

Desarrollo de aplicaciones con JAVA, JCreator, JDeveloper NetBeans



| | |
|-----------------|--|
| Familia: | Tecnología de la Información y Computación |
| Editorial: | Macro |
| Autor: | Manuel Torres Remon |
| ISBN: | 978-612-304-101-4 |
| N.º de páginas: | 448 |
| Edición: | 1.ª 2013 |
| Medida: | 17.5 x 24.8 |
| Colores: | 1 |
| Papel: | Bond de 75 g |

Material incluido:

Descripción

Este material resulta ser de medular importancia debido a que expone de manera didáctica las distintas aplicaciones con Java. A lo largo de sus capítulos se desarrollan temas relacionados a la clase swing, programación modular y estructura de secuencia, entre otros, que ayudarán al usuario a crear sus propias aplicaciones. Esta publicación está dirigida principalmente a desarrolladores, estudiantes de programación, ingenieros de sistemas y afines.

Índice general

Capítulo 1: Introducción al lenguaje Java

- 1.1. Introducción
- 1.2. Orígenes del lenguaje Java
 - 1.2.1. Comparando Java con C
- 1.3. Los programas en Java
- 1.4. Evolución de la plataforma Java
- 1.5. Tipos de aplicaciones Java
- 1.6. Requisitos para la instalación de Java
 - 1.6.1. Instalando Java
- 1.7. El JDK 7 y su instalación
- 1.8. El IDE JCreator y su instalación
- 1.9. El IDE NetBeans y su instalación
- 1.10. El IDE JDeveloper 11g y su instalación

Capítulo 2: Fundamentos de programación

- 2.1. Introducción
- 2.2. Los algoritmos
- 2.3. ¿Cómo solucionar un problema
- 2.4. Elementos que componen una aplicación Java
- 2.5. Los identificadores

- 2.6. Palabras reservadas por el lenguaje Java
- 2.7. Los operadores
 - 2.7.1. Operadores Aritméticos Binarios
 - 2.7.2. Operadores Aritméticos Unario
 - 2.7.3. Operadores de incremento y decremento
 - 2.7.4. Operadores de asignación
 - 2.7.5. Operadores de comparación
 - 2.7.6. Operadores lógicos
 - 2.7.7. Operador de concatenación
- 2.8. Orden de prioridad de los operadores
- 2.9. Los separadores
- 2.10. Los comentarios
- 2.11. Expresiones en Java
- 2.12. Tipos de datos
- 2.13. Los literales de los tipos de datos primitivos en Java
- 2.14. Las variables y su declaración en Java
- 2.15. La clase String
- 2.16. Conversiones entre tipos de datos en Java
- 2.17. Clases envoltorio o Wrapper
- 2.18. La clase Integer
- 2.19. Creando un proyecto con JCreator
- 2.20. Creando un proyecto con NetBeans
- 2.21. Creando un proyecto con JDeveloper

Capítulo 3: Clases de Swing

- 3.1. Introducción
- 3.2. Clases Swing
- 3.3. La clase JFrame
- 3.4. La clase JLabel
- 3.5. La clase JTextField
- 3.6. La clase JTextArea
- 3.7. La clase JPasswordField
- 3.8. La clase JButton
- 3.9. La clase JPanel
- 3.10. La clase JCheckBox
- 3.11. La clase JRadioButton
- 3.12. Construir GUI con JCreator
- 3.13. Visualizar Java con un Browser
- Caso desarrollado 1
- 3.14. Construir GUI con NetBeans
 - 3.14.1. Cambiando el contenido visual de los controles
- 3.15. Construir GUI con JDeveloper
 - 3.15.1. Cambiando el contenido visual de los controles

Capítulo 4: Estructura de secuencia

- 4.1. Introducción
- 4.2. Entrada y salida de datos
- 4.3. La clase Math
- 4.4. Métodos que representen a PI y E
- 4.5. Métodos de conversión entre grados y radianes
- 4.6. Métodos de la clase Math
- Caso desarrollado 1: Conversiones (JCreator, JDeveloper y NetBeans)
- Caso desarrollado 2: Casa de Cambio
- Caso desarrollado 3: Medidas

Capítulo 5: Estructura de selección

- 5.1. Introducción
- 5.2. Sentencia if simple
 - Caso desarrollado 1: Renta de autos
 - Caso desarrollado 2: Pago de trabajadores
- 5.4. Sentencia if doble
 - Caso desarrollado 3: Hectáreas de Algodón y Maíz
 - Caso desarrollado 4: Fiesta de 15 años
- 5.5. La clase JComboBox
- 5.6. Sentencia if doblemente enlazada
 - Caso desarrollado 5: Consumo de Agua
 - Caso desarrollado 6: Venta de Productos
- 5.7. Sentencia switch
- 5.8. ¿Cómo implementar un switch
 - Caso desarrollado 7: Hospital

Capítulo 6: Programación modular

- 6.1. Introducción
- 6.2. Variables locales y globales
- 6.3. Variables locales
 - Caso desarrollado 1: Lavandería con variables locales
- 6.4. Variable globales
 - Caso desarrollado 2: Lavandería con variables globales
- 6.5. Métodos void
 - 6.5.1. Métodos void sin parámetros
 - 6.5.2. Métodos void con parámetros
 - Caso desarrollado 3: Compra de piezas de refacción sin parámetros
 - Caso desarrollado 4: Alquiler de equipos de construcción con parámetros
- 6.6. Métodos con valor de retorno
 - 6.6.1. Métodos con valor de retorno sin parámetros
 - 6.6.2. Métodos con valor de retorno con parámetros
 - Caso desarrollado 5: Telas y moda de otoño sin parámetros
 - Caso desarrollado 6: Medio de publicidad con parámetros
- 6.7. Validación de datos
 - Caso desarrollado 7: Promedio de notas

Capítulo 7: Estructura de repetición

- 7.1. Contadores
 - Caso desarrollado 1: Contador de asistencia a una fiesta
- 7.2. Acumuladores
 - Caso desarrollado 2: Apoyo en recepción de hotel
 - Casos propuestos
- 7.3. Estructuras repetitivas
- 7.4. Clase DefaultListModel
- 7.5. Clase JList
- 7.6. Estructura de repetición for
 - Caso desarrollado 1: Venta de cuadernos escolares con for
- 7.7. Estructura de repetición While
 - Caso desarrollado 2: Venta de cuadernos escolares con while
- 7.8. Estructura de repetición While
 - Caso desarrollado 3: Venta de cuadernos escolares con Do While

Capítulo 8: Programación orientada a objetos

- 8.1. Introducción
- 8.2. Conceptos en programación orientada a objetos

8.2.1. Paquetes

8.3. Clases en Java

8.3.1. Partes de una clase

8.4. Objetos en Java

8.4.1 Formato de creación de objetos en Java

8.4.2. Formato para referenciar a los atributos de una clase

8.4.3. Formato para referenciar a los métodos de una clase

Caso desarrollado 1: Pago de pensión usando atributos públicos de clase

8.5. Métodos get y set

8.5.1. Formato para implementar un método set

8.5.2. Formato para implementar un método get

8.5.3. Implementación de métodos get y set con NetBeans

Caso desarrollado 2: Pago de pensión usando atributos privados de clase

8.6. Método constructor

8.6.1. Formato para la implementación de un método constructor

8.6.2. Creando un objeto de la clase Empleado usando método constructor

Caso desarrollado 3: Pago de pensión usando método constructor

8.7. Referencia this

8.8. Variables y métodos de clase: modificar static

8.8.1. Características del modificar Static

Caso desarrollado 4: Pago de empleados usando variables de clase

8.8.2. Métodos Estáticos

8.8.3. Inicializadores de variables de clase

Caso desarrollado 5: Pago de empleados usando métodos de clase e inicializadores

Capítulo 9: Manejo de Excepciones

9.1. Introducción

9.2. Try-Catch

9.3. Cláusula throw

9.4. Bloque finally

Caso desarrollado 1: Registro de libros

Capítulo 10: Arrays

10.1. Introducción

10.1.1. Ventajas y desventajas de usar arreglos

10.2. Arreglos

10.3. Arreglo Unidimensional

10.3.1. Formato de declaración de un arreglo unidimensional

Caso desarrollado 1: Listado de números básicos

Caso desarrollado 2: Listado de números usando clase

Caso desarrollado 3: Certamen de belleza

10.4. Arreglo Bidimensional

10.4.1. Formato de declaración de un arreglo Bidimensional

10.4.2. Operaciones sobre un arreglo Bidimensional

Caso desarrollado 4: Matriz de números enteros

Capítulo 11: Vector de objetos y arraylist

11.1. Vector de Objetos

11.1.1. Formato de declaración del vector de objetos

11.1.2. Formato de creación del vector de objetos

Caso desarrollado 1: Mantenimiento de empleados

11.2. Clase ArrayList

11.2.1. Formato para crear un ArrayList

11.2.2. Métodos que componen la clase ArrayList

Caso desarrollado 2: Mantenimiento de facturas

Capítulo 12: Archivos de texto

12.1. Configuración del JDK

12.2. Librerías a utilizar para el manejo de archivos

12.3. Clases y métodos para el manejo y control de archivos de texto

Caso desarrollado 1: Mantenimiento de estudiantes