



**BASIC
MICRO**

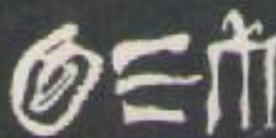
L'envie d'apprendre

Java™ 2

Edition 2001 SDK 1.3

WWW.OEMWEB.COM

Antoine Mirecourt et Pierre-Yves Saumont



OSMAN EYROLLES MULTIMEDIA

2-005-581-1

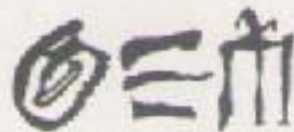
2-005-581-1



Java 2

Edition 2001 SDK 1.3

Antoine Mirecourt
et Pierre-Yves Saumont



OSMAN EYROLLES MULTIMEDIA

Sommaire

| | |
|--|-----------|
| Introduction | xv |
| Pourquoi Java ? | xv |
| Qu'est-ce que Java ? | xvi |
| Java est un langage orienté objets | xvi |
| Java est extensible à l'infini | xvii |
| Java est un langage à haute sécurité | xvii |
| Java est un langage simple à apprendre | xvii |
| Java est un langage compilé | xviii |
| Le premier langage du troisième millénaire ? | xviii |
| Les apports de la version 1.3 | xviii |
| Sources des programmes de ce livre | xix |
| Où se procurer le SDK Java 2 version 1.3 | xx |
| Les versions concernées par ce livre | xx |
| Contacter les auteurs | xx |
| Chapitre 1 : Installation du SDK | 1 |
| Installation du SDK | 2 |
| Configuration des chemins d'accès | 3 |
| Le chemin d'accès aux classes Java | 4 |
| Améliorer le confort d'utilisation | 5 |
| Installation de la documentation en ligne | 6 |
| Un éditeur pour la programmation | 8 |
| Ce que doit offrir un éditeur | 8 |
| UltraEdit | 8 |
| Quelques petits fichiers batch qui vous simplifieront la vie | 9 |
| Où trouver les informations ? | 10 |
| Le site Web de Sun | 10 |

| | |
|---|-----------|
| Le JDC | 10 |
| Les autres sites Web dédiés à Java | 11 |
| Les Newsgroups | 11 |
| Chapitre 2 : Votre premier programme | 13 |
| Première version : un programme minimal | 13 |
| Analyse du premier programme | 16 |
| Première applet | 23 |
| Analyse de la première applet | 25 |
| Une façon plus simple d'exécuter les applets | 29 |
| Résumé | 30 |
| Chapitre 3 : Les objets Java | 31 |
| Tout est objet | 31 |
| Les classes | 31 |
| Pertinence du modèle | 33 |
| Les instances de classes | 36 |
| Les membres de classes | 37 |
| Les membres statiques | 38 |
| Les constructeurs | 39 |
| Création d'un objet | 42 |
| La destruction des objets : le garbage collector | 45 |
| Comment retenir les objets : les handles | 45 |
| Création des handles | 45 |
| Modifier l'affectation d'un handle | 46 |
| Résumé | 48 |
| Chapitre 4 : Les primitives et les handles | 49 |
| Les primitives | 49 |
| Utiliser les primitives | 51 |
| Valeurs par défaut des primitives | 53 |
| Différences entre les objets et les primitives | 55 |
| Les valeurs littérales | 59 |
| Le casting des primitives | 60 |
| Le casting explicite | 61 |

| | |
|---|-----------|
| Casting d'une valeur entière en valeur flottante | 62 |
| Casting d'une valeur flottante en valeur entière | 63 |
| Formation des identificateurs | 63 |
| Portée des identificateurs | 63 |
| Portée des identificateurs de méthodes | 67 |
| Les objets n'ont pas de portée | 68 |
| Les chaînes de caractères | 68 |
| Constantes | 72 |
| Utilisation de final avec des objets | 72 |
| Accessibilité | 72 |
| Retour sur les variables statiques | 72 |
| Masquage des identificateurs de type static | 74 |
| Résumé | 76 |
| Chapitre 5 : Créez vos propres classes | 77 |
| ----- | |
| Tout est objet (bis) | 77 |
| L'héritage | 78 |
| Les constructeurs | 80 |
| Référence à la classe parente | 82 |
| La redéfinition des méthodes | 83 |
| Surcharger les méthodes | 84 |
| Signature d'une méthode | 85 |
| Surcharger les constructeurs | 87 |
| Les initialiseurs | 88 |
| Les initialiseurs de variables d'instances | 88 |
| Les initialiseurs d'instances | 89 |
| Méthodes : les valeurs de retour | 91 |
| Résumé | 93 |
| Chapitre 6 : Les opérateurs | 95 |
| ----- | |
| L'opérateur d'affectation | 95 |
| Raccourci | 96 |
| Les opérateurs arithmétiques à deux opérandes | 96 |
| Les opérateurs à deux opérandes et le sur-casting automatique | 97 |
| Les raccourcis | 99 |

| | |
|---|------------|
| Les opérateurs à un opérande | 100 |
| Les opérateurs relationnels | 102 |
| Les opérateurs logiques | 104 |
| Les opérateurs d'arithmétique binaire | 106 |
| Utilisation des opérateurs d'arithmétique binaire avec des valeurs logiques | 109 |
| L'opérateur à trois opérands ?: | 110 |
| Les opérateurs de casting | 110 |
| Les opérateurs de cast et les valeurs littérales | 111 |
| L'opérateur new | 111 |
| L'opérateur instanceof | 112 |
| Priorité des opérateurs | 112 |
| Résumé | 113 |
| Chapitre 7 : Les structures de contrôle | 115 |
| La séquence | 115 |
| Le branchement par appel de méthode | 116 |
| L'instruction de branchement return | 116 |
| L'instruction conditionnelle if | 116 |
| L'instruction conditionnelle else | 118 |
| Les instructions conditionnelles et les opérateurs ++ et -- | 119 |
| Les instructions conditionnelles imbriquées | 120 |
| La boucle for | 121 |
| L'initialisation | 121 |
| Le test | 122 |
| L'incrémentatation | 123 |
| Le bloc d'instructions | 123 |
| Imbrication des boucles for | 124 |
| Type des indices | 124 |
| Portée des indices | 124 |
| Sortie d'une boucle par return | 125 |
| Branchement au moyen des instructions break et continue | 125 |
| Utilisation de break et continue avec des étiquettes | 127 |
| L'instruction while | 128 |
| L'instruction switch | 128 |
| L'instruction synchronized | 134 |
| Résumé | 134 |

| | |
|---|------------|
| Chapitre 8 : L'accessibilité | 135 |
| Les packages (Où) | 138 |
| Chemins d'accès et packages | 141 |
| L'instruction package | 141 |
| Placement automatique des fichiers .class dans les répertoires correspondant aux packages | 143 |
| Chemin d'accès par défaut pour les packages | 144 |
| L'instruction import | 144 |
| Packages accessibles par défaut | 146 |
| Les fichiers .jar | 146 |
| Comment nommer vos packages ? | 147 |
| Ce qui peut être fait (Quoi) | 147 |
| static | 148 |
| Les variables static | 148 |
| Les méthodes static | 150 |
| Initialisation des variables static | 151 |
| Les initialiseurs statiques | 152 |
| final | 153 |
| Les variables final | 153 |
| Les variables final non initialisées | 154 |
| Les méthodes final | 154 |
| Les classes final | 156 |
| synchronized | 156 |
| native | 157 |
| transient | 157 |
| volatile | 157 |
| abstract | 158 |
| Les interfaces | 159 |
| Qui peut le faire (Qui) | 160 |
| public | 160 |
| protected | 160 |
| package | 161 |
| private | 161 |
| Autorisations d'accès aux constructeurs | 162 |
| Résumé | 162 |

| | |
|--|------------|
| Chapitre 9 : Le polymorphisme | 163 |
| Le sur-casting des objets | 164 |
| Retour sur l'initialisation | 165 |
| Le sur-casting | 167 |
| Le sur-casting explicite | 170 |
| Le sur-casting implicite | 171 |
| Le sous-casting | 171 |
| Le "late binding" | 173 |
| Les interfaces | 175 |
| Utiliser les interfaces pour gérer des constantes | 175 |
| Un embryon d'héritage multiple | 177 |
| Le polymorphisme et les interfaces | 178 |
| Pas de vrai héritage multiple | 179 |
| Quand utiliser les interfaces ? | 180 |
| Héritage et composition | 182 |
| Résumé | 183 |
| Chapitre 10 : Les tableaux et les collections | 185 |
| Les tableaux | 185 |
| Déclaration | 185 |
| Initialisation | 186 |
| Initialisation automatique | 186 |
| Les tableaux littéraux | 187 |
| Les tableaux de primitives sont des objets | 188 |
| Le sur-casting implicite des tableaux | 189 |
| Les tableaux d'objets sont des tableaux de handles | 192 |
| La taille des tableaux | 195 |
| Les tableaux multidimensionnels | 197 |
| Les tableaux et le passage d'arguments | 199 |
| Copie de tableaux | 200 |
| Les vecteurs | 204 |
| Les collections | 207 |
| Les itérateurs | 208 |
| Les comparateurs | 210 |
| Résumé | 211 |

| | |
|--|------------|
| Chapitre 11 : Les objets meurent aussi | 213 |
| Certains objets deviennent inaccessibles | 213 |
| Que deviennent les objets inaccessibles ? | 217 |
| Le garbage collector | 219 |
| Principe du garbage collector | 219 |
| Optimiser le travail du garbage collector | 221 |
| Les finaliseurs | 222 |
| Contrôler le travail du garbage collector | 226 |
| Références et accessibilité | 229 |
| Les références faibles | 230 |
| SoftReference | 232 |
| WeakReference | 232 |
| PhantomReference | 233 |
| Les queues | 233 |
| Autres formes de contrôle du garbage collector | 234 |
| La finalisation et l'héritage | 239 |
| Résumé | 240 |
| Chapitre 12 : Les classes internes | 241 |
| Les classes imbriquées | 241 |
| Les classes membres | 247 |
| Classes membres et héritage | 249 |
| Remarque concernant les classes membres | 249 |
| Les classes locales | 249 |
| Les classes anonymes | 251 |
| Comment sont nommées les classes anonymes | 254 |
| Résumé | 254 |
| Chapitre 13 : Les exceptions | 255 |
| Stratégies de traitement des erreurs | 255 |
| Signaler et stopper | 256 |
| Corriger et réessayer | 256 |
| Signaler et réessayer | 256 |
| La stratégie de Java | 257 |

| | |
|---|------------|
| Les deux types d'erreurs de Java | 257 |
| Les exceptions | 258 |
| Attraper les exceptions | 260 |
| Dans quelle direction sont lancées les exceptions ? | 261 |
| Manipuler les exceptions | 261 |
| Modification de l'origine d'une exception | 263 |
| Exceptions et héritage | 270 |
| Résumé | 270 |
| Chapitre 14 : Les entrées/sorties | 271 |
| Principe des entrées/sorties | 271 |
| Les streams de données binaires | 272 |
| Les streams d'entrée | 272 |
| Streams de communication | 272 |
| Streams de traitement | 272 |
| Les streams de sortie | 273 |
| Streams de communication | 273 |
| Streams de traitement | 273 |
| Les streams de caractères | 274 |
| Les streams d'entrée | 275 |
| Streams de communication | 275 |
| Streams de traitement | 275 |
| Les streams de sortie | 276 |
| Streams de communication | 276 |
| Streams de traitement | 276 |
| Les streams de communication | 277 |
| Lecture et écriture d'un fichier | 278 |
| Les streams de traitement | 279 |
| Exemple de traitement : utilisation d'un tampon | 284 |
| Exemple de traitement : conversion des fins de lignes | 285 |
| Résumé | 285 |
| Chapitre 15 : Le multithreading | 287 |
| Qu'est-ce qu'un processus ? | 287 |
| Avertissement | 287 |

| | |
|---|------------|
| Créer explicitement un thread | 288 |
| Exécuter plusieurs threads simultanément | 290 |
| Caractéristiques des threads | 294 |
| Contrôler les threads | 295 |
| Utilisation des groupes pour référencer les threads | 297 |
| Gérer la répartition du temps entre les threads | 301 |
| La priorité | 303 |
| La synchronisation | 305 |
| Résumé | 311 |
| Chapitre 16 : Fenêtres et boutons | 313 |
| Les composants lourds et les composants légers | 313 |
| Les composants lourds | 314 |
| Les composants légers | 314 |
| Le look & feel | 315 |
| Les fenêtres | 315 |
| Hiérarchie des fenêtres Java | 316 |
| Structure d'une fenêtre | 317 |
| Les layout managers | 319 |
| Créer une application fenêtrée | 319 |
| Quel événement intercepter ? | 320 |
| Intercepter les événements de fenêtre | 321 |
| Les événements et les listeners | 321 |
| Utilisation des composants | 329 |
| Utilisation des layout managers | 331 |
| Philosophie des layout managers | 332 |
| Bien utiliser les layout managers | 336 |
| Utilisation du FlowLayout | 340 |
| Utilisation d'un layout à plusieurs niveaux | 343 |
| Utilisation du GridBagLayout | 346 |
| Rendre l'interface réellement indépendante du système | 352 |
| Les éléments de l'interface utilisateur | 354 |
| Les boutons | 355 |
| Les événements de souris | 360 |
| Les menus | 366 |

| | |
|--|------------|
| Les fenêtres internes | 370 |
| Résumé | 374 |
| Chapitre 17 : Les applets | 377 |
| Les applets | 377 |
| Création d'une applet | 377 |
| Problèmes de compatibilité entre les versions | 378 |
| Avertissement | 379 |
| Deuxième avertissement | 380 |
| Fonctionnement d'une applet | 380 |
| Passer des paramètres à une applet | 381 |
| Agir sur le navigateur | 385 |
| Afficher un nouveau document | 385 |
| Afficher un message dans la barre d'état | 388 |
| Afficher des images | 388 |
| Les sons | 389 |
| Optimiser le chargement des applets à l'aide des fichiers d'archives | 390 |
| Conclusion | 391 |
| Index | 393 |