



Presentación del Curso Presencial

**Programación en Java
Nivel Básico**



Tabla de contenido

Presentación del curso	3
Objetivos de aprendizaje	6
Contenidos del curso	7
Competencias previas	9
Recursos	9
Aspectos metodológicos	9
Criterios de aprobación	9



Presentación del curso

El curso de Programación en Java Básico consta de cinco unidades, el cual tiene como objetivo conseguir que los desarrolladores de software que sean nuevos en la tecnología Java estén listos y en movimiento con la programación orientada a objetos (OOP) y el desarrollo de aplicaciones del mundo real usando el lenguaje y la plataforma Java.

La primera parte es una introducción paso a paso a la OOP con el uso del lenguaje Java. El tutorial comienza con una visión general de la plataforma y el lenguaje Java y le siguen las instrucciones para establecer un entorno de desarrollo que consiste en un Kit de desarrollo de Java (JDK) y el Eclipse IDE. Una vez que se le hayan presentado los componentes de su entorno de desarrollo, comenzará a aprender la sintaxis Java básica en un modo práctico.

Como cualquier lenguaje de programación, el lenguaje Java tiene su propia estructura, reglas de sintaxis y paradigma de programación. El paradigma de programación del lenguaje Java se basa en el concepto de programación orientada a objetos (OOP), que las funciones del lenguaje soportan.

El lenguaje Java es un derivado del lenguaje C, por lo que sus reglas de sintaxis se parecen mucho a C: por ejemplo, los bloques de códigos se modularizan en métodos y se delimitan con llaves ({ y }) y las variables se declaran antes de que se usen.

Estructuralmente, el lenguaje Java comienza con *paquetes*. Un paquete es el mecanismo de espacio de nombres del lenguaje Java. Dentro de los paquetes se encuentran las clases y dentro de las clases se encuentran métodos, variables, constantes, entre otros. En este tutorial, aprenderá acerca de las partes del lenguaje Java.





El compilador Java

Cuando usted programa para la plataforma Java, escribe el código de origen en archivos .java y luego los compila. El compilador verifica su código con las reglas de sintaxis del lenguaje, luego escribe los *códigos byte* en archivos .class. Los *códigos byte* son instrucciones estándar destinadas a ejecutarse en una Java Virtual Machine (JVM). Al agregar este nivel de abstracción, el compilador Java difiere de los otros compiladores de lenguaje, que escriben instrucciones apropiadas para el chipset de la CPU en el que el programa se ejecutará.

La JVM

Al momento de la ejecución, la JVM lee e interpreta archivos .class y ejecuta las instrucciones del programa en la plataforma de hardware nativo para la que se escribió la JVM. La JVM interpreta los *códigos byte* del mismo modo en que una CPU interpretaría las instrucciones del lenguaje del conjunto. La diferencia es que la JVM es un software escrito específicamente para una plataforma particular. La JVM es el corazón del principio "escrito una vez, ejecutado en cualquier lugar" del lenguaje Java. Su código se puede ejecutar en cualquier chipset para el cual una implementación apropiada de la JVM está disponible. Las JVM están disponibles para plataformas principales como Linux y Windows y se han implementado subconjuntos del lenguaje Java en las JVM para teléfonos móviles y aficionados de chips.

4

El recolector de basura

En lugar de forzarlo a mantenerse a la par con la asignación de memoria (o usar una biblioteca de terceros para hacer esto), la plataforma Java proporciona una gestión de memoria lista para usar. Cuando su aplicación Java crea una instancia de objeto al momento de ejecución, la JVM asigna automáticamente espacio de memoria para ese objeto desde el *almacenamiento dinámico*, que es una agrupación de memoria reservada para que use su programa. El *recolector de basura* Java se ejecuta en segundo plano y realiza un seguimiento de cuáles son los objetos que la aplicación ya no necesita y recupera la memoria que ellos ocupan. Este abordaje al manejo de la memoria se llama *gestión de la memoria implícita* porque no le exige que escriba cualquier código de manejo de la memoria. La recogida de basura es una de las funciones esenciales del rendimiento de la plataforma Java.





El kit de desarrollo de Java

Cuando usted descarga un kit de desarrollo de Java (JDK), obtiene, — además del compilador y otras herramientas, — una librería de clase completa de programas de utilidad preconstruidos que lo ayudan a cumplir cualquier tarea común al desarrollo de aplicaciones. El mejor modo para tener una idea del ámbito de los paquetes y bibliotecas JDK es verificar la documentación API JDK (vea Recursos).

El Java Runtime Environment

El Java Runtime Environment (JRE, también conocido como el Java Runtime) incluye las bibliotecas de códigos de la JVM y los componentes que son necesarios para programas en ejecución escritos en el lenguaje Java. Está disponible para múltiples plataformas. Puede redistribuir libremente el JRE con sus aplicaciones, de acuerdo a los términos de la licencia del JRE, para darles a los usuarios de la aplicación una plataforma en la cual ejecutar su software. El JRE se incluye en el JDK.



Objetivos de aprendizaje**Objetivo general:**

- Conocer las bases conceptuales y técnicas para iniciar a desarrollar en Java en base a la metodología de orientación a Objetos.

Objetivos específicos:

El alumno adquirirá los fundamentos básicos de la programación orientada a objetos. Es de especial interés para esta asignatura que el alumno sea capaz de analizar un problema proponiendo la solución que mejor se adapta al mismo. Al mismo tiempo que adquiere estos conocimientos básicos aprenderá un lenguaje de programación orientado a objetos con el que desarrollar en práctica los conocimientos teóricos. Todos los conceptos sobre programación adquiridos (teóricos y prácticos) se aplicarán al desarrollo de programas complejos donde el alumno hará uso de la programación orientada a objetos: definiendo clases, aplicando herencia y desarrollando los métodos necesarios para el correcto funcionamiento del programa.

6

Los objetivos específicos de aprendizaje son:

- Adquirir los fundamentos básicos de la programación orientada a objetos. Aprender un lenguaje de programación orientado a objetos (JAVA) con el que desarrollar en la práctica los conocimientos teóricos.
- Ser capaz de escribir programas sencillos en los que se utilicen:
 - Encapsulación, herencia y polimorfismo.
 - Clases abstractas e interfaces.
 - Control de excepciones.
 - Ficheros de distintos tipos.
- Conocer los principios más importantes de la programación recursiva.
- Conocer y utilizar diversos algoritmos de ordenación y búsqueda.

Contenidos del curso**Unidad 1**

- 1.1. Introducción al lenguaje Java
 - 1.1.1. Historia
 - 1.1.2. Arquitectura
 - 1.1.3. Conceptos de la Programación Orientada a Objetos
 - 1.1.4. Programa "Hola Mundo"
- 1.2. Tipos de Datos
 - 1.2.1. Numéricos
 - 1.2.2. Alfanuméricos
 - 1.2.3. Lógicos
 - 1.2.4. Arreglos
- 1.3. Operadores
 - 1.3.1. Aritméticos
 - 1.3.2. Relacionales
 - 1.3.3. Lógicos
 - 1.3.4. De incremento
 - 1.3.5. De asignación
 - 1.3.6.

Unidad 2

- 2.1. Objetos y clases
 - 2.1.1. Declaración de clases
 - 2.1.2. Atributos y métodos
 - 2.1.3. Creación de objetos: Constructores
- 2.2. Referenciación a objetos
 - 2.2.1. Clases static
 - 2.2.2. Paso de parámetros
 - 2.2.3. Variables, clases y métodos final

Unidad 3

- 3.1. Estructuras de control del programa
 - 3.1.1. Estructura de decisión binaria if
 - 3.1.2. Estructuras de repetición for, while, do while
 - 3.1.3. Estructura de selección múltiple switch
 - 3.1.4. Operador ternario
 - 3.1.5. Comandos break y continue
- 3.2. Entrada y salida de datos
 - 3.2.1. La clase System
 - 3.2.2. Salida de datos por pantalla
 - 3.2.3. Ingreso de datos por teclado
 - 3.2.4. Uso simple de archivos para entrada y salida





Unidad 4

- 4.1. Excepciones
 - 4.1.1. Manejo de errores usando excepciones
 - 4.1.2. Bloque try
 - 4.1.3. Bloque catch
 - 4.1.4. Bloque finally

Unidad 5

- 5.1. Estructuras de datos
 - 5.1.1. ArrayLists
 - 5.1.2. Sets
 - 5.1.3. Lists
 - 5.1.4. Maps



Competencias previas

- Manejo de paquetes informáticos
- Este curso es para desarrolladores de software que todavía no tienen experiencia con el código Java o la plataforma Java. El curso incluye una visión general de los conceptos de OOP.

Recursos

- Equipo de computación
- Conexión a internet

Aspectos metodológicos

El curso se desarrolla de forma presencial en las aulas de clase, donde se realizan actividades teóricas y prácticas con un enfoque dinámico y participativo centrado en los participantes.

Cada semana se presentan lecciones junto con actividades individuales y colaborativas.

El seguimiento del curso efectuado es constante, lo que garantiza el éxito del proceso de aprendizaje.

Criterios de aprobación

- Revisión permanente de la información recibida durante el desarrollo del curso
- Cumplimiento de los trabajos y deberes propuestos
- Participación activa durante el desarrollo del curso
- Obtención de un rendimiento mínimo de 14/20 puntos
- Asistencia mínima de 80/100