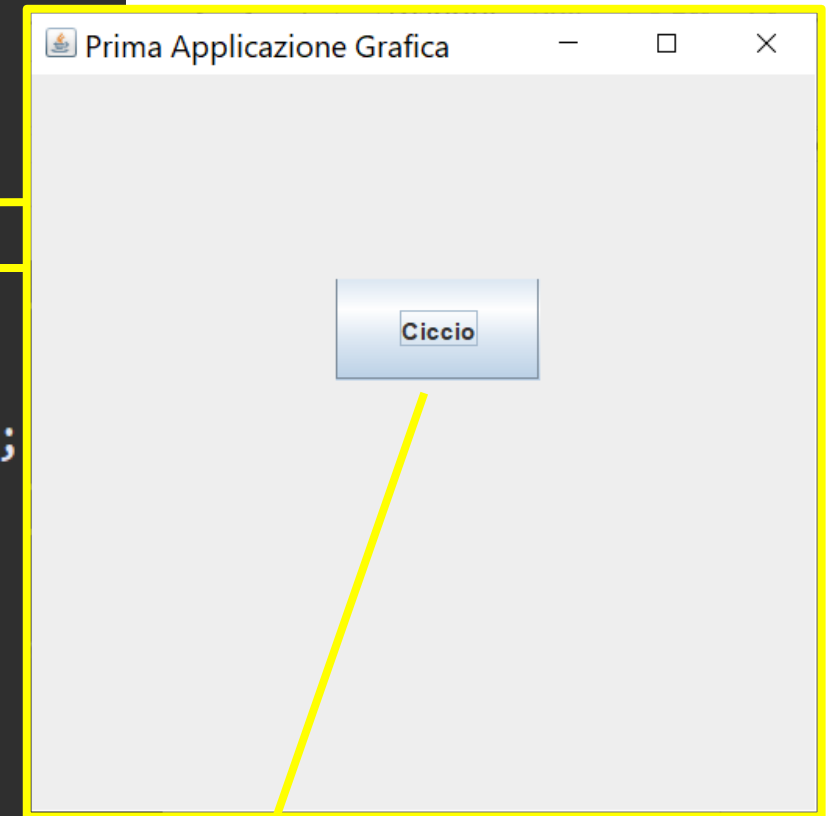
Azione associata a JButton

```
public class MyFrame extends JFrame implements ActionListener {
    JButton button;

    MyFrame(){
        button = new JButton();
        button.setBounds(150, 100, 100, 50);
        button.setText("Ciccio");
        button.addActionListener(this);

        this.setTitle("Prima Applicazione Grafica");
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setLayout(null); //no layout Manager set
        this.setSize(400,400);
        this.setVisible(true);
        this.add(button);
    } //MyFrame

    //implements actionPerformed
    public void actionPerformed(ActionEvent e){
        if (e.getSource()==button){
            System.out.print("Ciccio");
        } //if
    } //func
} //class
```



A screenshot of the IDE's Console window. The title bar shows "Problems", "Javadoc", "Declaration", and "Console". The console output shows "Main (2) [Java Application] C:\Program Files\Java\jdk-16" followed by "CiccioCiccioCiccio" on a new line. The text "CiccioCiccioCiccio" is highlighted in yellow.

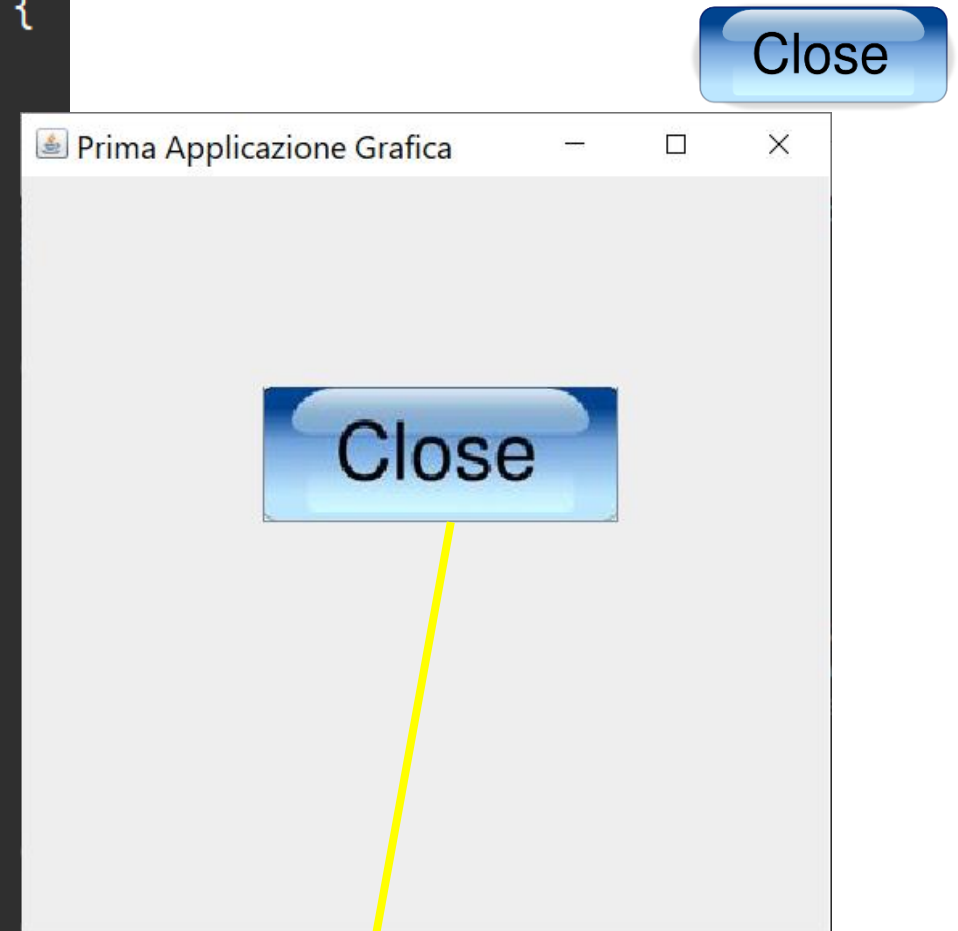
Inserire ImageIcon ad un JButton

```
public class MyFrame extends JFrame implements ActionListener {
    JButton button;
    ImageIcon icon;

    MyFrame(){
        icon = new ImageIcon("img/close.jpg");
        button = new JButton();
        button.setBounds(115, 100, 170, 65);
        button.addActionListener(this);
        button.setIcon(icon);

        this.setTitle("Prima Applicazione Grafica");
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setLayout(null); //no layout Manager set
        this.setSize(400,400);
        this.setVisible(true);
        this.add(button);
    } //MyFrame

    //implements actionPerformed
    public void actionPerformed(ActionEvent e){
        if (e.getSource()==button){
            System.out.print("Ciccio");
        } //if
    } //func
} //class
```



Problems @ Javadoc Declaration Console x
Main (2) [Java Application] C:\Program Files\Java\jdk-16
CiccioCiccioCiccio