



PROMETEO

PROGRAMACIÓN

Conceptos fundamentales



Williams Infanzón

williams.infanzon@ext.thepower.education

PROMETEO

Índice

01 Variables

02 Tipos de datos

03 Condicionales

04 Bucles

05 Scanner

01

Variables

¿Qué es una variable?

En programación, una variable es como un contenedor donde se almacena un dato. Este dato puede ser de distintos tipos (número, texto, etc.) y cambiar durante la ejecución del programa.

Propiedades de una variable

1. Nombre: Identifica a la variable (como una etiqueta única).
2. Tipo: Determina qué clase de datos puede almacenar.
3. Valor: El dato que contiene en un momento dado.

01

Variables de Entorno

DEMO

02

Tipos de Datos

En Java, los **tipos de datos** definen el tipo de valor que una variable puede almacenar. Estos se dividen principalmente en:

1. **Tipos primitivos:** Datos básicos y de tamaño fijo.
2. **Tipos por referencia:** Representan objetos y estructuras de datos más complejas, como cadenas (**String**) y listas.

02

Tipos de Datos

Tipos primitivos:

Tipo	Tamaño	Valores posibles	Ejemplo
<code>byte</code>	8 bits	-128 a 127	<code>byte edad = 25;</code>
<code>short</code>	16 bits	-32,768 a 32,767	<code>short año = 2023;</code>
<code>int</code>	32 bits	-2,147,483,648 a 2,147,483,647	<code>int numero = 100;</code>
<code>long</code>	64 bits	-9,223,372,036,854,775,808 a ...	<code>long distancia = 100000L;</code>
<code>float</code>	32 bits	Decimales simples	<code>float precio = 10.5f;</code>
<code>double</code>	64 bits	Decimales dobles	<code>double peso = 70.8;</code>
<code>char</code>	16 bits	Un carácter	<code>char inicial = 'A';</code>
<code>boolean</code>	1 bit	<code>true</code> o <code>false</code>	<code>boolean esMayor = true;</code>

02

Tipos de Datos

Los **tipos por referencia** apuntan a un objeto o estructura compleja en memoria. Los más comunes son:

1. **String**: Almacena texto.
2. **Arrays**: Almacenan múltiples valores del mismo tipo.
3. **Clases personalizadas**: Diseñadas por el programador.

02

Tipos de Datos

Conversión entre tipos

En ocasiones, es necesario convertir un tipo de dato a otro. Esto puede hacerse de dos maneras:

- **Conversión implícita (widening):** Java convierte automáticamente de un tipo más pequeño a uno más grande.
- **Conversión explícita (casting):** El programador indica la conversión.

02

Tipos de Datos

Comparación entre Tipos

El tipo de una variable determina:

- **El rango de valores que puede almacenar.**
- **Las operaciones que se pueden realizar.**

Ejemplo: Problema por desbordamiento

Un tipo primitivo tiene un tamaño fijo. Si intentamos almacenar un valor fuera de su rango, ocurre un **desbordamiento**.

02

Tipos de Datos

DEMO

03

Condicionales

Las estructuras de control son fundamentales en cualquier lenguaje de programación, ya que permiten alterar el flujo de ejecución de un programa. En Java, estas estructuras se clasifican en **selección**, **repetición** y **salto**.

Este bloque abordará en profundidad cada tipo de estructura, proporcionando ejemplos detallados, ejercicios prácticos, y un test para evaluar el aprendizaje.

Estructuras de Selección

Las estructuras de selección se utilizan para ejecutar diferentes bloques de código dependiendo de ciertas condiciones. En Java, estas incluyen:

1. **if**
2. **if-else**
3. **else if**
4. **switch**

03

Condicionales

DEMO

04

Bucles

Las estructuras de repetición permiten ejecutar un bloque de código varias veces, ya sea un número fijo de iteraciones o mientras una condición específica se cumpla. En Java, las estructuras de repetición principales son:

1. **for**: Ideal para un número fijo de iteraciones.
2. **while**: Ejecuta el bloque mientras la condición sea verdadera.
3. **do-while**: Similar a **while**, pero garantiza al menos una ejecución del bloque.

04

Bucles

DEMO

05

Scanner

La clase **Scanner** permite **leer datos introducidos por el usuario** desde diferentes fuentes, como el **teclado** o archivos.

Se utiliza para capturar entradas de distintos tipos (texto, números, decimales, etc.) de forma sencilla.

Métodos comunes:

- `nextLine()` → lee una cadena completa
- `nextInt()` → lee un número entero
- `nextDouble()` → lee un número decimal
- `next()` → lee una sola palabra

05

Scanner

DEMO

UT 2

PRÁCTICA



¡Primer programa individual!



Desarrollar una calculadora
básica con 4 métodos



Entrega del .java



PROMETEO



PROMETEO