



David Barnes, Michael Kölling

# Conception Objet en Java avec BlueJ

## Une approche interactive

2<sup>e</sup> édition



PEARSON  
Education

2-005-665-1



2-005-665-1

**David J. Barnes  
et Michael Kölling**

# Conception objet en Java avec BlueJ

*Une approche interactive*



**2<sup>e</sup> édition**

Coordination et validation technique :  
Patrice Moreaux, université de Reims

Traduction : Nathalie Le Guillou de Penanros,  
Chantal Kolb, Michaël Krajecki, Patrice Moreaux

**PEARSON**  
Education

# Table des matières

<b>Avant-propos</b> .....	XVII
<b>Préface pour les enseignants</b> .....	XIX
Java .....	XIX
BlueJ .....	XIX
De vrais objets d'abord .....	XX
Une approche itérative .....	XXI
Une étude non exhaustive du langage .....	XXII
Une approche par projet .....	XXII
Une progression par concept plutôt que par éléments du langage .....	XXIII
Contenu des chapitres .....	XXIV
Groupe de discussion .....	XXV
Compléments .....	XXVI
<b>Note des traducteurs</b> .....	XXVII
<b>Liste des projets</b> .....	XXIX
<b>Remerciements</b> .....	XXXIII
<b>PARTIE I. FONDEMENTS ET ORIENTATION OBJET</b> .....	1
<b>1. Objets et classes</b> .....	3
Objets et classes .....	3
Création d'objets .....	4
Appel des méthodes .....	6
Paramètres .....	7
Types de données .....	8
Instances multiples .....	9
État .....	10
Définition d'un objet .....	11

Interaction entre les objets .....	12
Code source .....	14
Un autre exemple .....	16
Valeurs de retour .....	16
Objets agissant comme paramètres .....	17
Résumé .....	18
Résumé des concepts .....	19
<b>2. Comprendre les définitions de classes</b> .....	21
Une billetterie automatique .....	22
Étudier le comportement d'une billetterie automatique .....	22
Étude d'une définition de classe .....	23
Champs, constructeurs et méthodes .....	25
Champs .....	27
Constructeurs .....	30
Transfert de données par le biais des paramètres .....	31
Affectation .....	33
Méthodes d'accès .....	34
Méthodes de modification .....	37
Impression à partir de méthodes .....	39
Résumé à propos de la billetterie élémentaire .....	42
Réflexions sur la conception de la billetterie .....	43
Les choix : l'instruction conditionnelle .....	43
Autre exemple d'instruction conditionnelle .....	47
Variables locales .....	48
Champs, paramètres et variables locales .....	50
Résumé à propos de la billetterie améliorée .....	51
Un exemple familier .....	52
Résumé .....	55
Résumé des concepts .....	55
<b>3. Interactions entre objets</b> .....	59
L'exemple de l'horloge .....	60
Abstraction et modularité .....	60

L'abstraction dans les logiciels .....	62
La modularité dans l'exemple de l'horloge .....	62
Implantation de l'affichage de l'horloge .....	63
Diagrammes de classes et diagrammes d'objets .....	64
Types primitifs et types d'objets .....	66
Le code source de <i>ClockDisplay</i> .....	68
La classe <i>NumberDisplay</i> .....	68
La concaténation .....	71
L'opérateur modulo .....	72
La classe <i>ClockDisplay</i> .....	73
Objets créateurs d'autres objets .....	76
Constructeurs multiples .....	78
Appels de méthodes .....	78
Appels de méthodes internes .....	78
Appels de méthodes externes .....	79
Résumé à propos de l'affichage de l'horloge .....	81
Autre exemple d'interaction entre les objets .....	81
Exemple du système de messagerie .....	83
Le mot clé <i>this</i> .....	84
Le débogueur .....	85
Définition de points d'arrêt .....	86
Pas à pas .....	88
Avancer dans les méthodes .....	89
Nouvelle expérience de l'appel de méthodes .....	90
Résumé .....	91
Résumé des concepts .....	91
<b>4. Groupement d'objets</b> .....	93
Groupement d'objets dans des collections de taille variable .....	93
Un agenda électronique .....	94
Un premier aperçu des classes de bibliothèques .....	95
Exemple d'utilisation d'une bibliothèque .....	95
Structures d'objets et collections .....	97
Numérotation au sein des collections .....	99

Suppression d'un élément d'une collection .....	100
Traitement de l'ensemble d'une collection .....	101
La boucle <i>while</i> .....	102
Parcourir une collection .....	105
Accès à l'index ou itérateurs .....	106
Résumé à propos de l'exemple de l'agenda .....	106
Autre exemple : un système d'enchères .....	108
La classe <i>Lot</i> .....	108
La classe <i>Auction</i> .....	109
Le transtypage .....	112
Les objets anonymes .....	112
Les collections .....	114
Résumé des collections variables .....	116
Collections de taille fixe .....	116
L'analyseur de fichier journal .....	117
Déclarer des variables de tableau .....	119
Créer des objets tableau .....	120
Utiliser des objets tableau .....	121
Analyser le fichier journal .....	122
La boucle <i>for</i> .....	123
Résumé .....	127
Résumé des concepts .....	128
<b>5. Comportements plus complexes</b> .....	131
Documentation pour les classes de bibliothèques .....	132
Le système TechSupport .....	133
Explorer le système TechSupport .....	133
Lire le code .....	135
Lecture de la documentation des classes .....	139
Interfaces ou implantation .....	141
Utiliser des méthodes de classe de bibliothèque .....	142
Vérifier l'égalité des chaînes .....	144
Ajout d'un comportement aléatoire .....	145
La classe <i>Random</i> .....	145
Nombres aléatoires dans un intervalle donné .....	147
Générer des réponses aléatoires .....	148

Paquetages et import .....	150
Des tableaux associatifs pour les associations .....	152
Le concept d'association .....	152
Utiliser un <i>HashMap</i> .....	153
Utiliser une association pour le système TechSupport .....	154
Les ensembles .....	156
Transformation de chaînes en suite de mots .....	157
Fin du système TechSupport .....	159
Rédaction d'une documentation de classe .....	161
<i>Javadoc</i> dans BlueJ .....	162
Éléments de la documentation de classe .....	162
Public ou privé .....	163
Masquer les informations .....	165
Méthodes privées et champs publics .....	166
Découvrir les classes à partir de leurs interfaces .....	167
Variables de classes et constantes .....	170
Le mot clé <i>static</i> .....	170
Constantes .....	172
Résumé .....	172
Résumé des concepts .....	173
<b>6. Des objets bien conçus .....</b>	<b>175</b>
Tester et déboguer .....	177
Tests unitaires dans BlueJ .....	177
Utilisation des Inspecteurs .....	181
Test positif versus test négatif .....	183
Automatisation des tests .....	184
Tests de non-régression .....	184
Vérification automatique des résultats de tests .....	187
Enregistrer un test .....	190
Configuration de test .....	192
Modularité et interfaces .....	194
Un scénario de débogage .....	196
Commentaires et style de programmation .....	196

Traces d'exécution manuelles .....	198
Une trace d'exécution de haut niveau .....	198
Vérification de l'état à l'aide d'une trace d'exécution .....	201
Traces d'exécution orales .....	203
Instructions d'affichage .....	204
Activation des informations de débogage .....	206
Choix d'une stratégie de test .....	207
Débogueurs .....	208
Mise en pratique des techniques .....	209
Résumé .....	209
Résumé des concepts .....	209
<b>7. Conception des classes .....</b>	<b>211</b>
Introduction .....	212
Exemple : le jeu du monde de Zuul .....	214
Introduction au couplage et à la cohésion .....	215
Duplication du code .....	217
Création d'extensions .....	220
La tâche .....	221
Découverte du code source pertinent .....	221
Couplage .....	223
Utilisation de l'encapsulation pour réduire le couplage .....	223
Conception dirigée par les responsabilités .....	228
Responsabilités et couplage .....	228
Localisation des modifications .....	231
Couplage implicite .....	232
Prévoir les évolutions futures .....	236
Cohésion .....	237
Cohésion des méthodes .....	237
Cohésion des classes .....	238
Cohésion et lisibilité .....	239
Cohésion et réutilisation .....	239
Réingénierie .....	241
Réingénierie et test .....	241
Un exemple de réingénierie .....	242



Conception : mode d'emploi .....	246
Exécution sans BlueJ .....	247
Méthodes de classe .....	247
La méthode principale .....	248
Restrictions des méthodes de classe .....	249
Résumé .....	249
Résumé des concepts .....	250
<b>PARTIE II. STRUCTURE DES APPLICATIONS</b> .....	<b>253</b>
<b>8. Améliorer la structuration avec l'héritage</b> .....	<b>255</b>
L'exemple DoME .....	256
Classes et objets DoME .....	257
Code source DoME .....	259
Étude de l'application DoME .....	265
Utilisation de l'héritage .....	266
Hiérarchies d'héritage .....	268
Héritage en Java .....	269
Héritage et droits d'accès .....	270
Héritage et initialisation .....	271
DoME : ajout d'autres types d'éléments .....	273
Avantages de l'héritage (pour l'instant) .....	275
Sous-typage .....	276
Sous-typage et sous-classes .....	277
Sous-typage et affectation .....	278
Sous-typage et passage de paramètre .....	280
Variables polymorphes .....	280
La classe <i>Object</i> .....	281
Collections polymorphes .....	283
Types d'éléments .....	283
Le transtypage revisité .....	283
Classes enveloppes .....	285
Hiérarchie de collection .....	286
Résumé .....	286
Résumé des concepts .....	287

<b>9. Héritage – Approfondissement</b> .....	289
Le problème : la méthode <i>print</i> de DoME .....	289
Type statique et type dynamique .....	291
Appeler <i>print</i> depuis <i>Database</i> .....	292
Surcharge .....	295
Recherche de méthode dynamique .....	296
Appel <i>super</i> dans les méthodes .....	300
Polymorphisme de méthode .....	301
Méthodes de la superclasse <i>Object</i> : <i>toString</i> .....	301
Accès protégé ( <i>protected</i> ) .....	304
Autre exemple d'héritage avec la surcharge .....	306
Résumé .....	310
Résumé des concepts .....	310
<b>10. De nouvelles techniques d'abstraction</b> .....	313
Simulations .....	314
La simulation renards-lapins .....	315
Le projet foxes-and-rabbits (renards et lapins) .....	315
La classe <i>Rabbit</i> .....	317
La classe <i>Fox</i> .....	320
La classe <i>Simulator</i> : initialisation .....	323
La classe <i>Simulator</i> : un pas de simulation .....	326
Améliorer la simulation .....	328
Classes abstraites .....	328
La superclasse <i>Animal</i> .....	329
Méthodes abstraites .....	329
Classes abstraites .....	332
Des méthodes plus abstraites .....	334
Héritage multiple .....	336
Une classe <i>Actor</i> .....	336
Flexibilité par abstraction .....	338
Dessin sélectif .....	338
Acteurs que l'on peut dessiner : héritage multiple .....	339
Interfaces .....	339
Une interface <i>Actor</i> .....	340
Héritage multiple d'interfaces .....	341

Interfaces comme types .....	342
Interfaces comme spécifications .....	343
Un nouvel exemple d'interface .....	344
Classe abstraite ou interface ? .....	345
Résumé sur l'héritage .....	346
Résumé .....	346
Résumé des concepts .....	347
<b>11. Construire des interfaces utilisateur graphiques</b> .....	<b>349</b>
Introduction .....	349
Composants, mise en page et gestion des événements .....	350
AWT et Swing .....	351
L'exemple <i>ImageViewer</i> .....	352
Premiers essais : créer un cadre .....	353
Ajout de composants simples .....	355
Ajout de menus .....	356
Gestion des événements .....	358
Réception centralisée des événements .....	359
Classes internes .....	361
Classes internes anonymes .....	363
ImageViewer 1.0 : la première version complète .....	365
Classes de traitement des images .....	366
Ajout de l'image .....	367
Mise en page .....	369
Conteneurs imbriqués .....	372
Filtres d'images .....	375
Un menu d'aide .....	378
Boîtes de dialogue .....	379
ImageViewer 2.0 : améliorer la structure du programme .....	381
ImageViewer 3.0 : d'autres composants d'interface .....	386
Boutons .....	386
Bordures .....	389
Autres extensions .....	390
Un autre exemple : <i>SoundPlayer</i> .....	393
Résumé .....	396
Résumé des concepts .....	397

<b>12. Gestion des erreurs</b> .....	399
Le projet address-book (carnet d'adresses) .....	400
Programmation défensive .....	404
Interactions client-serveur .....	404
Vérifier les arguments .....	406
Signaler les erreurs de serveur .....	407
Informé l'utilisateur .....	407
Informé l'objet client .....	408
Principes du déclenchement d'exception .....	411
Déclencher une exception .....	411
Classes d'exception .....	412
Effet du déclenchement d'une exception .....	414
Exceptions hors contrôle .....	415
Empêcher la création d'objet .....	416
Gestion des exceptions .....	417
Exceptions sous contrôle : la clause <i>throws</i> .....	417
Capturer une exception : le bloc <i>try</i> .....	418
Déclencher et capturer plusieurs exceptions .....	420
Propager une exception .....	421
La clause <i>finally</i> .....	422
Définir de nouvelles classes d'exception .....	423
Utiliser les assertions .....	424
Vérification de la cohérence interne .....	424
L'instruction <i>assert</i> .....	425
Instructions pour l'utilisation des assertions .....	427
Les assertions et le cadre de test unitaire de BlueJ .....	428
Correction et prévention des erreurs .....	428
Corriger les erreurs .....	429
Éviter les erreurs .....	430
Étude de cas : entrées/sorties de texte .....	431
Lecteurs ( <i>readers</i> ), écrivains ( <i>writers</i> ) et flux ( <i>streams</i> ) .....	432
Le projet address-book-io (E/S et carnet d'adresses) .....	432
Sortie de texte sur fichier avec <i>FileWriter</i> .....	435
Entrée de texte à partir de fichier avec <i>FileReader</i> .....	436
Sérialisation d'objet .....	437
Résumé .....	438
Résumé des concepts .....	438

<b>13. Conception des applications</b> .....	441
Analyse et conception .....	442
La méthode noms/verbes .....	442
Le système de réservation de places de cinéma .....	442
Découvrir les classes .....	443
Utiliser les fiches CRC .....	444
Scénarios .....	445
Conception des classes .....	448
Concevoir les interfaces de classes .....	449
Concevoir l'interface utilisateur .....	450
Documentation .....	450
Coopération .....	451
Prototypage .....	452
Croissance du logiciel .....	453
Le modèle de la chute d'eau .....	453
Développement itératif .....	453
Utiliser les patrons de conception .....	455
Structure d'un patron .....	456
Le décorateur .....	456
Le singleton .....	457
La méthode fabrique .....	458
L'observateur .....	458
Résumé sur les patrons .....	460
Résumé .....	461
Résumé des concepts .....	461
<b>14. Une étude de cas</b> .....	463
Étude de cas .....	463
Description du problème .....	464
Analyse et conception .....	464
Découvrir les classes .....	465
Utiliser les fiches CRC .....	465
Scénarios .....	466
Conception des classes .....	468
Conception des interfaces de classes .....	468
Collaborations .....	469
L'implantation résumé .....	470
Tests .....	474
Encore quelques remarques .....	474

- Développement itératif ..... 474
  - Étapes de développement ..... 475
  - Une première étape ..... 476
  - Tester la première étape ..... 480
  - Une étape ultérieure du développement ..... 480
  - D'autres idées de développement ..... 482
  - Réutilisation ..... 483
- Un autre exemple ..... 483
- Pour aller plus loin ..... 484

**PARTIE III. ANNEXES** ..... 485

**A. Travailler avec un projet BlueJ** ..... 487

- Installer BlueJ ..... 487
- Ouvrir un projet ..... 487
- Le débogueur BlueJ ..... 488
- Contenu du CD ..... 488

**B. Types de données Java** ..... 491

- Types primitifs ..... 491
- Types objet ..... 492
- Classes enveloppes ..... 493

**C. Structures de contrôle Java** ..... 495

- Instructions de sélection ..... 495
  - if-else* ..... 495
  - switch* ..... 496
- Boucles ..... 497
  - while* ..... 497
  - do-while* ..... 498
  - for* ..... 498
- Exceptions ..... 498
- Assertions ..... 499

**D. Opérateurs Java** ..... 501

- Expressions arithmétiques ..... 501
- Expressions booléennes ..... 502

<b>E. Utiliser Java sans BlueJ</b> .....	503
Exécution de programme Java sans BlueJ .....	503
Créer des fichiers <i>jar</i> exécutables .....	505
Développer sans BlueJ .....	505
<b>F. Configurer BlueJ</b> .....	507
Généralités .....	507
Changer la langue de l'interface .....	507
Utiliser la documentation locale de l'API .....	507
Modifier le modèle d'une nouvelle classe .....	508
<b>G. Utiliser le débogueur</b> .....	509
Les points d'arrêt .....	510
Les boutons de contrôle .....	510
Arrêt .....	510
Un pas (même niveau) .....	511
Un pas (développer) .....	511
Continuer .....	511
Terminer .....	511
Affichage des variables .....	511
L'écran Séquence d'appel .....	512
L'écran Fils d'exécution .....	512
<b>H. Outils de test unitaire JUnit</b> .....	513
Activation des fonctionnalités de test unitaire .....	513
Création d'une classe de test .....	513
Création d'une méthode de test .....	513
Assertions de test .....	514
Exécution de tests .....	514
Configurations de test .....	514
<b>I. Javadoc</b> .....	515
Commentaires de la documentation .....	515
La description principale .....	516
La section des balises .....	516
Prise en charge de <i>javadoc</i> par BlueJ .....	517

## XVI ♦ Conception objet en Java avec BlueJ

<b>J. Guide de style des programmes</b> .....	519
Dénomination .....	519
Mise en page .....	520
Documentation .....	521
Restrictions à l'utilisation du langage .....	521
Idiomes du code .....	522
<b>Index</b> .....	523