Utilisation de NetBeans

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

Extrait du site 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 coté '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 complèter.

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

ous ave	e <mark>z cliqué s</mark> u	r		
A	В	Effacer la zone de tex	xte Effacer le ch	amp de texte

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



Votre rôle consistera à complèter 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

L'interface graphique de NetBeans en mode Design	page 3
L'interface graphique de NetBeans en mode Source	page 4

Partie 2: Informations utiles pour tous les exercices

Démarrer NetBeans	page 5
Ouvrir un projet existant	page 5
Résoudre le problème d'accès à la librairie	page 6
Créer ou modifier la procédure évènementielle associée à un bouton	page 8
Exécuter l'application	page 9
Arrêter le programme	page 9
Sauvegarder	page 10

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

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

ojects II Services	Start Page # AddBon.java # es.java # AddBon.java # Ad	Palette # Swing Containers Parel Tabbed Pane Solt Pane Solt Pane Tool Bar Desktop Pane Internal Frame Layered Pane Palette des composant Combo Box Ust Text Field Text Area Ult Sorol Bar Solder Progress Bar Promatted Field Password Field
enêtre Projects	Fenêtre d'édition (mode Design)	ButonB (JButton) - Properties # Hierarchy Properties Bedrag Events Code Properties action cone> Code background [1240 240240] font Tahoma Li Man forcerrund 10.0.01 Cone Code Code Code Code Code Code Code Cod
Bouton8 [Fleuton] Bouton8 [Fleuton] BoutoneFfacerCT (JButton] Hell BoutoneFfacerCT (JButton] Boll jScrolPane1 [ScrolPane] Cenêtre Navigato	Putput #	lext 9

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

- la palette des composants permet d'y ajouter de nouveaux composants.
- la fenêtre properties 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):

(Pa	Rechercher dans :	PrepaCours	•	D 🗗	
Documents r		ko-Premieres-Notions ko-ProgBD	*	Project Name: AfficherEffacer	
Bureau	₽~}5 ₽~}5 ₽~}5	«o-ProgObjet1 «o-ProgObjet2 «o-Sous-Programme «o-StructControle		Open Required Projects:	
Documents	⊕) E ⊕) E ⊕) Jz ⊕ () Jz	ko-Tableaux ko-TypesStructures ava-Exo-Premieres-Notion AfficherEffacer rs	E		
Ordinateur	E Gestic E HTFile	ilairesPHP InProjet Is	*		
<u>.</u>	Nom du fichier :	F:\COURS-INFO\PrepaCours\Cours\Exo\Java-	Exo-Premiere	s-Notion\AfficherEffacer Open F	roject
Réseau	Type de fichier :	Project Folder		Ann	uler

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 ...

"JavaETBib jar" file/folder could not be found (in AfficherEffacer)	Recolve
escription:	
roblem: The project uses the file/folder called "JavaETBib.jar", but this file/folde not found.	er was
olution: Click Resolve and locate the missing file/folder.	

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 refédinissant 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**:

Rechercher dans	s: 📙 dist	- 🔊 🗊 🗈	🤊 🛄 -
C.	JavaETBib.ja	r	
Documents r			
Documents r	Nom du fichier :	JavaETBib.jar	Ouvrir

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

roject Problems:	
"JavaETBib.jar" file/folder could not be found (in AfficherEffacer)	Resolve
escription:	
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 évènementielle associée à un bouton

Méthode 1: Vous êtes en mode Design

Faites un double clic sur le bouton. Deux possibilités:

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



 la procédure évènementielle 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 évènementielle 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 évènementielle du bouton BoutonB :



Exécuter l'application

Cliquez sur l'icone suivant dans la barre d'outils:

	~	
V		
	V	

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

🕜 Run Pr	roject
1	Project Exemple does not have a main class set.
-	Select the main class:
	Exemple
	OK Cancel

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

			lci	
(Lire Afficher			
	Nom	eric	Lire	
Ou là	Age	53	Afficher	
Output-LireAfficher	Taille (m)	1.72	Effacer	
>>				

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 :

Files Services	ר Start Pane צ או איז
Steps	Choose Project
1. Choose Project 2	Q. Filter: Projects: Categories: Projects: Java Java Application JavaFX Java Class Library Java Web Java Project with Existing Sources
	Java EE J. HTML5 Java ME Embedded Java Card Maven T
rig	Creates a new Java SE application in a standard IDE project. You can also generate a main class in the project. Standard projects use an IDE-generated Ant build script to build, run, and debug your project.

la fenêtre suivante s'affiche alors:

Steps	Name and Locat	tion	
 Choose Project Name and Location 	Project Name:	JavaApplication7	
	Project Location:	F:\COURS-INFO\PrepaCours\Cours\Exemple	Browse
	Project Folder:	PrepaCours\Cours\Exemple\Java-Exemple-Pr	
	Use Dedicated Libraries Folde	Folder for Storing Libraries	Browse
		Different users and projects can share the same compilation libraries (see Help for details).	
	🔽 Create Main C	lass javaapplication7.JavaApplication7	

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 coté 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 apparait dans la fenêtre de **Projects** de NetBeans.

Ceci, par exemple:

teps	Name and Locat	ion	
Choose Project Name and Location	Project Name:	Exemple	
	Project Location:	F:\COURS-INFO\PrepaCours\Cours\Exemple\Java-Exemple-Premieres-Notions	Browse
	Project Folder:	F:\COURS-INFO\PrepaCours\Cours\Exemple\Java-Exemple-Premieres-Notions\Exemple	
	Crasta Maia C	here annuale Example	
	Create Main C	lass exemple.Exemple	

va nous créer un repertoire 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**:

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help Image: Start Page Projects Start Page Image:	🇊 Exemple -	NetBeans IDE 8.0.1		
Projects 28 Services Start Page 28 Projects 28 Services Folder Projects 28 Services Folder Projects 28 Services Folder Projects 20 Start Page 28 Folder Projects 28 Folde	File Edit Vie	ew Navigate Source Ref.	tor Run Debug Profile Team Tools Window Help :default.config> 💽 🌇 🏷 * 💷 * 🔇	D -
New ▶ □ Folder Build □ JFrame Form Clean and Build □ Java Package Generate Javadoc □ Java Interface	Projects %	Services St	tPage %	
Build Clean and Build Clean Generate Javadoc Build Clean Generate Javadoc Build Clean Generate Javadoc Build Clean Generate Javadoc Build Clean Generate Javadoc Build Clean Generate Javadoc Build Clean Generate Javadoc Build Clean		New	Folder	
	±- b	Build Clean and Build Clean Generate Javadoc	JFrame Form er Java Package er Java Class file Empty Java File java Interface	

la fenêtre suivante apparait alors:

🕽 New JFrame Form	
Steps	Name and Location
 Choose File Type Name and Location 	Class Name: New JFrame
	Project: Exemple
	Location: Source Packages
	Package:
	Created File: F:\COURS-INFO\PrepaCours\Cours\Exemple\Java-Exemple-Premieres-Notions\Exemple\src\NewJFrame.java
	A Warning: It is highly recommended that you do not place Java classes in the default package.
	< Back Next > Finish Cancel Help

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:

iteps	Name and L	ocation
. Choose File Type . Name and Location	Class Name:	Exemple
	Project:	Exemple
	Location:	Source Packages
	Package:	
	Created File:	$\label{eq:F:COURS-INFO} F: \COURS-INFO\PrepaCours\Cours\Exemple\Java-Exemple-Premieres-Notions\Exemple\src\Exemple.java-Exemple\Premieres-Notions\Exemple\Src\Exemple\Java-Exemple\Premieres-Notions\Exemple\Src\Exemple\Bar{Src}\Exemple\Bar{Src}\Exemple\Bar{Src}\Ba$
	🔔 Warning:	It is highly recommended that you do not place Java classes in the default package.

va nous créer un fichier **Exemple.java** dans le répertoire **src** (généré 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 <u>palette</u> <u>de composants</u>, 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):

Palette %			
Swing Container	ers		
Panel	Tabbed Pane	Split Pane	
Scroll Pane	🔟 Tool Bar	🔁 Desktop Pane	
Internal Frame	🔳 Layered Pane		
Swing Controls			
label Label	OK Button	In Toggle Button	
Check Box	e- Radio Button	[®] = Button Group	
💽 Combo Box	Eist	Text Field	
tx Text Area	I Scroll Bar	D Slider	
Progress Bar	Formatted Field	Password Field	
主 Spinner	Separator	T Text Pane	
🔯 Editor Pane	ੀ Tree	Table	
+ Swing Menus			
[JFrame] - Proper	ties 🛚 Hierarchy		

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 <u>fenêtre **Properties**</u>.

jLabel1 [JLabel] - Proper	ties 🛚 Hierarchy	
Properties Binding	Events Code	
font	Tahoma 11 Plain	
foreground	[0,0,0]	
horizontalAlignment	LEADING	▼ 🗉
icon		▼
labelFor	<none></none>	▼
text	Article	
toolTipText		
verticalAlignment	CENTER	▼ ▼

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:

	.	
Artic	le <mark>iTextField</mark> 2	a Baaraa
	-	

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

Properties Binding Events Code columns 0 document <default> font Tahoma 11 Plain foreground Image: [0,0,0] horizontalAlignment LEADING Image: [0,0,0]</default>	
columns 0 document <default> font Tahoma 11 Plain foreground III [0,0,0] horizontalAlignment LEADING III.</default>	
document <default> ▼ font Tahoma 11 Plain foreground ■ [0,0,0] horizontalAlignment LEADING ▼</default>	
font Tahoma 11 Plain foreground ■ [0,0,0] horizontalAlignment LEADING ▼	-
foreground ■ [0,0,0] … horizontalAlignment LEADING ▼…	=
horizontalAlignment LEADING 💌 🛄	
text	
toolTipText	
Other Properties	-

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 :

	Edit Text	
	Change Variable Name	
	Bind	+
	Events	•
	Align	Þ
	Anchor	+
	Auto Resizing	•
	Same Size	•
TextField2 [JTextField] - Na	Set to Default Size	
Form Exemple	Enclose In	+
····· 🔄 Other Components]···· 🛅 [JFrame]	Edit Layout Space	
label jLabel1 [JLabel]	Design Parent	+
- Frend Icide Dirend	Move Up	

et :

Rename		<u> </u>
New Name:	CT_Artide	

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:

[JFrame] - Properties 88	Hierarchy	
Properties Binding	Events Code	
Properties		*
defaultCloseOperation	EXIT_ON_CLOSE	: 🔹 🖬 🗐
title	Exemple	
Other Properties		
alwaysOnTop	1000	
alwaysOnTopSupport Other	properties of the Jav	vaBeans component
autoRequestFocus		
background	[240,240,24	01 🔻

donnera ceci à l'exécution:

🕌 Exemple		X
Article		

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 :

Ibraries Libraries Folder: .\ib \nblibraries.properties Build Compiling Compile Processor Run Tests Run Tests	Browse			
Compile Processor Run Compile Tests Run Tests				
• Fackaging				
Openover Compile-time Libraries:	Compile-time Libraries:			
	Add Project			
Application	Add Library			
Web Start O License Headers	Add JAR/Folder			
Formatting Hints	Edit			

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**:

echercher dans :	r dans : 🕕 dist			🥩 📂 🖽-		
9	📕 JavaETBib.ja	r -	Refe I	rence as Relative Path:		
Documents r				//ETBib/JavaETBib/dist/JavaETBib		
			OI	Path from Variable:		
-				<no suitable="" variable=""></no>		
Bureau			0	Copy to Libraries Folder:		
87-11				:xemple-Premieres-Notions\Exemple\ib		
			0	Absolute Path:		
Documents				F:\COURS-INFO\PrepaCours\Cours\ET		
Ordinateur						
<u>.</u>	Nom du fichier :	JavaETBib.jar		Ouvrir		
Réseau	Type de fichier :	Classpath Entry (folder, 7IP or 1AR file)		Annuler		

Cliquez ensuite sur Ouvrir, puis OK.