

MANIPULACIÓN DE NÚMEROS, CARACTERES Y OTROS VALORES



TIPOS DE DATOS PRIMITIVOS EN JAVA

Lógicos	boolean	true, false
Caracteres	char	caracteres unicode
Numéricos	byte	entero de 8 bits
	short	entero de 16 bits
	int	entero de 32 bits
	long	entero de 64 bits
	float	real de 32 bits
	double	real de 64 bits

Adicionalmente, tenemos la cadena de caracteres `java.lang.String`

VALORES LITERALES

- ▶ Representación de un valor fijo.
- ▶ Se escriben directamente en el código.

```
//Valor booleano VERDADERO
boolean resultado = true;
//Letra C
char letraMayuscula = 'C';
//Número 100
byte by = 100;
//Número 1000
short sh = 1000;
//Número 1000000
int in = 1000000;
```

```
//Valor 26, en decimal
int decVal = 26;
//Valor 26, en hexadecimal
int hexVal = 0x1a;
//Valor 26, en binario
int binVal = 0b11010;
```

VALORES LITERALES

- ▶ Desde Java SE 7, se puede usar el guión bajo (_) para delimitar parte de un literal. Java no lo procesará como parte de ese literal.

```
long creditCardNumber = 1234_5678_9012_3456L;
```

OPERADOR DE ASIGNACIÓN

- ▶ =
- ▶ Nos permite asignar un valor a una variable.
- ▶ El valor asignado debe ser del tipo sobre el que se ha definido la variable.

```
//Valor booleano VERDADERO
boolean resultado = true;
//Letra C
char letraMayuscula = 'C';
//Número 100
byte by = 100;
//Número 1000
short sh = 1000;
//Número 1000000
int in = 1000000;
```



1.

TIPOS DE DATOS NUMÉRICOS

NÚMEROS ENTEROS

- Números sin decimales, positivos y negativos.

Tipo de dato	Tamaño	Rango	Valor por defecto
byte	8 bits (1 byte)	De -128 a 127	0
short	16 bits	De -32,768 a 32,767	0
int	32 bits	De -2^{31} a $2^{31}-1$	0
long	64 bits	De -2^{63} a $2^{63}-1$	0L

NÚMEROS REALES

- ▶ Números con decimales, positivos y negativos.

Tipo de dato	Tamaño	Valor por defecto
float	32 bits (IEEE 754)	0.0f
double	64 bits (IEEE 754)	0.0d

IEEE 754

Núm = mantisa * **base**^{exponente}

$$345 = 0.345 * 10^3$$

OPERADORES NUMÉRICOS

Tipo	Operador	Precedencia	Operación realizada
Prefix, postfix	-- ++	expr++ expr-- ++expr --expr	Incremento/Decremento en una unidad.
Unarios	+ -	+ -	Cambio de signo
Multiplicativos	* / %	* / %	Multiplicación, división y resto
Aditivos	+ -	+ -	Suma, resta
De movimiento	<< >> >>>	<< >> >>>	Desplazamiento a nivel de bits.

OPERADORES NUMÉRICOS DE ASIGNACIÓN ABREVIADA

Operador	Uso	Equivalente a
<code>+=</code>	<code>op1 += op2</code>	<code>op1 = op1 + op2</code>
<code>-=</code>	<code>op1 -= op2</code>	<code>op1 = op1 - op2</code>
<code>*=</code>	<code>op1 *= op2</code>	<code>op1 = op1 * op2</code>
<code>/=</code>	<code>op1 /= op2</code>	<code>op1 = op1 / op2</code>
<code>%=</code>	<code>op1 %= op2</code>	<code>op1 = op1 % op2</code>

OPERADORES NUMÉRICOS A NIVEL DE BITS

Operador	Descripción
&	Realiza la operación lógica AND a nivel de bits
^	Realiza la operación lógica OR exclusivo a nivel de bits
	Realizar la operación lógica OR inclusivo a nivel de bits
~	Cambia cada 0 por un 1, y cada 1 por un cero
<<	Desplaza un número de bits hacia la izquierda. Rellena los huecos con ceros.
>>	(Signed). Desplaza un número de bits hacia la derecha. Rellena los huecos con el bit más significativo (el que indica el signo).
>>>	(Unsigned). Desplaza un número de bits hacia la derecha. Rellena con ceros a la izquierda



2.

TIPOS DE DATOS
LÓGICOS

VALORES **BOOLEANOS**

- ▶ Susceptibles de ser VERDADEROS (`true`) o FALSOS (`false`).
- ▶ Muy útiles en comparaciones y otras operaciones.

OPERADORES LÓGICOS CONDICIONALES

Operador	Descripción
!	Realiza la negación del operando
&&	Realiza la operación lógica condicional AND
	Realiza la operación lógica condicional OR
?:	(Ternario) Si el primer operando es verdadero, devuelve el valor del segundo; en otro caso, devuelve el tercero

OPERADORES LÓGICOS RELACIONALES

Operador	Descripción
==	Devuelve verdadero si ambos valores son verdaderos
!=	Devuelve el valor inverso a ==
>	Devuelve verdadero si el valor de la izquierda es mayor estricto que el de la derecha.
>=	Devuelve verdadero si el valor de la izquierda es mayor o igual que el de la derecha.
<	Devuelve verdadero si el valor de la izquierda es menor estricto que el de la derecha.
<=	Devuelve verdadero si el valor de la izquierda es menor o igual que el de la derecha.



3.

TIPOS DE DATOS DE CARACTERES

TIPOS DE DATOS DE CARACTERES

Tipo	Descripción	Literal
char	Nos permite almacenar un carácter UNICODE (16 bits)	'a'
java.lang.String	No es un tipo básico. Nos permite manejar cadenas de caracteres inmutables.	"Hola"

El operador más usual con cadenas de caracteres es +, que nos permite concatenar dos valores.



4.

CAMBIOS DE TIPOS DE DATOS (CASTINGS)

CASTINGS

- ▶ En ocasiones, nos puede interesar realizar un cambio *explícito* de un tipo de dato.

`System.out.println(5/9);` \longrightarrow 0

- ▶ A esta operación se le llama *casting*

`System.out.println((double)5/9);` \longleftarrow 0.5555555555555556

- ▶ Los tipos de datos deben ser *compatibles*.