

Hugues Bersini

Avec la collaboration d'Ivan Wellesz

L'orienté objet

**Cours et exercices
en UML, Java, C# et C++**

EYROLLES

2-005-628-1

L'orienté objet

**Cours et exercices
en UML, Java, C# et C++**

Hugues Bersini

Avec la collaboration d'Ivan Wellesz

EYROLLES



Table des matières

Avant-propos	1
L'orientation objet en deux mots	2
Objectifs de l'ouvrage	3
Plan de l'ouvrage	3
À qui s'adresse ce livre ?	4

CHAPITRE 1

Principes de base : quel objet pour l'informatique ?	5
Le trio <entité, attribut, valeur>	6
Stockage des objets en mémoire	7
L'objet dans sa version passive	11
L'objet dans sa version active	13
Introduction à la notion de classe	15
Des objets en interaction	16
Des objets soumis à une hiérarchie	19
Polymorphisme	21
Héritage bien reçu	22
Exercices	22

CHAPITRE 2

Un objet sans classe... n'a pas de classe	23
Constitution d'une classe d'objets	24
La classe comme module fonctionnel	27
La classe comme garante de son bon usage	28
La classe comme module opérationnel	29
Un premier petit programme complet dans les trois langages	31
La classe et la logistique de développement	36
Exercices	38

CHAPITRE 3

Du faire savoir au savoir-faire... ..du procédural à l'OO 45

Objectif objet : les aventures de l'OO 45

Mise en pratique 46

Analyse 48

Conception 49

Impacts de l'orientation objet 49

CHAPITRE 4

Ici Londres : les objets parlent aux objets 51

Envois de messages 52

Association de classes 53

Dépendance de classes 54

Réaction en chaîne de messages 55

Exercices 56

CHAPITRE 5

Collaboration entre classes 57

Pour en finir avec la lutte des classes 58

La compilation Java : effet domino 59

En C# et en C++ 61

De l'association unidirectionnelle à l'association bidirectionnelle 62

Auto-association 63

Assemblage 65

Exercices 66

CHAPITRE 6

Méthodes ou messages ? 67

Passage d'arguments prédéfinis dans les messages 68

Passage d'argument objet dans les messages 73

Une méthode est-elle d'office un message ? 76

La mondialisation des messages 78

Exercices 79

CHAPITRE 7

L'encapsulation des attributs 83

Accès aux attributs d'un objet 84

Encapsulation : pourquoi faire ? 88

Exercices 92

CHAPITRE 8

Les classes et leur jardin secret	93
Encapsulation des méthodes	94
Les niveaux intermédiaires d'encapsulation	95
Afin d'éviter l'effet papillon	99
Exercices	101

CHAPITRE 9

Vie et mort des objets	103
Question de mémoire	104
C++ : le programmeur est le seul maître à bord	112
En Java et C# : la chasse au gaspi	115
Exercices	118

CHAPITRE 10

UML	123
Diagrammes UML	125
Représentation graphique standardisée	125
Diagrammes de classe et diagrammes de séquence	126
Diagramme de classe	127
Les bienfaits d'UML	147
Diagramme de séquence	150
Exercices	154

CHAPITRE 11

Héritage	159
Comment regrouper les classes dans des superclasses	160
Héritage des attributs	161
Héritage ou composition ?	165
Économiser en rajoutant des classes ?	166
Héritage des méthodes	167
La recherche des méthodes dans la hiérarchie	174
Encapsulation protected	175
Héritage public en C++	176
Le multihéritage	177
Exercices	186

CHAPITRE 12

Redéfinition des méthodes	178
La redéfinition des méthodes	178
Beaucoup de verbiage mais peu d'actes véritables	188
Un match de football polymorphique	192
Exercices	194

CHAPITRE 13

Abstraite, cette classe est sans objet	228
De Canaletto à Turner	230
Des classes sans objet	230
Du principe de l'abstraction à l'abstraction syntaxique	230
Un petit supplément de polymorphisme	237
Exercices	241

CHAPITRE 14

Clonage, comparaison et assignation d'objets	251
Introduction à la classe Object	252
Décortiquons la classe Object	253
Test d'égalité de deux objets	255
Le clonage d'objets	259
Égalité, clonage et affectation d'objets en C++	263
En C#, un peu des deux	273
Exercices	278

CHAPITRE 15

Interfaces	281
Interfaces : favoriser la décomposition et la stabilité	283
Java et C# : interface via l'héritage	283
Les trois raisons d'être des interfaces	284
En C++ : fichiers .h et fichiers .cpp	295
Interfaces : du local à Internet	298
Exercices	299

CHAPITRE 16

Distribution gratuite d'objets : pour services rendus sur le réseau	303
Objets distribués sur le réseau : pourquoi ?	304
RMI (Remote Method Invocation)	306
Corba (Common Object Request Broker Architecture)	311
Rajoutons un peu de flexibilité à tout cela	318
Les services Web sur .Net	323
Exercices	330

CHAPITRE 17

Multithreading	333
Informatique séquentielle	335
Multithreading	337
Implémentation en Java	338
Implémentation en C#	340
L'impact du multithreading sur les diagrammes de séquence UML	342
Du multithreading aux applications distribuées	343
Des threads équirépartis	343
Synchroniser les threads	345
Exercices	350

CHAPITRE 18

Programmation événementielle	353
Des objets qui s'observent	354
En Java	356
En C# :	359
Exercices	363

CHAPITRE 19

Persistance d'objets	365
Sauvegarder l'état entre deux exécutions	366
Simple sauvegarde sur fichier	367
Sauvegarder les objets sans les dénaturer : la sérialisation	372
Les bases de données relationnelles	375
Les bases de données relationnelles-OO	387
Les bases de données OO	390
Exercices	392

CHAPITRE 20

Et si on faisait un petit flipper	395
---	-----

CHAPITRE 21

Les graphes	403
Le monde regorge de réseaux	404
Tout d'abord : juste un ensemble d'objets	405
Liste liée	407
La généricité en C++	414
La généricité en Java	417
Passons aux graphes	418
Exercices	423

CHAPITRE 22

Petite chimie OO amusante	427
Pourquoi de la chimie OO ?	428
Les diagrammes de classe du réacteur chimique	429
Quelques résultats du simulateur	443
Index	447