

Índice

!, 87
!= operador, 808
!=, 85
&&, 87
&, 89
&=, 90
-, 84
@ en desuso, 76
[: operador de indexación [], 159
^, 90
^=, 90
|, 89
||, 87
|=, 90
'+' operador + para String, 833
+, 84
<, 85
<<, 90
<<=, 90
<=, 85
== operador, 808
== vs equals(), 497
==, 85
>, 85
>=, 85
>>, 90
>>=, 90

A

Abstracción, 1
Abstract Window Toolkit (AWT), 535
AbstractButton, 572
AbstractSequentialList, 381
AbstractSet, 347
accept(), 715
acceso
 al constructor por defecto sintetizado, 527
 a paquetes y friendly, 179
 clase, 185

 clases internas & derechos de acceso., 277
 control, 188
 dentro de un directorio, vía el paquete por defecto, 180
 especificadores, 6, 169, 178, 9
Acoplamiento, 409
ActionEvent, 640
ActionListener, 553
actor, en casos de uso, 37
actualizaciones del libro, XLIII
adaptadores, listeners, 568
add(), ArrayList, 338
addActionListener(), 637
AddChangeListener, 603
AddListener, 562
addXXXListener() 562
Adición, 82
Adler
advertencia sobre copyright del código fuente, XLI
Agregación, 7
Alinear, 541
AlreadyBoundException, 771
ámbito
 anidamiento de clases internas en cualquier ámbito arbitrario, 273
 clases internas en métodos ámbitos, 272
análisis
 de requisitos, 36
 y diseño orientado a objetos, 33
 parálisis, 34
AND
 lógico (&&), 87
 operador de bits, 96
anidando interfaces, 265
aplicación
 applets y aplicaciones combinadas, 543
 constructor de aplicaciones, 628
aplicaciones con ventanas, 543
applet

- parámetro, 641
- y packages, 543
- alinear, 541
- aplicaciones y applets combinados., 543
- classpath, 542
- codebase, 541
- combinado con aplicación., 658
- empaquetando applets en un fichero JAR para optimizar la carga, 622
- mostrando una página Web dentro de un applet., 727
- name, 541
- parámetro, 541
- parámetros de inicialización., 658
- restricciones, 537
- ubicándolo en una página Web., 540
- ventajas de sistemas cliente/servidor., 538
- Applet, 537
- Appletviewer 541
- aplicación combinada con Applet, 658
- árbol, 612
- array
 - asociativo, 328, 361
 - de objetos, 302
 - de tipos de datos primitivos, 303
 - array asociativo, 361
 - array asociativo, Map, 330
 - comparación de arrays, 322
 - comparación de elementos, 322
 - comprobación de límites, 160
 - copia de un array, 320
 - devolver un array, 306
 - inicialización, 159
 - longitud, 160, 302
 - multidimensional, 164
 - objetos de primera clase, 302
 - sintaxis de inicialización de agretados dinámica, 305
- Array, 301
- ArrayList consciente de los tipos, 341
- ArrayList get(), 338, 343
- ArrayList size(), 338
- ArrayList usado con HashMap, 504
- ArrayList y copia en profundidad, 812
- ArrayList, 343, 348, 1352, 379, 383
 - sensible a tipos., 341
- arrays asociativos (Mapas), 330
 - multidimensionales, 164
- Arrays.asList(), 395
- Arrays.fill(), 318
- Arrayas.binarySearch(), 326
- ArrayList add(), 338
- asignación de objetos, 80
- asignación
 - de llamadas a métodos. 227
 - asignación dinámica tardía o en tiempo de ejecución. 223
 - asignación dinámica. 227
 - asignación en tiempo de ejecución. 227
 - asignación tardía. 227
 - temprana, 13
- Asignación, 80
- aspectos de rendimiento, 51
- atajo, teclado, 599
- atajos de teclado, 599
- available(), 459
- barra de progreso, 611
- base, 8, 16, 97, 98
- base de datos
 - de fichero plano, 734
 - en fichero plano, 734
 - relacional, 734
 - URL, 730
 - tipos, 8
- Basic: Microoft Visual Basic, 628
- BasicArrowButton, 573
- BeanInfo: custom BeanInfo, 643
- Beans
 - archivo manifiesto, 641
 - constructor de aplicaciones. 628
 - convención de nombres, 630
 - custom BeanInfo, 643
 - editor de propiedades natural. 643
 - eventos, 628
 - EventSetDescriptors, 634
 - FeatureDescriptor, 643
 - ficheros JAR y empaquetado, 641
 - getBeanInfo(), 632
 - getEventSetDescriptptors(), 635
 - getMethodDescriptors(), 635
 - getName(), 634
 - getPropertyDescriptors(), 634

- getPropertyType, 634
- getReadMethod(), 635
- getWriteMethod(), 635
- herramienta beanbox para probar Beans, 642
- hoja de propiedades natural, 643
- Introspector, 632
- MehodDescriptors, 635
- méthod, 635
- programación visual, 628
- PropertyChangeEvent, 643
- PropertyDescriptors, 634
- PropertyVetoException, 643
- propiedad indexada, 642
- propiedades de límites. 642
- propiedades, 628
- reflectividad, 629, 631
- Serializable, 639
- y el Delphi de Borland, 628
- y el multihilo, 670
- y el Visual Basic de Microsoft, 628

B

- barra de progreso, 611
- base 8, 16, 97, 98
 - de datos de fichero plano, 734
 - en fichero plano, 734
 - relacional, 734
 - URL, 730
 - tipos, 8
- Basic: Microoft Visual Basic, 628
- BasicArrowButton, 573
- BeanInfo: custom BeanInfo, 643
- Beans y el Delphi de Borland, 628
 - y el Visual Basic de Microsoft, 628
 - constructor de aplicaciones. 628
 - convención de nombres, 630
 - custom BeanInfo, 643
 - editor de propiedades natural. 643
 - eventos, 628
 - EventSetDescriptors, 634
 - FeatureDescriptor, 643
 - fichero manifest, 641
 - ficheros JAR y empaquetado, 641
 - getBeanInfo(), 632
 - getEventSetDescriptptors(), 635
 - getMethodDescriptors(), 635
 - getName(), 634
 - getPropertyDescriptors(), 634
 - getPropertyType, 634
 - getReadMethod(), 635
 - getWriteMethod(), 635
 - herramienta beanbox para probar Beans, 642
 - hoja de propiedades natural, 643
 - Introspector, 632
 - MehodDescriptors, 635
 - método, 635
 - programación visual, 628
 - PropertyChangeEvent, 643
 - PropertyDescriptors, 634
 - PropertyVetoException, 643
 - propiedad indexada, 642
 - propiedades de límites. 642
 - propiedades, 628
 - reflectividad, 629, 631
 - Serializable, 639
 - y el multihilo, 670
- Beck, Kent, 862
- biblioteca: creador vs. programador cliente, 169
 - diseño, 169
 - uso, 170
- Bill Joy, 85
- binario: números
 - operadores, 89
- binarySearch(), 326
- bind(), 770
- bit a bit: AND, 96
 - NOT ~, 90
 - operador AND (&), 89
 - operador OR (|), 89
 - operadores, 89
 - OR EXCLUSIVO XOR (^), 90
 - OR, 96
- BitSet, 398
- blanco final, 212
- bloque try en las excepciones, 408
- bloqueo, 686
 - e hilos, 675
 - para multihilado, 666
 - y available(), 459
- bloqueos en E/S, 682

Boolean vs. C y C++, 87
 Bolsa, 329
 Booch, Grady, 863
 Boolean, 109
 álgebra, 89
 Booleans y casting, 97
 BorderLayout, 554
 Borland Delphi, 628
 Borland, 644
 botón, Swing, 548
 creando tu propio, 569
 radio button, 586
 Botones, 572
 Box para BoxLayout, 558
 BoxLayout, 557
 break etiquetado, 116
 Buffered Writer, 453, 459
 BufferedInputStream, 449
 BufferedOutputStream, 451
 BufferedReader, 430, 453, 458
 búsqueda en un array, 409
 ordenamiento y búsqueda en Lists, 389
 ButtonGroup, 586
 ByteArrayInputStream, 446
 ByteArrayOutputStream, 447

C

C/ C++, interactuando con, 841
 C++, 85
 constructor de copias, 823
 estrategias para una transición hacia, 49
 la clase vector, vs. array y ArrayList. 302
 plantillas, 343
 por qué tiene éxito, 48
 Standard Container Library – STL, 329
 caja
 de diálogo, 604
 de mensaje, en Swing, 591
 callbacks: y clases internas, 289
 cambio: vector de cambio, 294
 campo
 TYPE para literales de clases primitivas, 514
 para reflectividad, 525
 campos, inicializando campos en interfaces, 264
 capacidad

 inicial, de un HashMap o HashSet, 372
 de un HashMap o un HashSet, 372
 capturando
 cualquier excepción, 414
 capturando una excepción, 407
 carga de una clase, 219
 inicialización y carga, 217
 cargando ficheros .class, 172
 caso de uso, 36
 caso de uso
 ámbito, 43
 iteración, 42
 CD ROM
 del libro, XLI
 multimedia del libro, XLI
 CGI: Common-Gateway Interface, 747
 cierre, y clases internas, 289
 clase, 184
 abstracta vs. Interfaz, 260
 abstracta, 235
 abstracta, 235
 acceso, 185
 anidamiento de clases internas dentro de cualquier
 ámbito arbitrario. 273
 anónima, 272, 441
 y constructores, 276
 Array, utilidad contenedora, 307
 base, 183, 195 226
 clase base abstracta, 235
 constructores y excepciones, 199
 inicialización, 197
 interfaz de clase base, 230
 carga, 219
 collection, 301
 creadores, 5
 derivada, 226
 diagramas de herencia, 209
 envoltorio Integer, 161
 equivalencia, y instanceof /isInstance(), 520
 estilo de creación de clases, 185
 File, 439
 heredando de clases internas, 283
 heredando de una clase abstracta, 235
 inicialización de datos miembro. 150
 inicialización de miembros, 192

- inicialización y carga de clases, 218
- inicializando la clase base, 197
- inicializando la clase derivada, 197
- inicializando miembros en el momento de la definición, 151
- interna, 267, 628
 - anidando dentro de cualquier ámbito arbitrario, 273
 - anónima, 441, 551,
 - y código dirigido por tablas, 381
 - y constructores, 276
 - y código dirigido por tablas, 381
 - clase interna anónima y constructores, 276
 - clases internas static, 279
 - en métodos ámbitos, 272
 - heredando de clases internas, 283
 - identificadores y ficheros .class, 286
 - llamada hacia atrás, 289
 - private, 654
 - referencia oculta al objeto de la clase envoltorio, 279
 - referenciando al objeto clase externo, 281
 - y conversión hacia arriba, 270
 - y sistemas de control, 291
 - y super, 284
 - y superposición, 284
 - y Swing, 561, 562
- internas privadas, 294
- jerarquías de clases y manejo de excepciones, 434
- literal clase, 517
- múltiplemente anidada, 282
- múltiplemente anidada, 282
- navegador, 185
- orden de inicialización, 153
- palabra clave, 8
- referenciando al objeto de la clase externa en una clase interna, 281
- subobjeto, 197
- clases
 - de sólo lectura, 827
 - finales, 216
 - identificadores y ficheros .class, 286
 - internas en métodos y ámbitos, 272
 - y conversiones hacia arriba, 270
 - y super, 284
 - y superposición, 284
 - y Swing, 561/562
 - contenedoras, utilidades para, 332
 - internas private, 294
 - oyente, 628
- Class, 575
 - clases internas estáticas, 279
 - forName(), 513, 566
 - getClass(), 415
 - getConstructors(), 527
 - getInterfaces(), 524
 - getMethods(), 527
 - getName(), 524
 - getSuperclass(), 524
 - is Instance(), 519
 - is Interface(), 524
 - newInstance(), 524
 - printInfo(), 524
- ClassCastException, 251, 515
- classpath y rmic, 772
- Classpath, 172, 543
- cliente, red, 713
- clone(), 805
 - eliminando / desconectando la clonabilidad, 818
 - soportando clonar clases derivadas, 818
 - y composición, 810
 - y herencia, 816
 - super.clone(), 808, 821
- CloneNotSupportedException, 808
- close(), 458
- Codebase, 541
- código
 - de uso de hash, 361, 370
 - dirigido por tablas, y clases internas anónimas, 381
 - no-Java, invocación, 841
 - estándares de codificación, 851
 - llamando a código no - Java, 841
 - organización, 179
 - reutilización, 191
- cola, 328, 357
- colisión: nombre, 174
- colisiones
 - de nombres al combinar interfaces, 260
 - de nombres, 174
 - durante el uso de hash, 370
- Collection, 329

- Collections, 389
- Collections.enumeration(), 396
- Collections.fill(), 331
- Collections.reverseOrder(), 324
- com.bruceekel.swing, 546
- coma flotante: verdadero y falso, 88
- comando de acción, 599
- combo box, 587
- comentarios: y documentación empotrada, 71
- Common-GatewayInterface(CGI), 747
- Comparable, 322, 359
- comparación de arrays, 322
- Comparator, 323, 359
- compareTo, en java.lang.Comparable, 322
- compilando un programa Java, 71
- complemento a dos con signo, 94
- componente, y JavaBeans, 629
- componentes, Swing, usando HTML con, 610
- composición
 - biblioteca de compresión, 465
 - combinando composición y herencia, 199
 - eligiendo composición vs. Herencia, 205
 - vs. Herencia, 210, 495
 - y clonación, 810
 - y diseño, 246
- Composición, 191
- comprobación de límites, array, 161
- concepto
 - elevado, 36
 - alto, 35
- conceptos básicos de programación orientada a objetos (POO), 1
- ConcurrentModificationException, 393
- condición
 - de muerte, y finalize(), 146
 - excepcional, 406
- Conectable, Look & Feel, 617
- conferencia, Software Development Conference, XXXIV
- conjuntos de constantes seguras en tipos, 264
- Consola: framework de com.bruceekel.swing para visualizado Swing, 545
- const, en C++, 832
- constante
 - de tiempo de compilación, 210
 - constante de tiempo de compilación, 210
 - constantes implícitas, y String, 832
 - grupos de valores constantes, 263
- constructor, 127
 - cláusula de construcción estática, 157
 - comportamiento de métodos polimórficos dentro de los
 - de copia de C++, 823
 - de la clase base, 240
 - inicialización durante la herencia y la composición, 199
 - invocando desde otros constructores, 138
 - llamando a constructores de clase base con parámetros, 198
 - orden de llamadas a constructor con herencia, 238
 - para reflectividad, 525
 - por defecto, 131, 136
 - sin sintetizando un constructor por defecto, 198
 - sin argumentos, 131
 - sin parámetros, 131
 - valor de retorno, 129
 - y clases internas anónimas, 272
 - y excepciones, 429
 - y finally, 430
 - y polimorfismo, 238
 - y sobrecarga, 129
- constructores, 244
 - de clase base y excepciones, 199
 - de IGUs, 536
 - por defecto, 131
- consultoría y guiado proporcionados por Bruce Eckel. XLIV
- contenedor
 - clase, 301, 328
 - de primitivas, 305
- contenedora y clases internas, 284
- contenedores
 - de fallo rápido, 392
 - synchronized, 392
- continue etiquetado, 116
- control: acceso, 6
- controlando el acceso, 188
- conversion, 136
 - automática, 193
 - de tipos, 192
 - de float o double a entero, truncamiento, 123

- extensora, 97
- hacia abajo, 29 , 209, 249
- conversión hacia abajo consecuidad de tipos en la identificación de tipos en tiepmo de ejecución, 514
- hacia arriba, 14, 223, 510
 - clases internas y conversión hacia arriba, 270
- reductora, 96, 136
- conversiones
 - y contenedores, 338
 - y tipos primitivos, 110
- cookies,
 - y servlets, 752
 - y JSP, 766
- copia
 - bit a bit, 808
 - en profundidad, 804, 810
 - utilizacion de la serialización para llevar a cabo y los ArrayLists, 812
 - superficial, 804, 810
- copias en profundidad, 814copiando un array, 320
- CORBA, 773
- correspondencia tadría 13, 223, 227
- cortocircuito y operadores lógicos, 88
- costes
 - de arranque, 51
 - comienzo, 51
- costructor por defecto, acceso igual que la clase, 528
- CRC, tarjetas clase-responsabilidad-colaboración, 39
- CRC32, 467
- CharArrayReader, 452
- CharArrayWriter, 584
- check box, 465
- CheckedInputStream, 465
- CheckedOutputStream, 467
- Checksum

D

- database: Java DataBase Connectivity (JDBC), 729
- DatabaseMetaData, 738
- DataFlavor, 622
- Datagrama, 726
- DataInput, 455
- DataInputStream, 449, 453, 458, 460
- DataOutput, 455
- DataOutputStream, 451, 454, 460

- Datos
 - equivalencia a clase, 4
 - final, 210
 - inicialización estática, 154
 - tipos de datos primitivos y uso con operadores, 100
- débil: lenguaje débilmente tipificado, 13
- DefaultMutableTreeNode, 614
- defaultReadObject(), 484
- DefaultTreeModel, 614
- defaultWriteObject(), 484
- DeflaterOutputStream, 465
- Delphi, de Borland, 628
- Demarco, Tom, 863
- derechos de acceso de una clase interna, 277
- derivada: clase derivada, 226
- derivados: tipos, 8
- desacoplamiento através del polimorfismo, 223
- desacoplamiento: vía polimorfismo, 14
- desarrollo incremental, 207
- desbordamiento: y tipos primitivos, 109
- desplazamiento de la pantalla en Swing, 553
- destroy(), 442
- destructor: Java no tiene, 201
- Destructor, 141, 142, 423
- Devolución
 - sobrecarga del valor de retorno, 136
 - valor devuelto por un constructor, 129
- devolver: un array, 306
- diagrama
 - de eherencias, 15
 - caso de uso, 37
 - diagramas de herencia de clases, 209
 - herencia, 15
- diálogo
 - etiquetado, 59
 - fichero, 608
- diálogos de Archivo, 608
- Dibujado de líneas en Swing, 601
- Dibujos, 601
- Diccionario, 361
- dinámica: correspondencia, 223, 227
- dinámico: cambio de comportamiento con composición, 247
- dirección del bucle IP local, 714
- directorio

- creando directorios y rutas, 443
- listador, 439
- y paquetes, 178
- diseño, 248
 - de bibliotecas, 169
 - de jerarquías de objetos, 220
 - y composición, 246
 - y errores, 189
 - y herencia, 246
 - análisis y diseño orientados a objetos, 33
 - añadiendo nuevos métodos a un diseño, 189
- dispose(), 604
- disposición
 - controlando la disposición con gestores, 554
 - controlando la disposición con gestores, 554
- dispositivos hardware, interactuando con, 841
- División, 82
- documentación: comentarios & documentación empotrada, 71
- Domain Name System (DNS), 712
- double, marcador de valor literal (D), 98
- do-while, 112

E

E/S

- available(), 459
- desde la entrada estándar, 462
- pushBack(), 505
- biblioteca de compresión, 465
- biblioteca, 439
- bloqueo en E/S, 682
- bloqueo, y available(), 459
- BufferedInputStream, 449
- BufferedOutputStream, 451
- BufferedReader, 430, 453, 458
- BufferedWriter, 453, 459
- ByteArrayInputStream, 446
- ByteArrayInputStream, 447
- características de los ficheros, 443
- clase File, 439
- close(), 458
- configuraciones de E/S típicas, 455
- controlando el proceso de serialización, 476
- CharArrayReader, 452
- CharArrayWriter, 452

- CheckedInputStream, 465
- CheckedOutputStream, 465
- DataInput, 455
- DataInputStream, 449, 453, 458, 460
- DataOutput, 455
- DataOutputStream, 451, 454, 460
- DeflaterOutputStream, 451, 454, 460
- directorio, creación de directorios y trayectorias, 443
- e hilos, bloqueo, 675
- entrada de consola, 458
- entrada, 445
- Externalizable, 477
- File, 446, 454, 506
- File.list(), 439
- FileDescriptor, 446
- FileInputStream, 458
- FileInputStream, 447
- FilenameFilter, 439, 504
- FileOutputStream, 447
- FilterReader, 453
- FilterWriter, 453
- flujo entubado, 682
- GZIPInputStream, 465
- GZIPOutputStream, 447
- InflaterInputStream, 465
- InputStream, 445, 718
- InputStreamReader, 452, 718
- internacionalización, 452
- leerLine(), 433, 453, 460, 463
- LineNumberInputStream, 456
- LineNumberReader, 454
- listador de directorios, 440
- mark(), 455
- mkdirs(), 445
- nextToken, 505
- ObjectOutputStream, 472
- OutputStream, 445, 447, 718
- OutputStreamWriter, 452, 718
- persistencia ligera, 471
- PipedInputStream, 447
- PipedOutputStream, 447
- PipedReader, 453
- PipedWriter, 453, 718
- PrintStream, 450
- PrintWriter, 453, 459, 718

- PushBackReader, 453
- PushbackInputStream, 449
- RandomAccessFile, 454, 460
- read(), 445
- readChar(), 460
- readDouble(), 460
- Reader, 445, 451, 452, 718
- readExternal(), 476
- readObject(), 472
- redirigiendo la E/S estándar, 464
- renameTo(), 445
- reset(), 455
- salida, 445
- seek(), 455, 461
- SequenceInputStream, 454
- Serializable, 476
- setErr(PrintStream), 464
- setIn(InputStream), 464
- setOut(PrintStream), 464
- StreamTokenizer, 453, 492, 504, 528
- StringBuffer, 446
- StringBufferInputStream, 446
- StringReader, 453, 458
- StringWriter, 453
- System.err, 462
- System.in, 458
- System.out, 462
- tubería, 446
- Unicode, 452
- write(), 453
- writeBytes(), 460
- writeChars(), 460
- writeDouble(), 460
- writeExternal, 476()
- writeObject(), 472
- Writer, 445, 451, 452, 718
- ZipEntry, 469
- ZipInputStream, 465
- ZipOutputStream, 465
- East, BorderLayout, 554
- editor, creación de uno usando el JTextPane de Swing.
583
- efecto lateral, 79, 85, 133, 802
- eficiencia
 - al usar la palabra clave synchronized, 670
 - e hilos, 650
 - y final, 216
 - y arrays, 301, 302
- EJB, 780
- Ejecutable, Hilo, 675
- ejecutando un programa Java, 71
- ejemplo de reflectividad, 573, 574
- ejemplos de componentes Swing, 571, 572
- elegancia, al programar, 45
- Encapsulación, 184
- encontrando ficheros .class durante la carga, 172
- Enterprise JavaBeans (EJB), 780
- entornos de programación visual, 536
- entrada
 - a la consola, 458
 - estándar: leyendo de la entrada estándar, 462
 - entrada de la consola, 458
- enum, grupos de valores constantes en C & C++, 263
- Enumeración, 396
- enviando un mensaje, 4
- envoltorio, manipulando la inmutabilidad de clases en-
voltorios primitivos, 827
- equals(), 86, 360
 - vs. ==, 497
 - y estructuras de datos con hashing, 367
 - superposición para HashMap, 366
- equivalencia
 - de objetos vs. equivalencia de referencias, 808, 809
 - ==, 85
 - de objetos, 85
- error
 - informando de errores del libro, XLIV
 - manejo con excepciones, 405
 - recuperación, 435
 - stream estándar de errores, 410
- errores, y diseño, 189
- escenario, 37
- es-como-un, 248
- espacio
 - de solución, 2
 - del problema, 2, 207
 - problema, 2
 - solución, 2
- espacios de nombres, 170
- especialización, 206

- especificación
 - del sitema, 36
 - especificadores de acceso, 6, 109, 178
 - excepción, 414
 - estándares: estándares de codificación, XLIII, 59
 - estilo de creación de clases, 184
 - es-un, 248
 - relación, herencia y conversión hacia arriba, 208
 - Etiqueta, 116
 - de archivo para ficheros HTML y JAR, 622
 - etiquetas de archivo de applet para ficheros HTML y JAR, 622
 - evento
 - multicast, y JavBeans, 670, 671
 - evento multifusión y JavaBeans, 671
 - JavaBeans, 629
 - modelo de eventos de Swing, 623
 - multicast, 624
 - orden de ejecución, 625
 - respondiendo a un evento Swing, 550
 - sistema dirigido por eventos, 292
 - unidifusión, 624
 - eventos
 - multifusión, 624
 - y listeners, 562
 - EventSetDescriptors, 635
 - evolución, en desarrollo de programas, 43
 - excepción:
 - aspectos de diseño, 432
 - bloque try, 408
 - capturando cualquier excepción., 414
 - capturando el punto de orientación de la excepción, 418
 - capturando una excepción, 407
 - clase Error, 419
 - clase Excepción, 419
 - constructores, 430
 - creando la tuya propia, 409
 - emparejamiento de excepciones, 433
 - especificación, 413
 - FileNotFoundException, 432
 - finally, 422
 - jerarquías de clases, 434
 - lanzando una excepción, 406
 - manejador de excepciones, 408
 - manejador, 405, 406
 - manejo de excepciones, 401
 - manejo, 201
 - periendo una excepción, error frecuente, 426
 - región protegida, 408
 - relanzando una excepción. 415
 - restricciones, 427
 - terminación vs. Reanudación, 409
 - Throwable, 414
 - try, 423
 - usos típicos de las excepciones, 435
 - y constructores de clase base, 199
 - y constructores, 22
 - y herencia, 427, 433
 - excepciones: y JNI, 848
 - exception:
 - NullPointerException, 420
 - printStackTrace(), 416
 - RuntimeException, 420
 - executeQuery(), 732
 - extendiendo una clase durante la herencia, 10
 - extensible: programa, 230
 - extensión
 - cero, 90
 - de signo, 90
 - e interfaces, 263
 - herencia pura vs. Extensión, 247
 - signo, 90
 - zero, 90
 - extenuación de la memoria, solución vía References, 315, 316
 - Externalizable, 477
 - Externalizable: un enfoque alenrativo al uso de, 482
 - Extiende, 183, 196, 248
 - Extreme Programming (XP), 45, 862
-
- ## F
-
- factor de carga de un HashMap o HashSet, 372
 - factores de azar, 34
 - False, 87
 - fallos frecuentes en el uso de operadores, 95
 - FeatureDescriptor
 - fichero
 - JAR, 171
 - características de ficheros, 443

ficheros de salida incompletos, errores y ráfagas, 460
 FIFO, 357
 File, 446, 454, 506
 File Transfer Protocol (FTP), 543
 File.list(), 439
 FileDecritptor, 446
 FileReader, 458
 FilenameFilter, 439, 652
 FileNotFoundException
 FileOutputStream, 447
 FileReader, 430, 453
 FileWriter, 453, 459
 FilterInputStream, 446
 FilterOutputStream, 447
 FilterReader, 453
 FilterWriter, 453
 final
 con referencias a objetos, 210
 y eficiencia, 216, 217
 y private, 214
 y static, 210
 blancos finales, 212
 clases, 216
 datos, 210
 método, 227
 métodos, 214, 446
 tipos primitivos estáticos, 211
 Final, 255
 finalize(), 141, 433
 y herencia, 240
 y super, 243
 llamando directamente, 142
 oren de finalización de objetos, 244
 Finally, 201, 203
 fallos frecuentes, 426
 flavor, portapapeles, 619
 FleInputStream, 446
 float, marcador de valor literal (F), 98
 FlowLayout, 555
 formación, 49
 forName(), 513, 566
 FORTRAN, 98
 Fowler, Martin, 34, 43, 862
 framework
 de aplicación, y applets., 538

 de control, y clases internas, 291
 de muestra, para Swing, 545
 framework de aplicación y applets, 538
 framework de control y clases internas, 291
 friendly, 179, 270
 e interfaz, 255
 y protected, 206
 menos accesible que protected, 243
 FTP: File Transfer Protocol (FTP), 543
 función
 de hashing perfecta, 370
 de uso de hash, 370
 función miembro, 5
 superposición, 11
 funciones JNI, 844

G

generador aleatorio de números, valores producidos por,
 123
 Generador, 331
 geString(), 732
 get(),
 ArrayList, 338, 343
 HashMap, 364
 getBeanInfo(), 632
 getBytes(), 459
 getClass(), 415, 522
 getConstructor(), 575
 getContentPane(), 540
 getContents(), 622
 getEventSetDescriptors(), 635
 getFloat(), 732
 getInputStream(), 715
 getInt(), 732
 getInterfaces(), 524
 getMethodDescriptors(), 635
 GetMethods, 527
 getModel(), 615
 getName(), 524, 635
 getOutputStream(), 715
 getPriority(), 690
 getPropertyDescriptors(), 634
 getPropertyType(), 634
 getSelectedValues(), 588
 getState(), 598

`getSuperclass()`, 524
`getTransferData()`, 622
`GetTransferDataFlavors`, 622()
`getWriteMethod()`, 635
 Glass, Robert, 863
 goto: falta de goto en Java, 116
`GridBagLayout`, 557
`GridLayout`, 556, 703
 guía: y formación, 51, 52
 guías

- desarrollo de objetos, 41
- estándares de programación, 851

H

`HaasMap`, 360, 379, 571
`hashCode()`, 358, 361

- y estructuras de datos con uso de hash, 367
- aspectos a tener en cuenta al escribir, 373
- superposición para `HashMap`, 366

`HashMap` usado con `ArrayList`, 504
`HashSet`, 358, 384
`Hashtable`, 388, 397
`hasNext()`, iterador, 344
 heredando de una clase abstracta, 235
 herencia

- de clases internas, 283
- de una clase abstracta, 235
- múltiple, en C++ y Java, 258
- vs. Composición, 209
- y clonado
- y final, 216
- y final, 240
- y sobrecarga de métodos vs. Superposición, 204
- y `synchronized`, 674
- combinando composición y herencia, 199
- diagramas de herencia de clases, 209
- diseñando con herencia, 246
- eligiendo composición vs. Herencia, 205
- especialización, 206
- extendiendo interfaces con herencia, 262
- extendiendo una clase durante, 10
- herencia múltiple en C++ y Java, 258
- herencia pura vs. Extensión, 247

inicialización con herencia, 217
 Herencia, 8, 183, 191, 194, 223
 herramienta `beanbox` para probar Beans, 642
 Hexadecimal, 97
 hilos demonio, 659
 HTML, 747

- en componentes Swing, 610
- nombre, 658
- parámetro, 658
- valor, 658

I

Icono, 575
 IDL, 774
`Idltojava`, 776
 IGU: interfaz gráfico de usuario, 291, 535
`IllegalMonitorStateException`, 681
`ImageIcon`, 575
 implementación, 5

- e interfaz, 205, 256
- separación, 184, 185
- ocultación, 5, 184, 270
- separación del interfaz y la implementación, 562

imprimiendo arrays, 309
`indexO:String`, 441, 527
`InflaterInputStream`, 465
 informando de errores en el libro, XLIV
 Inicialización

- agregada de arrays, 159
- con herencia, 218
- de arrays, 159
- de clase derivada, 197
- de constructores durante la herencia y la composición, 199
- de datos miembros de clases, 150
- de instancias, 158, 276
 - no static, 158
- de variables de métodos, 150
- perezosa, 193
- inicializando con el constructor, 127
- inicializando miembros de clase en el momento de la definición, 151
- miembro de la clase, 192
- orden de inicialización, 153, 166
- perezosa, 193

- static, 219
- y carga de clases, 217
- inicializadores miembro, 240
- inmodificable, haciendo que una Collection o Map sea inmodificable, 391
- InputStream, 445, 718
- InputStreamReader, 452, 718
- insertNodeInfo()
- instanceof: instanceof dinámico, 519
- instancia
 - de una clase, 2
 - inicialización de instancias, 276
- Integer: parseInt(), 608
- interactuando con dispositivos hardware, 841
- interbloqueo, utiilado, 680, 686
- Interface Definition Language (IDL), 774
 - user, 38
- interfaces: colisiones de nombres al usar interfaces, 260
- interfaz
 - anidando interfaces dentro de clases y otros interfaces, 265
 - cloneable, 806
 - común, 235
 - de un objeto, 3
 - de usuario de respuesta rápida, con hilos, 647
 - de usuario e hilos, para aumentar la velocidad de respuesta, 652
 - de usuario, 38
 - e implementación, separación, 184
 - gráfico de usuario (IGU), 291, 535
 - Runnable, 655
 - vs. Abstract, 260
 - vs. implementación, 205
 - y herencia, 262
 - conversión hacia arriba a interfaz, 258
 - definiendo la clase, 45
 - implementación, separación de, 6
 - inicializando campos de interfaces, 264
 - interfaz Cloneable usado como indicador, 806
 - común, 235
 - de la clase base, 230
 - gráfico de usuario (IGU), 291, 535
 - private, como interfaces anidados, 267
 - Runnable, 655
 - separación de interfaz e implementación, 562

- internacionalización en la biblioteca de E/S, 452
- Internet:
 - Internet Protocol, 712
 - Internet Service Provider (ISP), 542
- interrupt(), 689
- InterruptedException, 649
- Intranet, 538
 - y applets, 538
- Introspector, 632
- IP (Internet Protocol), 712
- isDaemon(), 659
- isDataFlavorSupported(), 622
- IsInstance, 519
 - (), 524
- ISP (Internet Service Provider), 543
- iteración, en desarrollo de programas, 42
- Iterador, 343, 349, 379
 - (), 349
 - hasNext(), 343
 - next(), 344

J

- Jacobsen, Ivar, 863
- Japplet, 554
- JAR, 641
 - etiqueta de archivo para ficheros HTML y JAR, 622
 - empaquetando applets para optimizar la carga, 622
 - ficheros JAR y el classpath, 174
- Java, 54
 - AWT, 535
 - Server Pages (JSP), 757
 - Virtual Machine, 512
 - y las set-top-boxes, 89
 - y los punteros, 799
 - biblioteca de contenedores, 329
 - compilando y ejecutando un programa, 71
 - herramienta de comprobación del uso de mayúsculas en el código fuente, 498
 - seminarios públicos de Java, XXXIV
 - versiones, XLII
- Java 1.1: streams de E/S, 452
- JavaBeans: ver Beans, 628
- Javac, 71
- JavaFoundation Classes (JFC/Seing), 535
- Javah, 842

JButton, 575
 Swing, 548
 JComboBox, 587
 Jcomponent, 577, 601
 JcheckBoxMenuItem, 594
 JCheckboxMenuItem, 598
 JDBC
 Java DataBase Connectivity, 729
 base de datos en fichero plano, 734
 base de datos relacional, 734
 createStatement(), 732
 DatabaseMetaData, 738
 executeQuery(), 732
 getFloat(), 732
 getInt(), 732
 getString(), 732
 join, 734
 procedimientos SQL almacenados, 736
 ResultSet, 732
 Statement, 732
 Structured Query Language (SQL), 729
 URL de base de datos, 730
 Jdialog, 604
 JDK: descarga e instalación, 71
 JFC: Java Foundation Classes (JFC/Swing), 535
 JfileChooser, 608
 Jframe, 546, 554
 Jini, 791
 JIT: compiladores Just-In Time, 53
 Jlist, 588
 Jmenu, 593, 599
 JmenuBar, 593, 599
 JMenuItem, 575, 593, 598, 599, 601
 JNICALL, 843
 JNIEnv, 845
 JNIEXPORT, 843
 Join, 734
 JoptionPane, 591
 Jpanel, 554, 573, 601, 704
 JpopupMenu, 599
 JprogressBar, 612
 JradioButton, 575, 586
 JscrollPane, 553, 581, 590, 614
 Jslider, 612
 JSP, 756

JtabbedPane, 590
 Jtable, 615
 JtextArea, 552, 620
 JtextField, 550, 577
 JtextPane, 583
 JtoggleButton, 573
 Jtree, 612
 JVM (Java Virtual Machine), 512

K

keySet(), 389
 Koenig, Andrew, 852

L

lanzando una excepción, 406
 lenguaje
 de programación Perl, 548
 de programación Simula, 3
 leyendo de la entrada estándar, 462
 libro:
 actualizaciones del libro, XLIII
 información de errores, XLIV
 LIFO, 356
 ligadura temprana, 13, 227
 ligeros: componentes Swing, 537
 limpieza
 con finally, 423
 llevándola a cabo, 142
 y el recolección de basura, 201
 LineNumberInputStream, 449
 LineNumberReader, 453
 LinkedList, 384
 list boxes, 588
 List, 302, 328, 329, 352, 588
 búsqueda y ordenación, 389
 lista
 desplegable, 587
 enlazada, 328
 lista desplegable, 587
 listener
 adapters, 568
 interfaces, 567
 y eventos, 562
 Lister, Timothy, 863
 ListIterator, 352

- literal:
 - double, 98
 - float, 98
 - literal e clase, 513, 517
 - long, 98
 - valores, 97
 - localhost y RMO, 770
 - localhost, 714
 - logaritmos
 - naturals, 98
 - logaritmos naturales, 98
 - lógico:
 - AND, 96
 - operador y cortocircuito, 88
 - operadores, 87
 - OR, 96
 - long, marcador literal del valor (L), 98
 - longitud,
 - de arrays, 302
 - miembro array, 160
 - Look & Feel: conectable, 617
 - llamada hacia atrás, 441, 550
 - llamadas a métodos inline, 214
-
- M**
-
- main(), 196
 - manejadorr, excepció, 408n
 - manejar, constante, 210
 - mantenimiento, programa, 43
 - Map, 302, 328, 329, 360, 386
 - Map.Entry, 368
 - mapa, 361
 - mark(), 455
 - Math.random(), 363
 - Math.random(): valores producidos por, 123
 - max(), 390
 - mayor
 - o igual que, 85
 - que (>), 85
 - mayúsculas: herramienta de comprobación del uso de
 - mayúsculas en código fuente Java, 498
 - menor
 - o igual que, 85
 - que, 85
 - mensaje, envío, 4
 - menu: JPopupMenu, 599
 - menus: JDialog, JAppl, JFram, 593
 - metacalse, 512
 - Method, 635
 - para reflectividad, 525
 - MethodDescriptors, 635
 - método:
 - añadiendo más métodos a un diseño, 189
 - clases internas en métodos & ámbitos, 272
 - comportamiento de métodos polimórficos dentro de constructores., 244
 - distinguiendo métodos sobrecargados, 131
 - herramienta de búsqueda, 564
 - inicialización de variables de método, 151
 - llamada a un método polimórfico, 223
 - llamadas inline a métodos, 214
 - método sincronizado y bloqueo, 675
 - métodos final, 214
 - métodos protected, 206
 - pasando una referencia un método, 799
 - private, 246
 - recursivo, 345
 - resolución de llamadas a métodos, 227
 - static, 140
 - uso de alias durante llamadas a métodos, 81
 - uso de alias durante una llamada a un método, 800
 - metodología: análisis y diseño, 33
 - métodos
 - opcionales, en los contenedores de Java 2, 393
 - synchronized, bloques synchronized, 669
 - Meyers, Scott, 5
 - Microsoft, 644
 - Visual Basic, 628
 - miembro:
 - función miembro, 5
 - objeto, 7
 - min(), 390
 - mkdirs(), 445
 - mnemónicos (atajos de teclado), 599
 - módulo, 82
 - de eventos Swing, 623
 - de eventos, Swing, 561
 - monitor, para multihilo, 666
 - muerto, Thread, 675
 - multicast, 640

multihilado, 647, 721
 multihilo
 y contenedores, 392
 y Java Beans, 670
 bloqueo, 675
 cuándo usarlo, 708
 decidiendo qué métodos sincronizar, 674
 desventajas, 708
 interbloqueo, 680
 Runnable, 701
 multiplicación, 82
 MultiStringMap, 504
 multitarea, 647

N

Name, palabra clave HTML, 658
 Naming:
 bind(), 770
 rebind(), 771
 unbind(), 771
 native meted interface (NMI) en Java 1.0, 841
 navegación con teclado, y Swing, 537
 navegador: navegador de clases, 185
 newInstance(), 575
 reflectividad, 524
 next(), Iterador, 344
 nextToken(), 505
 NMI: Kava 1.0 Native Method Interface, 841
 nombre, 541
 del constructor, 127
 creando nombres de paquete únicos, 172
 North, BorderLayout, 554
 NOT: lógico (!), 87
 notación exponencial, 98
 notify(), 675
 notifyAll(), 675
 notifyListeners(), 674
 nuevo operador, 140
 null, 59
 recolector de basura, permitiendo limpieza, 294
 NullPointerException, 420
 números,
 binario, 98
 binarios, impresión, 93

O

Object, 302
 Object.clone(), 808
 Object:
 clase raíz estándar, herencia por defecto, 194
 clone(), 805, 808
 getClass(), 522
 hashCode(), 361
 métodos wait() y notify(), 680
 ObjectOutputStream, 472
 gestión de obstáculos, 51
 objeto:, 2
 Class, 157, 488, 512, 666
 generator, para rellenar arrays y contenedores, 308
 asignación y copia de referencias, 80
 asignando objetos copiando referencias, 80
 bloqueo, para multihilo, 666
 cinco etapas del diseño de objetos, 41
 creación, 128
 equivalencia vs. equivalencia de referencias, 86
 equivalencia, 85
 final, 210
 guías para el desarrollo de objetos, 41
 interfaz hacia, 3
 método equals(), 86
 miembro, 7
 objeto Class, 488, 512, 666
 objeto/lógica de negocio, 625
 objetos inmutables, 827
 orden de finalización de los objetos, 244
 proceso de creación, 156
 serialización, 471
 uso de alias, 81
 web de objetos, 472, 804
 /lógica de negocio, 625
 accesibles y recolección de basura, 376
 inmutables, 827
 los arrays son objetos de primera clase, 302
 obstáculos de gestión, 51
 Octal, 98
 ocultamiento, implementación, 5, 184
 ODBC, 730
 OMG, 773
 onda seno, 601

- operador, 79
 - coma, 95, 114
 - complemento a uno, 90
 - de auto decremento, 84
 - de auto incremento, 84
 - de decremento, 84
 - de desplazamiento a la derecha (>>), 90
 - de desplazamiento a la izquierda, 90
 - de incremento, 84
 - de indexación [], 159
 - ternario, 94
 - y primitivos, array, 161
 - + y +=, sobrecarga para Strings, 195, 196
 - +, para String, 832
 - == y !=, 808
 - binario, 89
 - coma, 95
 - complemento a uno, 90
 - conversión, 96
 - de bits, 89
 - desplazamiento, 90
 - fallos frecuentes, 95
 - lógico, 87
 - operador coma, 114
 - operador de indexación [], 159
 - operadores lógicos y atajos, 88
 - precedencia, 79
 - precedencia, mnemónico, 99
 - relacional, 85
 - sobrecarga para String, 832
 - sobrecarga, 95
 - ternario, 94
 - unario, 84, 89
- operadores
 - booleanos que no funcionan con boolean, 85
 - de conversión, 96
 - de desplazamiento, 90
 - de Java, 79
 - matemáticos, 82
- OR, 96
 - (| |), 87
- orden
 - alfabético vs. Lexicográfico, 326
 - de finalización de objetos, 244
 - de inicialización, 217, 153, 245

- de llamadas a constructor, con herencia, 238
- ordenación, 322
 - y búsqueda en Listas, 389
- ordenado lexicográfico vs. Alfabético, 325, 326
- organización, código, 179
- orientación a objetos: conceptos básicos y objeto de la programación orientada a objetos (POO), 1
- OutputStream, 445, 447, 718
- OutputStreamWriter, 452, 718

P

- paintComponent(), 601, 608
- palabra
 - clave finally(), 422
 - clase instanceof, 515
 - clave abstract, 235
 - clave break, 114
 - clave catch, 408
 - clave continue, 144
 - clave default, en una sentencia switch, 121
 - clave else, 110
 - clave extends, 195
 - clave final, 210
 - clave for, 113
 - clave implements, 256
 - clave interfaz, 255
 - clave super, 196
 - clave switch, 120
 - clave this, 137
 - clave throw, 407
 - clave transient, 480
- palabras clave: class, 3, 8
- paquete, 170
 - por defecto, 181
 - acceso, y friendly, 179
 - creando nombres de paquete únicos, 172
 - nombres, uso de mayúsculas, 67
 - paquete por defecto, 181
 - visibilidad, friendly, 270
 - y applets, 543
 - y estructura de subdirectorios, 178
- parálisis, análisis, 34
- param, palabra clave HTML, 658
- parámetro:
 - constructor, 128

- final, 213
- lista variable de parámetros (cantidad y tipo de los parámetros desconocida), 163
- paso de una referencia a un método., 799
- parámetros del constructor, 128
- parseInt(), 608
- paso:
 - pasando una referencia a un método, 799, 800
 - paso por valor, 802
- Patrón
 - Comando, 441
 - Command Pattern, 441
 - de diseño decorador, 448
- patrones
 - de diseño, 44, 50, 187
 - de diseño: decorador, 448
 - de diseño: singleton, 187
 - diseño, 441, 450
 - patrones de diseño, 187
- pegamento, en BorderLayout, 557
- persistencia, 485
 - ligera, 471
- petición, en POO, 4
- PhantomReference, 375
- pintar en un JPanel en Swing, 601
- piped
 - stream, 683
 - streams, 462
- PipedInputStream, 446
- PipedOutputStream, 446, 447
- PipedReader, 452
- PipedWriter, 452
- planificación, 38
 - desarrollo de software, 35
- plantilla, en C++, 343
- Plauger, P.J., 863
- polígono:
 - ejemplo y run-time type identification, 509
 - ejemplo, 8, 228
- polimorfismo, 121, 223, 252, 510, 531
 - y constructors, 238
 - comportamiento de métodos polimórficos dentro de los constructors, 244
- POO, 184
 - análisis y diseño, 33
 - características básicas, 2
 - conceptos básicos de programación orientada a objetos, 1
 - lenguaje de programación Simula, 3
 - protocolo, 255
 - sustituibilidad, 2
- portabilidad en C, C++ y Java, 99
- portapapeles
 - del sistema, 619
 - portapapeles del sistema, 619
- posición, absoluta al disponer componentes Swing, 557
- prámeto final, 213, 442
- precedencia: mnemónico de precedencia de operadores, 99
- prerrequisitos, para este libro, 1
- principio de sustitución, 11
- println(), 524
- println(), 345
- printStackTrace(), 441, 416
- PrintStream, 451
- PrintWriter, 453, 460, 718
- prioridad,
 - traed???, 689
 - prioridad por defecto para un grupo de Threads, 693
- private, 6, 169, 179, 181, 665
 - interfaces cuandos e anidan, 267
 - clase interna, 654
 - clases internas, 294
 - ilusión de superposición de métodos private, 214
 - métodos, 246
- procedimientos almacenados en SQL, 736
 - hash, 368
 - e hilado, 647
- Proceso hash: encadenamiento externo, 370
- programa: mantenimiento, 43
- programación
 - conceptos básicos de programación orientada a objetos (POO), 1
 - dirigida por eventos, 549
 - en red, 712
 - en red: accept(), 715
 - en red: cliente, 714
 - en red: Common Gateway Interface (CGI), 747
 - en red: conexión dedicada, 721
 - en red: datagramas, 726

- en red: dirección IP del bucle local, 714
- en red: DNS (Domain Name System), 712
- en red: `getInputStream()`, 715
- en red: `getOutputStream()`, 715
- en red: HTML, 715
- en red: identificando máquinas, 712
- en red: Internet Protocol (IP), 712
- en red: Java DataBase Connectivity, 729
- en red: localhost, 714
- en red: mostrando una página Web desde un applet, 726
- en red: multihilo, 721
- en red: probando programas sin una red, 714
- en red: protocolo no seguro, 726
- en red: protocolo seguro, 726
- en red: puerto, 715
- en red: servidor, 714
- en red: `showDocument()`, 727
- en red: sirviendo a múltiples clientes, 721
- en red: Socket, 720
- en red: sockets basados en streams, 726
- en red: Transmisión Control Protocol (TCP), 726
- en red: URL, 728
- en red: User Datagram Protocol (UDP), 726
- estándares de codificación, 851
- Extreme Programming (XP), 45, 862
- multiparadigma*, 2
- multiparadigma*, 2
 - orientación a objetos, 510
- orientada a objetos, 510
- par, 47
- par, 47
- programación orientada a eventos, 549, 550
- programador
 - cliente vs. creador de bibliotecas, 169
 - cliente, 5
- programas Java: ejecución desde el Explorador de Windows, 547
- promoción:
 - de tipos primitivos, 109
 - promoción de tipos, 98
- `PropertyChangeEvent`, 643
- `PropertyDescriptor`, 634
- `PropertyVetoException`, 643
- propiedad, 628
 - indeseada, 642
 - editor por defecto de propiedades, 643
 - propiedad indexada, 643
- propiedades, 504
 - de límites, 643
 - limitadas, 643
 - propiedades limitadas, 643
 - propiedades vinculadas, 643
- `protected`, 6, 169, 178, 183, 206
 - friendly, 206
 - es también friendly, 183
 - más accesible que friendly, 243
 - uso en `clone()`, 805
- protocolo, 255
 - orientado a la conexión, 726
 - seguro, 726
- prototipado: rápido, 44
- prototipo rápido, 44
- prueba:
 - automatizada, 46
 - Extreme Programming (XP), 45
 - prueba de unidad, 196
- pruebas de clase, 196
- `public`, 6, 169, 178, 179
 - clase y unidades de compilación, 170
 - e interfaz, 255
- puerto, 715
- puntero: exclusión de punteros en Java, 289
- punteros, and Java, 799
- pura:
 - herencia vs. extensión, 247
 - sustitución, 11
- `pushback()`, 505
- `PushBackInputStream`, 449
- `PushBackReader`, 453
- `put()`, `HashMap`, 364
- Python, 40, 54

R

- RAD (Rapid Application Development), 525
- radio button, 586
- `random()`, 363
- `RandomAccessFile`, 454, 460
- `read()`, 453
- `readChar()`, 460

- readDouble(), 460
- Reader, 453, 452, 682, 718
- readExternal(), 476
- readLine(), 433, 453, 460, 463
- readObject() con Serializable, 483
- readObject(), 472
- reanudación: terminación vs. reanudación, manejo de excepciones, 409
- rebind(), 771
- recolección
 - de basura, 140, 142
 - de basura: cómo funciona el recolector, 147
 - de basura y ejecución de métodos nativos, 48
 - de basura y limpieza, 201
 - de basura: objetos alcanzables, 375
 - de basura: orden de reclamación de objetos, 203
 - de basura: poniendo referencias a null para permitir la limpieza, 294
- recursión, inintencionada vía toString(), 345
- redirigiendo la E/S estándar, 464
- Reference, de java.lang.ref, 375
- referencia
 - hacia delante, 152
 - asignando objetos copiando referencias, 80
 - averiguando el tipo exacto de una referencia base, 511, 512
 - equivalencia de referencias vs. equivalencia de objetos, 808
 - equivalencia vs. equivalencia de objetos, 86
 - final, 210
- referenciado, referenciado hacia adelante, 152
- reflectividad, 524, 525, 564, 631
 - y Beans, 629
 - diferencia entre RTTI y reflectividad, 526
- región protegida, en manejo de excepciones, 408
- registro: registro de objetos remotos, 770
- relación
 - es-un vs. es-como-un, 11
 - es-un, herencia, 206
 - tiene un, composición, 206
- relacional:
 - base de datos, 734
 - operadores, 85
- relanzando una excepción, 416
- Remote Meted Invocation (RMI), 767
 - removeXXXListener(), 562
 - renameTo(), 445
 - rendimiento y final, 216
 - recolección de basura: forzando la finalización, 203
 - reset(), 455
 - ResultSet, 732
 - resume(), 675, 679
 - e interbloques, 686
 - abolición en Java 2, 687
 - reusabilidad, 7
 - reutilizar, 41
 - bibliotecas de clases ya existentes, 50
 - código reutilizable, 628
 - reutilizando código, 191
- RMI
 - y CORBA, 780
 - AlreadyBoundException, 771
 - bind(), 770
 - interfaz remoto, 768
 - localhost, 770
 - parámetros Serializable, 771
 - rebind(), 771
 - registro de objetos remotos, 770
 - Remote, 768
 - Meted Invocation, 767
 - RemoteException, 768, 773
 - rmic y classpath, 772
 - rmic, 771
 - rmiregistry, 770
 - RMI Security Manager, 770
 - skeleton, 771
 - stub, 771
 - TCP/IP, 770
 - unbind(), 771
 - UnicastRemoteObject, 768
- rmic, 771
- rmiregistry, 770
- RMI Security Manager, 769
- RTTI:
 - Class, 575
 - ClassCastException, 515
 - Constructor, 525
 - conversión hacia abajo segura en tipos, 514
 - conversión hacia abajo, 514
 - conversion, 511

diferencia entre RTTI y la reflectividad, 526
 Field, 525
 getConstructor(), 575
 isInstane, 519
 metacalse, 512
 Method, 525
 newInstance(), 574
 objeto Class, 511
 palabra clave instnaceof, 515
 reflectividad, 524
 run-time type identification (RTTI), 250
 usando el objeto Class, 522
 y clonado, 809
 Rumbaugh, James, 863
 runFinalizersOnExit(), 243
 Runnable, 701
 run-time
 type identification (RTTI), 250
 type identification (RTTI): cuándo usarla, 531
 type identification (RTTI): ejemplo polígono, 509
 type identification (RTTI): mal uso, 531
 RuntimeException Excepción en tiempo de ejecución,
 302, 420

S

sección
 crítica, y bloque sincronizado, 669
 sección crítica y bloque synchronized, 669
 secciones de código críticas en el tiempo, 841
 select(), 455, 461
 seguimiento de sesiones con servlets, 752
 seguridad, y restricciones de los applets, 537
 seminaries: formación proporcionada por Bruce Eckel,
 XLIV
 seminarios
 de formación proporcionados por Bruce Eckel, XLIV
 seminarios públicos de Java, XXXIV
 sentencia
 case, 121
 if-else, 94, 110
 mission, 35
 misión, 35
 separación de interfaz e implementacion, 6, 184, 562
 separando
 la implementación y el interfaz, 6
 la lógica de negocio de la lógica del IU, 625
 SequenceInputStream, 446, 454
 Serializable: 471, 476, 481, 491, 639
 readObject(), 483
 writeObject(), 483
 serialización:
 controlando el proceso de serialización, 476
 para llevar a cabo copias en profundidad, 814
 parámetros RMI, 771
 versionado, 485
 y almacenamiento de objetos, 485
 defaultReadObject(), 484
 defaultWriteObject(), 484
 servidor, 713
 servlet: 747
 ejecutando servlets con Tomcat, 756
 multihilo, 751
 servlets: seguimiento de sesión, 752
 session: y JSP, 764
 Set, 302, 328, 329, 357, 584
 setActionCommand(), 599
 setBorder(), 579
 setContents(), 621
 setDaemon(), 659
 setDefaultCloseOperation(), 547
 setErr(PrintStream), 464
 setIcon(), 577
 setIn(InputStream), 464
 setLayout(), 554
 setMnemonic(), 599
 setOut(PrintStream), 464
 setPriority(), 690
 settoolTipText(), 577
 show(), 605
 showDocument(), 727
 shuffle(), 390
 Simula-67, 184
 singleton: patron de diseño, 187
 sintaxis de inicialización de agregación dinámica para
 arrays, 305
 Sistema de aplicación, 291, 292
 sistemas multihilo, 625
 size(), ArrayList, 338
 sizeof(), falta de, en Java, 99
 skeleton, RMI, 771

- `sleep()`, 649, 664, 675, 677
- `slider`, 611
- Smalltalk, 3, 140
- sobrecarga
 - de métodos, 129
 - de operadores para Strings, 833
 - de operadores, 95
 - en valores de retorno, 136
 - vs. superposición, 203 233
 - y clases internas, 284
 - y constructores, 129
 - distinguiendo métodos sobrecargados, 131
 - falta de ocultación de nombres durante la herencia, 204
 - función, 10
 - operador `+` y `+=` sobrecargados para Strings, 195, 196
- Socket, 720
- sockets
 - basados en flujos, 726
 - basados en flujos, 726
- SoftReference, 375
- software: metodología de desarrollo, 33
- Software: Development Conference, XXXIV
- South, BorderLayout, 554
- SQL: procedimientos almacenados, 736
- SQL: Structured Query Language, 730
- Stack, 356, 397
- `stateChanged()`, 603
- Statement, 732
- static, 255
 - y final, 210
 - bloque, 157
 - clases internas, 279
 - cláusula de construcción, 157
 - cláusula, 513
 - inicialización, 219
 - de datos, 154
 - `synchronized static`, 666
 - tipos primitivos final static, 211
- STL: C++, 329
- `stop()`
 - abolición en Java 2, 686
 - e interbloqueos, 686
- stream, E/S, 445
- StreamTokenizer, 453, 492, 504, 528
- String:
 - concatenación con el operador `+`, 95
 - conversion automática de tipos, 341
 - `IndexOf()`, 441, 527
 - inmutabilidad, 831
 - métodos, 834
 - de la clase, 831
 - operador `+`, 341
 - ordenamiento lexicográfico vs. alfabético, 325, 326
 - sobrecarga de los operadores `+` y `+=`, 195, 196
 - `toString()`, 192, 340
- StringBuffer, 446
 - métodos, 836
- StringBufferInputStream, 446
- StringReader, 453, 458
- StringSelection, 621
- StringTokenizer, 495
- StringWriter, 452
- stub, RMI, 771
- subobjeto, 197, 205
- subrecarga: sobrecarga vs. superposición, 203
- substracción, 82
- super, 198
 - y `finalize()`, 243
- `super.clone()`, 805, 808, 821
- superclase, 196
- superposición vs. sobrecarga, 233
- `suspend()`, 675, 678
 - e interbloqueos, 686
 - abolición en Java 2, 688
- sustitución pura, 169
- sustituibilidad, en POO, 2
- Swing, 535
- `synchronized`, 24, 665
 - y herencia, 674
 - y `wait()` `notify()`, 680
 - decidiendo qué métodos sincronizar, 674
 - eficiencia, 670
- `System.arraycopy()`, 320
- `System.err`, 410, 462
- `System.gc()`, 145
- `System.in`, 458, 462
- `System.out`, 462
- `System.out.println()`, 345
- `System.run.Finalization()`, 145

T

-
- tabla, 615
 - tamaño de un HashMap o un HashSet, 372
 - tarjetas clase-responsabilidad-colaboración (CRC), 39
 - TCP(IP y RMI), 770
 - TCP, Transmisión Control Protocol, 726
 - técnicas de prueba, 281
 - terminación vs. reanudación, gestión de excepciones, 635
 - Thread, 647, 649
 - bloqueado, 675
 - combinado con clase main, 654
 - compartiendo recursos limitados, 661
 - cuándo pueden suspenderse, 665
 - cuándo usar hilos, 708
 - decidiendo qué métodos sincronizar, 674
 - destroy(), 689
 - desventajas, 708
 - deteniéndolos y y reanudándolos de forma correcta, 687
 - E/S e hilos, bloqueo, 675
 - estados, 675
 - getPriority(), 690
 - grupo de hilos, 693
 - grupo de hilos, prioridad por defecto, 694
 - hilos demonio, 659
 - hilos y eficiencia, 649
 - interbloqueo, 686
 - interfaz Runnable, 655
 - interrupt(), 686
 - isDaemon(), 659
 - método sincronizado y bloqueo, 675
 - muerto, 675
 - notify(), 675
 - notifyAll(), 675
 - nuevo Thread, 675
 - orden de ejecución de los hilos, 651
 - prioridad, 690
 - resume(), 687
 - abolición en Java 2, 675
 - e interbloques, 686
 - run(), 650
 - Runnable, 675
 - setDaemon(), 659
 - setPriority(), 690
 - sleep(), 664
 - start(), 651
 - stop()
 - abolición en Java 2, 686
 - e interbloques, 686
 - suspend(), 675, 678
 - abolición en Java 2, 687
 - e interbloques, 686
 - wait(), 675, 680
 - y JavaBeans, 670
 - y Runnable, 701
 - yield(), 675
 - Throwable, 418
 - clase base de Exception, 414
 - tiene un, 7
 - tipo base, 8
 - comprobación de tipos y arrays, 301
 - conversión hacia abajo segura durante la identificación de tipos en tiempo de ejecución, 514
 - débilmente tipificado, 13
 - derivado, 8
 - encontrando el tipo exacto de la referencia base, 511, 512
 - equivalencia de tipos de datos a clase, 4
 - parametrizado, 342
 - seguridad de tipos en Java, 96
 - tipos primitivos, 58
 - comparación, 86
 - contenedores, 305
 - de datos, y uso con operadores, 100
 - envoltorio, 364, 827
 - final, 210
 - inicialización de miembros de datos de clases, 150
 - static final, 211
 - toArray(), 389
 - token, 492
 - Tokenizing, 492
 - Tomcat, contenedor estándar de servlets, 756
 - TooManyListenersException, 624, 640
 - toString(), 192, 340, 345, 379

Transferable, 622
 tTransmisión Control Protocol (TCP), 727
 TreeMap, 360, 388, 495
 TreeSet, 358, 384
 trucos, 577
 true, 87
 try, 203, 423
 Tubería, 446

U

UDP, User Datagram Protocol, 726
 UML: 40
 Indicando composición, 7
 Unified Modeling Language, 511, 862
 unario: menos (-), 84
 adición (+), 84
 operador, 89
 operadores, 84
 unbind (), 771
 unicast: eventos unidifusión, 624
 UnicastRemoteObject, 768
 Unicode, 452
 unidad
 de compilación, 170
 de traducción, 170
 unidifusión, 640
 Unified Modeling Language (UML), 5, 862
 UnsupportedOperationException, 208, 393
 URL, 728
 Usando RTTI haciendo uso del objeto Class. 522
 User Datagram Protocol (UDP), 726
 uso
 de alias, 81
 durante llamadas a métodos, 800
 y String, 833
 Uso de hash: función de Hashing perfecta, 370
 utilidad JAR, 469

V

vaciando ficheros de salida, 460
 valor,
 palabra clave HTML, 658
 evitando su cambio en tiempo de ejecución, 210
 value, 80
 variable:

definición de una variable, 113
 variable:
 inicialización de variables de métodos, 150
 listas de parámetros variables (tipo y cantidad de parámetros desconocidos)., 163
 Vector, 384, 396, 397
 de cambio, 44
 versionado: serialización, 485
 versions de Java, XLIII
 visibilidad, visibilidad de paquete (friendly), 270
 Visual BASIC, Microsoft, 628
 visual: programación, 628
 wait (), 675, 681

W

Waldrop, M. Mitchell, 864
 WeakHashMap, 378
 Web:
 colocando un applet dentro de una página Web, 540
 de objetos, 472, 804
 mostrando una página Web desde dentro de un applet, 726
 seguridad y restricciones de applets, 537
 WindowAdapter, 547
 windowClosing (), 547, 604
 Windows Explorer, ejecutando programas Java desde, 547
 write (), 445
 writeBytes (), 460
 writeChars (), 460
 writeDouble (), 460
 writeExternal (), 476
 writeObject () con Serializable, 482, 483
 writeObject (), 472
 Writer, 445, 451, 452, 682, 718

X

XOR, 90
 XP, Extreme Programming, 45
 yield (), 675

Z

ZipEntry, 469
 ZipInputStream, 465
 ZipOutputStream, 465