



Desarrollo de aplicaciones con Java 8

Orientado a objetos

Autor: Manuel Torres Remon

© Derechos de autor registrados:
Empresa Editora Macro EIRL

© Derechos de edición, arte gráfico y diagramación reservados:
Empresa Editora Macro EIRL

Jefe de edición:

Magaly Ramon Quiroz

Diseño de portada:

Fernando Cavassa Repetto

Corrección de estilo:

Dina Dueñas Calderón

Diseño y diagramación:

Fernando Cavassa Repetto

Edición a cargo de:

© Empresa Editora Macro EIRL

Av. Paseo de la República N.° 5613, Miraflores, Lima, Perú

☎ Teléfono: (511) 748 0560

✉ E-mail: proyectoeditorial@editorialmacro.com

🌐 Página web: www.editorialmacro.com

Primera edición: Febrero 2018

Tiraje: 2000 ejemplares

Impresión:

Talleres gráficos de la Empresa Editora Macro EIRL

Jr. San Agustín N.° 612-624, Surquillo, Lima, Perú

Febrero 2018

ISBN N.° 978-612-304-552-4

Hecho el Depósito Legal en la Biblioteca Nacional del Perú N.° 2018-01252

Prohibida la reproducción parcial o total, por cualquier medio o método, de este libro sin previa autorización de la Empresa Editora Macro EIRL.

Índice

Introducción.....	11
■ Capítulo 1: Introducción al lenguaje Java	19
1.1 Generalidades	21
1.2 Lenguajes de programación	22
1.3 Java	23
1.4 Versiones Java	24
1.5 Orígenes del lenguaje Java.....	26
1.6 Características de la tecnología Java.....	27
1.7 Los programas en Java	28
1.8 La máquina virtual Java (JVM)	31
1.9 JDK	32
1.10 JRE.....	33
1.11 Aplicaciones Java.....	34
1.12 Instalando Java	35
1.12.1 Instalación de JDK.....	36
1.12.2 Instalación de NetBeans (versión 8.2)	38
1.12.3 Descarga de Eclipse (versión Mars 2)	42
1.13 El IDE NetBeans y su primera aplicación de plataforma.....	43
1.13.1 Crear una nueva aplicación.....	43
1.13.2 Añadir paquetes al proyecto.....	44
1.13.3 Añadir un formulario al proyecto	45
1.14 El IDE Eclipse Mars y su primera aplicación de plataforma	46
1.14.1 Crear una nueva aplicación.....	46
1.14.2 Añadir paquetes al proyecto.....	47
1.14.3 Añadir un formulario al proyecto	48
CASO DESARROLLADO 1: Monto de capital usando NetBeans.....	50
CASO DESARROLLADO 2: Venta de productos usando Eclipse Mars 2.....	52
■ Capítulo 2: Fundamentos de programación	57
2.1 Generalidades	59
2.2 Proceso de solución mediante una aplicación Java	59
2.2.1 Fase 1: Análisis del problema	59
2.2.2 Fase 2: Diseño del algoritmo.....	61
2.2.3 Fase 3: Codificación del problema	63
2.2.4 Fase 4: Ejecución del programa	66

2.3	Asociar el pseudocódigo al formulario en Java	66
2.4	Metodología de programación con Java	68
2.4.1	Programación estructurada	68
2.4.2	Programación modular	69
2.4.3	Programación orientada a objetos	70
2.5	Elementos que componen una aplicación Java	71
2.6	Los comentarios	72
2.7	Elementos léxicos en Java	74
2.7.1	Palabras reservadas	74
2.7.2	Identificadores	75
2.7.3	Literales	76
2.7.4	Operadores	77
2.7.5	Delimitadores	82
2.8	Los paquetes y las importaciones	84
2.8.1	java.lang	85
2.8.2	Java.awt	85
2.8.3	Javax.swing	85
2.8.4	java.io	85
2.8.5	Java.util	86
2.9	Tipos de datos	86
2.10	Las variables	87
2.10.1	Inicialización de variables	89
2.10.2	Conversiones de tipo	89
2.11	La clase Integer	91
2.12	La clase String	92
2.13	Manejo de la clase JOptionPane	97
2.14	La clase Math	102
2.14.1	Métodos que representan a PI y a E	102
2.14.2	Métodos de conversión entre grados y radianes	103
2.14.3	Métodos de la clase Math	103
	CASO DESARROLLADO 1: Modelo de cotización bancaria (programación estructurada)	108
	CASO DESARROLLADO 2: Modelo de cotización bancaria (programación modular)	110
	CASO DESARROLLADO 3: Modelo de cotización bancaria (programación orientada a objetos)	114
■ Capítulo 3: Clase Swing		119
3.1	Generalidades	121
3.2	Clases Swing	121
3.3	La clase JFrame	124
3.4	La clase JLabel	125
3.5	La clase TextField	126
3.6	La clase TextArea	128
3.7	La Clase PasswordField	131
3.8	La clase Button	132

3.9	La clase JCheckBox	133
3.10	La clase JRadioButton	135
3.11	La clase JComboBox	136
3.12	La clase DefaultListModel	139
3.13	La clase JList	140
3.14	La clase JMenuBar	141
3.15	La clase JMenuItem	142
3.16	Clase Desktop Pane	143
	CASO DESARROLLADO 1: Manejo de la clase JList	144
	CASO DESARROLLADO 2: Manejo de la clase JComboBox	145
	CASO DESARROLLADO 3: Manejo de la clase JMenuBar	146

■ Capítulo 4: Programación orientada a objetos..... 149

4.1	Generalidades	151
4.2	Conceptos iniciales de programación orientada a objetos	152
4.2.1	Abstracción	152
4.2.2	Encapsulamiento	152
4.2.3	Herencia	153
4.2.4	Polimorfismo	153
4.3	Paquetes Java	154
4.3.1	Creación de un paquete	155
4.3.2	Importación de paquetes	157
4.3.3	Alcance de los elementos contenidos en un paquete	157
4.4	Clases en Java	158
4.4.1	Estructura de una clase	158
4.4.2	Agregar una clase a un paquete en NetBeans	159
4.4.3	Atributos de una clase	160
4.4.4	Métodos en las clases	161
4.4.5	Métodos sin valor de retorno	162
4.4.6	Métodos con valor de retorno	163
4.5	Método constructor	164
4.6	Referencia this	165
4.7	Objetos en Java	166
4.7.1	Referenciar a los atributos de clase	167
4.7.2	Referenciar a los métodos de clase	168
4.8	Métodos get y set	168
4.8.1	Método set	168
4.8.2	Método get	170
4.8.3	Implementación de métodos get y set con NetBeans	171
	CASO DESARROLLADO 1: Casa de cambio (con atributos públicos de clase)	173
	CASO DESARROLLADO 2: Casa de cambio (con atributos privados de clase)	176
	CASO DESARROLLADO 3: Casa de cambio (con métodos get y set)	179
	CASO DESARROLLADO 4: Casa de cambio (con método constructor)	182

■ Capítulo 5: Estructura de secuencia 187

5.1	Generalidades	189
5.2	Estructura secuencial.....	189
5.3	Componentes de una estructura secuencial	190
5.3.1	Declaración.....	190
5.3.2	Asignación	190
5.3.3	Lectura u obtención de datos.....	190
5.3.4	Escritura de datos.....	191
	CASO DESARROLLADO 1: Cambio de temperaturas	192
	CASO DESARROLLADO 2: Conversión de medidas	195
	CASO DESARROLLADO 3: Pago de empleados	198
	CASO DESARROLLADO 4: Venta de productos	201

■ Capítulo 6: Estructura de selección..... 207

6.1	Generalidades	209
6.2	Operadores de relación	209
6.3	Operadores lógicos	211
6.4	Instrucción if simple.....	212
6.5	Instrucción if doble	213
6.6	Instrucción if doblemente enlazada.....	214
6.7	Sentencia switch	215
6.8	Implementar una condición lógica	216
	CASO DESARROLLADO 1: Renta de autos (usando if doblemente enlazado)	220
	CASO DESARROLLADO 2: Pago de trabajadores (usando if simple)	224
	CASO DESARROLLADO 3: Control de hectáreas de algodón y maíz (usando if doble).....	228
	CASO DESARROLLADO 4: Control de consumo de agua (usando if doblemente enlazado).....	232
	CASO DESARROLLADO 5: Control de pensión de estudiantes (usando switch).....	235

■ Capítulo 7: Estructura de repetición 241

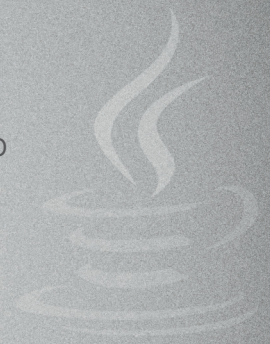
7.1	Generalidades	243
7.2	Contadores.....	243
7.3	Acumuladores	244
7.4	Estructuras repetitivas.....	244
7.4.1	Estructura de repetición for	245
7.4.2	Estructura de repetición while	247
7.4.3	Estructura de repetición do... while	248
7.5	Variables y métodos de clase: modificador static	250
7.5.1	Variable de clase.....	250
7.5.2	Métodos estáticos.....	251
7.5.3	Inicializadores de variables de clase	252
	CASO DESARROLLADO 1: Control de registro de participantes (usando contadores).....	253
	CASO DESARROLLADO 2: Registro de alquiler (usando acumuladores)	258

CASO DESARROLLADO 3: Registro de venta (usando for)	264
CASO DESARROLLADO 4: Control de pago (usando modificar static)	270
■ Capítulo 8: Lambda y manejo de excepciones	279
8.1 Generalidades	281
8.2 Expresión Lambda	281
8.3 Bloque try-catch	283
8.4 Cláusula throw	284
8.5 Bloque finally	286
CASO DESARROLLADO 1: Registro de libros	286
CASO DESARROLLADO 2: Control de pago de empleados usando Lambda	291
■ Capítulo 9: Arreglos.....	297
9.1 Generalidades	299
9.2 Arreglos	300
9.3 Arreglo unidimensional.....	301
9.4 Arreglo bidimensional.....	303
CASO DESARROLLADO 1: Listado de números básico.....	306
CASO DESARROLLADO 2: Listado de números usando clase	309
CASO DESARROLLADO 3: Certamen de belleza.....	314
CASO DESARROLLADO 4: Matriz de números enteros.....	319
■ Capítulo 10: Vector de objetos y ArrayList.....	327
10.1 Vector de objetos.....	329
10.2 Clase ArrayList	330
CASO DESARROLLADO 1: Mantenimiento de empleados (vector de objetos).....	331
CASO DESARROLLADO 2: Mantenimiento de facturas (ArrayList)	339
CASO DESARROLLADO 3: Mantenimiento de productos (ArrayList y Lambda)	349
■ Capítulo 11: Archivos de texto	357
11.1 Configuración del JDK	359
11.2 Librerías a utilizar para el manejo de archivos.....	362
11.3 Clases y métodos para el manejo y control de archivos de texto	363
CASO DESARROLLADO 1: Mantenimiento de alumnos.....	365
CASO DESARROLLADO 2: Mantenimiento de productos	374
Referencias bibliográficas.....	383

Introducción al lenguaje Java

CAPÍTULO

1



Competencias

- Reconoce los conceptos básicos del lenguaje Java.
- Identifica los componentes de la máquina virtual de Java.
- Configura e instala las aplicaciones Java en una computadora personal.
- Implementa aplicaciones básicas usando NetBeans y Eclipse Mars.

Contenido

- 1.1 Generalidades
- 1.2 Lenguajes de programación
- 1.3 Java
- 1.4 Versiones Java
- 1.5 Orígenes del lenguaje Java
- 1.6 Características de la tecnología Java
- 1.7 Los programas en Java
- 1.8 La máquina virtual Java (JVM)
- 1.9 JDK
- 1.10 JRE
- 1.11 Aplicaciones Java
- 1.12 Instalando Java
- 1.13 El IDE NetBeans y su primera aplicación de plataforma
- 1.14 El IDE Eclipse Mars y su primera aplicación de plataforma
 - ▶ Caso desarrollado 1: Monto de capital usando NetBeans
 - ▶ Caso desarrollado 2: Venta de productos usando Eclipse Mars 2

1.1 Generalidades

Los tiempos han cambiado gracias a la nueva era digital en la que se vive, en la cual los aparatos electrónicos han tomado mucho valor. Hoy se puede ver una nueva generación de equipos inteligentes (televisores, autos, lavadoras, etc.), capaces de hacer la vida de sus usuarios mucho más cómoda. Uno de los puntos importantes para esta revolución es la programación, ya que es percibida como un segundo lenguaje en la era digital. Así, se puede afirmar que en el futuro todos los aparatos electrónicos se manejarán mediante aplicaciones desarrolladas que permitirán tener una personalización de las actividades que estas realizan.

Para lograr estos objetivos es necesario desarrollar aplicaciones, y esto solo se logrará dominando un lenguaje de programación, lo cual —en realidad— no es una tarea difícil, pues es evidente en la actualidad el gran avance de la informática; no obstante, la gran mayoría de las personas aún está logrando adaptarse a ello. Así pues, en la era del desarrollo, desde temprana edad, ya se podrán crear aplicaciones, a medida, que solucionen algún proceso por más simple o complejo que este sea. Uno de los lenguajes de programación más conocidos es Java, pues se adapta mucho a los cambios y presenta una mejor performance al momento de aplicarse en aparatos electrónicos: televisores, relojes o aparatos móviles, como es el caso de Android.

Nuestro futuro está marcado por la programación, por eso nos vemos en la necesidad de aprender a desarrollar algo que nos permita dar solución a un determinado problema. Por ejemplo, en el Reino Unido se ha introducido la programación de *software* en el currículo de los colegios, la cual se desarrolla de forma obligatoria a partir de la educación primaria; este hecho se enfoca en tres fines que resulta importante analizar: el primero es para que los escolares sean capaces de entender el código fuente de cualquier programa; el segundo, para conocer los pilares fundamentales de la programación; y, finalmente, el tercero, para desarrollar sus propias aplicaciones enfocadas a sus necesidades. Esto significa que dentro de unos años se tendrá una nueva generación de programadores que poseerán una visión distinta con respecto a la programación; para ellos será como aplicar matemáticas al desarrollar una aplicación. Asimismo, Estados Unidos, mediante su organización Code.org, promueve un programa llamado la Hora del Código (del inglés *Hour of Code*), que incentiva a los estudiantes para que opten por un curso alternativo de desarrollo de aplicaciones. En el Perú, Microsoft promueve su campaña #YoPuedoProgramar, dirigida a estudiantes de primaria en un intento por difundir las posibilidades de programación informática con un programa llamado Una Hora de Código. Así pues, se puede advertir que todo lo dicho es un pequeño paso a la formación de futuros programadores.

Finalmente, se puede decir que el desarrollo del *software* es uno de los principales impulsores de la innovación en el mundo, lo cual se debe a la nueva generación de aparatos inteligentes y muchas cosas más que vendrán en un futuro no muy lejano. Todos los profesionales de distintas carreras deberán estar en un continuo proceso de búsqueda y aprendizaje de nuevos lenguajes de programación que se adapten a sus necesidades; esto se puede dar usando lenguajes como C#, PHP o Java, los cuales en la actualidad dominan el futuro de la programación. También se debe considerar que detrás de todos los lenguajes habidos y por haber se encuentran empresas muy influyentes como Google, Microsoft, Oracle, que siempre estarán presentes en la evolución de la programación.