



Ressources **informatiques**

# **Apprendre la Programmation Orientée Objet avec le langage **Java****

**(avec exercices pratiques et corrigés)**

**Luc GERVAIS**

Téléchargement  
[www.editions-eni.fr](http://www.editions-eni.fr)



Les éléments à télécharger sont disponibles à l'adresse suivante :

<http://www.editions-eni.fr>

Saisissez la référence ENI de l'ouvrage **RIJAPOO** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

## Avant-propos

### Chapitre 1

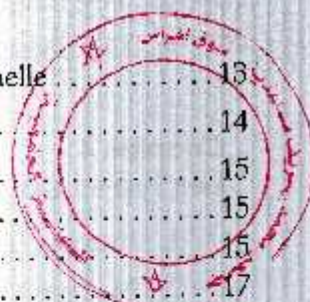
#### Introduction à la POO

- |                                     |    |
|-------------------------------------|----|
| 1. Histoire de la POO .....         | 9  |
| 2. Historique du langage Java ..... | 12 |

### Chapitre 2

#### La conception orientée objet

- |  |    |
|--|----|
| 1. Approche procédurale et décomposition fonctionnelle ..... | 13 |
| 2. La transition vers l'approche objet .....                 | 14 |
| 3. Les caractéristiques de la POO .....                      | 15 |
| 3.1 L'objet, la classe et la référence .....                 | 15 |
| 3.1.1 L'objet .....  | 15 |
| 3.1.2 La classe .....  | 17 |
| 3.1.3 La référence .....                                     | 18 |
| 3.2 L'encapsulation .....                                    | 18 |
| 3.3 L'héritage .....   | 19 |
| 3.4 Le polymorphisme .....                                   | 21 |
| 3.5 L'abstraction .....                                      | 22 |
| 4. Le développement objet .....                              | 23 |
| 4.1 Cahier des charges du logiciel .....                     | 23 |



4.2	Modélisation et représentation UML .....	23
4.2.1	Diagrammes de cas d'utilisation .....	26
4.2.2	Diagrammes de classes .....	27
4.2.3	Les énumérations .....	33
4.2.4	Diagrammes de séquences .....	34
4.3	Codage, intégration et mise en production .....	36

**Chapitre 3****Introduction à la plate-forme Java**

1.	Introduction .....	39
2.	Environnement d'exécution .....	41
3.	Une librairie très complète .....	42
4.	Des outils de développement performants .....	43
5.	Téléchargement puis installation de NetBeans .....	43
6.	L'incontournable Hello World .....	46

**Chapitre 4****Les types en Java**

1.	Introduction .....	55
2.	Les types "primitifs" .....	56
3.	Les types "référence" .....	58
4.	Pour nous aider... .....	60
5.	La superclasse java.lang.Object .....	64
5.1	equals .....	64
5.2	hashCode .....	67
5.3	toString .....	69
5.4	finalize .....	71
5.5	getClass, .class et l'opérateur instanceof .....	71
5.6	clone .....	73

5.7	notify, notifyAll et wait	78
6.	La classe java.lang.String	78
7.	Exercice corrigé	82
7.1	Énoncé	82
7.2	Correction	83

**Chapitre 5**  
**Création de classes**

1.	Introduction	87
2.	Package	87
3.	Déclaration d'une classe	94
3.1	Accessibilité des membres	96
3.2	Attributs	96
3.2.1	Attributs constants	97
3.3	Accesseurs	99
3.4	Constructeurs	104
3.4.1	Étapes de la construction d'un objet	104
3.4.2	Surcharge de constructeurs	106
3.4.3	Chaînage de constructeurs	107
3.4.4	L'initialiseur static	110
3.4.5	L'initialiseur dynamique	111
3.4.6	Les constructeurs de type private	113
3.4.7	Le "builder pattern"	115
3.5	Destructeurs	118
3.6	Le mot clé this	123
3.7	Méthodes	125
3.7.1	Déclaration	126
3.7.2	Passages de paramètres par valeur	129
3.8	Surcharge des méthodes	136

3.9	Mécanisme des exceptions	138
3.9.1	Présentation	138
3.9.2	Principe de fonctionnement des exceptions	140
3.9.3	Prise en charge de plusieurs exceptions	150
3.10	Exercice	151
3.10.1	Énoncé	151
3.10.2	Conseils	152
3.10.3	Correction	153
4.	Les interfaces	156
4.1	Introduction	156
4.2	Le contrat	156
4.3	Déclaration d'une interface	157
4.4	Implémentation	159
4.5	NetBeans et les interfaces	161
4.6	Représentation UML d'une interface	163
4.7	Interfaces et polymorphisme	163
4.8	Exercice	164
4.8.1	Énoncé	164
4.8.2	Conseils	165
4.8.3	Correction	168
4.9	Les interfaces de la machine virtuelle Java	172
5.	Association, composition et agrégation	175
5.1	Les tableaux	182
5.2	Les collections	189
5.2.1	ArrayList<E> et LinkedList<E>	195
5.2.2	Queue<T> et Stack<T>	196
5.2.3	HashMap<K, V>	196
5.2.4	Les "itérateurs"	197
5.3	Exercice	199
5.3.1	Énoncé	199
5.3.2	Correction	200
6.	Les classes imbriquées	202
7.	Quelques différences avec le C#	206

## Chapitre 6

### Héritage et polymorphisme

1. Comprendre l'héritage .....	207
2. Codage de la classe de base (superclasse) et de son héritière (sous-classe) .....	208
2.1 Interdire l'héritage .....	208
2.2 Définir les membres héritables .....	209
2.3 Syntaxe de l'héritage .....	210
2.4 Exploitation d'une classe héritée .....	210
3. Communication entre classe de base et classe héritière .....	212
3.1 Les constructeurs .....	212
3.2 Accès aux membres de la classe de base depuis l'héritier .....	216
3.3 Méthodes virtuelles .....	219
3.4 Méthodes de type "final" .....	224
4. Exercice .....	226
4.1 Énoncé .....	226
4.2 Corrigé .....	227
5. Les classes abstraites .....	230
6. Le polymorphisme .....	231
6.1 Comprendre le polymorphisme .....	231
6.2 Exploitation du polymorphisme .....	232
6.3 Les opérateurs instanceof et () .....	233

## Chapitre 7

### Communication entre objets

1. L'événementiel : être à l'écoute .....	235
2. Le pattern Observateur .....	236
2.1 Généralités .....	236
2.2 Implémentation en langage Java .....	237
2.3 Les "listeners" .....	243
2.4 Utilisation d'un listener dans une application graphique .....	244

3. Exercices .....	251
3.1 Exercice 1 : Énoncé .....	251
3.2 Exercice 1 : Correction .....	252
3.3 Exercice 2 : Énoncé .....	255
3.4 Exercice 2 : Correction .....	255
4. Appels synchrones, appels asynchrones .....	258

## Chapitre 8

### Le multithreading

1. Introduction .....	261
2. Comprendre le multithreading .....	261
3. Multithreading et Java .....	264
4. Implémentation des threads en Java .....	265
4.1 Étendre la classe Thread .....	265
4.2 Implémenter l'interface Runnable .....	268
4.3 S'endormir et S'attendre .....	270
4.4 Abandon depuis le thread primaire .....	272
4.5 Threads et classes anonymes .....	275
4.5.1 Avec l'interface Runnable .....	275
4.5.2 Avec la classe Thread .....	276
4.5.3 Accès aux variables et aux données membres simplifié .....	277
5. Synchronisation entre threads .....	281
5.1 Nécessité de la synchronisation .....	281
5.2 Les méthodes "synchronized" .....	283
5.3 Les traitements "synchronized" .....	285
5.4 La classe Semaphore .....	286
6. Communication interthread .....	289
6.1 La méthode join .....	289
6.2 Les objets de synchronisation .....	291

7. Exercice .....	301
7.1 Énoncé .....	301
7.2 Correction .....	302

Index .....	309
-------------	-----