

# Création d'une applet Java avec JBuilder 9 sous Linux

## I - Avant de commencer

Avant de commencer, vous devez connaître le répertoire de travail dans lequel sera créé les fichiers de votre applet. Pour connaître le répertoire de travail de votre machine, cliquez sur l'icône « Répertoire perso » situé sur le bureau de Linux. Votre répertoire de travail est écrit dans la barre d'adresse, en haut à gauche de la fenêtre, à côté de **Emplacement**. Exemple de répertoire de travail : **/home/elevés**

Dans la suite de ce tutorial, le répertoire de travail sera mentionné par **< répertoire de travail >**

Par exemple, si votre répertoire de travail est **/home/elevés**, la ligne suivante indique le répertoire **/home/elevés/projet** :

**< répertoire de travail >/projet**

Vous pouvez maintenant lancer JBuilder 9. Pour cela, cliquez sur l'icône JBuilder située sur le bureau de Linux.

## II - Création d'une applet de base avec JBuilder 9

Si un projet est déjà ouvert lors du lancement de JBuilder, il faut le fermer :

- \* Menu **Fichier** + **Fermer les projets...**
- \* Bouton **Tous**
- \* Bouton **OK**

Création d'un nouveau projet :

- \* Menu **Fichier** + **Nouveau projet...**
- \* Une boîte de dialogue « Expert projet » s'ouvre sur l'écran.
- \* Dans le champ **Nom**, entrez le nom du projet : **mon\_applet**
- \* Puis cliquez sur le bouton **Terminer**

Le nouveau projet a été créé dans le répertoire **< répertoire de travail >/jbproject/mon\_applet**

Création d'une nouvelle Applet :

- \* Menu **Fichier** + **Nouveau...**
- \* Une boîte de dialogue « Galerie d'objets » s'ouvre sur l'écran.
- \* Cliquez sur l'onglet « **Web** », puis sur l'icône « **Applet** », et enfin sur le bouton « **OK** ».
- \* Une boîte de dialogue « **Expert applet** » s'ouvre sur l'écran.
- \* Dans le champ **Nom de classe**, entrez le nom de la nouvelle classe : **ma\_classe**
- \* Puis cliquez sur le bouton **Terminer**

Le programme source de l'applet s'affiche alors à l'écran en langage Java.

Pour exécuter le projet, il faut aller dans le menu **Exécuter**, puis cliquer sur **Exécuter le projet**.

Si on exécute le projet maintenant, on constate que notre applet ressemble pour l'instant à un rectangle gris, sans texte ni bouton. Pour arrêter l'exécution du projet, il faut cliquer sur le bouton **Quitter** de la fenêtre de l'applet.

## III - Ajout d'un texte sur l'applet

Nous allons rajouter un texte sur notre applet. En Java, un message textuel est appelé un label. Pour l'ajouter :

- \* Cliquez sur l'onglet **Conception**, situé en dessous du code source : on retrouve notre applet sous forme d'un rectangle gris, vide pour l'instant
- \* Au dessus de l'applet se trouvent différents onglets : *Swing, Conteneurs Swing, DataExpress, etc.* Tout à droite de ces onglets se trouvent 2 petites flèches permettant de faire défiler les onglets. Cliquez sur la flèche droite à plusieurs reprise pour faire défiler tous les onglets vers la gauche, jusqu'au dernier [*Autres*].
- \* Cliquez sur l'onglet **AWT** : cet onglet contient tous les objets [texte, boutons, cases à cocher, etc.] que nous pouvons ajouter sur notre applet. En positionnant le curseur de la souris sur l'icône d'un objet, le nom

- de l'objet apparaît dans une bulle d'aide.
- \* Cliquez sur l'objet **Label**, dont l'icône est une lettre A majuscule.
- \* Cliquez dans le rectangle gris de l'applet : un texte vient d'être rajouté sur l'applet
- \* Pour modifier ce texte, cliquez dans le tableau de droite, sur l'avant dernière ligne, sur le mot **label1** [à droite de **text**]
- \* Effacez label1 et entrez la phrase **Bonjour, je suis une applet JAVA** à la place
- \* Cliquez dans le rectangle gris de l'applet pour valider : le texte est aussitôt modifié sur l'applet
- \* Exécutez le projet pour voir le résultat [menu **Exécuter** + **Exécuter le projet**]

Le but d'une applet JAVA est d'être exécuté sur une page web, dans un navigateur Internet. Voyons comment afficher notre applet dans une page web :

- \* Allez dans le menu **Projet** puis cliquez sur **Construire le projet** : JBuilder crée alors automatiquement une page web [au format HTML] sur le disque dur, contenant l'applet
- \* Pour aller voir cette page web, cliquez sur l'icône **Répertoire perso** située sur le bureau de Linux
- \* Ouvrez le dossier **jbproject**
- \* Ouvrez le dossier portant le nom de votre applet [**mon\_applet**]
- \* Ouvrez le dossier **classes**
- \* Puis cliquez sur la page web pour l'afficher [**ma\_classe.html**] : on peut apercevoir notre applet, dans le navigateur Internet, comme le verrait un visiteur de notre site Internet.

Remarque : pour modifier rapidement la page web avec le navigateur Mozilla, on pourra cliquer sur **Éditer la page** dans le menu **Fichier**.

#### **IV - Ajout d'un bouton sur l'applet**

Refermez **tous les navigateurs Internet ouverts** et retourner sous JBuilder 9.

Pour ajouter un bouton sur l'applet :

- \* Cliquez sur l'objet **Button** [dont l'icône est un bouton OK] dans l'onglet AWT
- \* Cliquez sur le fond gris de l'applet : un bouton nommé **button1** a été rajouté à l'applet
- \* Pour changer le texte afficher sur le bouton, modifier le champ **label** dans le tableau de droite : entrez **coucou** comme nouveau label, puis cliquez sur le fond gris pour valider

Exécutez le projet pour voir le résultat dans JBuilder.

Construisez le projet puis allez voir le résultat dans la page web.

#### **V - Modifier l'action du bouton**

Nous allons programmer le bouton **coucou** afin qu'il affiche **Vous venez de cliquer sur le bouton !** à la place du texte **Bonjour, ...**. Pour cela, nous allons faire 3 modifications dans le code source en Java. Cliquez sur l'onglet « Source », situé dans JBuilder 9 en dessous de l'Applet, à côté de l'onglet « Conception », puis :

- \* A la ligne 16, remplacez la ligne `public class ma_classe extends Applet {` par la ligne suivante :  
`public class ma_classe extends Applet implements ActionListener {`
- \* Sur les lignes 36 et 37 se trouvent 2 accolades fermantes. Sautez une ligne entre les 2 accolades, puis ajoutez la ligne ci-dessous sur la ligne 38 :  

n° de ligne ↓	
36	}
37	<code>button1.addActionListener(this) ;</code>
38	}
- \* Tout à la fin du programme, juste **avant la dernière accolade** fermante, rajoutez les 4 lignes suivantes :  

```
public void actionPerformed(ActionEvent e) {
Object source = e.getSource();
if (source == button1) {
label1.setText("Vous venez de cliquer sur le bouton"); } }
```

En suivant le même principe, on pourra rajouter 3 labels [label2, label3, et label4] sur l'applet, et 3 boutons [button2, button3, et button4] afin que chacun des 4 boutons modifie un label particulier parmi les 4.