

# Utilisation de NetBeans

Auteur : E. Thirion - Dernière mise à jour : 20/01/15

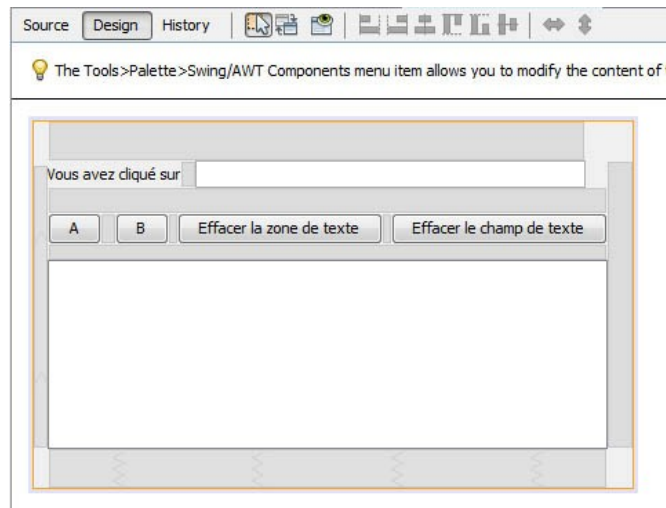
Extrait du site [cours.thirion.free.fr](http://cours.thirion.free.fr)

## Organisation des exercices

### Exercices de programmation pure

L'objectif de ce cours sur les bases de la programmation est avant tout de vous apprendre à programmer. La grande majorité des exercices de ce cours sont donc des exercices de programmation pure. C'est à dire que l'accent est mis sur l'écriture du code et que le côté 'design' de l'interface graphique est totalement ignoré. Un exercice sera donc un projet Java dont l'interface graphique est déjà réalisée et dont le code source est à compléter.

Prenons l'exemple du premier exercice: le projet **AfficherEffacer**. En ouvrant le projet avec NetBeans, vous aurez l'interface graphique du projet:



ainsi que le code source incomplet, dans le fichier **AfficherEffacer.java** :

```

Source Design History
1  /**
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  /**
7  *
8  * @author eric
9  */
10 public class AfficherEffacer extends javax.swing.JFrame {
11
12     /**
13     * Creates new form AfficherEffacer
14     */
15     public AfficherEffacer() {
16         initComponents();
17     }
18
19     /**
20     * This method is called from within the constructor to initialize the form.
21     * WARNING: Do NOT modify this code. The content of this method is always
22     * regenerated by the Form Editor.
23     */
24     @SuppressWarnings("unchecked")
25     Generated Code
118

```

Votre rôle consistera à compléter ce code source de manière à faire fonctionner le programme. La majorité des exercices sont organisés de cette manière.

### **Exercice de programmation et de design**

Exceptionnellement, il se peut que le projet soit à réaliser de A à Z, interface graphique incluse. C'est par exemple le cas de l'exercice de prise en main de NetBeans.

## **Organisation de ce document**

Ce document est divisé en trois parties:

### **Partie 1: L'interface graphique de NetBeans**

<a href="#">L'interface graphique de NetBeans en mode Design</a>	page 3
<a href="#">L'interface graphique de NetBeans en mode Source</a>	page 4

### **Partie 2: Informations utiles pour tous les exercices**

<a href="#">Démarrer NetBeans</a>	page 5
<a href="#">Ouvrir un projet existant</a>	page 5
<a href="#">Résoudre le problème d'accès à la librairie</a>	page 6
<a href="#">Créer ou modifier la procédure événementielle associée à un bouton</a>	page 8
<a href="#">Exécuter l'application</a>	page 9
<a href="#">Arrêter le programme</a>	page 9
<a href="#">Sauvegarder</a>	page 10

### **Partie 3 : Informations utiles uniquement pour les exercices incluant le design**

<a href="#">Créer une nouvelle application</a>	page 11
<a href="#">Ajouter une fenêtre</a>	page 13
<a href="#">Déposer des composants sur la fenêtre de l'application</a>	page 14
<a href="#">Modifier le libellé d'une étiquette ou d'un bouton</a>	page 15
<a href="#">Vider le contenu initial d'un champ de texte</a>	page 15
<a href="#">Changer le nom d'un composant</a>	page 16
<a href="#">Donner un titre à la fenêtre</a>	page 17
<a href="#">Importer la librairie JavaEtBib</a>	page 18

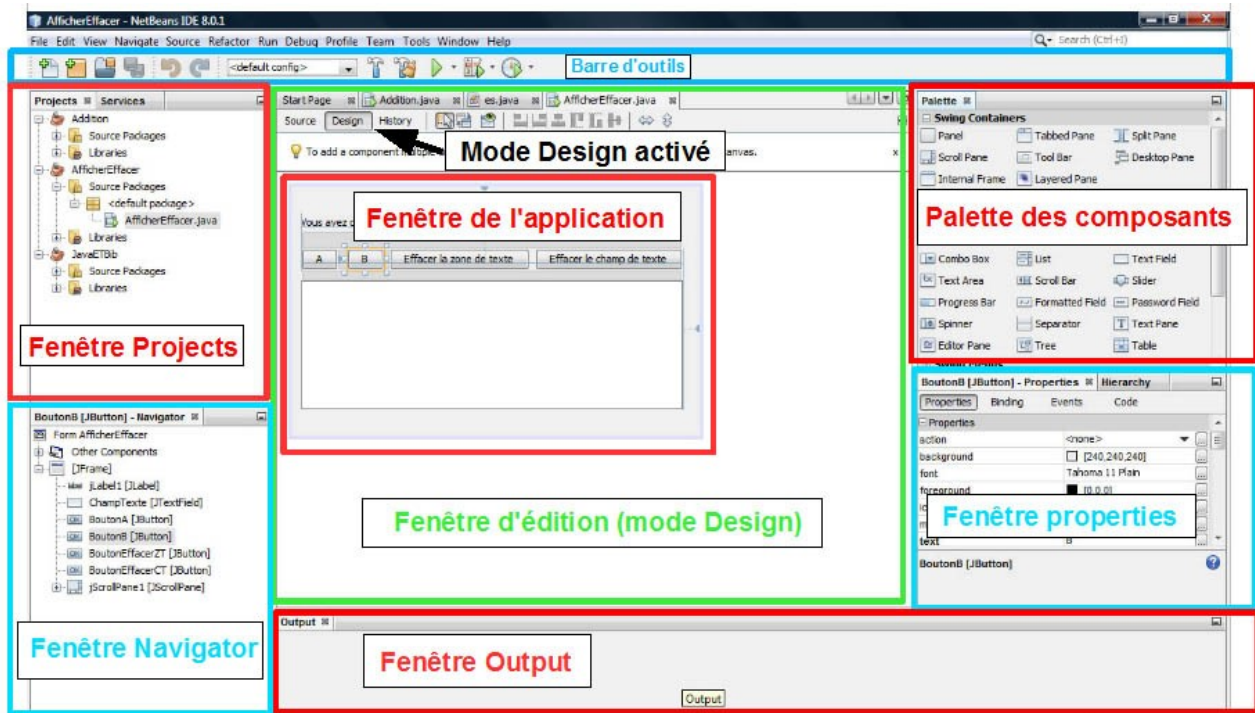
## Partie I : L'interface graphique de NetBeans

L'interface graphique de NetBeans se décompose en différentes fenêtres, menus et barres d'outils. Elle n'a pas la même apparence en mode **Design**, qu'en mode **Source**.

### Apparence de l'interface en mode Design

Le mode **Design**, comme son nom l'indique, est prévu pour permettre le développement de l'interface graphique d'une application.

Dans ce mode, la fenêtre de l'application est affichée à l'intérieur de la fenêtre d'édition:

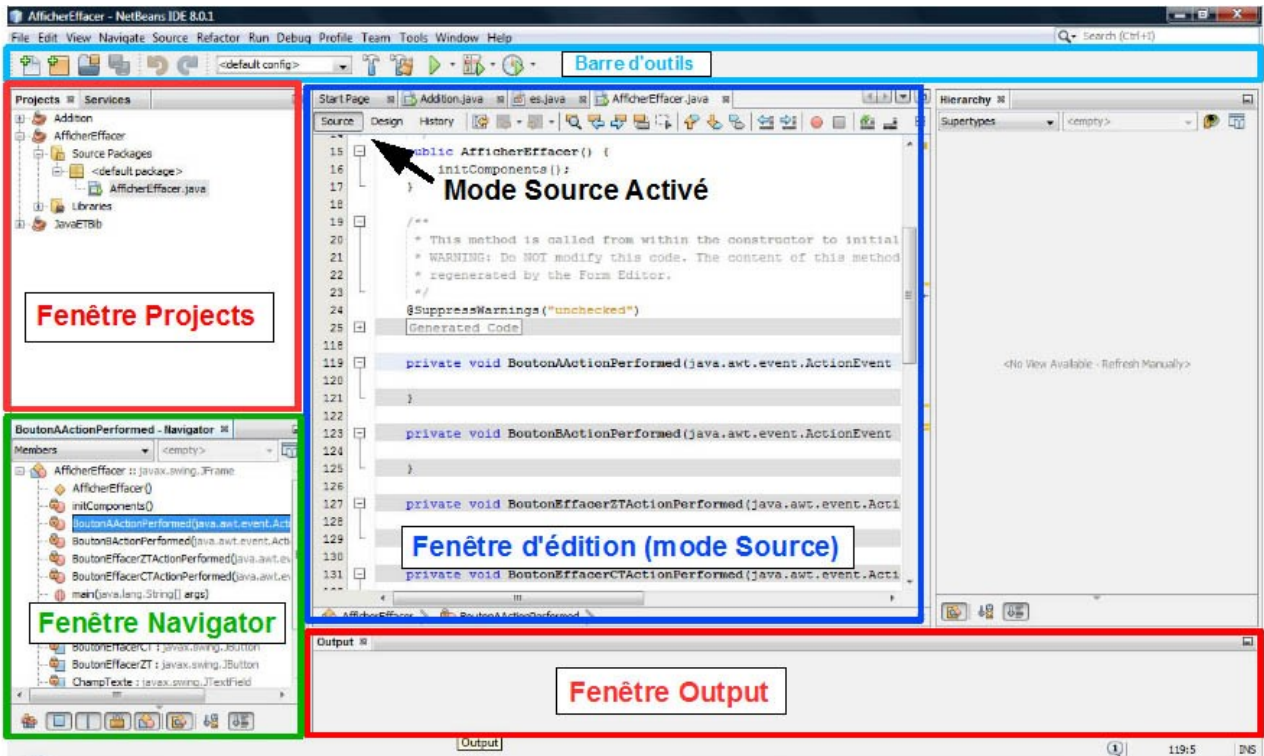


Certaines des fenêtres présentées ci-dessus sont spécialement prévues pour afficher ou modifier les propriétés graphiques de la fenêtre de l'application:

- la **palette des composants** permet d'y ajouter de nouveaux composants.
- la **fenêtre propriétés** ainsi que la **fenêtre Navigator** permettent d'afficher ou de modifier les propriétés d'un composant particulier de l'application.

## L'interface de NetBeans en mode Source

Le mode source est fait pour la programmation. Dans ce cas, NetBeans affiche le texte du fichier source dans la fenêtre d'édition:



La **barre d'outil** située tout en haut vous permettra en particulier d'ouvrir, exécuter ou sauvegarder un projet existant.

## Partie 2: Informations utiles pour tous les exercices

### Démarrer NetBeans

Double cliquez sur l'icone de NetBeans :

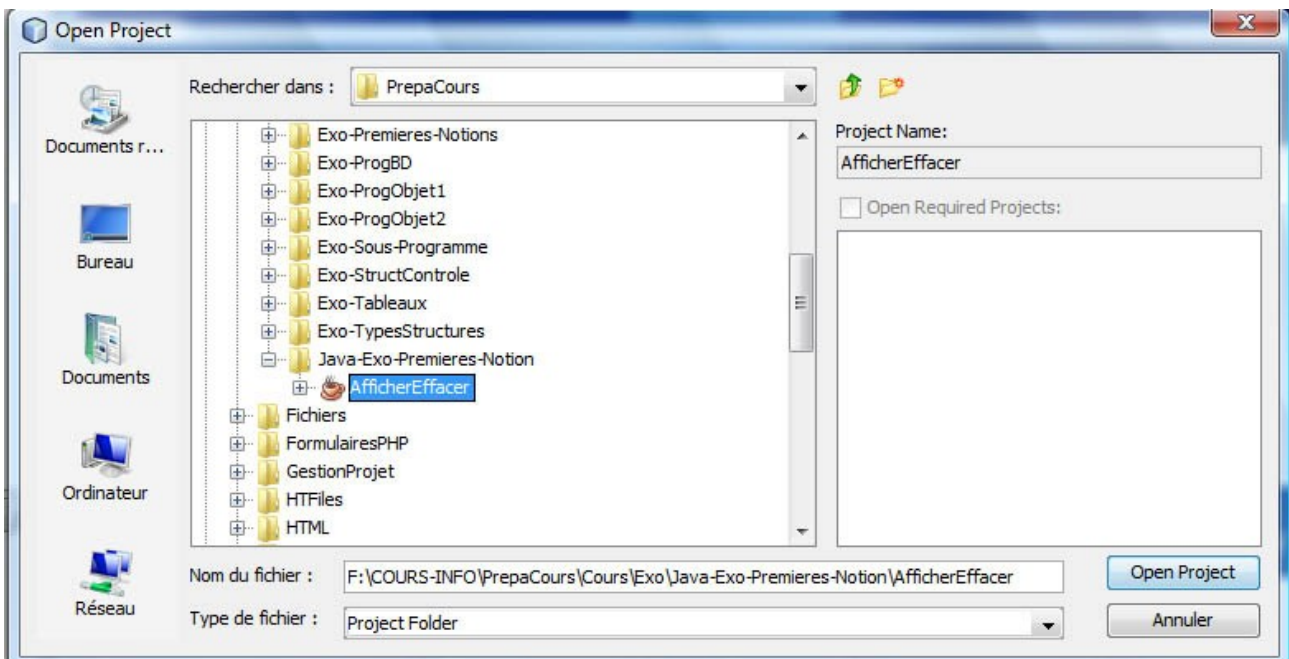


### Ouvrir un projet existant

Cliquez sur l'icone d'ouverture de projet dans la [barre d'outils](#):

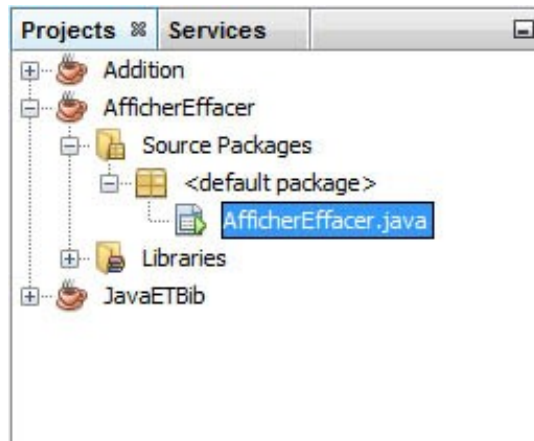


Ensuite, dans la fenêtre suivante (intitulée **Open Project**), sélectionner le repertoire contenant le projet à ouvrir. Il apparait en principe avec une petite tasse de café à gauche (le symbole de Java):



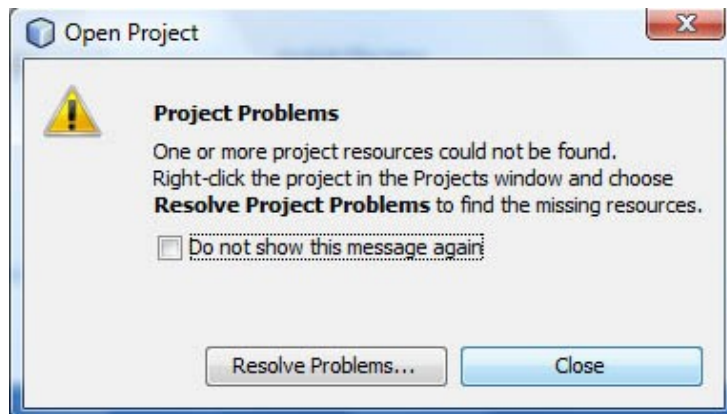
Puis, cliquez sur le bouton **Open Project**.

Si NetBeans n'affiche ni le code source du projet, ni sa fenêtre, double-cliquez sur le fichier source dans la fenêtre Project. Par exemple, avec l'ouverture du projet **AfficherEffacer**, il faudra double-cliquer sur **AfficherEffacer.java** :



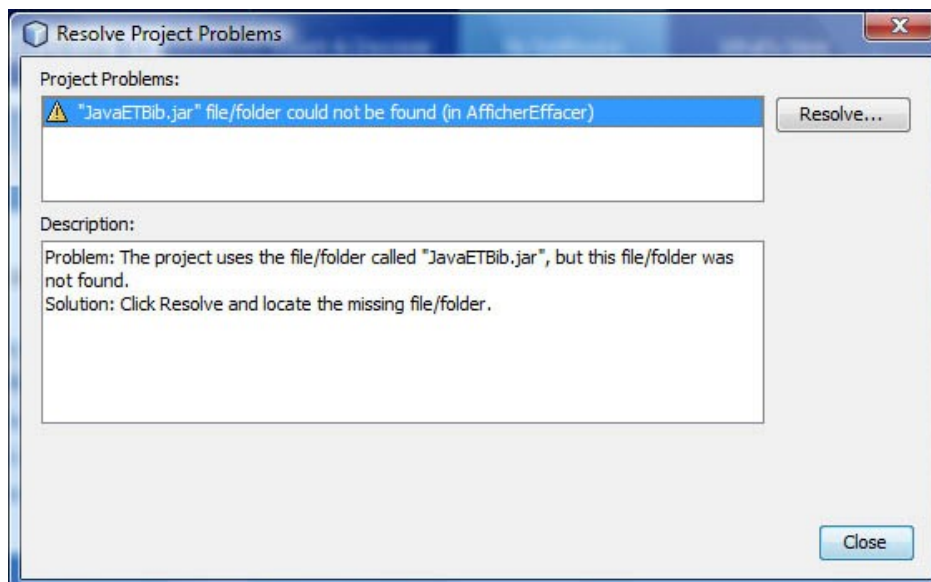
## Résoudre le problème d'accès à la librairie

Si vous obtenez le message d'erreur suivant à l'ouverture du projet:



cliquez sur le bouton **Resolve Problems ...**

S'il s'agit bien d'un problème d'accès à la librairie, vous devez avoir une message d'erreur **JavaETBib.jar file/folder could not be found ...**

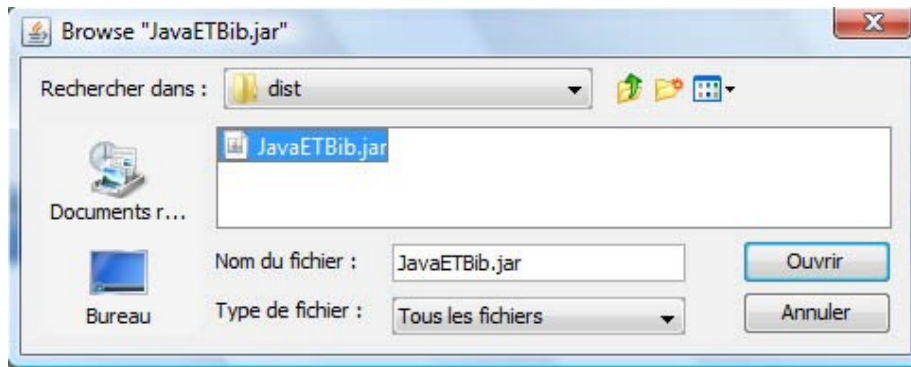


L'origine de ce problème est que votre répertoire d'exercice est mal positionné par rapport à la librairie **JavaETBib**.

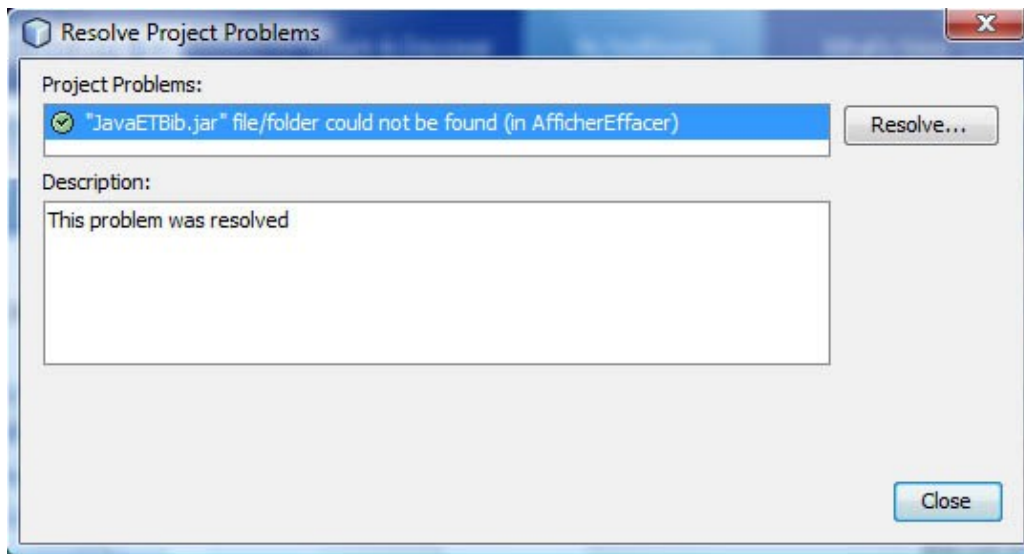
Vérifiez tout d'abord que vous avez bien respecté les consignes d'installation. Si ce n'est pas le cas, cliquez sur **Close** et corrigez la position du répertoire du projet ou de la librairie.

Si les consignes d'installation sont bien respectées, vous pouvez résoudre le problème en refédisant le chemin d'accès à la librairie.

Pour cela, cliquez sur le bouton **Resolve ...** et sélectionnez le fichier **JavaETBib.jar** (il se trouve en principe dans le répertoire **dist**, lui même contenu dans le répertoire **JavaETBib**):



Cliquez sur le bouton **Ouvrir**. Si le problème est bien résolu, NetBeans vous affiche 'This problem was resolved' :



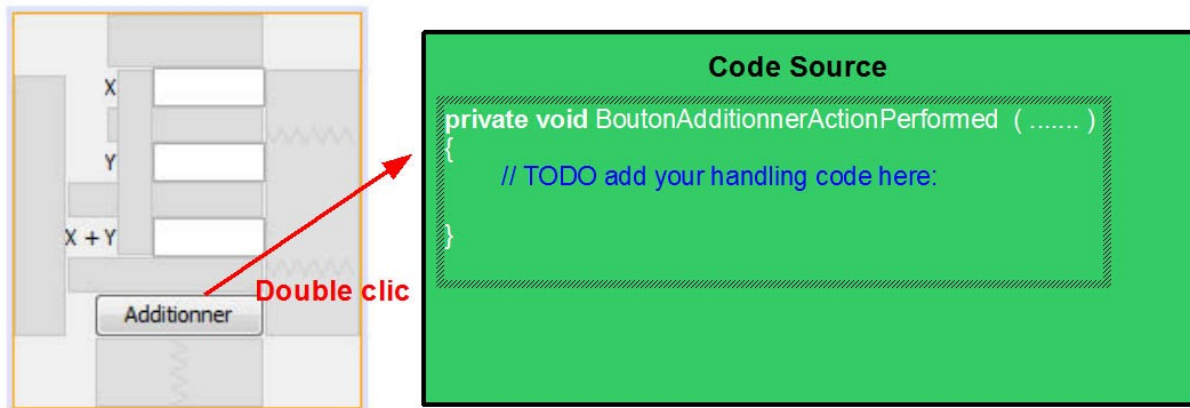
cliquez sur le bouton **Close**. La librairie **JavaETBib** est à présent correctement importée dans votre projet.

## Créer ou modifier la procédure événementielle associée à un bouton

### Méthode 1: Vous êtes en mode Design

Faites un double clic sur le bouton. Deux possibilités:

- la procédure événementielle n'existe pas. Dans ce cas, NetBeans va générer automatiquement une partie du code de la procédure événementielle. Par exemple, si le nom de votre bouton est **BoutonAdditionner**, vous obtiendrez ceci:



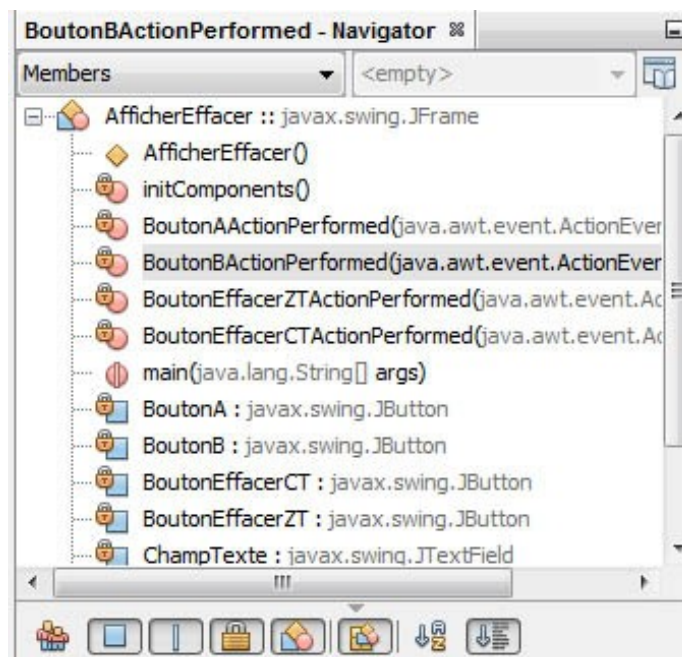
- la procédure événementielle existe. Dans ce cas, NetBeans vous positionne directement à l'intérieur de la procédure pour vous permettre de modifier son code.

### Méthode 2: Vous êtes en mode Source

Cette méthode ne permet que de modifier une procédure événementielle qui existe déjà. Pour la créer, utilisez la méthode 1.

Double-cliquez sur le nom de la procédure dans l'onglet **Members** la fenêtre **Navigator**. Rappelez vous que le nom de cette procédure est formée du nom du bouton suivi de **ActionPerformed**.

La copie d'écran suivante illustre par exemple l'accès à la procédure événementielle du bouton **BoutonB** :



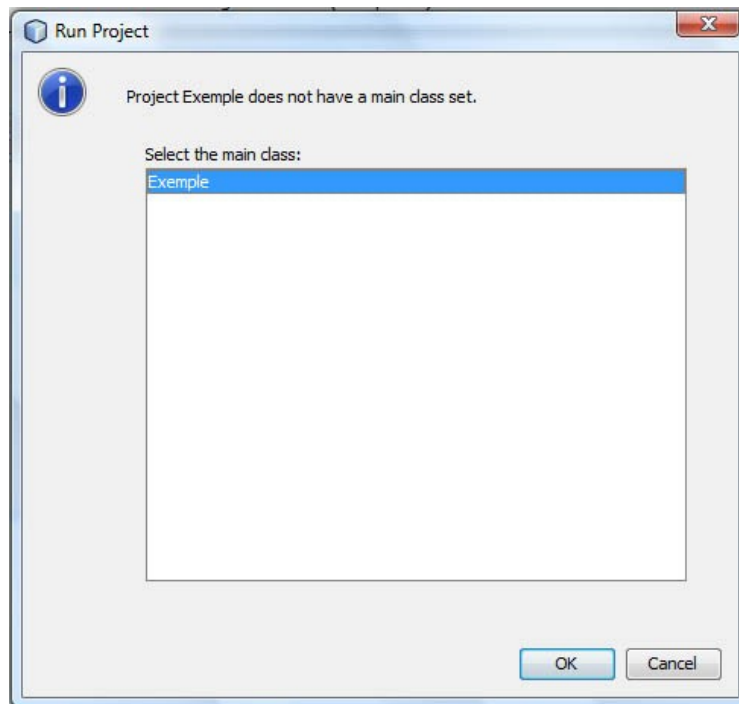


## Exécuter l'application

Cliquez sur l'icone suivant dans la [barre d'outils](#):



Si vous êtes dans un exercice à réaliser de A à Z, vous aurez probablement le message suivant à la première exécution:

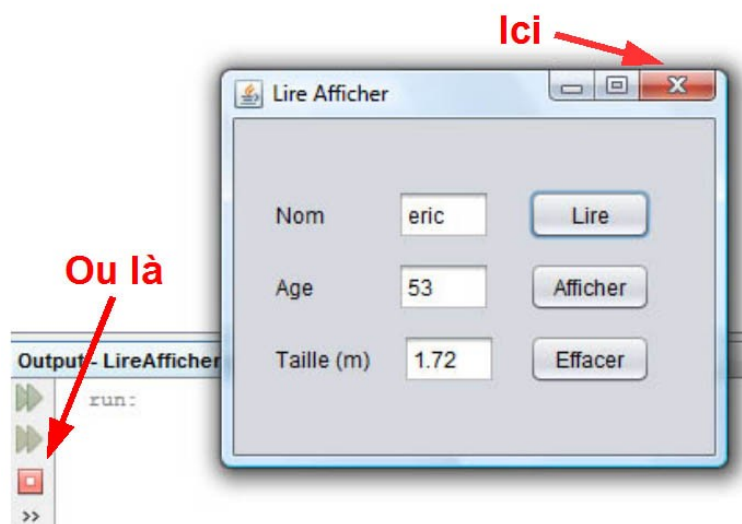


Cliquez simplement sur OK.

## Arrêter le programme

Pour arrêter un programme en cours d'exécution, vous pouvez soit fermer la fenêtre de l'application ou cliquer sur le carré rouge dans la fenêtre **Output**.

Par exemple, pour arrêter l'application **LireAfficher**, on peut cliquer



## **Sauvegarder l'application**

Cliquez sur l'icone suivant dans la [barre d'outils](#):



S'il apparait en grisé, c'est que vous n'avez fait aucune modification depuis la dernière sauvegarde et que par conséquent, il n'y a rien à sauvegarder.

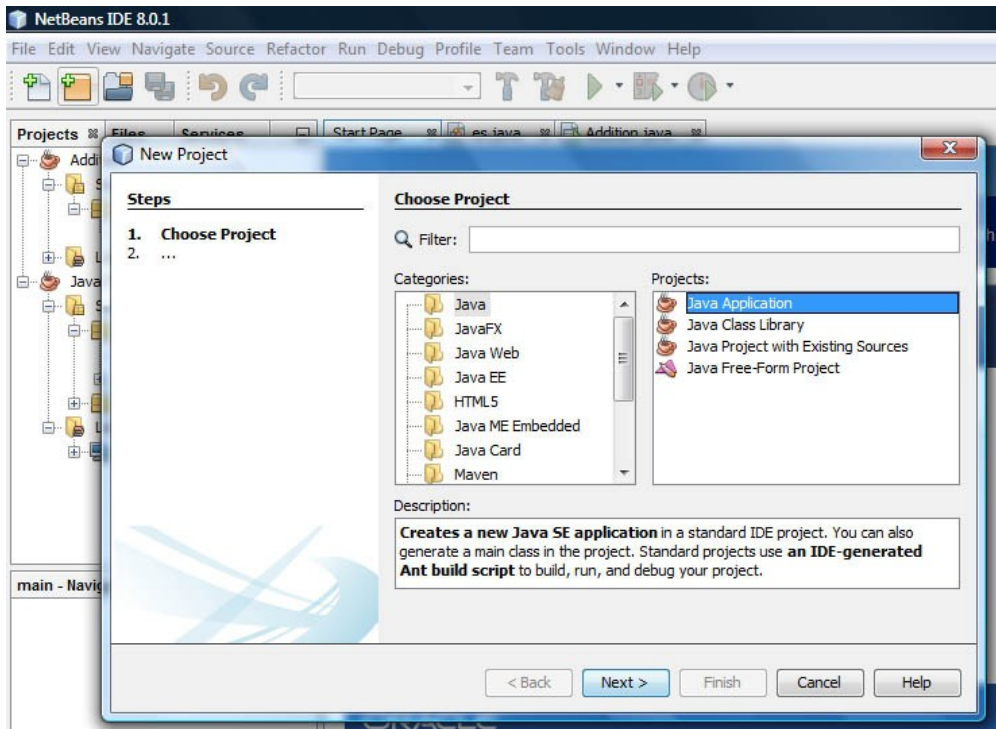
## Partie 3 : Informations utiles pour les exercices incluant le Design

### Créer une nouvelle application (ou autrement dit, un nouveau projet)

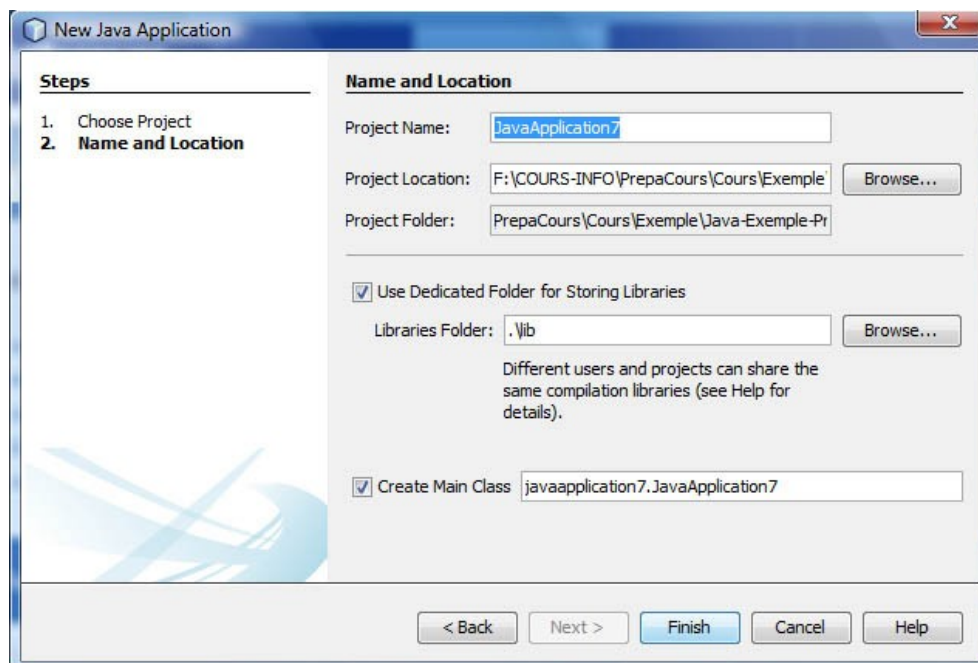
Cliquez sur l'icone suivant dans la [barre d'outils](#):



Sélectionnez ensuite **Java Application** et cliquez sur **Next** :



la fenêtre suivante s'affiche alors:



Vous devez préciser ici le nom de votre projet dans le champ de texte étiqueté **Project Name**. Choisissez le

nom qui vous conviendra, mais de préférence un nom en relation avec la finalité de votre projet.

Vous devez également sélectionner un répertoire dans lequel sera stocké votre projet. Cliquez sur le bouton **Browse** à côté de **Project Location** et sélectionnez votre répertoire.

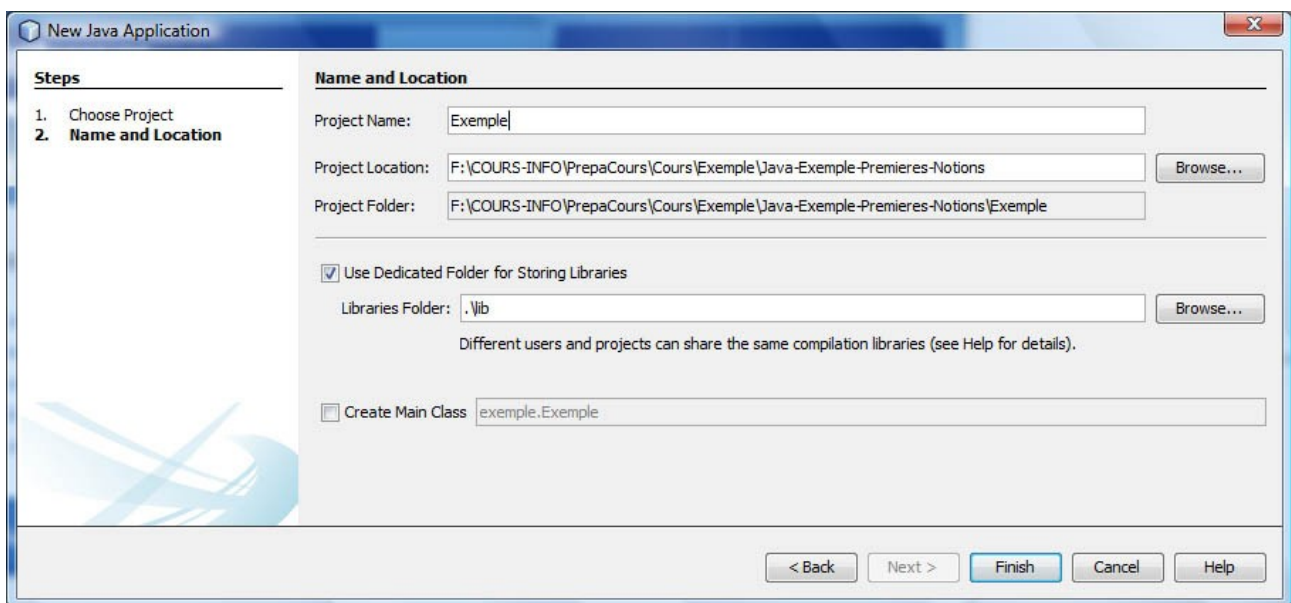
Enfin, il est nécessaire à cette étape de décocher **Create Main Class**.

Lorsque vous avez fait ceci, cliquez sur le bouton **Finish**.

Vous constaterez alors:

- qu'un répertoire de même nom que votre projet aura été créé à l'intérieur du répertoire que vous avez sélectionné.
- que votre projet apparaît dans la fenêtre de **Projects** de NetBeans.

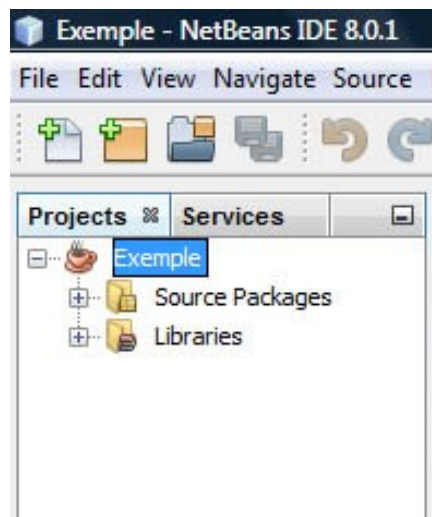
Ceci, par exemple:



va nous créer un répertoire **Exemple** (nom du projet) à l'intérieur du répertoire

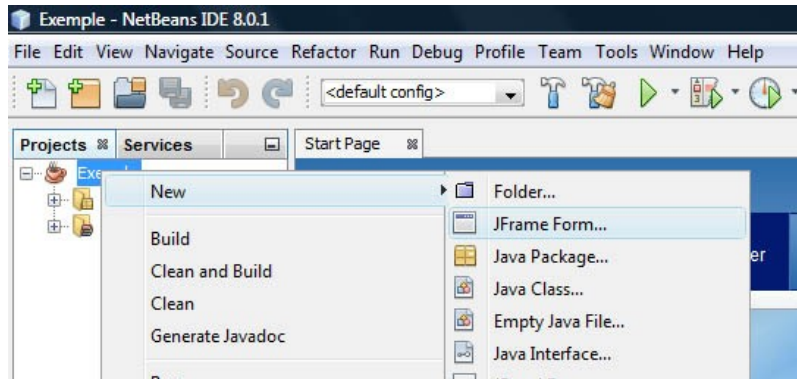
F:\COURS-INFO\PrepaCours\Cours\Exemple\Java-Exemple-Premieres-Notions

et nous afficher ceci:

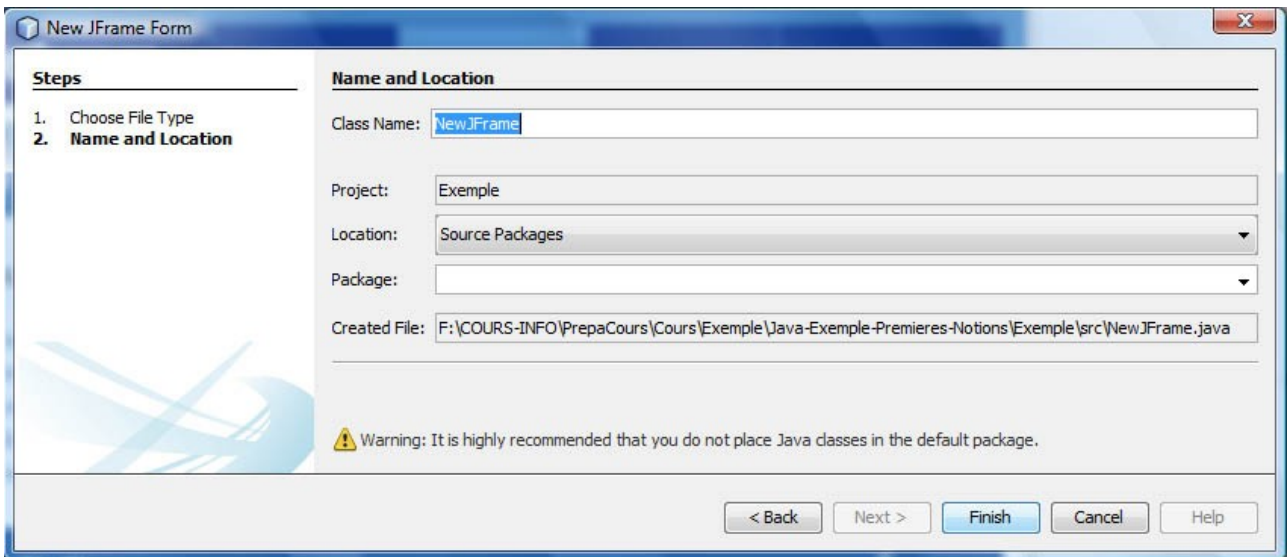


## Ajouter une fenêtre à l'application

Faite un clic droit sur le nom de votre application dans la fenêtre **Projects** et sélectionnez, **New** et **JFrame Form**:



la fenêtre suivante apparaît alors:

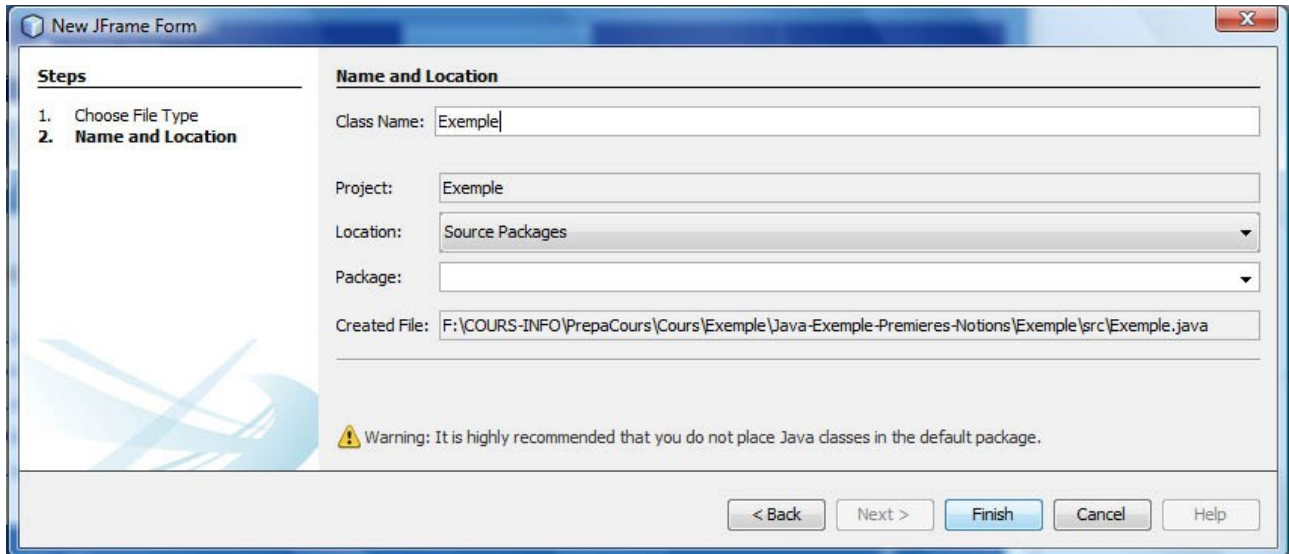


le champ de texte étiqueté **Class Name** sert (entre autres) à définir le nom du fichier dans lequel figurera le code source associé à la nouvelle fenêtre. Le nom par défaut est **NewJframe**.

Les applications que nous allons développer dans la partie 'Base de la programmation' seront toujours des applications à une seule fenêtre et un seul fichier source. Il est assez judicieux dans ce cas (mais non obligatoire !) de donner le même nom à ce fichier qu'au projet.

Lorsque vous aurez saisi ce nom et cliqué sur le bouton **Finish**, vous constaterez qu'un fichier d'extension **.java** de même nom aura été créé dans le répertoire **src** de votre projet.

Ceci, par exemple:

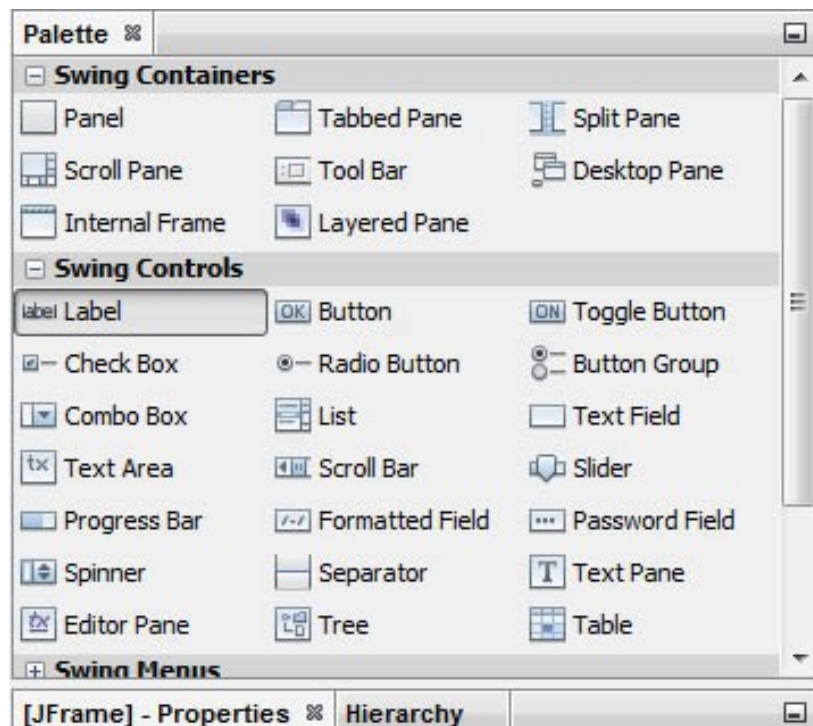


va nous créer un fichier **Exemple.java** dans le répertoire **src** (génééré automatiquement par NetBeans) situé à l'intérieur du repertoire du projet **Exemple**.

## Déposer des composants sur la fenêtre de l'application

Pour ajouter un composant sur la fenêtre de l'application, il suffit de cliquer sur ce composant dans la [palette de composants](#), puis de cliquer sur la fenêtre de l'application à l'endroit où vous souhaitez qu'il apparaisse.

Voici par exemple la sélection d'une étiquette (**Label** en anglais):



Lorsque l'on clique ensuite sur la fenêtre de l'application, on obtient ceci:

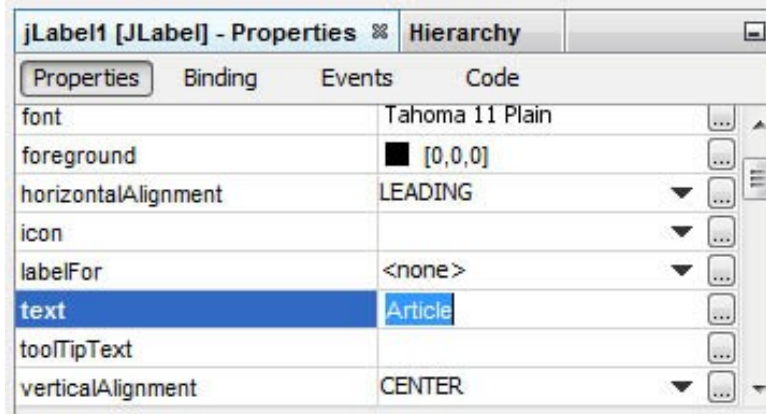


On procédera de même pour un bouton (composant **Button**), un champ de texte (composant **TextField**) ou une zone de texte (composant **TextArea**).

### **Modifier le libellé d'une étiquette ou d'un bouton**

Pour modifier le libellé d'une étiquette ou d'un bouton, cliquez dessus et mettez le libellé voulu dans la propriété **text** de la [fenêtre Properties](#).

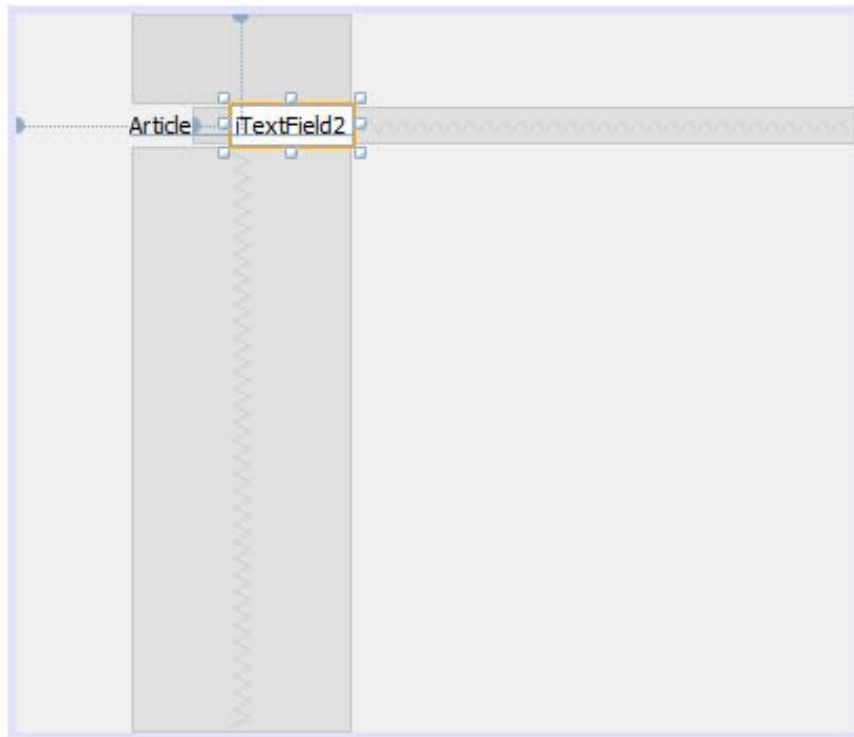
Par exemple, pour que le libellé de notre étiquette devienne 'Article', au lieu de **jLabel1** :



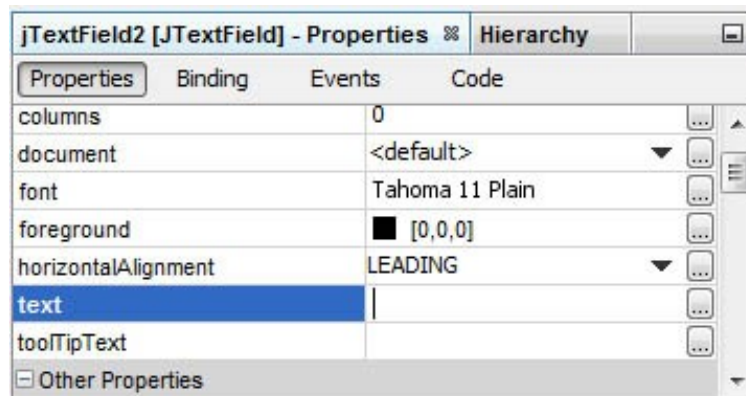
### **Vider le contenu initial d'un champ de texte**

En général, lorsqu'on lance une application, les champs de texte doivent être vides. Malheureusement, NetBeans génère automatiquement du texte à l'intérieur des nouveaux champs de texte déposés sur la fenêtre de l'application. Vous constaterez qu'un nouveau champ de texte contient initialement son nom par défaut. Pour le supprimer, il suffit d'effacer le contenu de la propriété **text**.

Dans l'exemple suivant, nous avons ajouté un champ de texte à droite de l'étiquette 'Article'. Ce champ de texte contient le nom par défaut du champ de texte, c'est à dire **jTextField2** dans ce cas:



Pour que ce texte n'apparaisse pas, on efface la propriété **text** de ce composant:



### Changer le nom d'un composant

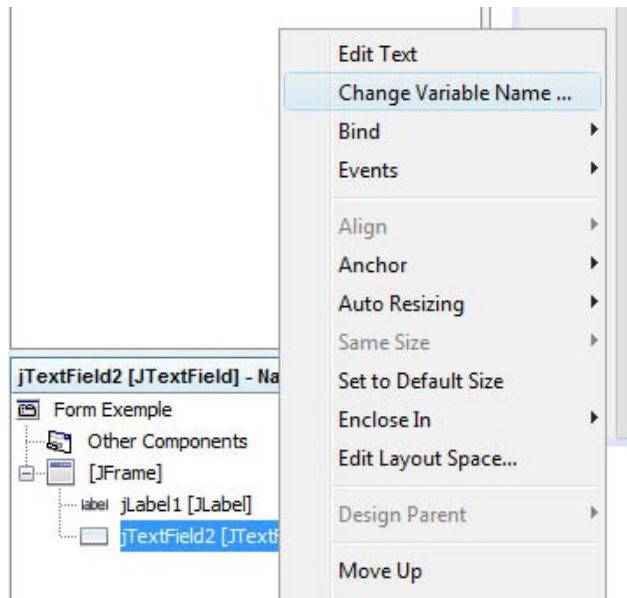
Pour rendre les programmes plus lisibles, il vaut mieux ne pas utiliser les noms par défaut des composants. Si votre programme comporte par exemple une dizaine de champ de texte nommé **JTextField1**, **JTextField2**, ... vous aurez du mal à vous en sortir. Il vaut donc mieux donner des noms évocateurs, en rapport avec l'utilité du composant dans l'application.

Je vous conseil également de préfixer les noms des composants en fonction du type de composant. Par exemple, vous pouvez faire commencer les nom des champs de texte par **CT\_**, les noms des zones de texte par **ZT\_**, et ceux des boutons par **BT\_**

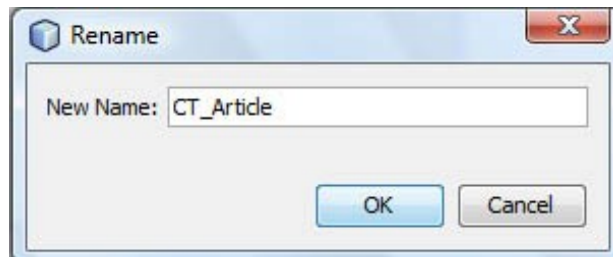
Pour modifier le nom d'un composant, faite un clic droit sur le nom du composant dans la [fenêtre Navigator](#) et sélectionnez **Change Variable Name ...** dans le menu qui apparait. Vous obtiendrez alors une boite de dialogue qui vous permettra de saisir le nouveau nom du composant.

Par exemple, pour changer le nom du champ de texte **JTextField2** en **CT\_Article** :





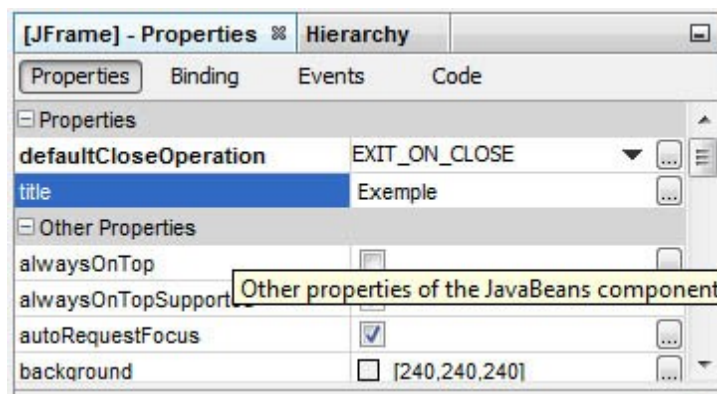
et :



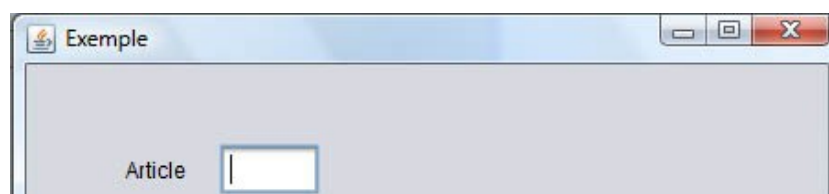
### Donner un titre à la fenêtre

Pour donner un titre à votre fenêtre, cliquez sur le fond et tapez le titre dans la propriété **title** de la **fenêtre Properties**. Le titre de la fenêtre n'est pas visible en mode Design, mais le sera dès que vous aurez lancé le programme.

Par exemple:



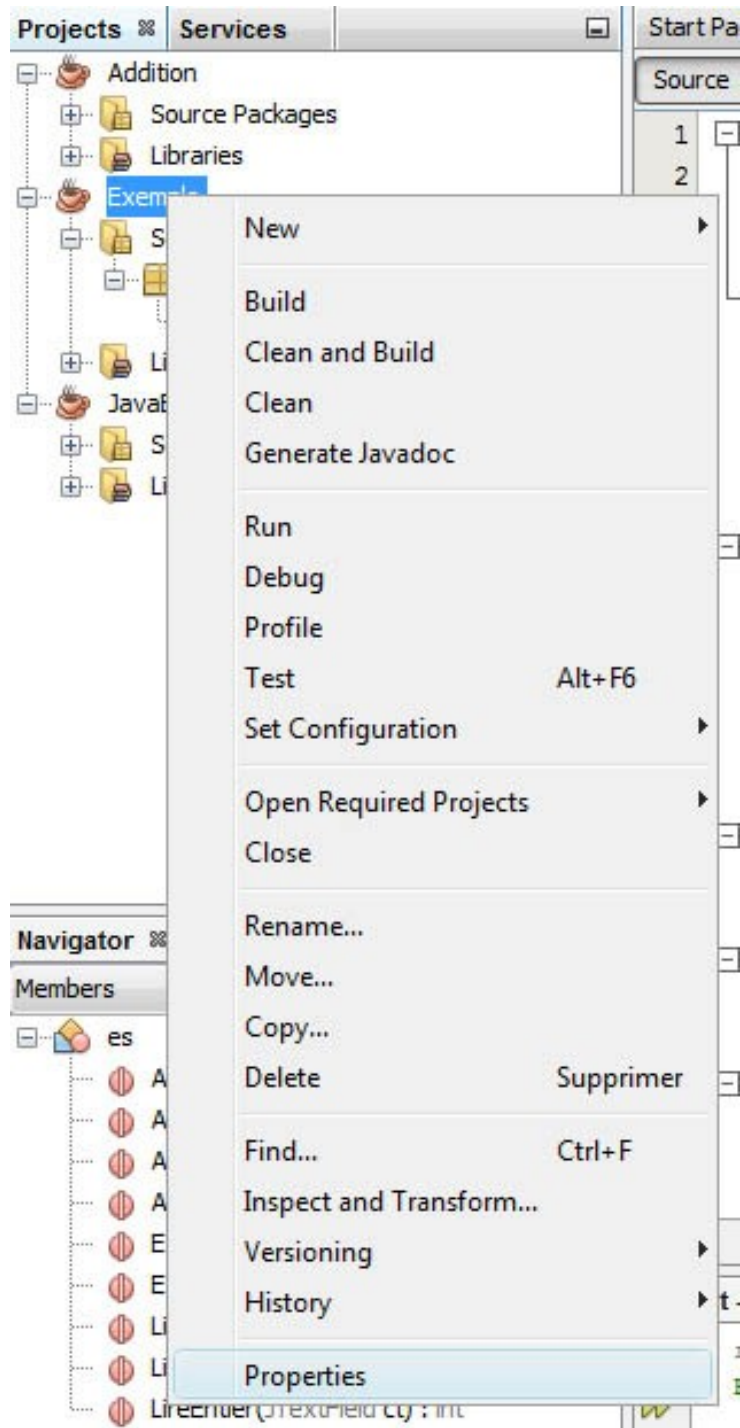
donnera ceci à l'exécution:



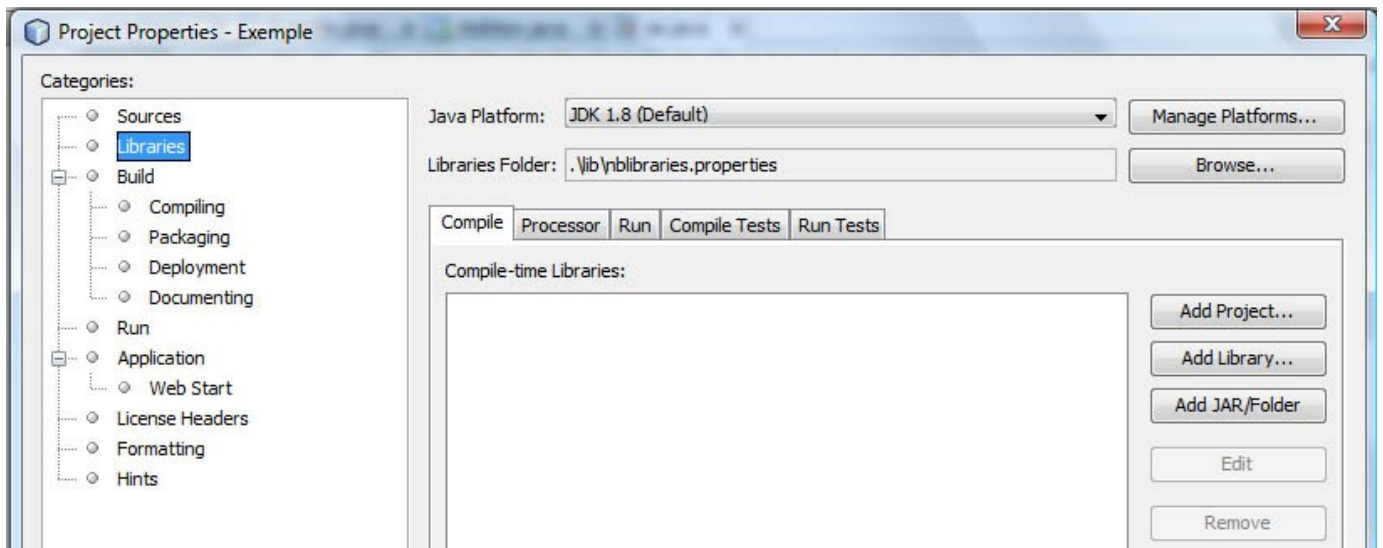
## Importer la librairie JavaETBib

Pour pouvoir utiliser les procédures de lecture et d'affichage de la librairie **JavaETBib** (es.Afficher, es.Lire, es.LireEntier, .... etc), vous devez inclure la librairie dans votre projet.

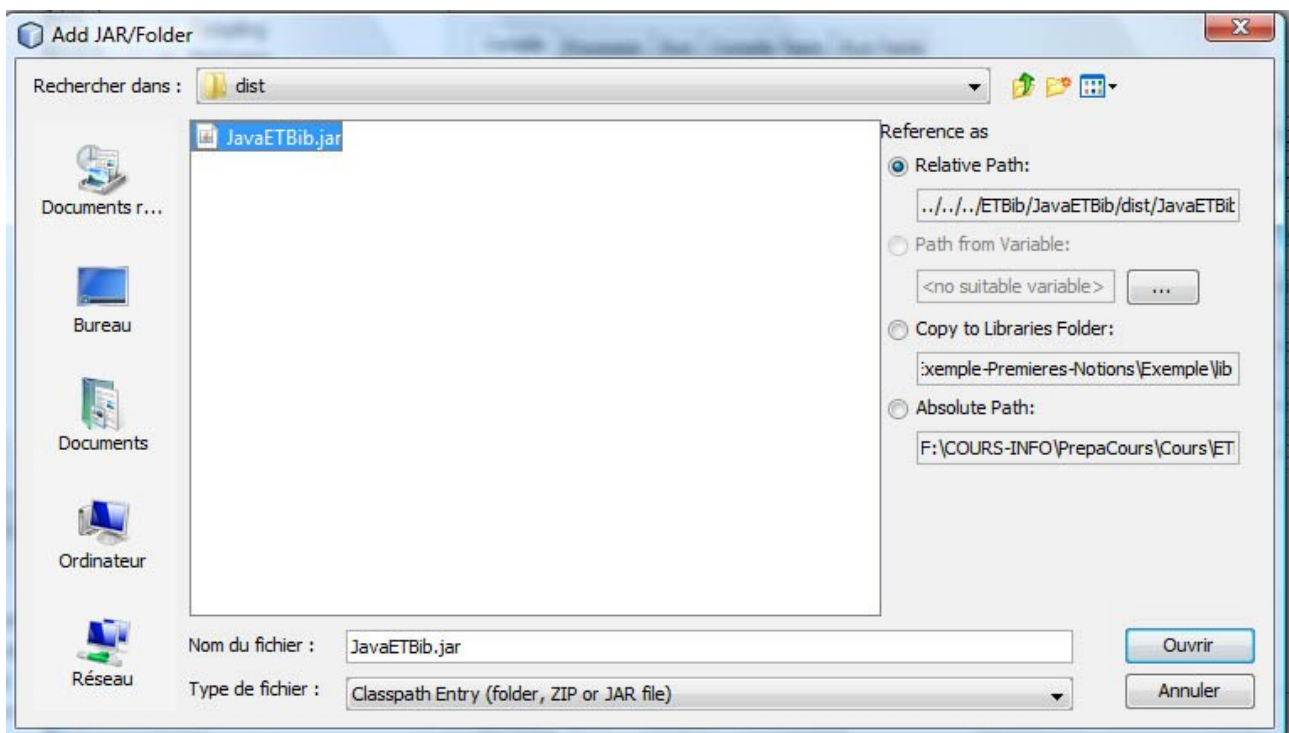
Pour cela, faites un clic droit sur le nom du projet dans la fenêtre **Projects** et sélectionner **Properties** :



Dans la fenêtre qui apparait, cliquez sur **Librairies** :



Puis sur le bouton **Add JAR/Folder**. Vous allez alors avoir une nouvelle fenêtre qui vous permettra définir l'emplacement de la librairie. Il s'agit plus précisément de définir l'emplacement du fichier **JavaETBib.jar**. Il doit se trouver à l'intérieur du répertoire **dist**, contenu lui même dans le répertoire **JavaETBib**. Sélectionnez de préférence **Relative Path**:



Cliquez ensuite sur **Ouvrir**, puis **OK**.