

## **Programación - Java**

Laura Gabriela Vélez Mesa  
Juan Jose Naranjo Bocanegra

Corporación Universitaria Empresarial Alexander Von Humboldt

Facultad de Ingeniería

Arle Morales Ortiz

Armenia, Quindio

13 de Julio de 2024

## 1. ¿Qué es un paradigma de programación?

Un paradigma de programación es un enfoque o estilo de programación que proporciona una manera particular de construir y estructurar el software. Los paradigmas de programación más comunes son el procedural, el orientado a objetos, el funcional y el lógico. Cada uno de estos paradigmas ofrece diferentes técnicas y estructuras para la resolución de problemas en el desarrollo de software.

## 2. ¿Qué es la programación orientada a objetos?

La programación orientada a objetos con siglas (POO) utiliza "objetos" y sus interacciones para diseñar aplicaciones y programas informáticos. Se basa en cuatro principios fundamentales: encapsulamiento, abstracción, herencia y polimorfismo.

### 3. a. Resumen de las preguntas frecuentes de la página 9 “Frequently Asked Questions”

**What is a virtual machine?** Una máquina virtual (VM) es un entorno de software que emula a una computadora física, permitiendo que se ejecuten programas como si estuvieran en una máquina física. En Java, la Java Virtual Machine (JVM) permite que el código Java se ejecute en cualquier plataforma que tenga una JVM, proporcionando independencia de la plataforma.

b. Cuadro comparativo de diferencias entre Java, Python y JavaScript

Característica	Java	Python	JavaScript
Paradigma	Orientado a objetos, imperativo, funcional	Orientado a objetos, imperativo, funcional	Orientado a objetos, imperativo, funcional
Tipado	Estático	Dinámico	Dinámico
Ejecución	Compilado y ejecutado	Interpretado	Interpretado
Uso Principal	Aplicaciones empresariales, Android	Ciencia de datos, Scripting, aplicaciones web	Desarrollo web, aplicaciones del lado cliente y servidor
Sintaxis	Verbosa y explícita	Concisa y flexible	Concisa y flexible
Gestión de Memoria	Automática (Garbage Collection)	Automática (Garbage Collection)	Automática (Garbage Collection)

Concurrencia	Multithreading	Multithreading y async	Event-driven, async
--------------	----------------	------------------------	---------------------

### c. Explicaciones

#### ¿Cómo hacer un comentario en Java?

En Java, los comentarios se pueden hacer de tres formas:

- Comentarios de una sola línea usando //
- Comentarios de múltiples líneas usando `/* ... */`
- Comentarios de documentación usando `/** ... */` para generar documentación con Javadoc.

#### ¿Qué es una palabra reservada?

Una palabra reservada en Java es una palabra que tiene un significado especial para el compilador y no se puede usar como nombre de variables, métodos, clases o cualquier otro identificador. Ejemplos incluyen `if`, `else`, `while`, `for`, etc

#### ¿Qué es un identificador?

Un identificador es el nombre que se da a elementos como variables, métodos, clases y otros objetos definidos por el usuario en un programa. Debe comenzar con una letra, un signo de dólar \$ o un guión bajo `_`, seguido por cualquier combinación de caracteres.

#### ¿Qué es un literal?

Un literal es una representación fija de un valor en el código fuente. Ejemplos de literales en Java incluyen números (42, 3.14), caracteres ('a'), cadenas ("Hello, World!"), y valores booleanos.

### 5. ¿Qué son los datos primitivos en Java?

Los datos primitivos en Java son los tipos de datos básicos que están predefinidos en el lenguaje. Incluyen:

- `byte`: Entero de 8 bits
- `short`: Entero de 16 bits
- `int`: Entero de 32 bits
- `long`: Entero de 64 bits
- `float`: Número de punto flotante de 32 bits
- `double`: Número de punto flotante de 64 bits

- char: Carácter Unicode de 16 bits
- boolean: Valores true o false

## 6. ¿Qué tipos de operadores existen en java?

Según el libro en la página 35 (aunque no he podido verificar la página exacta en el PDF), los tipos de operadores en Java incluyen:

- Operadores aritméticos: +, -, \*, /, %
- Operadores lógicos: &&, ||, !
- Operadores de comparación: ==, !=, >, <, >=, <=

## 7. ¿Cómo funcionan los operadores de incremento y decremento?

Los operadores de incremento (++) y decremento (--) se usan para incrementar o decrementar en 1 el valor de una variable.

## 8. ¿Cuáles son los operadores de comparación?

- == (igual a)
- != (diferente de)
- > (mayor que)
- < (menor que)
- >= (mayor o igual que)
- <= (menor o igual que)

## 11. ¿Cómo funciona la sentencia “The switch Statement”?

La sentencia switch en Java se utiliza para ejecutar un bloque de código entre múltiples opciones según el valor de una variable. Puede ser más legible que una serie de sentencias if-else cuando se trata de múltiples valores discretos.

## 14. ¿Qué hace la sentencia break y continue en un ciclo?

break: Esta sentencia se utiliza para salir completamente del bucle. Cuando se encuentra una instrucción break, el bucle se detiene inmediatamente, y no se ejecuta ningún código posterior dentro de esa iteración ni en las iteraciones restantes.

continue: Esta sentencia permite saltar a la siguiente iteración sin ejecutar el código restante dentro de la iteración actual. Es útil cuando queremos omitir ciertas comprobaciones dentro del bucle sin anidar más condicionales.

**15. Con el ejemplo de depuración en java, demuestre con pantallazos el uso del depurador en intelliJ IDEA, puede utilizar otro IDE.**



