

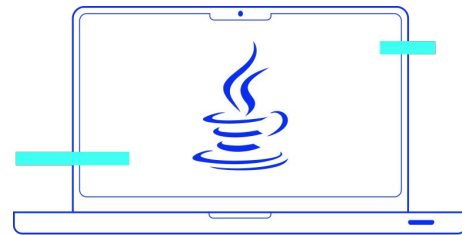
# Bootcamp Java Developer

Fase 1 - Java Analyst  
Módulo 7

# Operadores

# Introducción

En esta sección, aprenderemos a manipular la información y a realizar preguntas a las variables que almacenamos y para eso tenemos los **operadores: símbolos matemáticos** que indican que debe ser llevada a cabo una **operación específica** sobre un cierto número de operandos (cantidad de elementos).

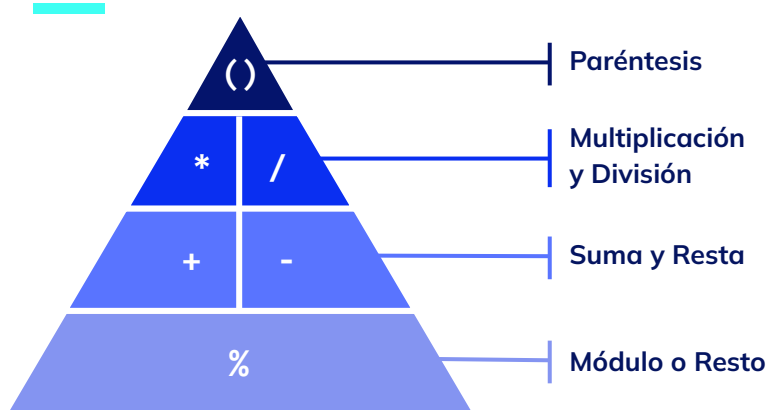


# Operadores

## Operadores

Operador	Símbolo	Ejemplo
Suma	+	num1 + num2
Resta	-	num1 - num2
Multiplicación	*	num1 * num2
División	/	num1 / num2
Módulo o Resto	%	num1 % num2

## Jerarquías



**Recuerda:**

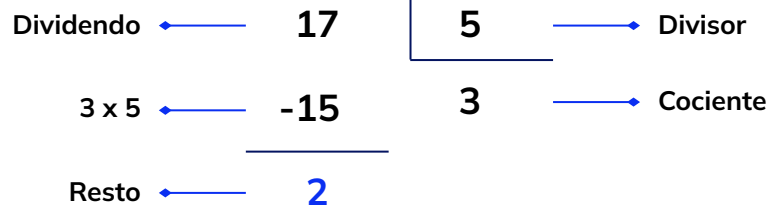
El símbolo de suma “+” sirve también para **concatenar (“pegar”) cadenas de texto.**

```
String identificadorString = "Cadena de " + "Caracteres";
```



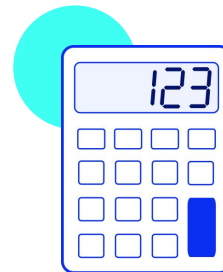
## Módulo o resto

Es el número que queda después de haber dividido un número no divisible.



The diagram illustrates a division problem with labels and arrows pointing to the corresponding parts of the calculation:

Dividendo	←	17		5	→	Divisor
				3	→	Cociente
		3 x 5	←	-15		
		<hr/>				
Resto	←	2				



# Operadores de asignación

Esencialmente, son un grupo que permite asignar un **valor a una variable**:

Símbolo	Descripción	Ejemplos
=	Asignación	num1 = num2 → num1 = 5
+=	Suma y asignación	num1 += num2 → (num1 = num1 + num2)
-=	Resta y asignación	num1 -= num2 → (num1 = num1 - num2)
*=	Multiplicación y asignación	num1 *= num2 → (num1 = num1 * num2)
/=	División y asignación	num1 /= num2 → (num1 = num1 / num2)
%=	Resto y asignación	num1 %= num2 → (num1 = num1 % num2)

# Operadores de comparación

Comparan dos valores y devuelven **Verdadero** o **Falso**.



Símbolo	Descripción	Ejemplo
<code>==</code>	Igualdad	<code>num1 == num2</code>
<code>!=</code>	Distinto	<code>num1 != num2</code>
<code>&lt;</code>	Menor que	<code>num1 &lt; num2</code>
<code>&gt;</code>	Mayor que	<code>num1 &gt; num2</code>
<code>&lt;=</code>	Menor o igual que	<code>num1 &lt;= num2</code>
<code>&gt;=</code>	Mayor o igual que	<code>num1 &gt;= num2</code>





# Operadores incrementales

Son operadores **unarios** (un único operando). Ayudan a incrementar y a disminuir el valor de una variable numérica.



Símbolo	Descripción	Ejemplos
++	Para incrementar de uno en uno	num ++    primero utiliza la variables y después incrementa.
		++ num    primero incrementa el valor y después la usa.
--	Para disminuir de uno en uno	num --    primero utiliza la variables y después la disminuye.
		-- num1    primero disminuye el valor y después la usa.



# Operadores lógicos

Son usados para **concatenar dos o más expresiones con operadores relacionales**.

Símbolo	Descripción	Ejemplo
<code>&amp;&amp;</code>	And	op1 <code>&amp;&amp;</code> op2
<code>  </code>	Or	op1 <code>  </code> op2
<code>&amp;</code>	And Inclusivo	op1 <code>&amp;</code> op2
<code> </code>	Or Inclusivo	op1 <code> </code> op2
<code>!</code>	Not	<code>!op1</code>

En algunos casos no es necesario evaluar la segunda condición, pero de ser así utilizaríamos los operadores ***inclusivos*** para ello.

Para saber qué valores booleanos pueden devolver estas preguntas anidadas con los operadores lógicos, debemos saber la **tabla de la verdad**.

Condición 1	Condición 2	And	Or
Verdadero	Verdadero	Verdadero	Verdadero
Verdadero	Falso	Falso	Verdadero
Falso	Verdadero	Falso	Verdadero
Falso	Falso	Falso	Falso



**¡Sigamos  
trabajando!**