

# Java: Tipos Primitivos



Facultad de Ciencias  
Diana Isabel Ramírez García

Introducción a Ciencias de la Computación 2021-1

# Procesos de Programación

1. **Definición del programa**
2. Diseño
3. Implementación
4. Depuración
5. Mantenimiento
6. Documentación



# Procesos de Programación

1. Definición del programa
2. **Diseño**
3. Implementación
4. Depuración
5. Mantenimiento
6. Documentación



# Procesos de Programación

1. Definición del programa
2. Diseño
3. **Implementación**
4. Depuración
5. Mantenimiento
6. Documentación





# Palabras reservadas

¿Toda palabra puede ser un identificador?

abstract	else	interface	static
boolean	extends	long	super
break	false	main	switch
byte	final	native	synchronized
case	finally	new	this
catch	float	null	throw
char	for	package	throws
class	if	private	true
continue	implements	protected	try
default	import	public	void
do	instanceof	return	volatile
double	int	short	while

# Primeros Pasos en Java

## ¿Qué son los tipos primitivos?

Son tipos ya definidos en Java que representan los datos más simples , mientras que los objetos son creados por el programador con una estructura más compleja.

# Tipos Primitivos

Tipo	Descripción
byte	Entero de 8 bits
short	Entero de 16 bits
int	Entero de 32 bits
long	Entero de 64 bits
float	Real en 32 bits con 7 dígitos en la parte decimal
double	Real en 64 bits con 15 dígitos en la parte decimal
char	Caracter en 16-bits (Unicode)
boolean	Booleano



# Declaración de una variable

Antes poder utilizar una variable o asignarle un valor, es necesario declararla con el identificador adecuado.

## Ejemplos:

- `int entero;`
- `boolean verdadero;`
- `char caracter;`

# Operador de Asignación

Utilizamos el operador " = " para asignarle un valor a toda variable.

## Asignación Simple:

```
int a = 2020;
```

## Asignación Múltiple:

```
int b = 2020 , c = 2019 , d = 2018 ;
```

## Asignación Compuesta:

```
a = a + 1 ;
```

# Operadores Aritméticos

Operador	Descripción
+	Suma
-	Resta
*	Multiplicación
/	División
%	Residuo de la división

Operador	Descripción
+	Más unario
-	Menos unario
++	Autoincremento
--	Autodecremento

# Operadores Relación

Estos operadores nos devuelven un valor booleano de acuerdo a la comparación

Operador	Descripción
<	Menor que
<=	Menor o igual que
==	Igual que
>=	Mayor o igual que
>	Mayor que
!=	Diferente de

# Operadores Lógicos

Estos operadores únicamente son utilizados por valores booleanos y devuelven uno del mismo tipo

Operador	Descripción
&&	Conjunción
	Disyunción
!	Negación

# Tablas de verdad

&&	true	false
true	true	false
false	false	false

	true	false
true	true	true
false	true	false

!	
false	true
true	false

# Concatenación

Existe el tipo primitivo de caracter sin embargo , no existe uno para las cadenas, estos se manejan como objetos de la clase String.

## Ejemplo

```
String nombre = "Diana"
```

```
String apellido = "Ramírez"
```

```
String nombreCompleto = nombre + apellido
```

# Precedencia y Asociación

Operador	Descripción	Asoc
() [] .	paréntesis y punto	I
++ --	Incremento y decremento (Post)	I
++ --	Incremento y decremento (Pre)	D
!	Negación	D
- +	Menos y más unarios	D
new	Operador para crear objetos	D
( <i>tipo</i> )	Conversión explícita de tipo	D
* / %	Multiplicación, división, residuo	I
+ -	Suma, resta	I
instanceof <, <=, >, >=	Operadores de relación	I
== !=	Operadores de igualdad	I
&&	Conjunción	I
	Disyunción	I
?:	Operador condicional	D
= <i>op</i> =	Asignación	D



# Conversión

**Conversión Implícita:** Cuando tenemos un tipo de menor y otro de mayor prioridad.

**double a = 2 + 3.4 ;**

Al asignar un tipo no se puede asignar uno de menor prioridad si el valor no lo cumple, ya se pierde información.

**int i = 3.4 ;**



**double i = 3.4;**



# Conversión

**Conversión explícita:** Cuando no es posible la conversión implícita, sin embargo necesitamos su conversión (cast llamado en inglés).

**Ejemplo:**

```
double real = 5.345679;
```

```
int entero = (int) real;
```

# Bibliografía

López, A.. (2014). Introducción al desarrollo de programas con Java. Mexico: Las prensas de Ciencias.