

Programmation Orientée Objet avec JAVA

Plan

- Introduction
- Le langage JAVA
- La programmation objet
- Les bases du langage
- Les classes et les objets
- L'héritage et le polymorphisme
- La gestion des exceptions
- Le graphisme

Chapitre 7 – Le graphisme

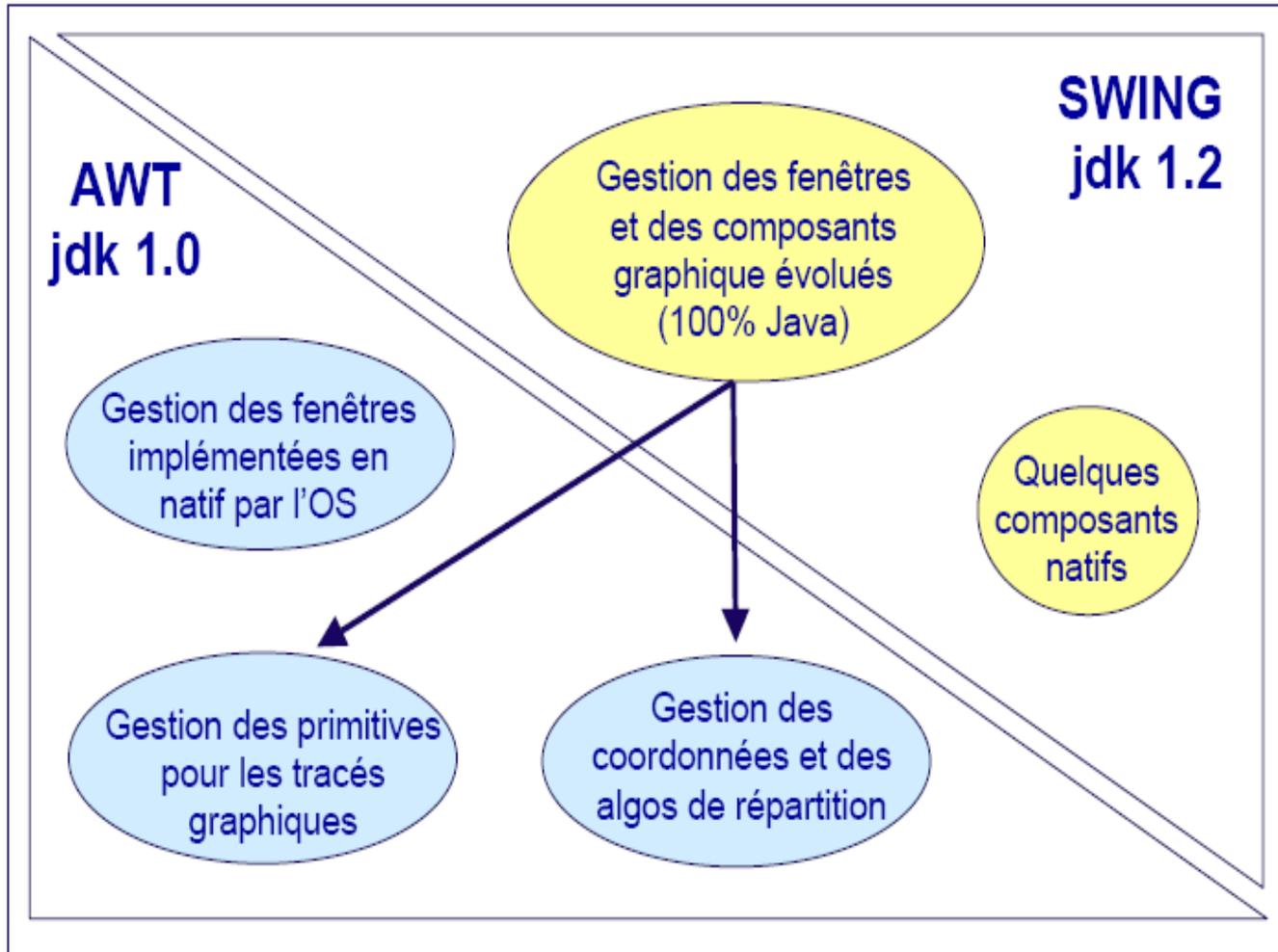
Chapitre 7 – Le graphisme

- **Les Interfaces Homme- Machines**
 - ✦ Le package « AWT » (`java.awt.*`)
 - ✦ Le package « SWING » (`javax.swing.*`)

Le dessin

- ✦ La classe « Graphics » (`java.awt.Graphics`)
- ✦ La classe « Graphics2D »

Chapitre 7 – Le graphisme



Chapitre 7 – Le graphisme

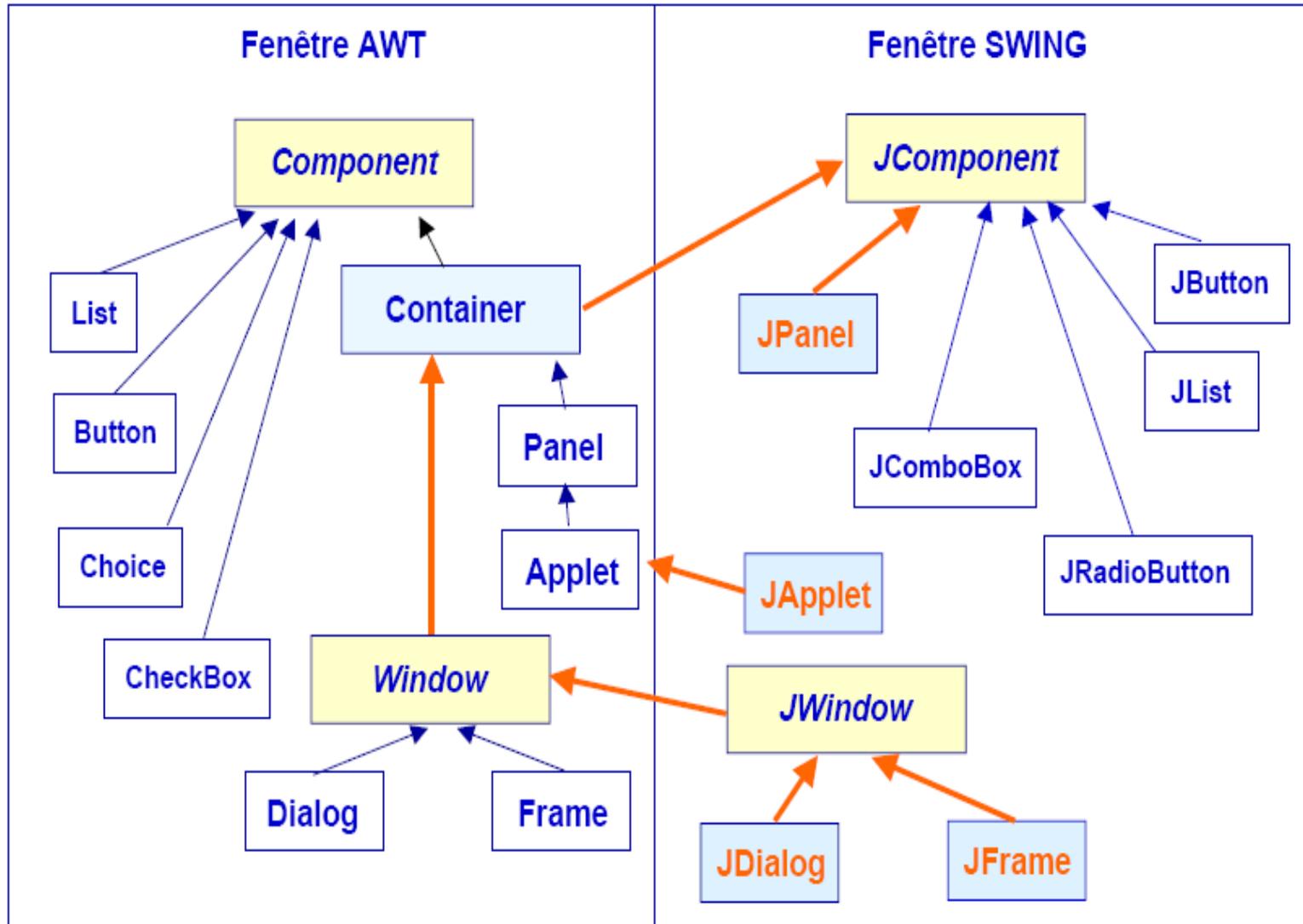
La partie « statique »

- ✦ La notion de « conteneurs »
- ✦ Les contrôles (composants graphiques)
- ✦ La politique de placement des composants dans les « conteneurs »

La partie « dynamique »

- ✦ Le système d'événements
- ✦ L'affectation d'un événement à un contrôle

Chapitre 7 – Le graphisme



Chapitre 7 – Le graphisme

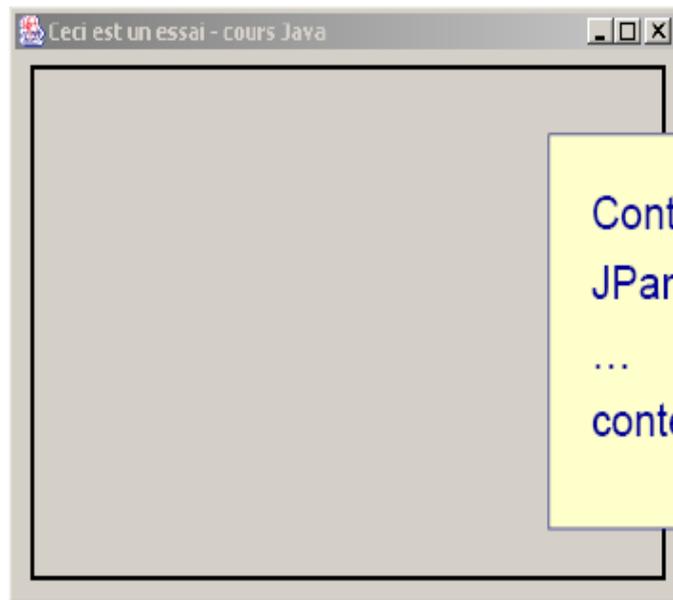
Les classes Component et JComponent

- ✦ Héritent de la classe « Object »
- ✦ Il est possible d'utiliser des objets dérivant de « Component » et de « JComponent » dans le même programme mais c'est déconseillé
- ✦ Préférer les composants « SWING » aux composants « AWT »
 - ✦ 100% Java (indépendant du système de fenêtrage de l'OS)
 - ✦ Plus riche en terme de fonctionnalités
- ✦ Composants « AWT » : Applets devant fonctionner avec certains navigateurs

Chapitre 7 – Le graphisme

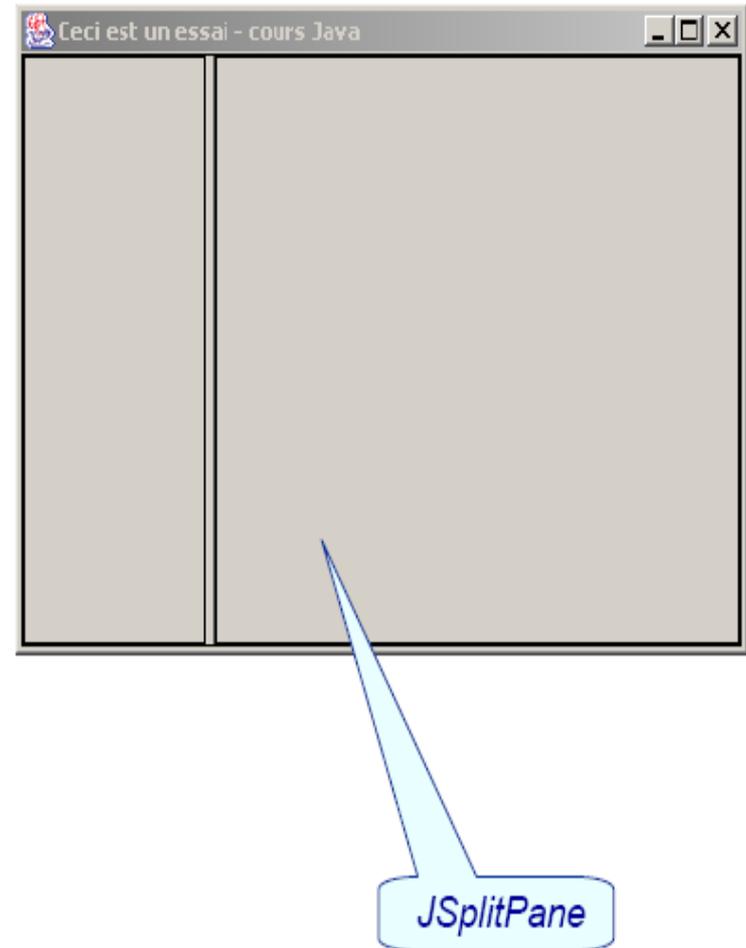
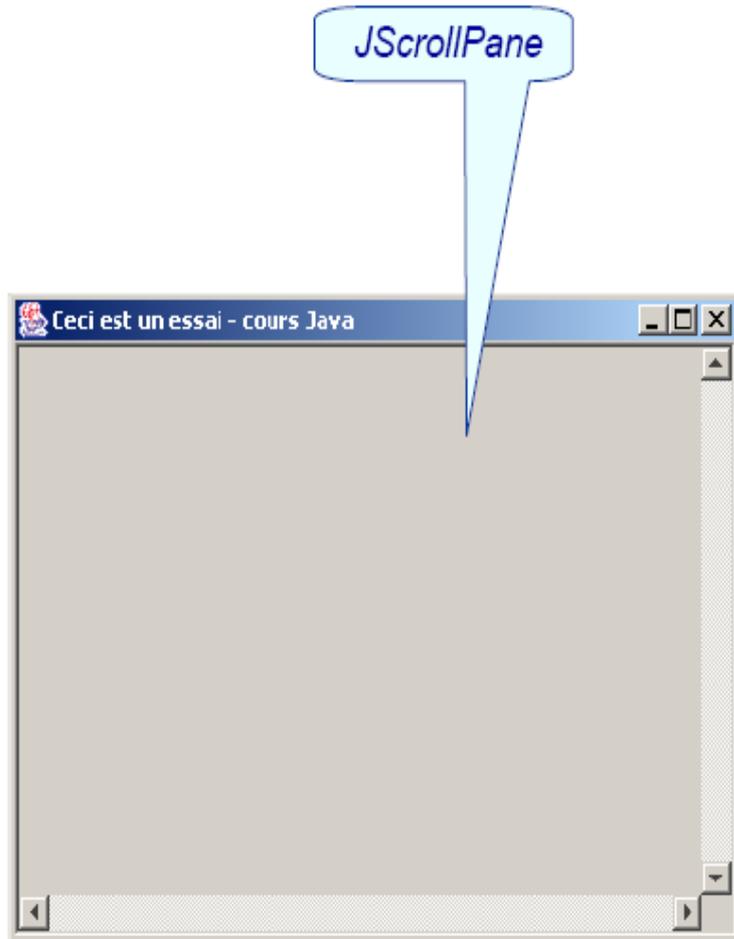
Les « conteneurs »

- ✦ La classe « Container »
- ✦ Les classes « JPanel », « JScrollPane » et « JSplitPane »



```
Container conteneur = this.getContentPane ( ) ;  
JPanel panneau= new JPanel ( ) ;  
...  
conteneur.add ( panneau ) ;
```

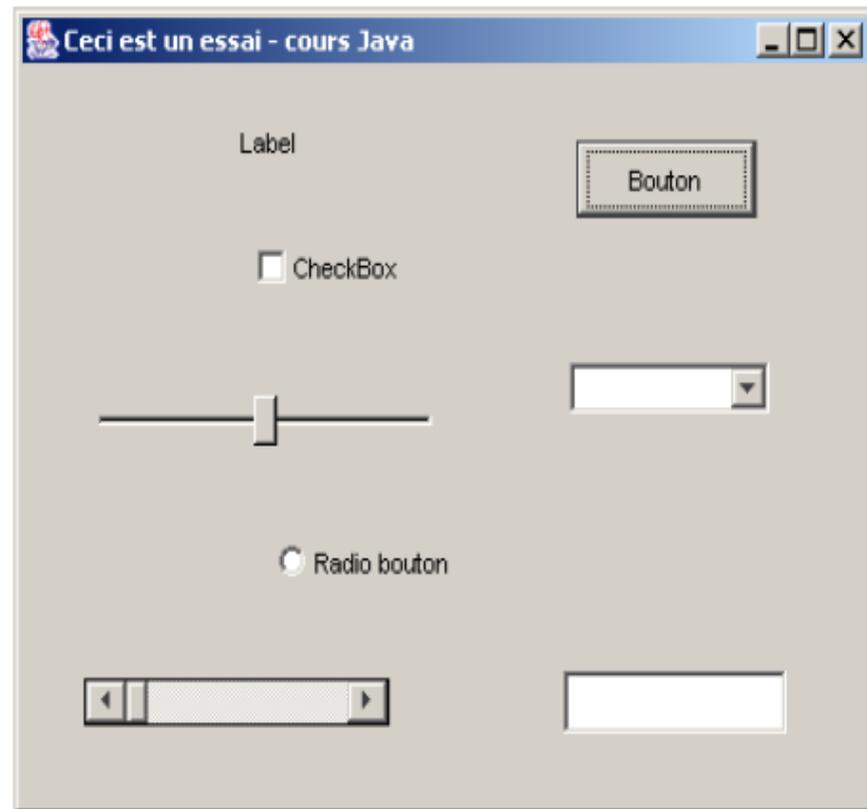
Chapitre 7 – Le graphisme



Chapitre 7 – Le graphisme

Les « contrôles » de SWING

- ✦ JLabel
- ✦ JTextField
- ✦ JButton
- ✦ Canvas ou Canvas3D
- ✦ JCheckBox
- ✦ JRadioButton
- ✦ JScrollBar
- ✦ JComboBox



Chapitre 7 – Le graphisme

Les « boîtes de dialogue »

✦ JDialog

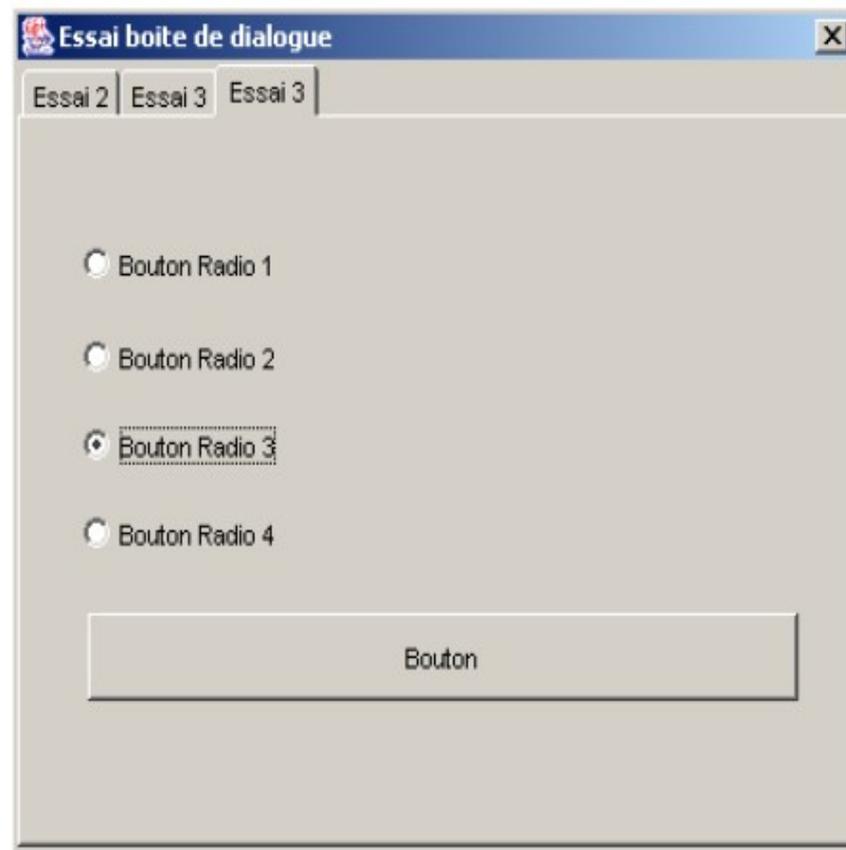
- ✦ Dépend d'une « frame »
- ✦ Modale (doit être fermée)
- ✦ Non modale



```
JDialog dial = new JDialog ( frame, "essai" , true ) ;
```

Chapitre 7 – Le graphisme

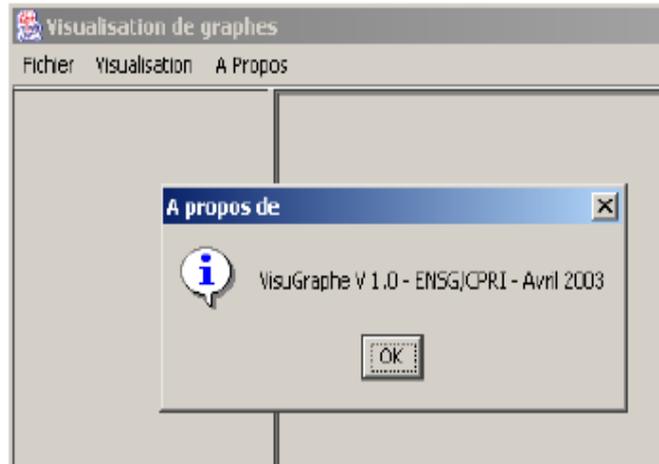
Les « boites de dialogue »



Chapitre 7 – Le graphisme

Les « boites de dialogue »

✦ JOptionPane (toujours modale)



QUESTION_MESSAGE



WARNING_MESSAGE



ERROR_MESSAGE

```
JOptionPane.showMessageDialog ( frame, "VisuGraphe V 1.0 - ENSG/CPRI - Avril 2003",  
                                "A propos de", JOptionPane.INFORMATION_MESSAGE );
```

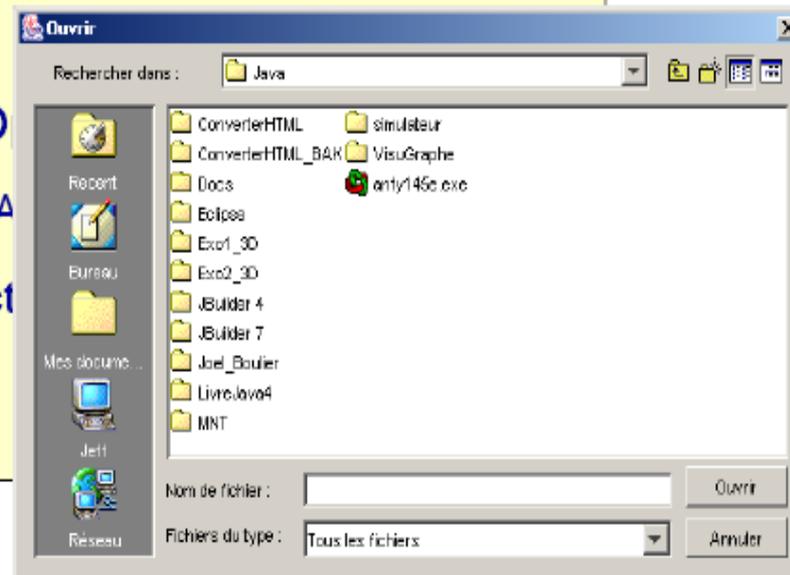
Chapitre 7 – Le graphisme

Les « boites de dialogue »

✦ JFileChooser

```
nomfic = new String ( ) ;  
JFileChooser chooser = new JFileChooser ("D:/Java") ;
```

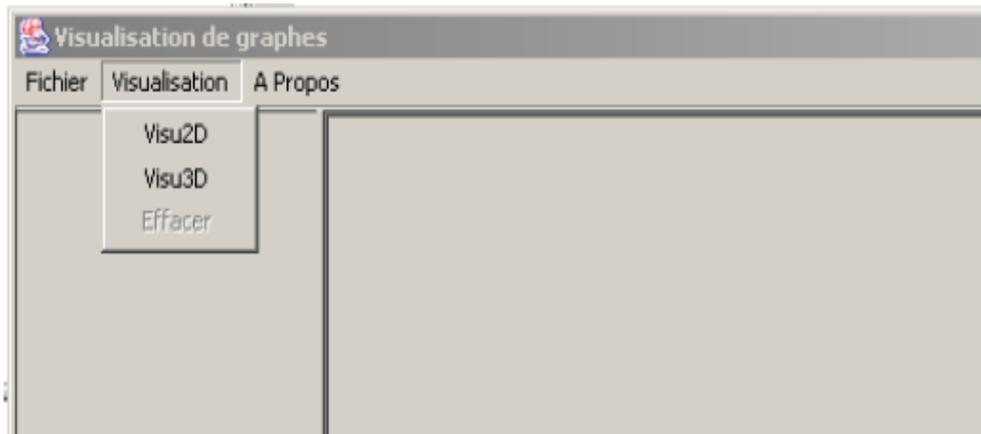
```
int returnVal = chooser.showO  
if ( returnVal == JFileChooser.A  
    nomfic = chooser.getSelect  
    ...  
}
```



Chapitre 7 – Le graphisme

Les « menus »

- ✦ JMenuBar, JMenu, JMenuItem
- ✦ JPopupMenu



Chapitre 7 – Le graphisme

Les composants « évolués »

- ✦ JList
- ✦ JTree
- ✦ JTabbedPane (Onglets)
- ✦ JColorChooser
- ✦ JOptionPane

Chapitre 7 – Le graphisme

- **Exemple**

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

public class ButtonTest extends JFrame {
    JButton MonBouton ;
    public ButtonTest ( String titre ) {
        setTitle ( titre ) ;
        Container contentPane = this.getContentPane ( ) ;
        contentPane.setLayout ( new FlowLayout ( ) ) ;
        MonBouton = new JButton ( "Mon Bouton" ) ;
        ContentPane .add ( MonBouton ) ;
        setSize ( 200, 100 ) ;
        setVisible ( true ) ;
    }
}
```

Chapitre 7 – Le graphisme

```
public static void main (String argv [])  
{  
    new ButtonTest ( "Mon Bouton" );  
}  
}
```

Chapitre 7 – Le graphisme



Chapitre 7 – Le graphisme

Le « FlowLayout »

```
panneau.setLayout ( new FlowLayout ( ) );  
this.getContentPane ( ).add ( panneau );
```

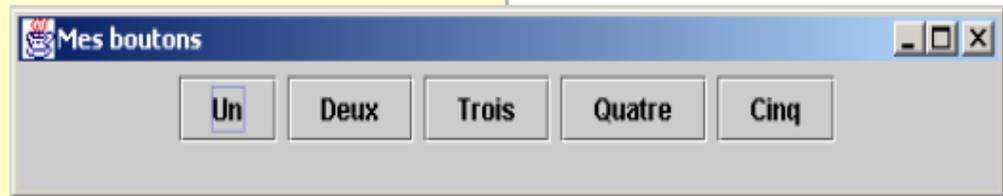
```
Bouton1.setText ("Un");  
Bouton2.setText ("Deux");
```

...

```
panneau.add ( Bouton1 );  
panneau.add ( Bouton2 );
```

...

```
this.setTitle ("Mes boutons");  
this.setSize ( new Dimension ( 496, 78 ) );  
this.setVisible ( true );
```



Chapitre 7 – Le graphisme

Le « GridLayout »

```
panneau.setLayout ( new GridLayout ( 2, 3 ) );  
this.getContentPane ( ).add ( panneau );
```

```
Bouton1.setText ("Un");  
Bouton2.setText ("Deux");
```

...

```
panneau.add ( Bouton1 );  
panneau.add ( Bouton2 );
```

...

```
this.setTitle ("Mes boutons");  
this.setSize ( new Dimension ( 496, 78 ) );  
this.setVisible ( true );
```



Chapitre 7 – Le graphisme

Le « BorderLayout »

```
panneau.setLayout ( new BorderLayout ( ) );  
this.getContentPane ( ).add ( panneau );
```

```
Bouton1.setText ( "Un" );  
Bouton2.setText ( "Deux" );
```

...

```
panneau.add ( " North " , Bouton1 );  
panneau.add ( " East " , Bouton2 );
```

...

```
this.setTitle ( "Mes boutons" );  
this.setSize ( new Dimension ( 496, 78 ) );  
this.setVisible ( true );
```



Center, West, South

Chapitre 7 – Le graphisme

```
panneau.setLayout ( null );  
this.getContentPane ( ).add ( panneau );  
  
Bouton1.setBounds ( new Rectangle ( 34, 20, 90, 35 ) );  
Bouton1.setText ( "Un" );  
Bouton2.setBounds ( new Rectangle ( 115, 82, 134, 36 ) );  
Bouton2.setText ( "Deux" );  
...  
  
panneau.add ( Bouton1 );  
panneau.add ( Bouton2 );  
...  
  
this.setTitle ( "Mes boutons" );  
this.setSize ( 300,300 );  
this.setVisible ( true );
```



Chapitre 7 – Le graphisme

Les événements

- ✦ Préviennent le programmeur et les composants de l'interface graphique que quelque chose s'est produit
- ✦ Leur détection permet de réagir aux actions de l'utilisateur en adaptant son comportement à ces actions
- ✦ Une gestion en trois étapes :
 - ✦ Déterminer les événements et les associer à des « écouteurs »
 - ✦ Écrire le code des « écouteurs » et de chaque événement
 - ✦ Enregistrer les « écouteurs »

Chapitre 7 – Le graphisme

La notion « d'écouteur » ou « Listener »

- ✦ Classe possédant au moins une méthode, appelée si l'événement attendu apparaît
- ✦ Une classe « écouteur » implémente une interface
- ✦ Plusieurs types « d'écouteur » :
 - ✦ « ActionListener » écoute tous les événements (menus)
 - ✦ « MouseListener » écoute les événements souris
 - ✦ « WindowListener » écoute les événements sur les fenêtres
 - ✦ ...

Chapitre 7 – Le graphisme

Écriture du code des « écouteurs »

- ✦ Exemple : traitement de l'événement « `ActionEvent` » déclenché par pression d'un bouton souris (ou touche clavier)
- ✦ Pour traiter un tel événement on doit implémenter l'interface « `ActionListener` » qui contient la méthode « `actionPerformed` »
- ✦ La méthode « `actionPerformed` » doit contenir le code à exécuter à chaque « clic souris »

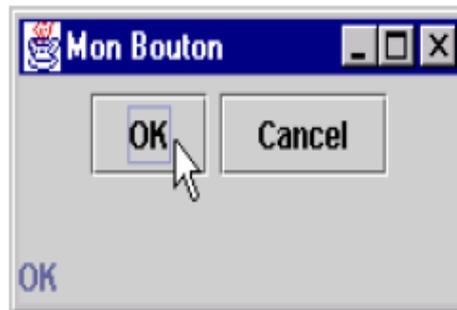
Chapitre 7 – Le graphisme

```
public class EventFrame extends JFrame implements ActionListener {
    JButton BoutonOk, BoutonCancel;
    JLabel statusLabel;
    /** Creates a new instance of EventFrame */
    public EventFrame(String titre) {
        setTitle(titre);
        Container contentPane = getContentPane();
        JPanel ButtonPanel = new JPanel();
        ButtonPanel.setLayout (new FlowLayout());
        BoutonOk = new JButton("Ok");
        ButtonPanel.add(BoutonOk);
        BoutonCancel = new JButton("Cancel");
        BoutonOk.addActionListener(this);
        BoutonCancel.addActionListener(this);
        ButtonPanel.add(BoutonCancel);
        contentPane.add(ButtonPanel, BorderLayout.CENTER);
        statusLabel = new JLabel("Zone de texte");
        contentPane.add(statusLabel, BorderLayout.SOUTH);
        setSize(200, 100);
        setVisible(true);
    }

    public void actionPerformed (ActionEvent event){
        if (event.getSource() == BoutonOk){
            statusLabel.setText("Ok détecté");
        }
        if (event.getSource() == BoutonCancel) {
            statusLabel.setText("Cancel détecté");
        }
    }
}
```

Chapitre 7 – Le graphisme

```
public class MonTest {  
    public static void main ( String[ ] args ) {  
        EventFrame ef = new EventFrame ( "Mon Bouton " );  
    }  
}
```



Chapitre 7 – Le graphisme

- **Récapitulatif des principaux événements**

Composant	Événement créé	Méthode invoquée
JButton	ActionEvent	actionPerformed
JCheckBox	ItemEvent	itemStateChanged
JList	ListSelectionEvent	valueChanged
JTextField	ActionEvent	actionPerformed
JWindow	WindowEvent	WindowOpen/Close ...
JComponent	ComponentEvent	componentMoved/Resized ...
JComponent	FocusEvent	focusLost, focusGained
JComponent	KeyEvent	keyPressed/Released/Typed
...