

# Contenido

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introducción al lenguaje de programación Java</b> .....  | <b>1</b>  |
| 1.1      | Introducción.....   | 2         |
| 1.2      | Comencemos a programar.....                                 | 2         |
| 1.2.1    | El Entorno Integrado de Desarrollo (IDE).....               | 3         |
| 1.3      | Estructuras de control.....                                 | 4         |
| 1.3.1    | Entrada y salida de datos por consola.....                  | 4         |
| 1.3.2    | Definición de variables.....                                | 5         |
| 1.3.3    | Comentarios en el código.....                               | 5         |
| 1.3.4    | Estructuras de decisión.....                                | 6         |
| 1.3.5    | Estructuras iterativas.....                                 | 10        |
| 1.4      | Otros elementos del lenguaje.....                           | 13        |
| 1.4.1    | Tipos de datos.....   | 13        |
| 1.4.2    | Algunas similitudes y diferencias con C y C++.....          | 13        |
| 1.4.3    | Definición de constantes.....                               | 14        |
| 1.4.4    | Arrays.....   | 15        |
| 1.4.5    | Matrices.....   | 18        |
| 1.4.6    | Literales de cadenas de caracteres.....                     | 20        |
| 1.4.7    | Caracteres especiales.....                                  | 22        |
| 1.4.8    | Argumentos en línea de comandos.....                        | 22        |
| 1.5      | Tratamiento de cadenas de caracteres.....                   | 23        |
| 1.5.1    | Acceso a los caracteres de un String.....                   | 24        |
| 1.5.2    | Mayúsculas y minúsculas.....                                | 24        |
| 1.5.3    | Ocurrencias de caracteres.....                              | 25        |
| 1.5.4    | Subcadenas.....   | 25        |
| 1.5.5    | Prefijos y sufijos.....                                     | 26        |
| 1.5.6    | Posición de un substring dentro de la cadena.....           | 27        |
| 1.5.7    | Concatenar cadenas.....                                     | 27        |
| 1.5.8    | La clase <code>StringBuffer</code> .....                    | 27        |
| 1.5.9    | Conversión entre números y cadenas.....                     | 29        |
| 1.5.10   | Representación numérica en diferentes bases.....            | 30        |
| 1.5.11   | La clase <code>StringTokenizer</code> .....                 | 31        |
| 1.5.12   | Usar expresiones regulares para particionar una cadena..... | 33        |
| 1.5.13   | Comparación de cadenas.....                                 | 33        |
| 1.6      | Operadores.....   | 35        |
| 1.6.1    | Operadores aritméticos.....                                 | 35        |
| 1.6.2    | Operadores lógicos.....                                     | 36        |
| 1.6.3    | Operadores relacionales.....                                | 36        |
| 1.6.4    | Operadores lógicos de bit.....                              | 36        |
| 1.6.5    | Operadores de desplazamiento de bit.....                    | 36        |
| 1.7      | La máquina virtual y el JDK.....                            | 36        |
| 1.7.1    | El JDK (Java Development Kit).....                          | 37        |
| 1.7.2    | Versiones y evolución del lenguaje Java.....                | 37        |
| 1.8      | Resumen.....  | 38        |
| <b>2</b> | <b>Programación orientada a objetos</b> .....               | <b>39</b> |
| 2.1      | Introducción.....   | 40        |
| 2.2      | Clases y objetos.....                                       | 40        |
| 2.2.1    | Los métodos.....  | 41        |
| 2.2.2    | Herencia y sobrescritura de métodos.....                    | 43        |
| 2.2.3    | El método <code>toString</code> .....                       | 43        |

|            |  |            |
|------------|--|------------|
| 2.2.4      | El método equals .....   | 44         |
| 2.2.5      | Definir y “crear” objetos .....                                  | 45         |
| 2.2.6      | El constructor .....   | 46         |
| 2.2.7      | Un pequeño repaso de lo visto hasta aquí .....                   | 47         |
| 2.2.8      | Convenciones de nomenclatura .....                               | 49         |
| 2.2.9      | Sobrecarga .....   | 50         |
| 2.2.10     | Encapsulamiento .....  | 53         |
| 2.2.11     | Visibilidad de métodos y atributos .....                         | 55         |
| 2.2.12     | Packages (paquetes).....   | 56         |
| 2.2.13     | La estructura de los paquetes y la variable CLASSPATH.....       | 57         |
| 2.2.14     | Las APIs (Application Programming Interface).....                | 58         |
| 2.2.15     | Representación gráfica UML.....                                  | 58         |
| 2.2.16     | Importar clases de otros paquetes .....                          | 60         |
| <b>2.3</b> | <b>Herencia y polimorfismo .....</b>                             | <b>60</b>  |
| 2.3.1      | Polimorfismo.....  | 63         |
| 2.3.2      | Constructores de subclases .....                                 | 65         |
| 2.3.3      | La referencia super .....  | 65         |
| 2.3.4      | La referencia this .....   | 68         |
| 2.3.5      | Clases abstractas.....   | 69         |
| 2.3.6      | Constructores de clases abstractas .....                         | 72         |
| 2.3.7      | Instancias.....  | 76         |
| 2.3.8      | Variables de instancia .....                                     | 77         |
| 2.3.9      | Variables de la clase .....                                      | 79         |
| 2.3.10     | El Garbage Collector (recolector de residuos) .....              | 79         |
| 2.3.11     | El método finalize.....  | 80         |
| 2.3.12     | Constantes .....   | 81         |
| 2.3.13     | Métodos de la clase .....  | 81         |
| 2.3.14     | Clases utilitarias.....  | 83         |
| 2.3.15     | Referencias estáticas .....                                      | 84         |
| 2.3.16     | Colecciones (primera parte) .....                                | 85         |
| 2.3.17     | Clases genéricas .....   | 90         |
| 2.3.18     | Implementación de una pila (estructura de datos).....            | 93         |
| 2.3.19     | Implementación de una cola (estructura de datos).....            | 94         |
| <b>2.4</b> | <b>Interfaces .....</b>  | <b>95</b>  |
| 2.4.1      | Desacoplamiento de clases .....                                  | 97         |
| 2.4.2      | El patrón de diseño: factory method.....                         | 99         |
| 2.4.3      | Abstracción a través de interfaces.....                          | 99         |
| 2.4.4      | La interface Comparable.....                                     | 100        |
| 2.4.5      | Desacoplar aún más .....   | 104        |
| <b>2.5</b> | <b>Colecciones .....</b>   | <b>108</b> |
| 2.5.1      | Cambio de implementación .....                                   | 110        |
| <b>2.6</b> | <b>Excepciones .....</b>   | <b>111</b> |
| 2.6.1      | Excepciones declarativas y no declarativas.....                  | 114        |
| 2.6.2      | El bloque try-catch-finally.....                                 | 116        |
| <b>2.7</b> | <b>Resumen .....</b>   | <b>118</b> |
| <b>3</b>   | <b>Acceso a bases de datos (JDBC).....</b>                       | <b>119</b> |
| <b>3.1</b> | <b>Introducción .....</b>  | <b>120</b> |
| <b>3.2</b> | <b>Conceptos básicos sobre bases de datos relacionales .....</b> | <b>120</b> |
| 3.2.1      | Relaciones foráneas y consistencia de datos.....                 | 121        |
| 3.2.2      | Diagrama Entidad-Relación (DER) .....                            | 122        |
| 3.2.3      | SQL – Structured Query Language.....                             | 122        |
| 3.2.4      | Ejecutar queries.....  | 122        |

|            |   |            |
|------------|---|------------|
| 3.2.5      | Unir tablas (join) .....  | 123        |
| 3.2.6      | Ejecutar updates .....  | 124        |
| <b>3.3</b> | <b>Conectar programas Java con bases de datos .....</b>                                 | <b>125</b> |
| 3.3.1      | Invocar un query con un join .....  | 130        |
| 3.3.2      | Updates .....   | 130        |
| 3.3.3      | Ejecutar un INSERT .....  | 130        |
| 3.3.4      | Ejecutar un DELETE .....  | 132        |
| 3.3.5      | Ejecutar un UPDATE .....  | 132        |
| 3.3.6      | El patrón de diseño "Singleton" (Singleton Pattern) .....                               | 132        |
| 3.3.7      | Singleton Pattern para obtener la conexión .....  | 133        |
| 3.3.8      | El shutdown hook .....  | 135        |
| 3.3.9      | Inner classes (clases internas) .....   | 136        |
| 3.3.10     | Manejo de transacciones .....   | 136        |
| <b>3.4</b> | <b>Uso avanzado de JDBC .....</b>   | <b>138</b> |
| 3.4.1      | Acceso a la metadata (ResultSetMetaData) .....  | 138        |
| 3.4.2      | Definir el "query fetch size" para conjuntos de resultados grandes .....                | 140        |
| 3.4.3      | Ejecutar batch updates (procesamiento por lotes) .....                                  | 141        |
| <b>3.5</b> | <b>Resumen .....</b>  | <b>142</b> |
| <b>4</b>   | <b>Diseño de aplicaciones Java (Parte I) .....</b>                                      | <b>143</b> |
| 4.1        | Introducción .....  | 144        |
| 4.2        | Atributos de una aplicación .....   | 144        |
| 4.2.1      | Casos de uso .....  | 144        |
| 4.3        | Desarrollo de aplicaciones en capas .....   | 145        |
| 4.3.1      | Aplicación de estudio .....   | 146        |
| 4.3.2      | Análisis de los objetos de acceso a datos (DAO y DTO) .....                             | 147        |
| 4.3.3      | Análisis del façade .....   | 151        |
| 4.3.4      | Diagrama de secuencias de UML .....   | 154        |
| 4.4        | Portabilidad entre diferentes bases de datos .....                                      | 155        |
| 4.4.1      | DAOs abstractos e implementaciones específicas para las diferentes bases de datos ..... | 156        |
| 4.4.2      | Implementación de un factory method .....   | 159        |
| 4.4.3      | Combinar el factory method con el singleton pattern .....                               | 160        |
| 4.4.4      | Mejorar el diseño de los DAOs abstractos .....  | 162        |
| 4.5        | Diseño por contratos .....  | 164        |
| 4.5.1      | Coordinación de trabajo en equipo .....   | 164        |
| 4.6        | Resumen .....   | 166        |
| <b>5</b>   | <b>Interfaz gráfica (GUI) .....</b>   | <b>167</b> |
| 5.1        | Introducción .....  | 168        |
| 5.2        | Componentes y contenedores .....  | 168        |
| 5.2.1      | Distribución de componentes (layouts) .....   | 169        |
| 5.2.2      | AWT y Swing .....   | 169        |
| 5.3        | Comenzando a desarrollar GUI .....  | 170        |
| 5.3.1      | Distribuciones relativas .....  | 170        |
| 5.3.2      | Combinación de layouts .....  | 176        |
| 5.4        | Capturar eventos .....  | 182        |
| 5.4.1      | Tipos de eventos .....  | 186        |
| 5.4.2      | Eventos de acción .....   | 187        |
| 5.4.3      | Eventos de teclado .....  | 190        |
| 5.5        | Swing .....   | 192        |
| 5.5.1      | Cambiar el LookandFeel .....  | 196        |

|          |   |            |
|----------|---|------------|
| 5.6      | Model View Controller (MVC)                           | 197        |
| 5.6.1    | Ejemplo de uso: ListModel                             | 198        |
| 5.6.2    | Ejemplo de uso: TableModel                            | 200        |
| 5.7      | Resumen   | 203        |
| <b>6</b> | <b>Multithreading (Hilos)</b>                         | <b>205</b> |
| 6.1      | Introducción  | 206        |
| 6.2      | Implementar threads en Java                           | 207        |
| 6.2.1    | La interface Runnable                                 | 208        |
| 6.2.2    | Esperar a que finalice un thread                      | 209        |
| 6.2.3    | Threads e interfaz gráfica                            | 210        |
| 6.2.4    | Sistemas operativos multitarea                        | 213        |
| 6.2.5    | Ciclo de vida de un thread                            | 213        |
| 6.2.6    | Prioridad de ejecución                                | 215        |
| 6.3      | Sincronización de threads                             | 216        |
| 6.3.1    | Monitores y sección crítica                           | 216        |
| 6.3.2    | Ejemplo del Productor/Consumidor                      | 217        |
| 6.4      | Resumen   | 221        |
| <b>7</b> | <b>Networking</b>                                     | <b>223</b> |
| 7.1      | Introducción  | 224        |
| 7.2      | Conceptos básicos de networking                       | 224        |
| 7.2.1    | TCP - "Transmission Control Protocol"                 | 224        |
| 7.2.2    | UDP - "User Datagram Protocol"                        | 224        |
| 7.2.3    | Puertos   | 225        |
| 7.2.4    | Dirección IP  | 225        |
| 7.2.5    | Aplicaciones cliente/servidor                         | 225        |
| 7.3      | TCP en Java   | 225        |
| 7.3.1    | El socket   | 225        |
| 7.3.2    | Un simple cliente/servidor en Java                    | 225        |
| 7.3.3    | Serialización de objetos                              | 228        |
| 7.3.4    | Implementación de un servidor multithread             | 229        |
| 7.3.5    | Enviar y recibir bytes                                | 231        |
| 7.3.6    | Enviar y recibir valores de tipos de datos primitivos | 234        |
| 7.4      | UDP en Java   | 234        |
| 7.5      | Remote Method Invocation (RMI)                        | 237        |
| 7.5.1    | Componentes de una aplicación RMI                     | 237        |
| 7.5.2    | Ejemplo de una aplicación que utiliza RMI             | 237        |
| 7.5.3    | Compilar y ejecutar la aplicación RMI                 | 240        |
| 7.5.4    | RMI y serialización de objetos                        | 240        |
| 7.6      | Resumen   | 241        |
| <b>8</b> | <b>Diseño de aplicaciones Java (Parte II)</b>         | <b>243</b> |
| 8.1      | Introducción  | 244        |
| 8.2      | Repaso de la aplicación de estudio                    | 244        |
| 8.2.1    | El modelo de datos                                    | 244        |
| 8.2.2    | La funcionalidad                                      | 244        |
| 8.2.3    | El backend y el frontend                              | 245        |
| 8.2.4    | Los métodos del façade                                | 245        |
| 8.3      | Capas lógicas vs. capas físicas                       | 246        |
| 8.3.1    | Desventajas de un modelo basado en dos capas físicas  | 246        |
| 8.3.2    | Modelo de tres capas físicas                          | 247        |

|           |  |            |
|-----------|--|------------|
| 8.4       | Desarrollo de la aplicación en tres capas físicas .....    | 248        |
| 8.4.1     | Desarrollo del servidor.....                               | 248        |
| 8.4.2     | Desarrollo de un cliente de prueba .....                   | 252        |
| 8.4.3     | El service locator (o ubicador de servicios).....          | 254        |
| 8.4.4     | Integración con la capa de presentación .....              | 259        |
| 8.5       | Implementación del servidor con tecnología RMI.....        | 262        |
| 8.5.1     | El servidor RMI .....                                      | 262        |
| 8.5.2     | El service locator y los objetos distribuidos .....        | 264        |
| 8.5.3     | Desarrollo de un cliente de prueba .....                   | 264        |
| 8.5.4     | Integración con la capa de presentación.....               | 265        |
| 8.5.5     | El Business Delegate.....                                  | 268        |
| 8.6       | Concurrencia y acceso a la base de datos.....              | 268        |
| 8.6.1     | El pool de conexiones .....                                | 268        |
| 8.6.2     | Implementación de un pool de conexiones .....              | 269        |
| 8.6.3     | Integración con los servidores TCP y RMI .....             | 274        |
| 8.7       | Resumen .....  | 275        |
| <b>9</b>  | <b>Estructuras de datos dinámicas .....</b>                | <b>277</b> |
| 9.1       | Introducción.....  | 278        |
| 9.2       | Estructuras dinámicas .....                                | 278        |
| 9.2.1     | El nodo .....  | 278        |
| 9.2.2     | Lista enlazada (linked list).....                          | 279        |
| 9.2.3     | Pila (stack) .....   | 283        |
| 9.2.4     | Cola (queue) .....   | 284        |
| 9.2.5     | Implementación de una cola sobre una lista circular .....  | 285        |
| 9.2.6     | Clases LinkedList, Stack y Queue .....                     | 288        |
| 9.2.7     | Tablas de dispersión (Hashtable).....                      | 288        |
| 9.2.8     | Estructuras de datos combinadas.....                       | 290        |
| 9.2.9     | Árboles.....   | 293        |
| 9.2.10    | Árbol Binario de Búsqueda (ABB).....                       | 293        |
| 9.2.11    | La clase TreeSet.....                                      | 294        |
| 9.3       | Resumen .....  | 295        |
| <b>10</b> | <b>Parametrización mediante archivos XML.....</b>          | <b>297</b> |
| 10.1      | Introducción.....  | 298        |
| 10.2      | XML - “Extensible Markup Language” .....                   | 298        |
| 10.3      | Estructurar y definir parámetros en un archivo XML.....    | 299        |
| 10.3.1    | Definición de la estructura de parámetros .....            | 300        |
| 10.3.2    | Leer y parsear el contenido de un archivo XML .....        | 302        |
| 10.3.3    | Acceder a la información contenida en el archivo XML ..... | 304        |
| 10.4      | Resumen .....  | 312        |
| <b>11</b> | <b>Introspección de clases y objetos .....</b>             | <b>313</b> |
| 11.1      | Introducción.....  | 314        |
| 11.2      | Comenzando a introspectar .....                            | 315        |
| 11.2.1    | Identificar métodos y constructores.....                   | 315        |
| 11.2.2    | Acceso al prototipo de un método .....                     | 317        |
| 11.3      | Annotations.....   | 320        |
| 11.4      | Resumen .....  | 322        |
| <b>12</b> | <b>Generalizaciones y desarrollo de frameworks .....</b>   | <b>323</b> |
| 12.1      | Introducción.....  | 324        |

|           |  |            |
|-----------|--|------------|
| 12.2      | ¿Qué es un framework?  | 324        |
| 12.2.1    | ¿Frameworks propios o frameworks de terceros?                  | 325        |
| 12.2.2    | Reinventar la rueda  | 325        |
| 12.3      | Un framework para acceder a archivos XML                       | 326        |
| 12.3.1    | Diseño de la API del framework                                 | 327        |
| 12.3.2    | Análisis del elemento a generalizar                            | 329        |
| 12.3.3    | Parsear el archivo XML y cargar la estructura de datos         | 330        |
| 12.4      | Un framework para acceder a bases de datos                     | 336        |
| 12.4.1    | Identificación de la tarea repetitiva                          | 337        |
| 12.4.2    | Diseño de la API del framework                                 | 338        |
| 12.4.3    | Java Beans   | 340        |
| 12.4.4    | Transacciones  | 350        |
| 12.4.5    | Mappeo de tablas usando annotations                            | 355        |
| 12.5      | El bean factory  | 358        |
| 12.6      | Integración  | 360        |
| 12.6.1    | Los objetos de acceso a datos                                  | 360        |
| 12.6.2    | El façade  | 362        |
| 12.6.3    | El archivo de configuración                                    | 363        |
| 12.6.4    | El cliente   | 363        |
| 12.7      | Resumen  | 364        |
| <b>13</b> | <b>Entrada/Salida</b>  | <b>365</b> |
| 13.1      | Introducción   | 366        |
| 13.2      | I/O streams (flujos de entrada y salida)                       | 366        |
| 13.2.1    | Entrada y salida estándar                                      | 366        |
| 13.2.2    | Redireccionar la entrada y salidas estándar                    | 368        |
| 13.2.3    | Cerrar correctamente los streams                               | 369        |
| 13.2.4    | Streams de bytes (InputStream y OutputStream)                  | 370        |
| 13.2.5    | Streams de caracteres (readers y writers)                      | 371        |
| 13.2.6    | Streams bufferizados   | 372        |
| 13.2.7    | Streams de datos (DataInputStream y DataOutputStream)          | 373        |
| 13.2.8    | Streams de objetos (ObjectInputStream y ObjectOutputStream)    | 374        |
| 13.3      | Resumen  | 376        |
| <b>14</b> | <b>Consideraciones finales</b>                                 | <b>377</b> |
| 14.1      | Introducción   | 378        |
| 14.2      | Consideraciones sobre multithreading y concurrencia            | 378        |
| 14.2.1    | Clases con o sin métodos sincronizados                         | 378        |
| 14.2.2    | El singleton pattern en contextos multithreaded                | 378        |
| 14.3      | Consideraciones sobre clases “legacy”                          | 380        |
| 14.3.1    | La clase StringTokenizer y el método split                     | 380        |
| 14.4      | Resumen  | 380        |
| <b>15</b> | <b>Object Relational Mapping (ORM) y persistencia de datos</b> | <b>381</b> |
| 15.1      | Introducción   | 382        |
| 15.2      | Hibernate framework  | 383        |
| 15.2.1    | El modelo de datos relacional                                  | 383        |
| 15.2.2    | ORM (Object Relational Mapping)                                | 384        |
| 15.2.3    | Configuración de Hibernate                                     | 384        |
| 15.2.4    | Mappeo de tablas   | 385        |
| 15.2.5    | La sesión de Hibernate   | 387        |

|  |            |
|--|------------|
| 15.3 Asociaciones y relaciones .....                                       | 387        |
| 15.3.1 Asociación many-to-one .....  | 388        |
| 15.3.2 Asociación one-to-many .....  | 389        |
| 15.3.3 P6Spy .....   | 391        |
| 15.3.4 Lazy loading vs. eager loading .....                                | 392        |
| 15.4 Recuperar colecciones de objetos .....                                | 394        |
| 15.4.1 Criterios de búsqueda vs. HQL .....                                 | 394        |
| 15.4.2 Named queries .....   | 396        |
| 15.4.3 Ejecutar SQL nativo .....   | 397        |
| 15.4.4 Queries parametrizados .....  | 397        |
| 15.5 Insertar, modificar y eliminar filas .....                            | 397        |
| 15.5.1 Transacciones .....   | 397        |
| 15.5.2 Insertar una fila .....   | 398        |
| 15.5.3 Estrategia de generación de claves primarias .....                  | 398        |
| 15.5.4 Modificar una fila .....  | 399        |
| 15.5.5 Múltiples updates y deletes .....                                   | 399        |
| 15.6 Casos avanzados .....   | 400        |
| 15.6.1 Análisis y presentación del modelo de datos .....                   | 400        |
| 15.6.2 Asociaciones many-to-many .....                                     | 401        |
| 15.6.3 Claves primarias compuestas (Composite Id) .....                    | 403        |
| 15.7 Diseño de aplicaciones .....  | 405        |
| 15.7.1 Factorías de objetos .....  | 406        |
| 15.8 Resumen .....   | 412        |
| <b>16 Inversión del control por inyección de dependencias .....</b>        | <b>415</b> |
| 16.1 Introducción .....  | 416        |
| 16.2 Spring framework .....  | 416        |
| 16.2.1 Desacoplar el procesamiento .....                                   | 418        |
| 16.2.2 Conclusión y repaso .....   | 422        |
| 16.3 Spring y JDBC .....   | 424        |
| 16.4 Integración Spring + Hibernate .....                                  | 426        |
| 16.5 Resumen .....   | 430        |
| <b>17 Actualización a Java 8 .....</b>                                     | <b>431</b> |
| 17.1 Introducción .....  | 432        |
| 17.2 Novedades en Java 7 .....   | 432        |
| 17.2.1 Literales binarios .....  | 432        |
| 17.2.2 Literales numéricos separados por “_” (guion bajo) .....            | 432        |
| 17.2.3 Uso de cadenas en la sentencia switch .....                         | 433        |
| 17.2.4 Inferencia de tipos genéricos .....                                 | 433        |
| 17.2.5 Sentencia try con recurso incluido .....                            | 434        |
| 17.2.6 Atrapar múltiples excepciones dentro de un mismo bloque catch ..... | 434        |
| 17.2.7 Nuevos métodos en la clase File .....                               | 435        |
| 17.3 Novedades en Java 8 .....   | 435        |
| Bibliografía .....   | 439        |

## Material disponible en la Web de apoyo

---

Todos los capítulos cuentan con autoevaluaciones online, además los docentes tienen presentaciones para apoyar el dictado de las clases.

### Encontrará videotutoriales sobre:

- Instalar Java y Eclipse.
- Crear y ejecutar nuestro primer programa en Eclipse.
- Pasar argumentos en línea de comandos en Eclipse.
- Compilar y ejecutar un programa Java desde la línea de comandos (sin utilizar Eclipse).
- Utilizar la herramienta "javadoc" para documentar nuestro código fuente.
- Empaquetar clases utilizando la herramienta "jar".
- Usar Eclipse como cliente SQL.
- Utilizar herramientas visuales para diseño y desarrollo de interfaz gráfica.
- Compilar y ejecutar una aplicación RMI.

### Lecturas complementarias:

- Apéndice A Nociones básicas de programación
- Apéndice B Applets

### Además:

- Código fuente de cada capítulo.
- Hipervínculos de interés.
- Fe de erratas.