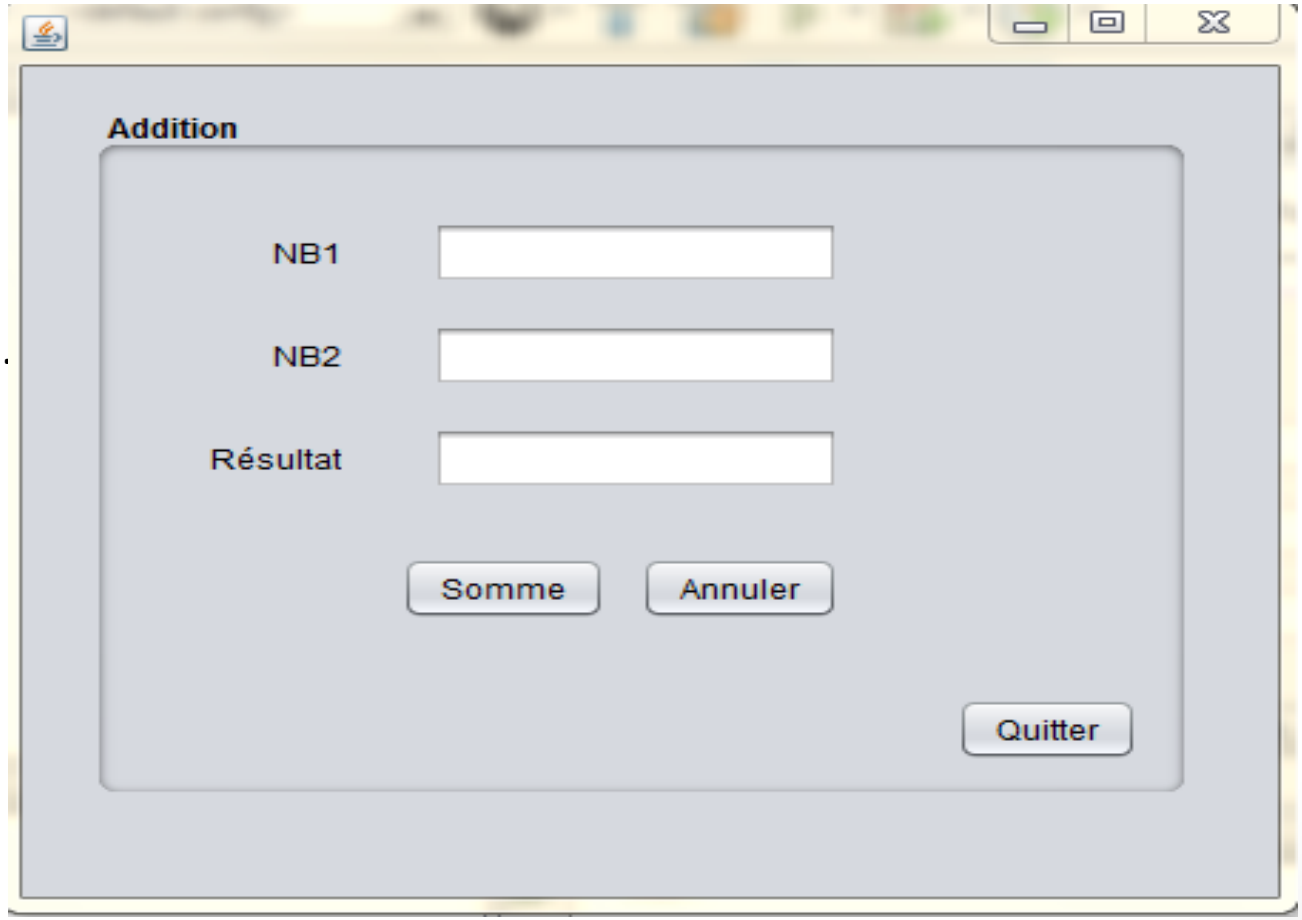
Université de Jijel
Faculté des sciences exactes et de l'informatique
Département d'informatique

Application avec interface graphique (API Swing)

Objectif de ce TP

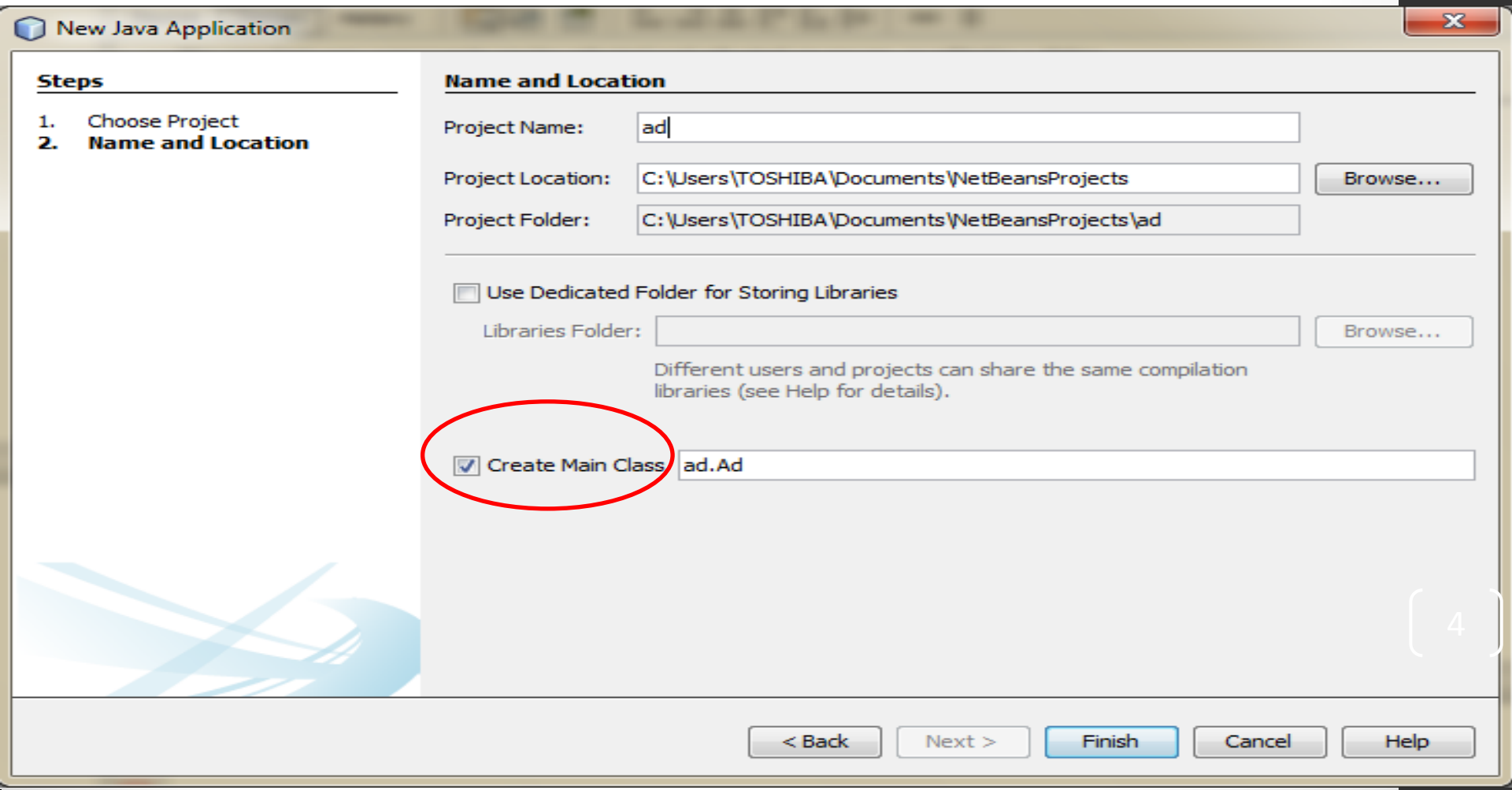
Construire un programme à interface graphique qui somme deux nombres réels.



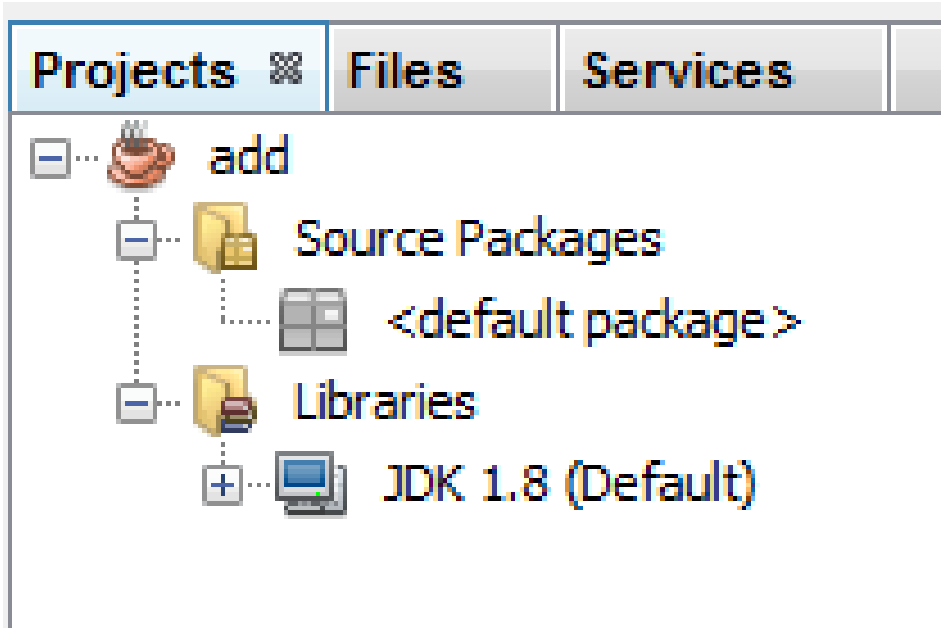
Etape 1

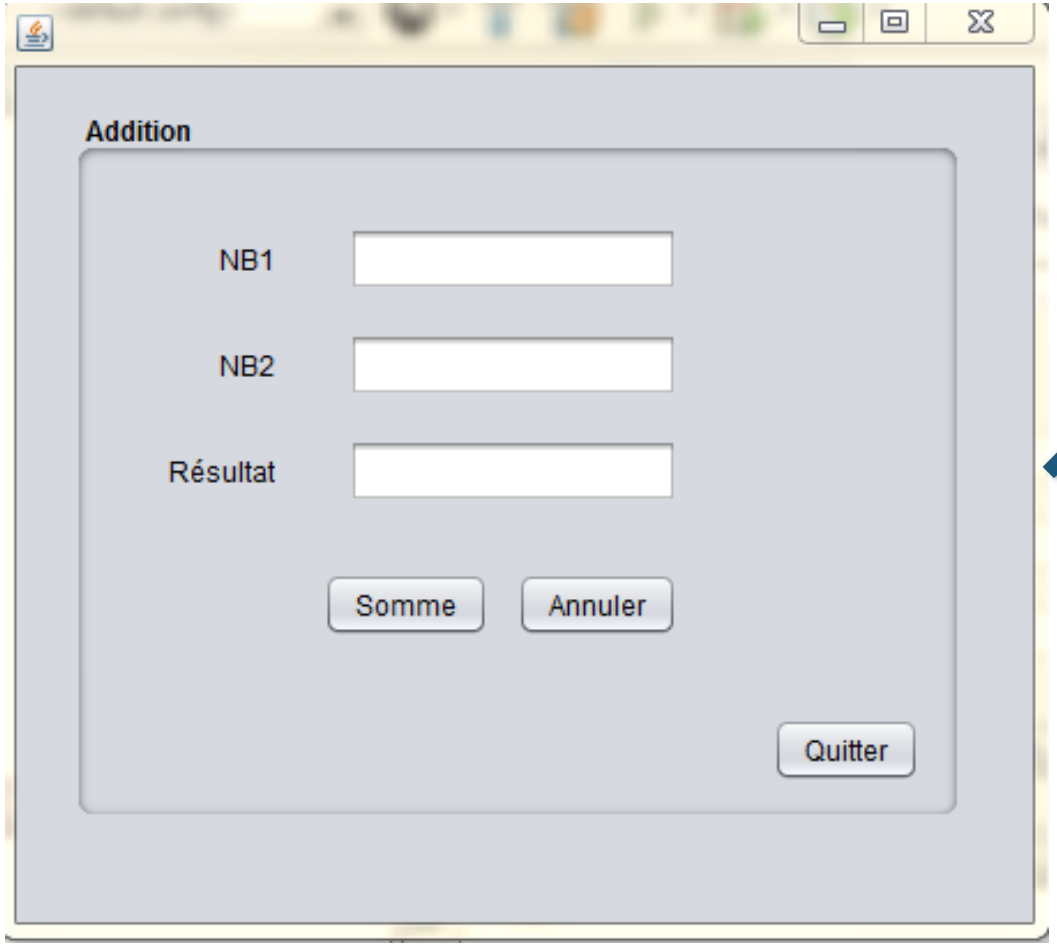
- Création du projet

- File > New Project
- Categories / Java & Projects/ Java Application > Next
- Project Name ……Nommer le programme (eg. Add)
- Décocher (Create Main Class)
- Finish



Ce qu'on obtient





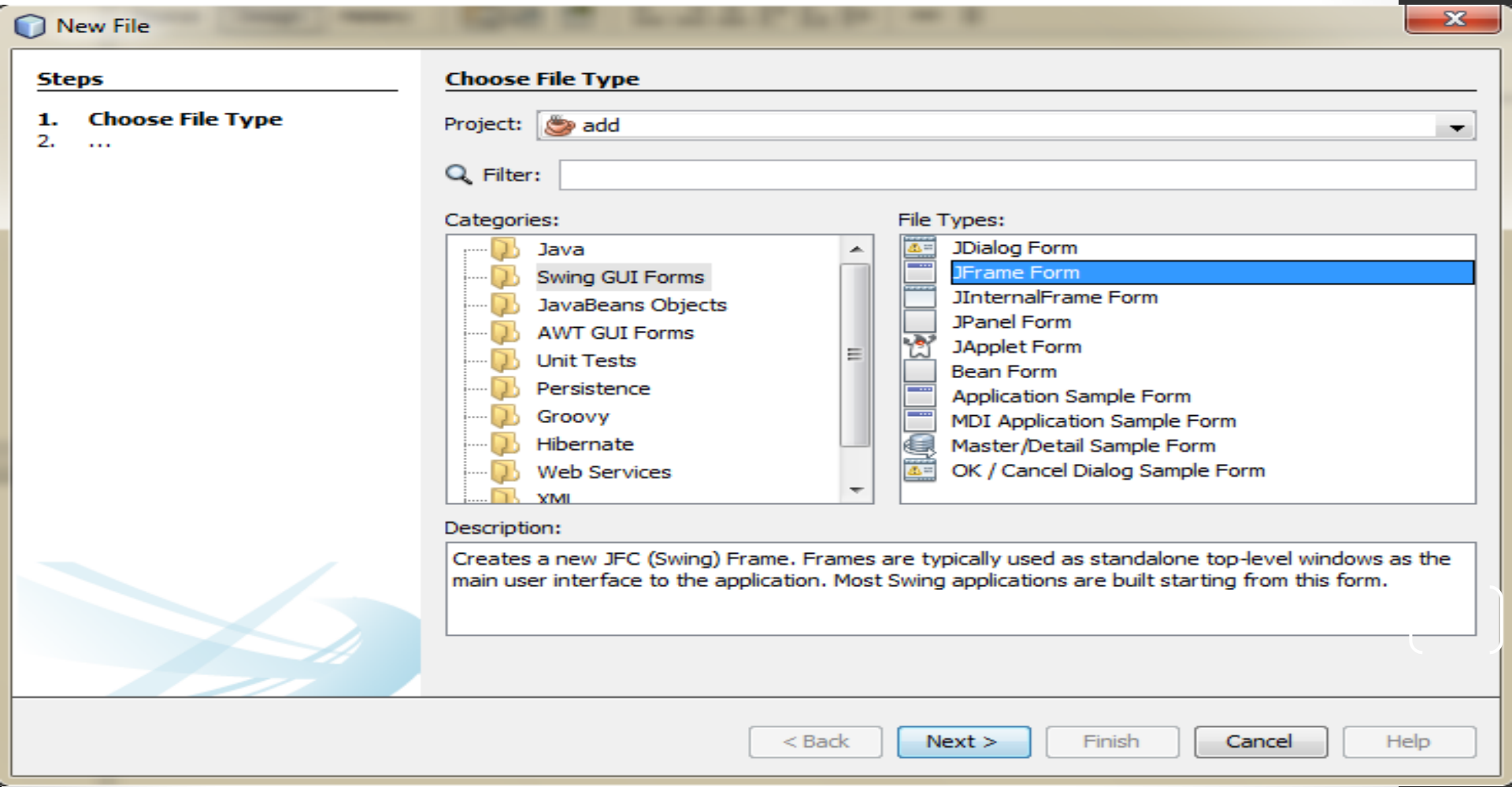
Fenêtre : Jframe

Rôle: conteneur de composants

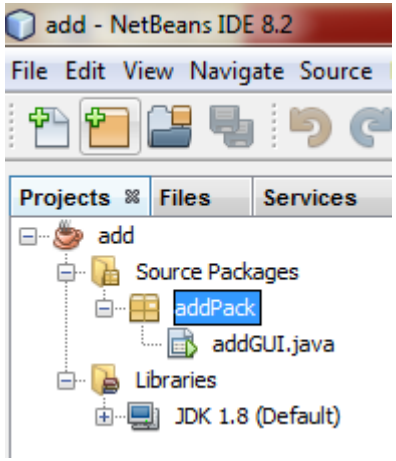
Etape 2

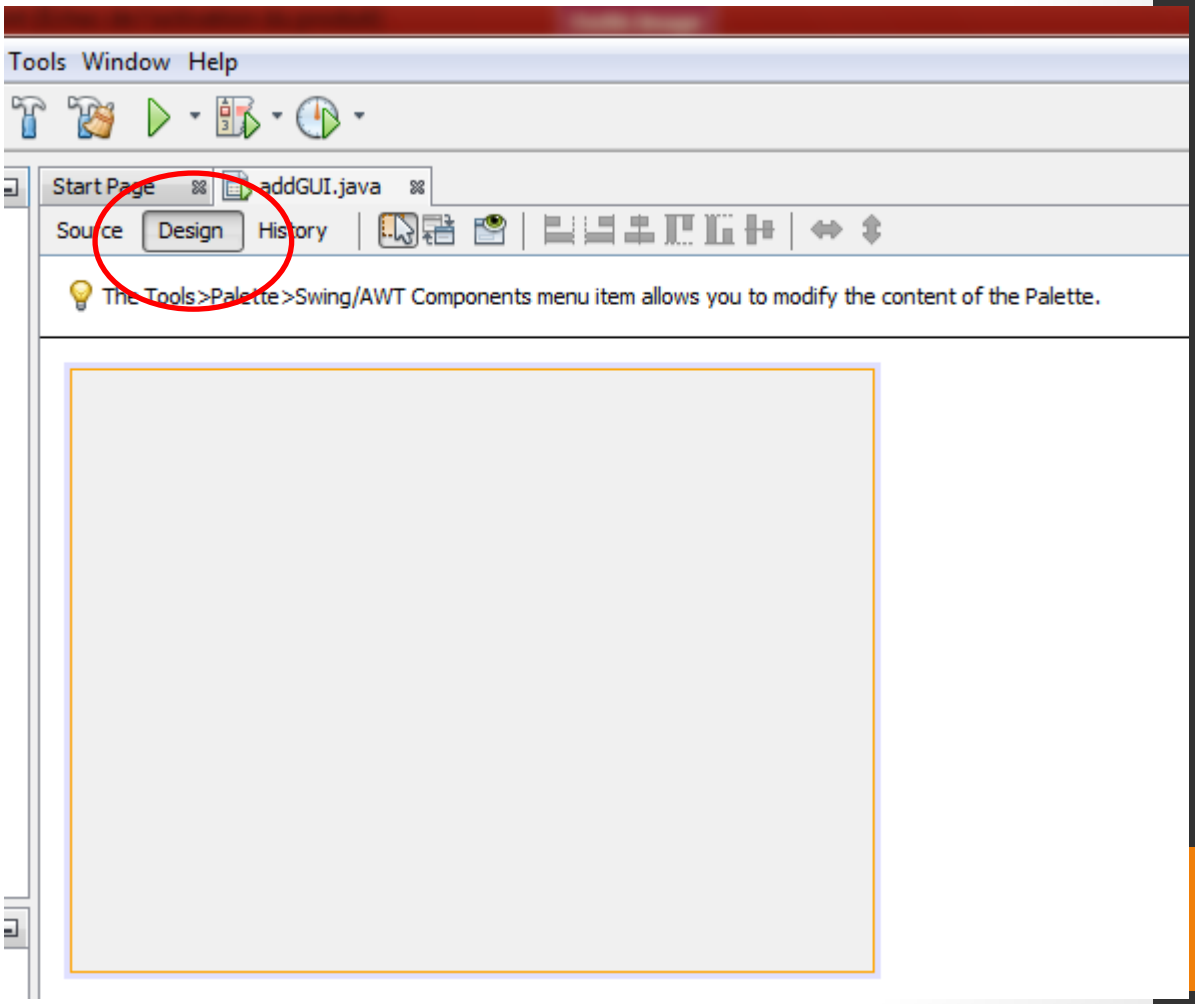
- Création de l’interface graphique

- File > New File
- Categories /Swing Gui Forms & File Types/ Jframe Form > Next
- Class Name …… Nommer la classe (eg, AddGUI)
- Package …… Nommer le package (eg, AddPack)
- Finish



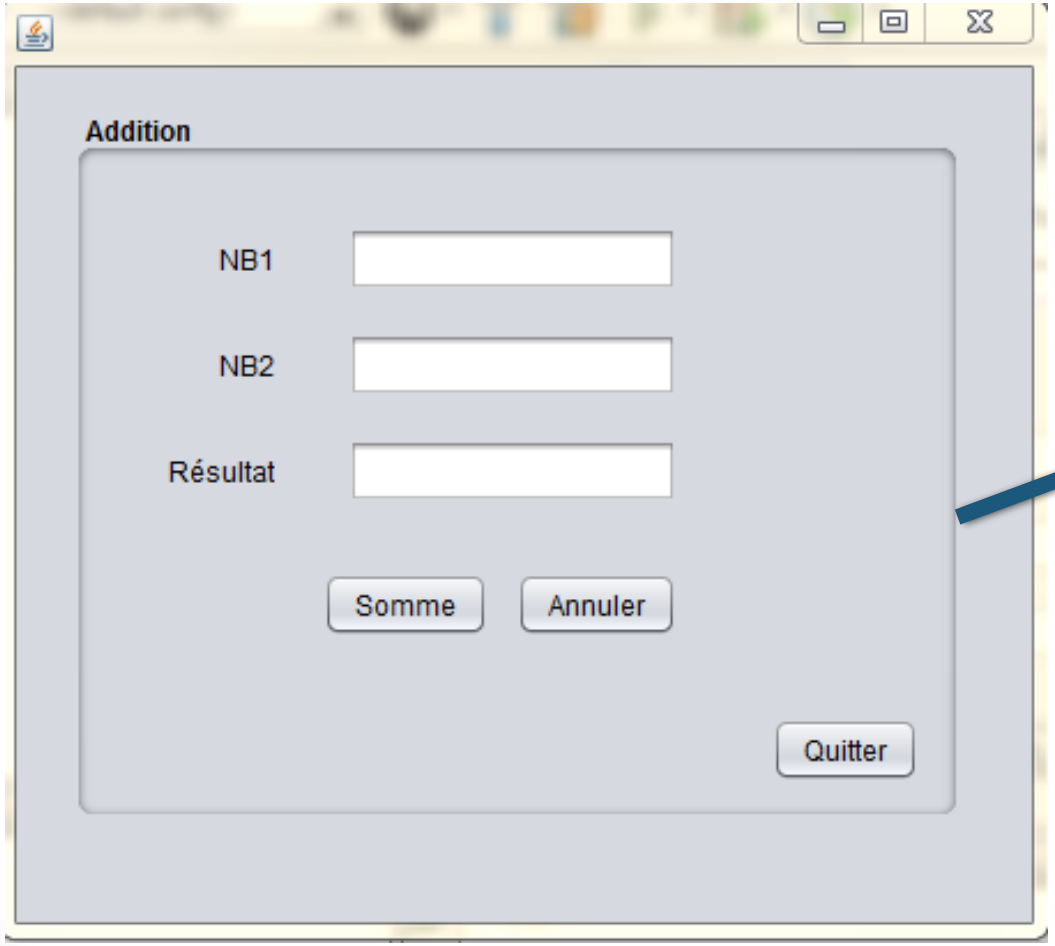
Ce qu'on obtient





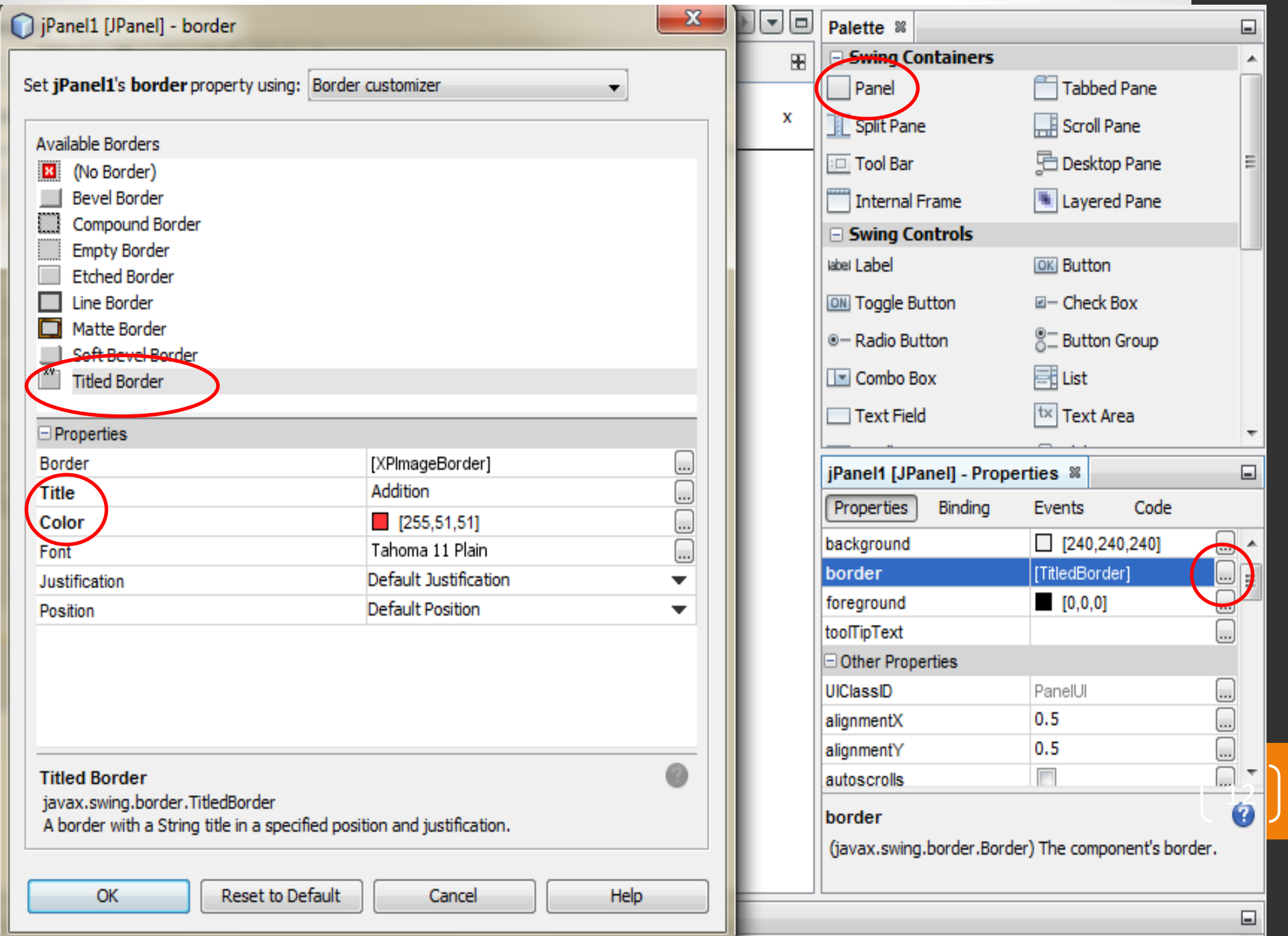
Etape 3

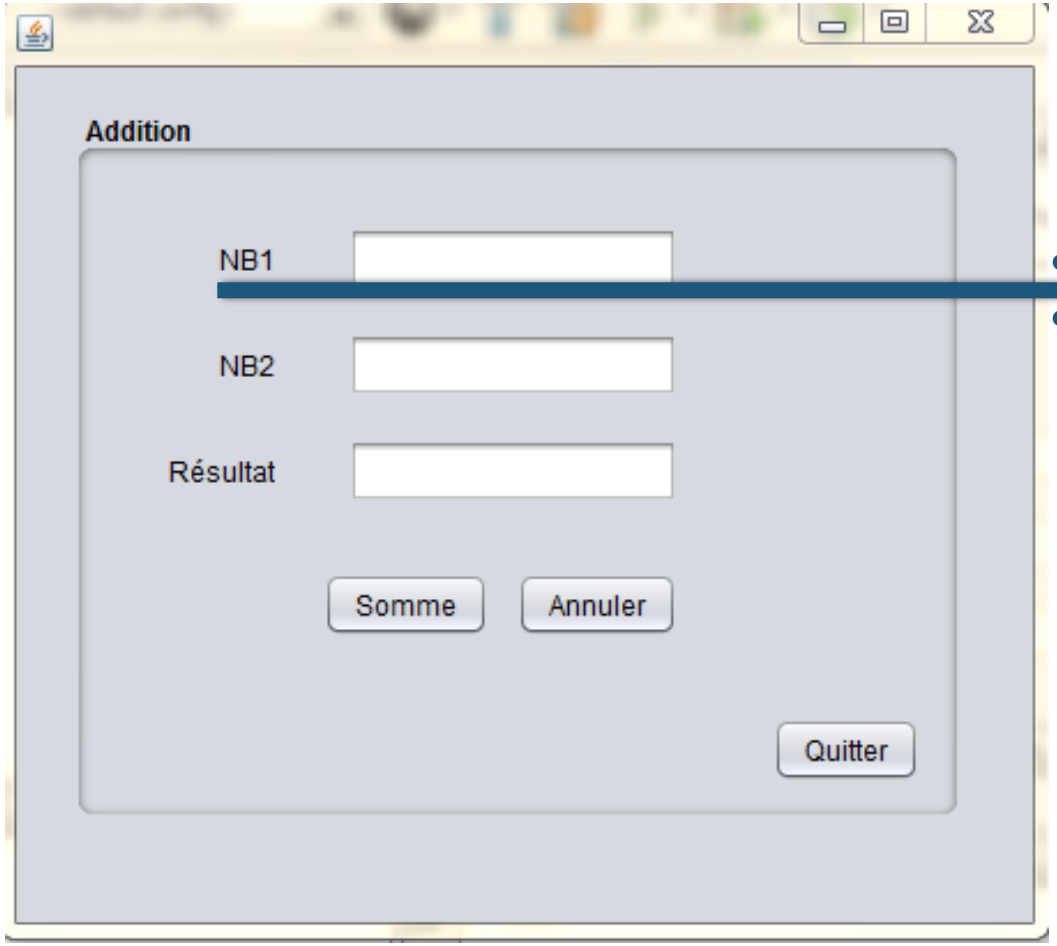
- Insertion des différents composants



Panneau (avec un titre) : Panel

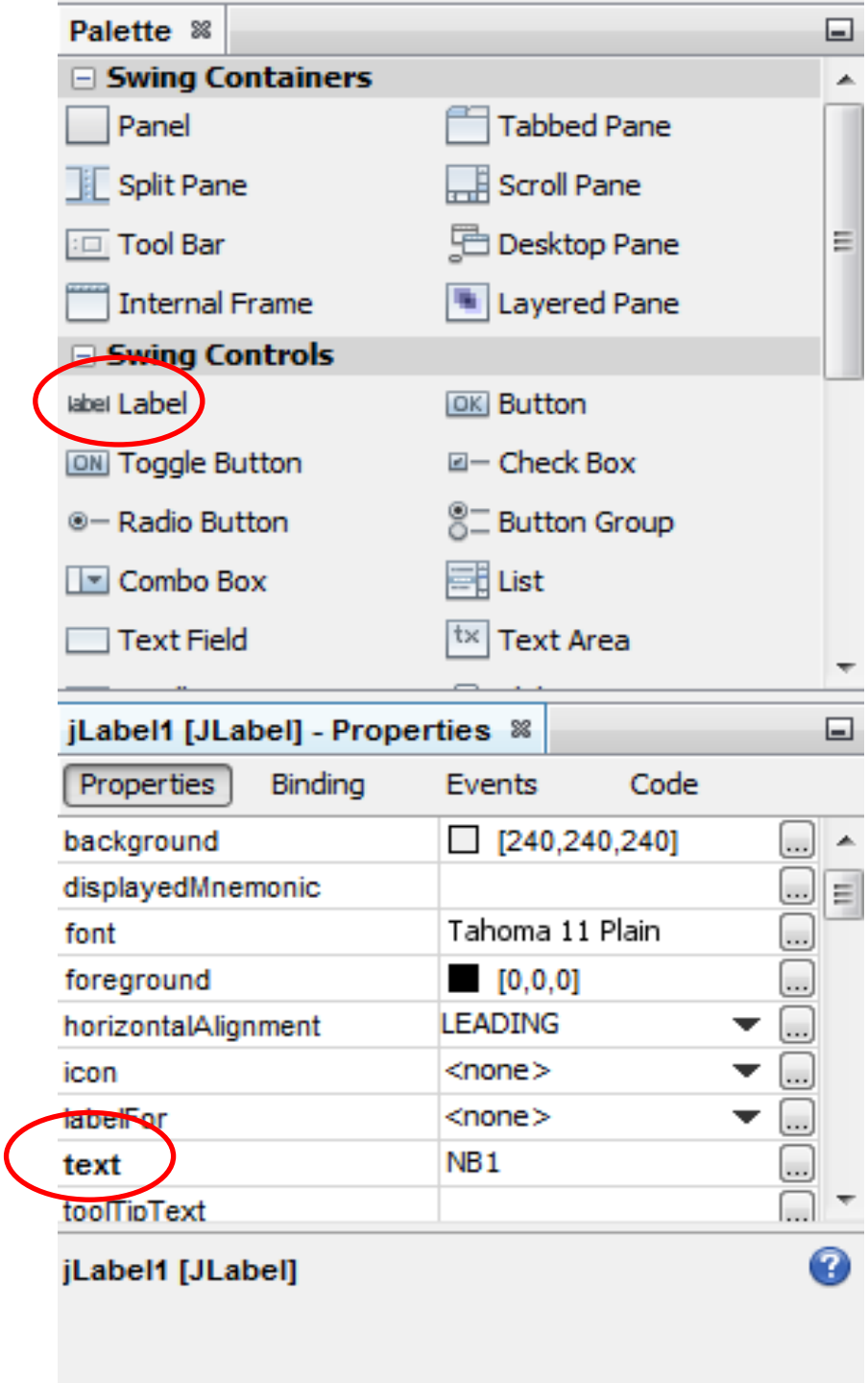
Rôle: Sous fenêtre


jPanel1 [JPanel] - border
Set jPanel1's border property using: Border customizer
Available Borders
(No Border)
Bevel Border
Compound Border
Empty Border
Etched Border
Line Border
Matte Border
Soft Bevel Border
Titled Border
Properties
Border [XPImageBorder]
Title Addition
Color [255,51,51]
Font Tahoma 11 Plain
Justification Default Justification
Position Default Position
Titled Border
javax.swing.border.TitledBorder
A border with a String title in a specified position and justification.
OK
Reset to Default
Cancel
Help
Palette
Swing Containers
Panel
Tabbed Pane
Split Pane
Scroll Pane
Tool Bar
Desktop Pane
Internal Frame
Layered Pane
Swing Controls
Label
Button
Toggle Button
Check Box
Radio Button
Button Group
Combo Box
List
Text Field
Text Area
jPanel1 [JPanel] - Properties
Properties
Binding
Events
Code
background [240,240,240]
border [TitledBorder]
foreground [0,0,0]
toolTipText
Other Properties
UIClassID PanelUI
alignmentX 0.5
alignmentY 0.5
autoscrolls
border
(javax.swing.border.Border) The component's border.



3 étiquettes : Jlabel

Non modifiable par l ’utilisateur

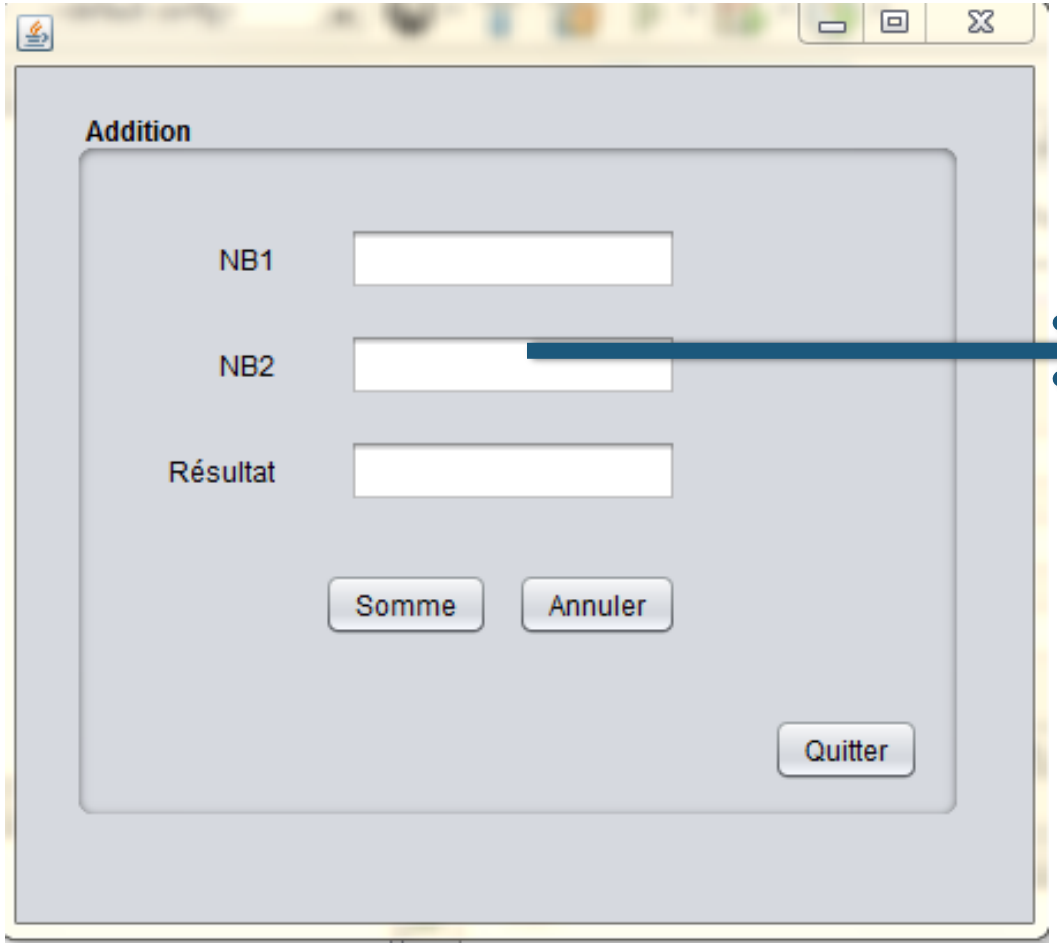


De même pour

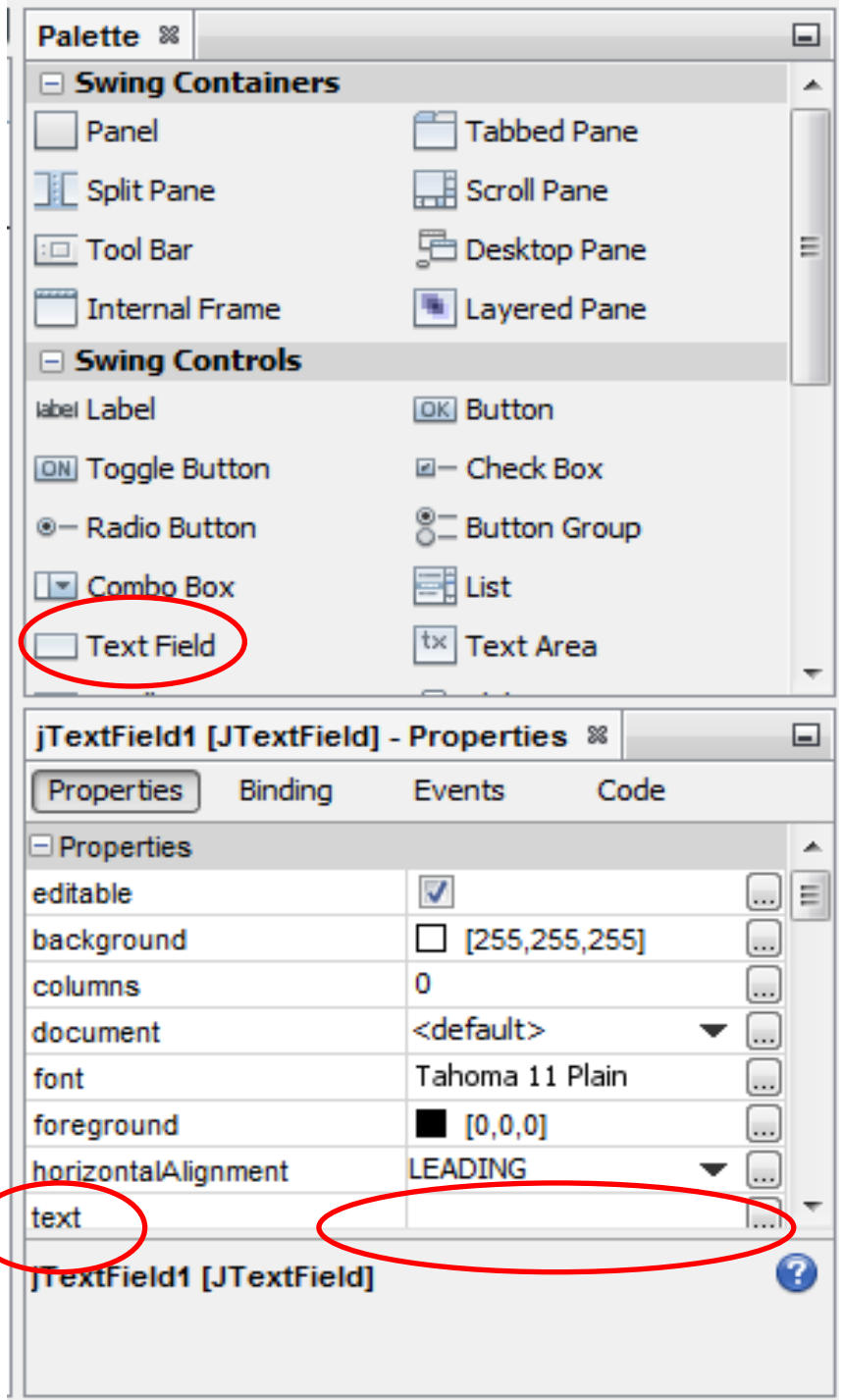
NB2

Résultat

À quoi ressemble notre interface ?



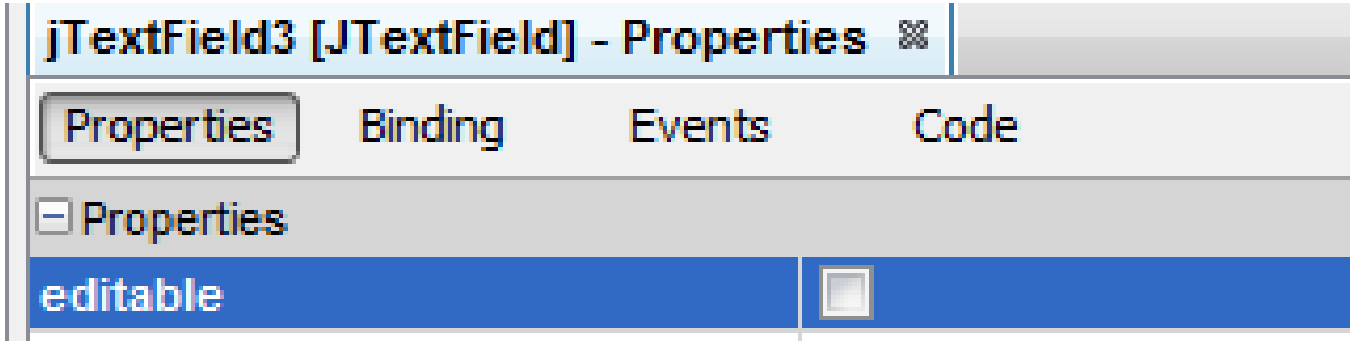
3 champs de texte:
JTextField



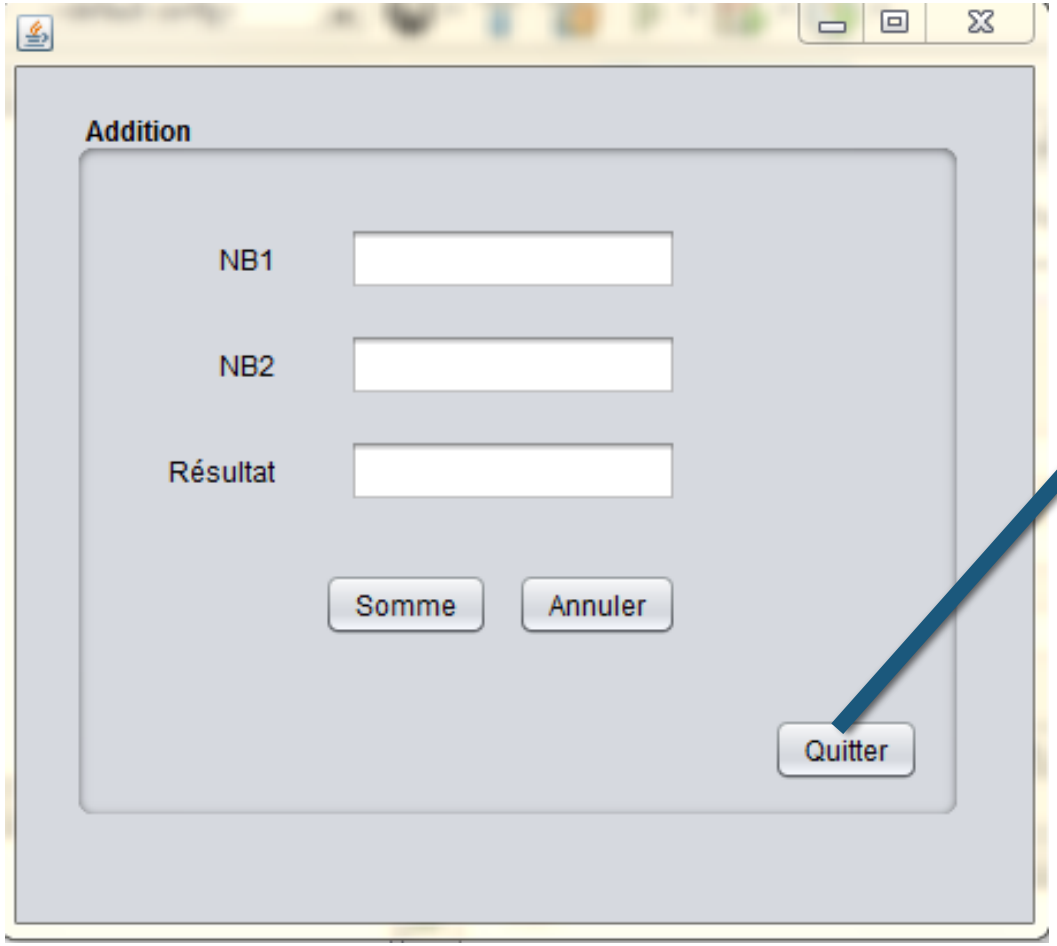
De même pour

Les deux autres champs

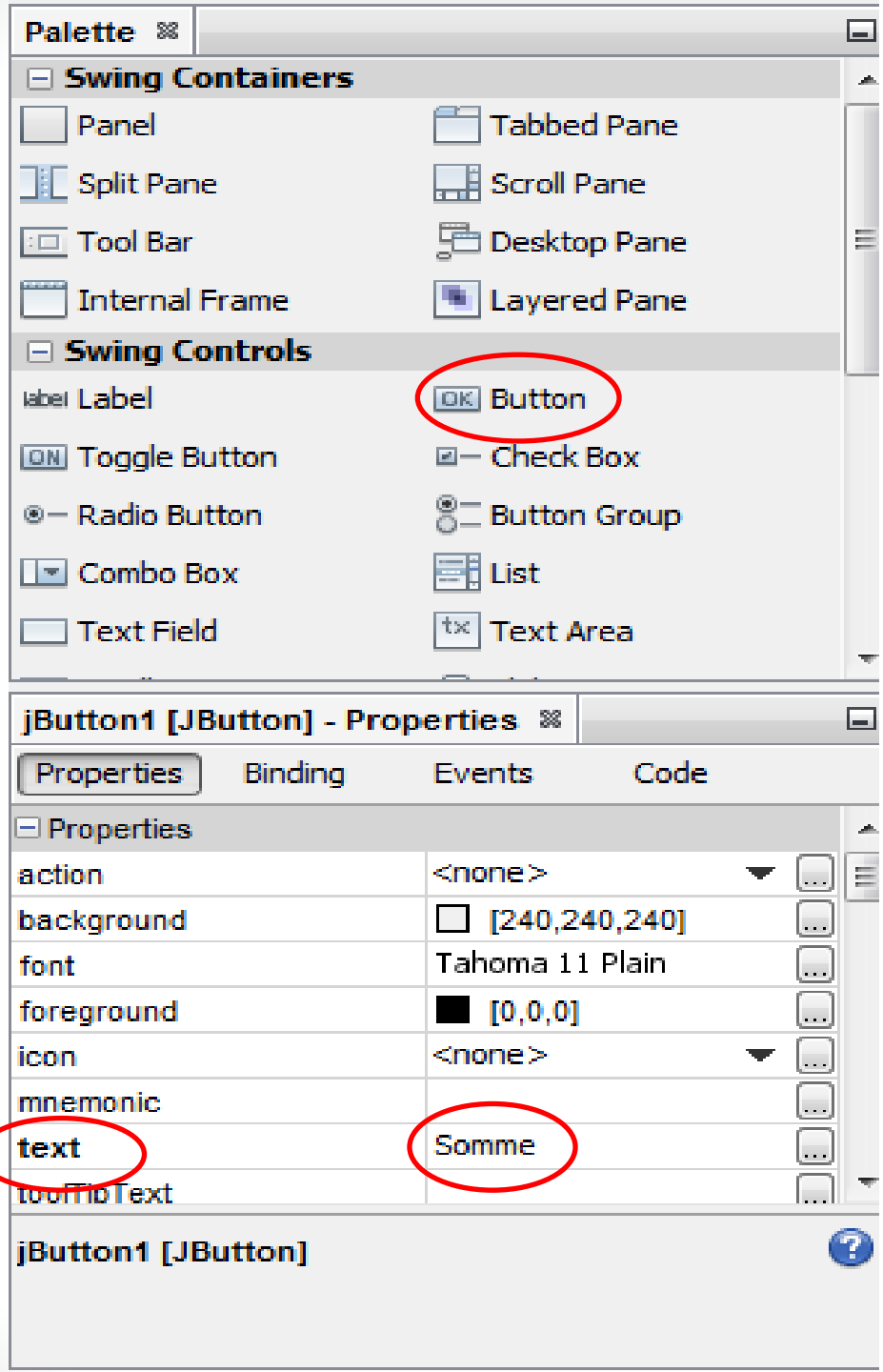
Pour un champ résultat , on peut désactiver la saisie en allant dans Propriétés/ (décocher editable) :



À quoi ressemble notre interface ?



3 boutons: Button



De même pour

Les deux autres boutons

Etape 4

- Écriture du Code

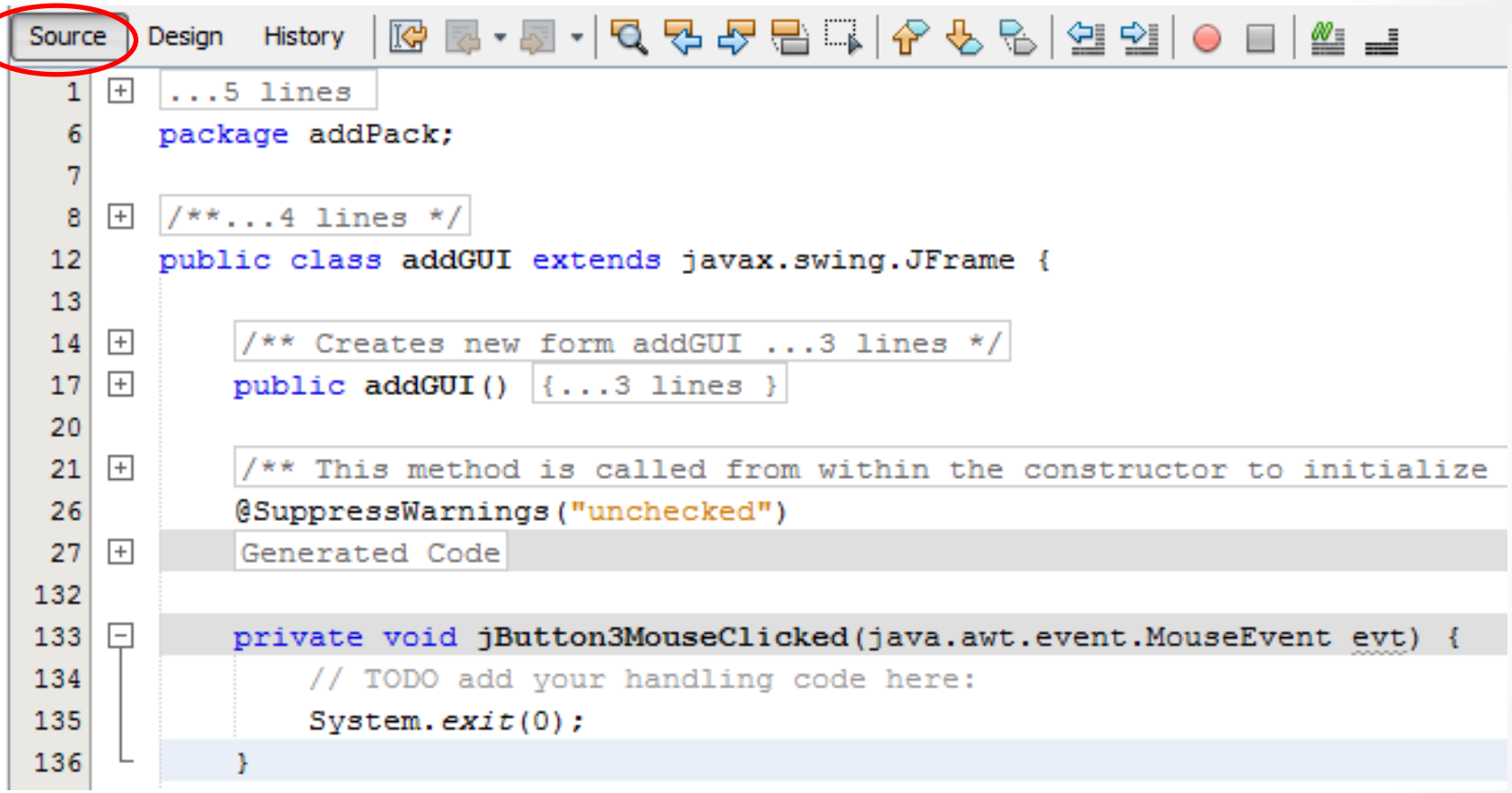
Bouton Quitter

- Double clic sur le bouton

```
private void jButton3ActionPerformed(java.awt.event.MouseEvent evt) {

System.exit(0);

  }
```


Source
Design
History
...5 lines
package addPack;
/**...4 lines */
public class addGUI extends javax.swing.JFrame {
/** Creates new form addGUI ...3 lines */
public addGUI() {...3 lines }
/** This method is called from within the constructor to initialize
@SuppressWarnings("unchecked")
Generated Code
private void jButton3MouseClicked(java.awt.event.MouseEvent evt) {
// TODO add your handling code here:
System.exit(0);
}

Bouton Annuler

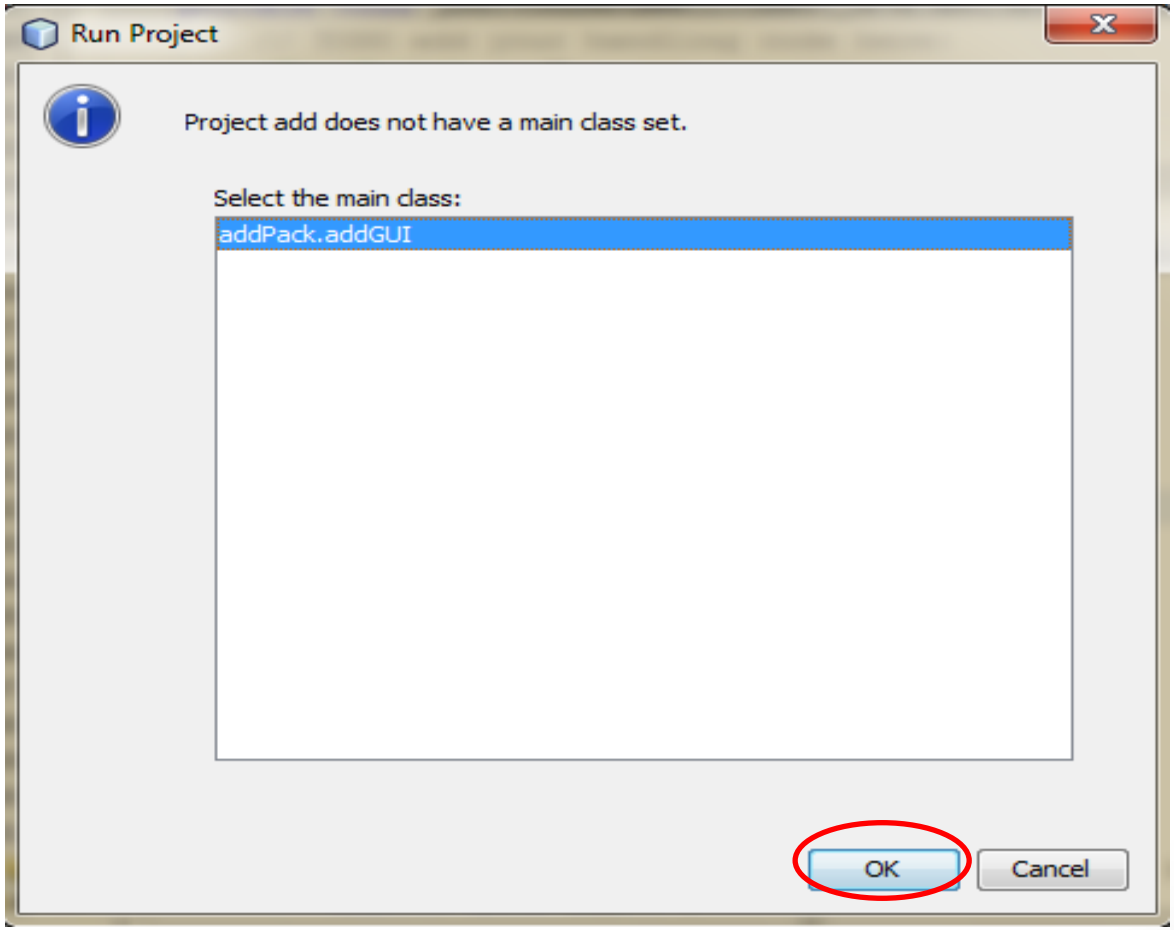
```
private void jButton2ActionPerformed(java.awt.event.MouseEvent evt) {
        jTextField1.setText("");
        jTextField2.setText("");
        jTextField3.setText("");
    }
```

Bouton Somme

```
float n1,n2,res;
n1=Float.parseFloat(jTextField1.getText());
n2=Float.parseFloat(jTextField2.getText());
res=n1+n2;
jTextField3.setText(String.valueOf(res));
```

Exécution

- Run / Run Project (Add)

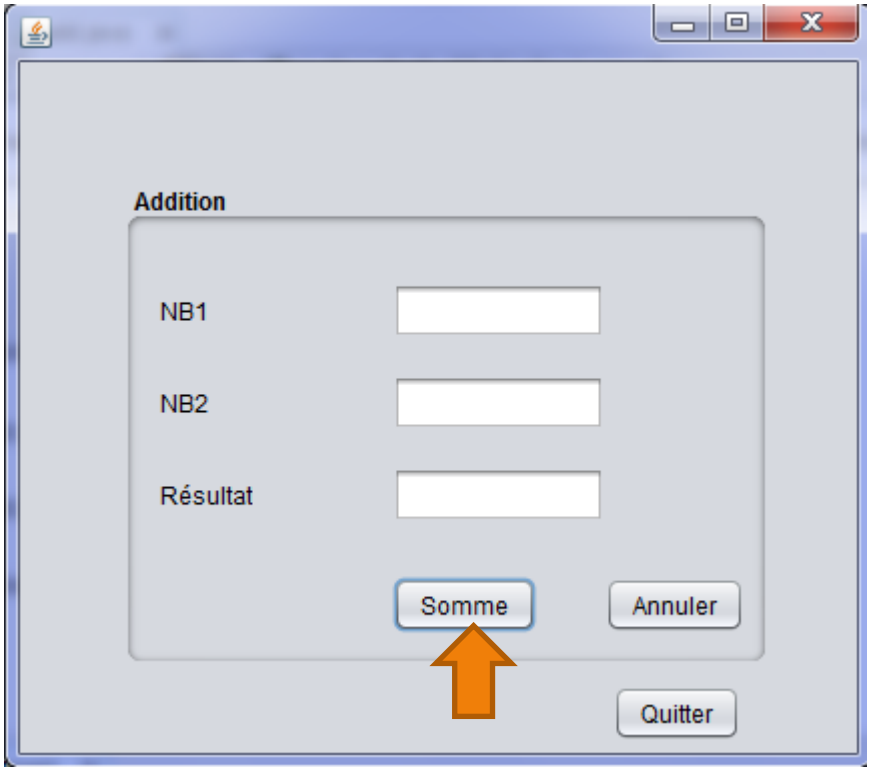


Exemple d'exécution

Traitement des exceptions

Exceptions ?

Le terme *exception* désigne tout événement arrivant durant l'exécution d'un programme interrompant son fonctionnement normal. Autrement, les exceptions sont des erreurs survenant lors de l'exécution d'un programme.



Exemple 1:

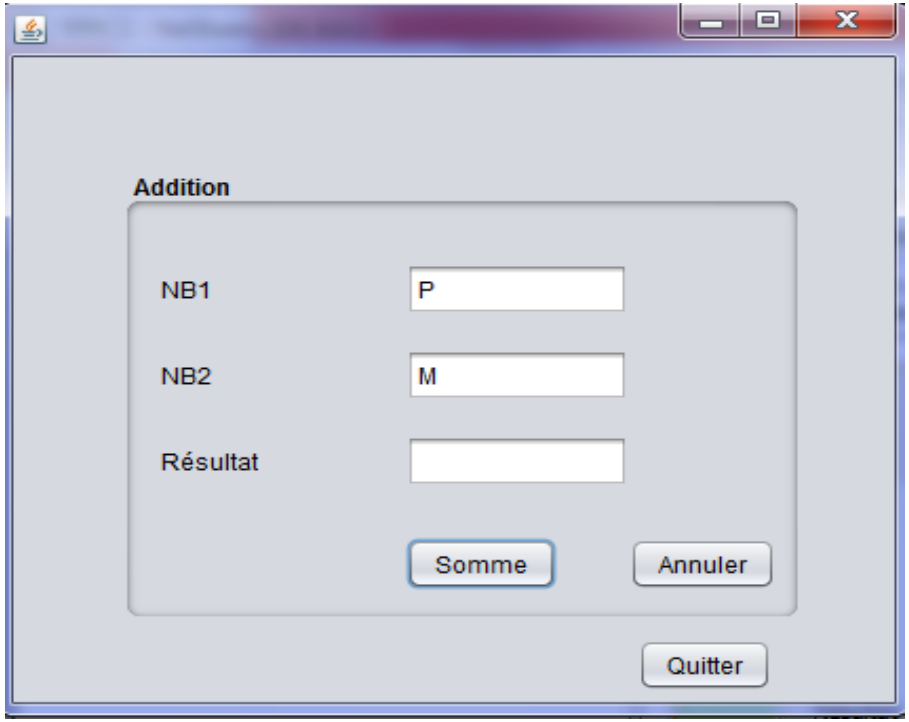
Lacer le calcul sans données d'entrée

Exception in thread "AWT-EventQueue-0" java.lang.**NumberFormatException**:

empty String at …. java.lang.**Float.parseFloat** ….

Exemple 2:

Saisir des données non numériques

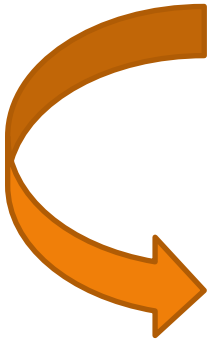


Exception in thread "AWT-EventQueue-0" java.lang.**NumberFormatException**:

For input string: "P" at …. java.lang.**Float.parseFloat** ….

Traitement de l'exception

```
Try {
    Instructions susceptibles de lever des exceptions
}
Catch ( Exception e) {
    Traitement de l'exception
}
```

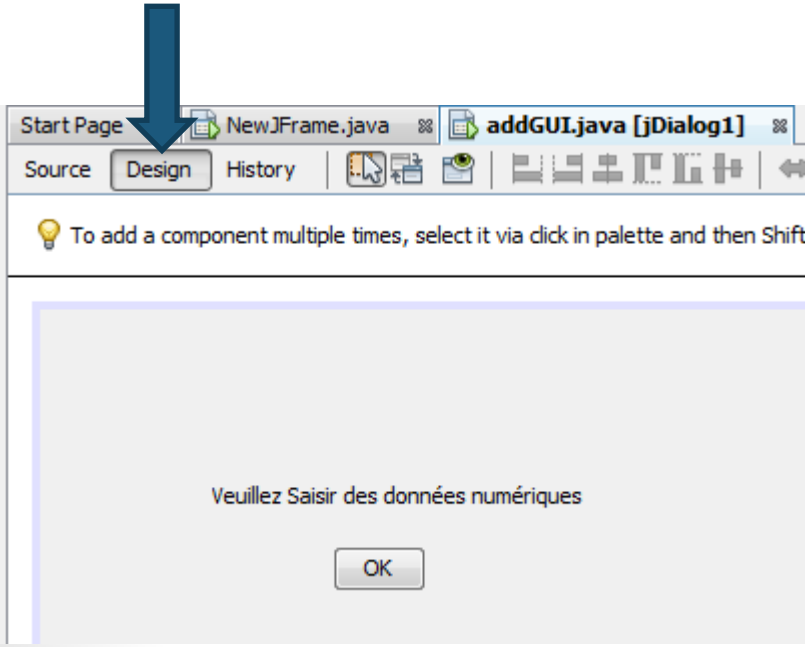


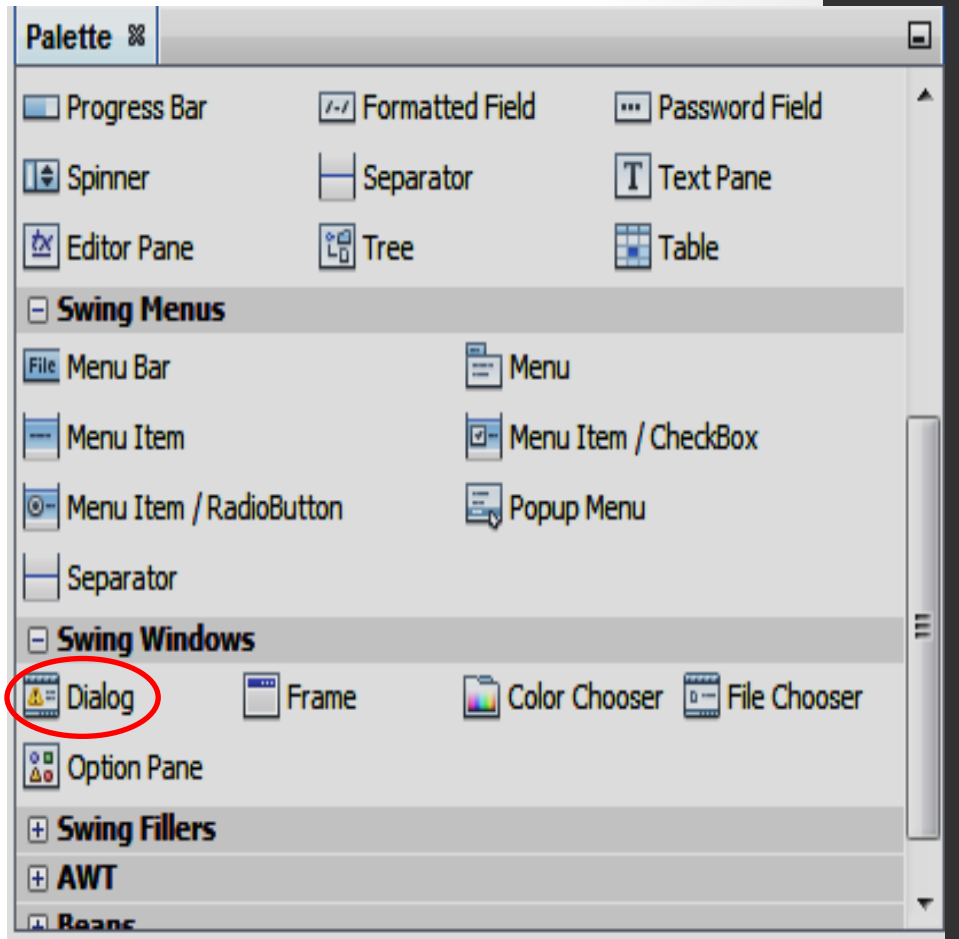
On veut afficher **un message**

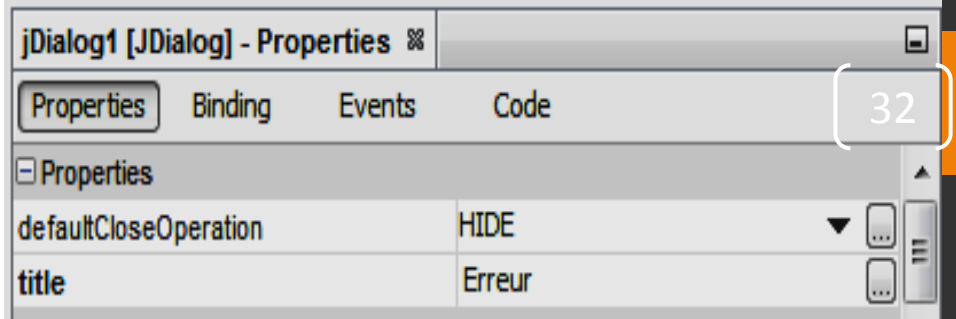
(une boite de dialogue)

pour avertir l'utilisateur

Création de la boite de dialogue







Code pour le bouton Ok

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    jDialog1.setVisible(false);
}
```

Masquer ou fermer la boite de dialogue

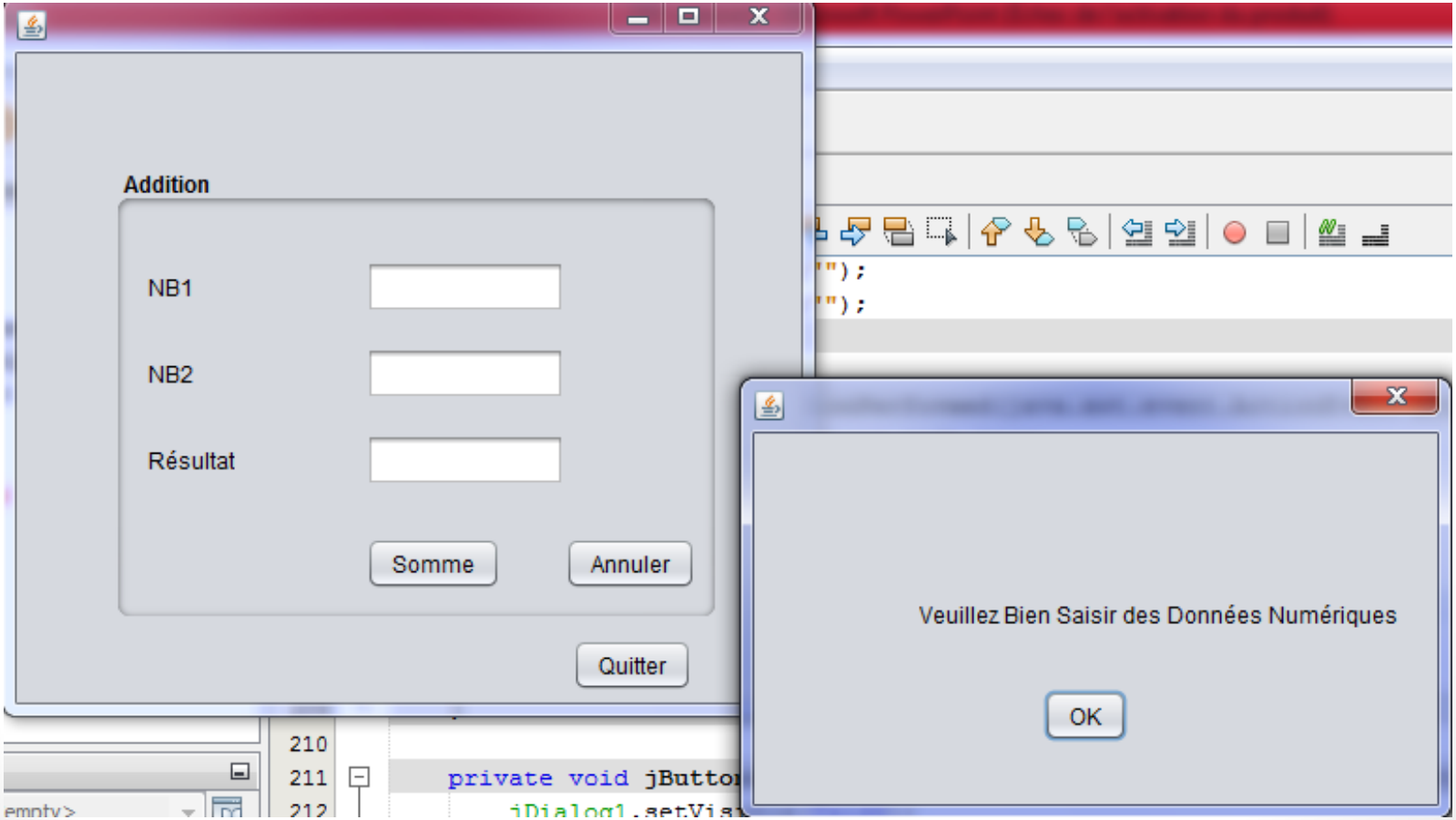
Bouton Somme : Insertion du try - catch

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    |
     try{
    float n1,n2,res;
    n1=Float.parseFloat(jTextField1.getText());
    n2=Float.parseFloat(jTextField2.getText());
    res=n1/n2;
    jTextField3.setText(String.valueOf(res));
    } catch (NumberFormatException e){
    jDialog1.setVisible(true);
                }
}
```

Type de l'exception

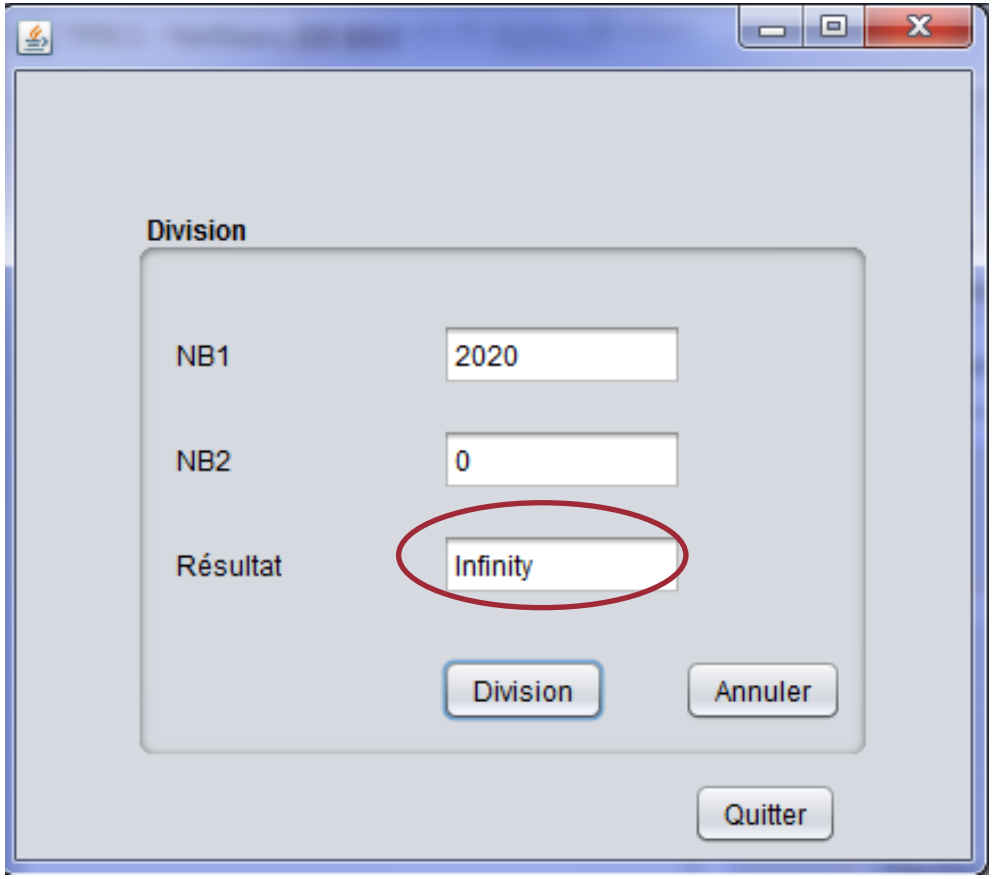
Visualiser la boite de dialogue

Exécuter …


Addition
NB1
NB2
Résultat
Somme
Annuler
Quitter
Veuillez Bien Saisir des Données Numériques
OK
210
211
212
private void jButto

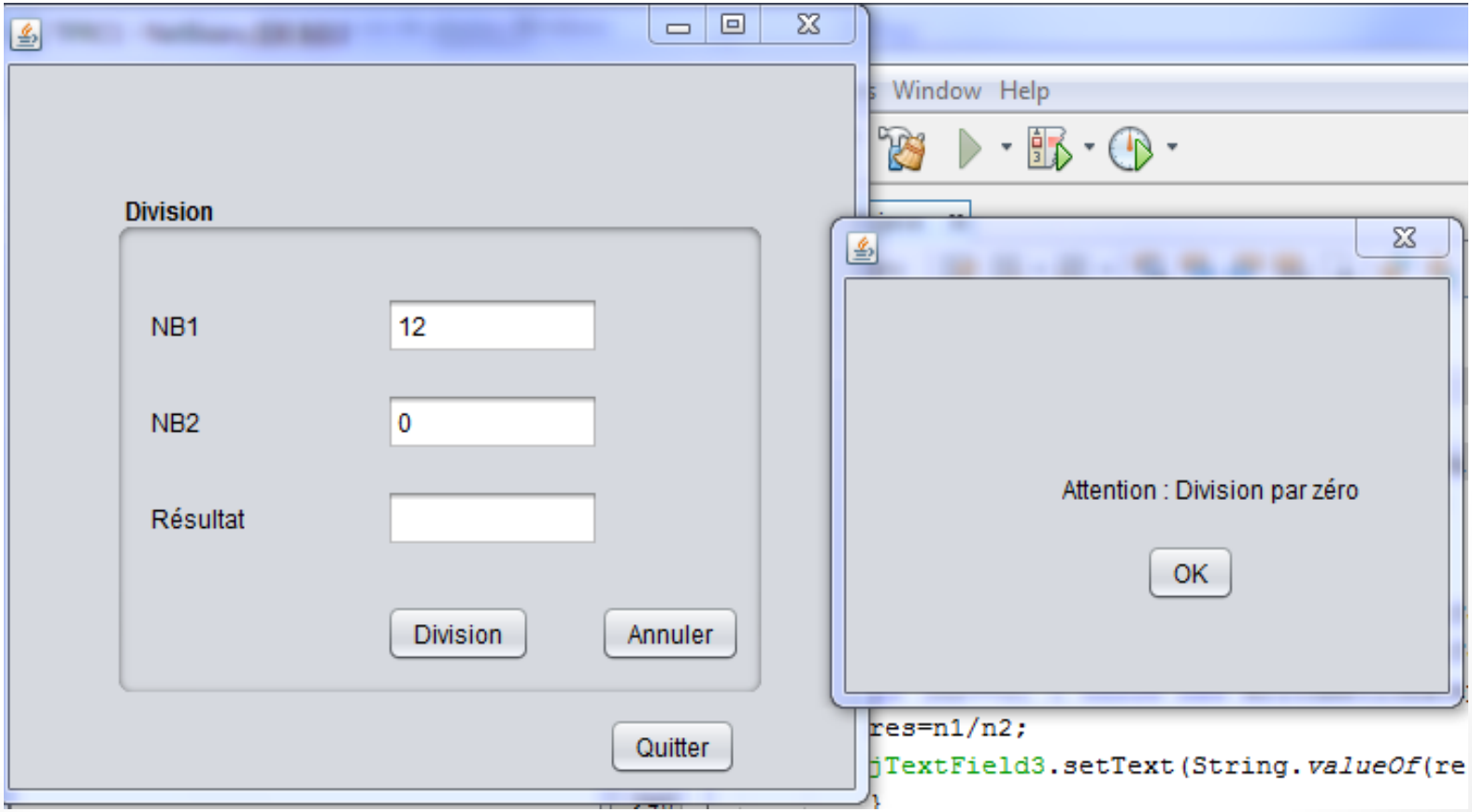
Remplacer la somme par une division

- Essayer la division par zéro

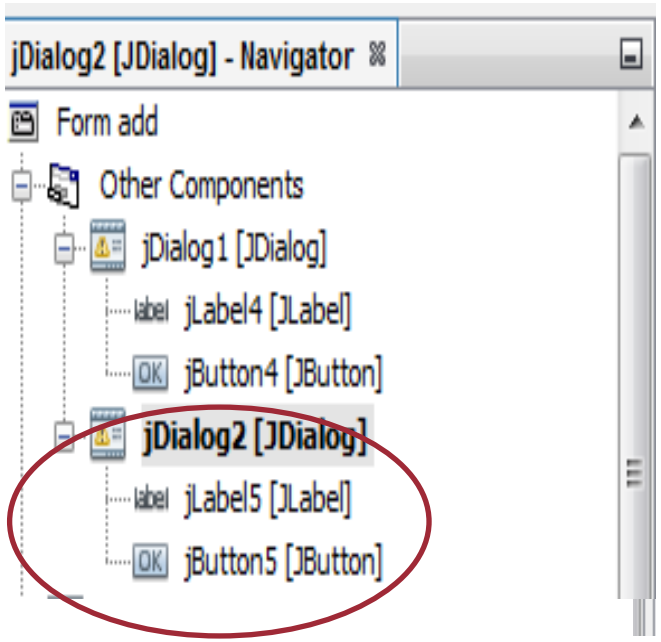


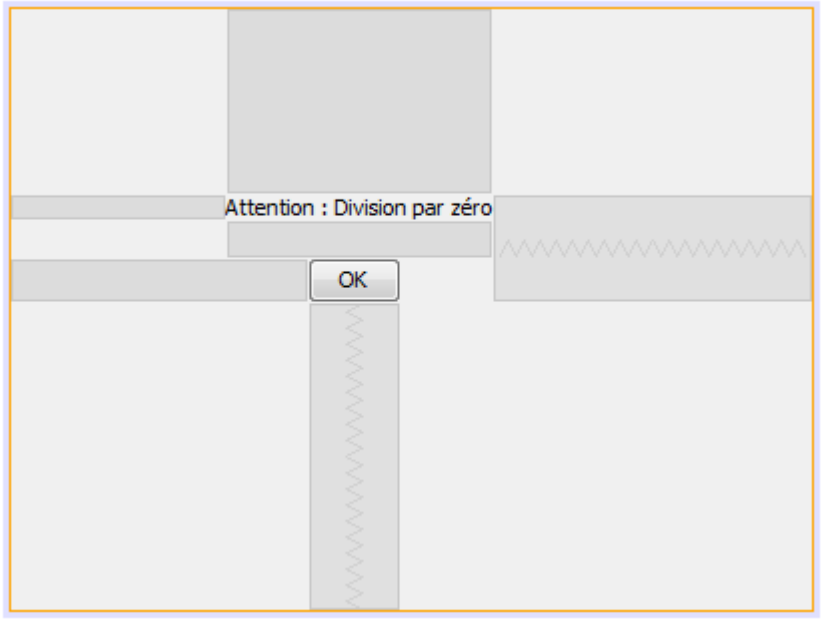
On veut signaler cette exception

On veut afficher une boite de dialogue pour avertir l'utilisateur



Design & code






```
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
    jDialog2.setVisible(false);
}
```

throw = lancer ou signaler

```
try{
float n1,n2,res;
n1= Float.parseFloat(jTextField1.getText());
n2= Float.parseFloat(jTextField2.getText());
if (n2==0) { throw new ArithmeticException ();}
res=n1/n2;
jTextField3.setText(String.valueOf(res));
}
catch (ArithmeticException e1) {jDialog2.setVisible(true);}
catch(NumberFormatException e2){jDialog1.setVisible(true);}
```