

Programación I

# Operadores

Introducción al Lenguaje Java

# Operadores

Los operadores habituales  $+$   $-$   $*$   $/$  se utilizan en Java para denotar suma, resta, multiplicación y división.

El resto entero, llamado módulo, se denota mediante el signo  $\%$ . Por ejemplo  $15 \% 2 = 1$ .

La división por cero da lugar a una excepción (un error), mientras que la división de un coma flotante por cero produce un resultado infinito o NaN ( Not a Number ).

Existe una abreviatura para emplear operadores en la asignación. Por ejemplo,  $x+=4$ ; que es equivalente a  $x = x + 4$ ;

# Operadores de asignación

El operador más importante y más frecuentemente usado es el operador asignación =, que hemos empleado para la inicialización de las variables y constantes.

```
//VARIABLES:
```

```
double sueldo;  
int dias_de_vacaciones;  
long poblacion_terrestre;  
boolean hecho = false;  
int i, j;
```

```
//CONSTANTES
```

```
final double DOLAR = 89.78;
```

a=b;

que asigna a a el valor de b. A la izquierda siempre tendremos una variable tal como a, que recibe valores, a la derecha otra variable b, o expresión que tiene un valor. Por tanto, tienen sentido las expresiones.


```
int ancho, alto;  
int a=1234;  
double area= calculaArea(radio);  
double superficie=ancho*alto;
```

# Operadores de incremento y decremento

Java tiene operadores tanto de incremento como decremento para realizar estas sencillas operaciones.

`n++` o `n--` - incrementa o decrementa en 1 el valor de la variable `n`.

```
int ancho, alto;  
int a=1234;  
a++;  
double area= calculaArea(rad
```



Hay dos formas de utilizarlos, en forma de sufijo o en forma de prefijo.

La única diferencia entre ambas formas surge cuando se utilizan los operadores dentro de una expresión.

```
int m = 7; int n = 7; int a = 2 * ++m;  
// en este caso a = 16 y m = 8  
int b = 2 * n++;  
// en este caso b = 14 y n = 8
```

# Concatenación de Strings

En Java se usa el operador + para concatenar cadenas de caracteres o Strings.

```
System.out.println(nombre + " tiene " + edad + " años.");
```

El operador + cuando se utiliza con Strings y otros objetos, creando un solo String que contiene la concatenación de todos sus operandos.

Si alguno de los operandos no es una cadena, se convierte automáticamente en una cadena.



# Operadores relacionales y booleanos

SIGNIFICADO	OPERADOR
Igual que	==
Distinto de	!=
Mayor que	>
Menor que	<
Mayor o igual que	>=
Menor o igual que	<=
Operador lógico Y	&&
Operador lógico O	
Operador ternario	Condición?expresión1:expresión2

# Síntesis de los operadores

- **Aritméticos**

- + : suma
- - : resta
- \* : multiplicación
- / : división

- **Lógicos, relacionales y booleanos**

- > : mayor que
- < : menor que
- <> : mayor o menor que
- != : distinto que
- == : igual que
- && : y lógico
- || : o lógico

- **Incremento y decremento**

- ++ : Incremento
- -- : Decremento
- += n<sup>º</sup> : incremento
- -= n<sup>º</sup> : decremento

- **Concatenación**

- + : une o concatena.

# Síntesis de los operadores

- **Aritméticos**

- + : suma
- - : resta
- \* : multiplicación
- / : división

- **Lógicos, relacionales y booleanos**

- > : mayor que
- < : menor que
- <> : mayor o menor que
- != : distinto que
- == : igual que
- && : y lógico
- || : o lógico

- **Incremento y decremento**

- ++ : Incremento
- -- : Decremento
- += n<sup>º</sup> : incremento
- -= n<sup>º</sup> : decremento

- **Concatenación**

- + : une o concatena.