



CONTENIDO

Prólogo	xiii
Capítulo 1. Introducción a Java	1
1.1. La historia de Java	2
1.2. ¿Qué es Java?	3
1.2.1. Java como lenguaje de Internet	3
1.2.2. Java como lenguaje de propósito general	4
1.3. Características de Java	5
1.3.1. Sencillo	5
1.3.2. Orientado a objetos	6
1.3.3. Distribuido	7
1.3.4. Interpretado	7
1.3.5. Robusto	8
1.3.6. Seguro	8
1.3.7. Arquitectura neutral	9
1.3.8. Portable	9
1.3.9. Alto rendimiento	10
1.3.10. Multihilo	10
1.3.11. Dinámico	11
1.4. La programación orientada a objetos como base de Java	11
1.5. Especificaciones del lenguaje Java	12
1.6. Aplicaciones y <i>applets</i>	13
1.6.1. Semejanzas y diferencias entre aplicaciones y <i>applets</i> ..	13
1.7. Creación de programas	15
1.7.1. Etapas para crear un programa	15
1.8. Componentes de una aplicación	18
1.9. Herramientas de desarrollo Java	21
1.9.1. El entorno de desarrollo JDK	21
1.10. Una aplicación práctica de Java	22

1.11. Estructura de un programa aplicación en Java	
1.11.1. Referencia a miembros de una clase	
1.12. Errores de programación	
1.12.1 Errores de compilación (sintaxis)	
1.12.2. Errores de ejecución	
1.12.3. Errores lógicos	

Capítulo 2. Características del lenguaje Java

2.1. Palabras reservadas	
2.2. Identificadores	
2.3. Tipos de datos	
2.4. Tipos simples (primitivos)	
2.5. Variables	
2.6. Constantes	
2.7. La biblioteca de clases de Java	
2.8. Conceptos básicos sobre excepciones	
2.9. La clase Number y sus subclases	
2.10. Las clases Character y Boolean	
2.11. Entrada y salida básicas	
2.12. Operadores	
2.12.1. Operadores sobre enteros	
2.12.2. Operadores sobre valores en coma flotante	
2.13. La sentencia de asignación	
2.14. Expresiones	
2.15. Clase Math	
2.16. Paquete java.math	
2.17. Conversiones de tipos. Operadores molde	
2.18. Operadores aritméticos	
2.19. Operadores relacionales	
2.20. Operadores lógicos	
2.21. Operadores de manipulación de bits	
2.22. Operadores de asignación adicionales	
2.23. Operador condicional	
2.24. Prioridad de los operadores	

Capítulo 3. Decisiones y bucles

3.1. La sentencia if	
3.2. La sentencia if-else	
3.3. Las sentencias if e if-else anidadas	
3.4. La sentencia switch	
3.5. La sentencia for	
3.6. La sentencia break	
3.7. La sentencia continue	

3.8. Diferencias entre <code>continue</code> y <code>break</code>	86
3.9. La sentencia <code>while</code>	88
3.10. La sentencia <code>do-while</code>	91
Capítulo 4. Clases, objetos y métodos	95
4.1. Objetos y clases	96
4.2. Declaración y creación de un objeto	98
4.3. Acceso a datos y métodos	100
4.4. Utilización de métodos	100
4.5. Paso de parámetros	102
4.6. Paso de parámetros por valor	103
4.7. Paso de parámetros por referencia	103
4.8. Constructores	104
4.9. Modificadores de acceso	107
4.10. <code>private</code>	108
4.11. <code>protected</code>	109
4.12. <code>public</code>	111
4.13. Recursividad	111
Capítulo 5. Herencia	117
5.1. Descripción de herencia	118
5.2. La clase <code>Object</code>	119
5.3. El método <code>clone</code>	122
5.4. El método <code>equals</code>	122
5.5. El método <code>finalize</code>	122
5.6. El método <code>toString</code>	123
5.7. El método <code>getClass</code>	123
5.8. Ventajas de la herencia	124
5.9. Superclases y subclases	124
5.10. Modificadores y herencia	125
5.11. Clases abstractas	127
5.12. Métodos abstractos	129
5.13. Interfaces	134
5.14. Definición de una interfaz	135
Capítulo 6. Encapsulamiento y polimorfismo	141
6.1. Encapsulamiento	142
6.2. Modificadores de clase	142
6.3. Modificadores de variables	143
6.4. Modificadores de métodos	144
6.5. Clases internas	144
6.6. Paquetes	149
6.7. Declaración de un paquete	150
6.8. Paquetes incorporados	151
6.9. Acceso a los elementos de un paquete	152

6.10. Importación de paquetes	153
6.11. Control de acceso a paquetes	153
6.12. Polimorfismo	154
6.13. Ligadura	156
6.14. Ligadura dinámica	157
Capítulo 7. Arrays	161
7.1. Concepto de array	162
7.2. Proceso de arrays	163
7.2.1 Declaración	164
7.2.2. Creación	164
7.2.3. Inicialización y utilización	165
7.3. Arrays de objetos	167
7.4. Copia de arrays	169
7.5. Arrays multidimensionales	171
7.5.1. Declaración de arrays multidimensionales	172
7.6. Ordenación de arrays	178
7.7. Selección	179
7.8. Burbuja	180
7.9. Inserción	181
7.10. Shell	183
7.11. Ordenación rápida	185
7.12. Búsqueda	186
7.13. Implementación genérica de los métodos de ordenación	188
Capítulo 8. Cadenas y fechas	193
8.1. Creación de cadenas	194
8.2. Comparación de cadenas	197
8.3. Concatenación	201
8.4. Otros métodos de la clase String	202
8.5. La clase StringTokenizer	208
8.6. La clase StringBuffer	209
8.7. Métodos de la clase StringBuffer	210
8.8. La clase Date	212
8.9. Los formatos de Fechas	213
8.10. La clase Calendar	214
Capítulo 9. Interfaces gráficas de usuario	217
9.1. El AWT	218
9.2. Realización de dibujos: clase Graphics	219
9.3. La clase Component	222
9.4. La clase Container	224
9.5. Ventanas	225
9.5.1. Clase Frame	225
9.5.2. Clase Dialog	229

9.5.3. Clase <code>FileDialog</code>	232
9.6. Clase <code>Panel</code>	234
9.7. Clase <code>Label</code>	234
9.8. Clase <code>Button</code>	236
9.9. Clase <code>TextComponent</code>	238
9.10. Clase <code>Canvas</code>	242
9.11. Clase <code>Choice</code>	244
9.12. Clase <code>Checkbox</code>	246
9.13. Listas	249
9.14. Clase <code>Scrollbar</code>	252
9.15. Menús	254
9.16. Administradores de diseño	256
9.16.1. <code>FlowLayout</code>	256
9.16.2. <code>BorderLayout</code>	257
9.16.3. <code>GridLayout</code>	258
9.17. <code>swing</code>	262
Capítulo 10. Gestión de eventos	263
10.1. Tipos de eventos	264
10.2. Los componentes del AWT como fuente de eventos	265
10.3. Receptores de eventos	266
10.4. Procesamiento de eventos	269
10.5. Clases adaptadoras	272
10.6. Clases receptoras anónimas	274
10.7. Problemas comunes en el tratamiento de eventos	276
Capítulo 11. Applets	285
11.1. Introducción a HTML	286
11.2. Incorporación de <i>applets</i> a páginas Web	289
11.2.1. Edición de un documento HTML y ejecución de applets	291
11.3. Estructura de un <i>applet</i>	294
11.4. Transformación de aplicaciones en <i>applets</i>	298
11.5. Incorporación de sonido	306
11.6. Incorporación de imágenes	306
Capítulo 12. Programación concurrente: Hilos de ejecución	311
12.1. La programación multihilo en Java	312
12.2. Estados de un hilo	313
12.3. Creación de hilos	314
12.4. Planificación y prioridades	316
12.5. Hilos de tipo demonio	317
12.6. Grupos de hilos	318
12.7. Sincronización	319
12.8. Animaciones	326
12.9. Doble <i>buffer</i>	327

Capítulo 13. Manejo de excepciones	339
13.1. Conceptos generales	340
13.2. Manejo de excepciones	343
13.3. Captura y tratamiento de excepciones	343
13.4. Lanzar la excepción	345
13.5. Declarar la excepción	347
13.6. El bloque finally	348
13.7. Creación de excepciones	353
13.8. Métodos de la clase Throwable	358
Capítulo 14. Archivos	361
14.1. La clase File	362
14.2. Flujos	368
14.3. Apertura de archivos	370
14.4. Encadenamiento de flujos	373
14.5. Excepciones en archivos	375
14.6. Métodos de InputStream	378
14.7. Métodos de OutputStream	379
14.8. Métodos de Reader	379
14.9. Métodos de Writer	379
14.10. Métodos de DataInputStream	380
14.11. Métodos de DataOutputStream	381
14.12. Métodos de RandomAccessFile	381
14.13. Serialización de objetos	382
14.14. StringTokenizer y StreamTokenizer	383
14.15. Operaciones con archivos y mantenimientos de los mismos	383
14.16. Archivos secuenciales	384
14.17. Archivos directos	395
14.18. Funciones de transformación de clave y tratamiento de colisiones	396
Capítulo 15. Estructuras de datos definidas por el programador	411
15.1. Listas	412
15.2. Implementación de una lista	415
15.3. Lista ordenada	417
15.4. Listas genéricas y uso de interfaces	422
15.5. Listas doblemente enlazadas	434
15.6. Pilas	435
15.7. Colas	438
15.8. Colas circulares	442

APÉNDICES

A.	Palabras reservadas Java	445
B.	Prioridad de operadores	447
C.	Guía de sintaxis	451
D.	Paquetes de la plataforma Java 2. Versiones 1.3 y 1.4 Beta	475
E.	Java versus C++	501
F.	Contenido del CD	513
G.	Recursos: libros, revistas y WEB	525
	Índice analítico	529

Cinco años después de su lanzamiento, Java se ha convertido en un estándar de la industria, en un lenguaje de programación para desarrollo de aplicaciones tanto de propósito general como de Internet, y también en un lenguaje para comenzar la formación en programación, al tener características excelentes para el aprendizaje.

Java, desarrollado por Sun Microsystems en 1995, es un magnífico y completo lenguaje de programación orientado a objetos diseñado para distribuir contenidos a través de una red. Una de sus principales características es que permite operar de forma independiente de la plataforma y del sistema operativo que se está utilizando. Esto quiere decir que permite crear una aplicación que podrá descargarse de la red y funcionar posteriormente en cualquier tipo de plataforma de hardware o software. Generalmente, y al contrario, todo programa o aplicación queda atado a dos cosas: al hardware y al sistema operativo. Así, por ejemplo, una aplicación Windows sólo funcionará en plataforma Wintel (equipada con procesadores Intel y sistema operativo Windows) igual que una versión creada para Mac sólo funciona sobre Power PC o Imac y Mac OS o la misma aplicación desarrollada para Unix, sólo lo hace sobre plataformas Unix y no hay forma de que funcione sobre otra máquina.

La idea de Java, por el contrario, es poner una capa sobre cualquier plataforma de hardware y sobre cualquier sistema operativo que permite que cualquier aplicación desarrollada en Java quede ligada únicamente a Java, independizada por lo tanto de la