

Programación Orientada a Objetos con

Incluye CD-ROM



Ejemplos y Aplicaciones
Software de Desarrollo

5.117
1556

UNIVERSIDAD
CONTINENTAL



B111124329

CENTRO DE
DOCUMENTACIÓN

JAVA

Fondo Editorial



Juan Andrés Chipana Rodríguez



Programación Orientada a Objetos con JAVA

Programación Orientada a Objetos con JAVA

Juan Andrés Chipana Rodríguez

jchipana@continental.edu.pe

jchipana@eco.unc.edu.ar

Universidad Continental de Ciencias e Ingeniería

Fondo Editorial

Calle Real N° 125 Huancayo – Perú

Teléfonos: (064)221580

Web <http://www.continental.edu.pe>

1ª Edición: Septiembre de 2004

Editorial e Imprenta Ríos S.A.C.

Impreso en Perú.



Acerca del Autor

Juan Chipana Rodríguez, Economista e Ingeniero de Sistemas ha hecho sus estudios en La Universidad Nacional de Córdoba y en la Universidad Tecnológica Nacional, Republica Argentina.

En el ámbito laboral se ha desempeñado como asesor informático en diversas instituciones y empresas nacionales y extranjeras.

Actualmente es catedrático de la Universidad Continental de Ciencias e Ingeniería e imparte las asignaturas de Técnicas de Programación, Elementos de Computación, Investigación de Operaciones, entre otras.

Dedicatoria:

Siguiendo las enseñanzas de mi madre y mi familia que me acompaña sobre nuestro Señor Jesucristo, dedico esta obra a los más necesitados y desamparados.

Contenido

1.- Programación Orientada a Objetos

1.1	Introducción	14
1.2	Java.	14
1.3	Conceptos Fundamentales	15
1.4	Funciones	15
1.5	Parámetros por valor e instrucción return	15
1.6	Parámetros por Referencia	16
1.7	Mostrar Resultados	17
1.8	Resumen de Formas de comunicación	18

2.- El entorno de Desarrollo y POO

2.1	Introducción	22
2.2	Ventana principal de JCreate	22
2.3	Creación de Archivos	23
2.4	Aplicaciones	24
2.5	Clases	25
2.6	Niveles de protección y encapsulamiento	27
2.7	Clases contenedoras	29
2.8	Instrucción new	30

3.- Arrays

3.1	Introducción	36
3.2	Declaración y acceso	36
3.3	Recorrido de arrays	36
3.4	Búsqueda en arrays	38
3.5	Arrays multidimensionales	39

4.- Ingreso de Datos por Teclado

4.1	Introducción	44
4.2	Clases InputStreamReader y BufferedReader	44
4.3	Funciones miembro static	46
4.4	Excepciones	46
4.5	Constructores y destructores	46
4.6	Constructores sobrecargados	48

5.- Arrays de Objetos	
5.1 Introducción	50
5.2 Declaración, Creación y Acceso	50
5.3 Ventajas en el uso de arrays de objetos	53
5.4 Objetos como parámetros en funciones miembro	57
5.5 Datos miembro static	58
6.- Herencia	
6.1 Introducción	62
6.2 Niveles de	64
6.3 Protected	64
6.4 Constructores por Defecto en Herencia	65
6.5 Constructores con parámetros en herencia e instrucción super	66
6.6 Redefinición de funciones e instrucción super	68
7.- Polimorfismo	
7.1 Introducción	72
7.2 Sobrecarga de Funciones	72
7.3 Funciones Abstractas	73
8.- Applets	
8.1 Introducción	78
8.2 Proyectos Java Applets	78
8.3 La clase Applet	79
8.3.1 Método init	79
8.3.2 Método resize	79
8.3.3 Método getsize	79
8.4 Método paint y la clase Graphics	80
8.8 La clase Color	84
8.9 Método setColor	84
8.10 La clase Font	85
8.11 El método setFont	85
9. Interfaz gráfica	
9.1 Introducción	88
9.2 Label (Rótulos o Etiquetas)	88
9.3 Button (Botones)	89
9.4 TextField (Campos de Texto)	89

9.5	Choice (Lista de Opciones)	90
9.6	Casilla de Verificación	90
9.7	CheckboxGroup (Grupo de opciones)	91
9.8	List (Cuadro de lista o Listas)	92

10.- Eventos

10.1	Introducción	96
10.2	Eventos de Teclado	96
10.3	Eventos del Ratón	97
10.4	Métodos paint, repaint y update	98
10.5	Programación de Objetos visuales - Método action	101
10.6	La Clase Event	101

11.- Menús

11.1	Introducción	106
11.2	Frames	106
11.3	Menús	107
11.4	Diálogos	108

12.- Ejercicios Resueltos

12.1	Introducción	112
12.2	Ejercicios Complementarios	112
12.3	Ejercicios sobre interfaz gráfica	116
12.4	Ejercicios sobre Programación de objetos visuales	124
12.5	Ejercicios sobre manejo de menús	129

Ejercicios Propuestos	132
-----------------------------	-----

Diccionario de Términos	142
-------------------------------	-----

PRÓLOGO

Actualmente una de las áreas con más desarrollo en la industria del software y en el ámbito académico es la orientación a objetos. La orientación a objetos promete mejoras de amplio alcance. Nos referimos al diseño, desarrollo y mantenimiento del software, ofreciendo una solución a largo plazo a los problemas y preocupaciones que han existido desde el comienzo en el desarrollo de software: la falta de portabilidad del código y reutilización, código que es difícil de modificar, ciclos de desarrollo largos y técnicas de codificación no intuitivas. Un lenguaje orientado a objetos ataca estos problemas.

Conociendo las ventajas de la programación orientada a objetos frente a otros paradigmas de programación, nos preguntamos ¿Qué lenguajes orientado a objetos existen?

Existe un amplio número de lenguajes orientados a objetos, cada uno con una sus propias características. Dentro de este repertorio de lenguajes, tenemos a Java.

Cuando Java se introdujo en forma importante, allá por 1995, fue cuando su uso en el diseño de páginas Web revolucionó la naturaleza de estas, sobre todo mediante applets, (aplicaciones que podemos incluir en nuestras páginas Web). Sin embargo Java es un lenguaje de programación orientada a objetos que no sólo nos permite diseñar aplicaciones para la Web sino que nos permite desarrollar aplicaciones de uso general. Por ejemplo podemos desarrollar un sistema con bases de datos, interfaces y cualquier otro elemento de otros lenguajes.

¿Por qué elegir a Java como lenguaje de desarrollo?

Sabemos que las aplicaciones desarrolladas para la Web día a día son más importantes. Y en este aspecto, Java presenta ventajas sobre los demás lenguajes de programación orientada a objetos. Una aplicación compilada en Java puede ejecutarse en cualquier sistema operativo y en cualquier procesador. Y esa diversidad de plataformas es una característica de la Web.

Java por lo tanto a diferencia de otros lenguajes nos garantiza que nuestra aplicación se va poder ejecutar en cualquier equipo con requerimientos mínimos.

Por lo tanto el presente libro desarrolla un paradigma de programación con diversas ventajas, en un lenguaje de programación con características similares.

En la primera parte desarrollamos los conceptos básicos que incluye un repaso sobre conceptos de programación estructurada necesarios para introducirnos a la programación orientada a objetos.

La segunda parte desarrolla en forma detalla los conceptos fundamentales de la POO y sus diversas aplicaciones.

La tercera parte abarca temas avanzados sobre interfaz gráfica con el usuario, incluyendo temas como applets, objetos visuales, menús, entre otros.

En la última parte hemos incluido un conjunto de ejercicios complementarios totalmente desarrollados, sobre los temas desarrollados y sobre la conjunción o interrelación de los mismos.

El CD que acompaña al presente libro contiene:

- Código de todos los Ejemplo del libro
- Código de los Ejercicios Complementarios
- Software freeware para trabajar con Java.

Agradecimientos

He recibido ayuda de algunas personas en el desarrollo de este libro, colegas y alumnos, por lo que estoy muy agradecido; pero en especial quiero agradecer a mi colega Alicia Tello por sus correcciones en la presentación de este libro. También expresar mi agradecimiento a la Universidad Continental de Ciencias e Ingeniería y a sus autoridades por el apoyo en al publicación de este libro.

Juan Andrés Chipana Rodríguez



1.- Conceptos Fundamentales

1.1 Introducción

Las técnicas o paradigmas de programación han evolucionado con el paso del tiempo. Dentro de estos paradigmas podemos mencionar a la programación estructura, programación lógica, programación orientada a objetos (POO) y otras.

Tengamos en cuenta que la industria del software es una de las más importantes de la economía, y como tal debe ser eficiente. ¿Cómo se logra la eficiencia? La respuesta es simple. Trabajando como lo hacen las otras industrias convencionales. En estas últimas la fabricación de los productos es por partes. Por ejemplo en la industria automotriz, no se realiza la fabricación del auto (producto final) en un solo paso, sino que se divide en etapas de fabricación. Primero se fabrican objetos (partes) para luego a partir de ellos armar el producto final. En la programación orientada a objetos se trabaja de forma similar. Desarrollamos un sistema definiendo objetos. Luego de probar estos objetos se combinan y relacionan formando un producto final, que en este caso es el sistema o aplicación. Trabajar de esta forma tiene muchas ventajas. La más importante es la reutilización de los objetos. En la industria automotriz una llanta puede ser usada en diversos tipos de vehículos. En programación esta propiedad permite que cierto código pueda utilizarse en diversas aplicaciones.

Otras ventajas que tenemos al trabajar con POO es el mantenimiento y la confiabilidad de los sistemas. Es decir cuando los requerimientos del sistema cambian, es mucho más sencillo y confiable hacer los cambios cuando hemos utilizado la POO.

1.2 JAVA

Existen diversos lenguajes de programación orientados a objetos. Java es uno de ellos. Se diferencia del resto en cuanto a la transportabilidad, la cual es una ventaja realmente significativa. Esta característica hace que el código producido por el compilador de Java pueda transportarse a cualquier plataforma, es decir a cualquier procesador con cualquier sistema operativo. Y esto es realmente importante si consideramos la Web donde podemos encontrar una diversidad de equipos de cómputo. Entonces trabajar con Java nos asegura que cualquier máquina en la Web con requerimientos mínimos, pueda ejecutar nuestras aplicaciones.