

# Chuleta de Java

Jose Jesus García Rueda  
(Adaptado de Jialong He)

## Tipos de datos básicos

<b>Byte</b>	8	-128..127
<b>short</b>	16	-32,768..32,767
<b>int</b>	32	-2,147,483,648.. 2,147,483,647
<b>long</b>	64	-9,223,372,036,854,775,808.. 9,223,372,036,854,775,807
<b>float</b>	32	3.4e-038.. 3.4e+038
<b>double</b>	64	1.7e-308.. 1.7e+308
<b>char</b>	16	Carácter Unicode
<b>Boolean</b>		true, false

Declaración de variable:  
*tipo* identificador [= *valor*];

## Arrays

**int** *nombre\_array*[ ];                    declara un array de enteros  
*nombre\_array*=**new int**[100 ];            de tamaño 100

**int** *nombre\_array*[ ] = **new int** [            declara un array de enteros  
100];                    en una sola línea  
**int** *nombre\_array* [ ] = { 1, 2, 3, 4};

**int** *nombre\_array*[] = **new**                    array multidimensional  
**int**[10][20];

**null**

## Clase

```
{public|final|abstract} class nombre
{ [declaracion_variables_clase]
public static void main(String[] args) {código}
[métodos]
}
```

this, super

## Método

```
{public | private } [static] {tipo | void} nombre(arg, ..., arg )
{código}
```

## Variable

```
{public | private } [static]tipo name [= expresión];
```

## Interface

```
interface nombre
```

```
{
Lista de métodos declarados.
}
```

implements

## Operadores aritméticos

**+, -, \*, /**                    suma, resta, multiplicación, división  
**%, ++, --**                modulo, incremento, decremento

## Operadores Relacionales

**==, !=, >, <, >=, <=**            igual, distinto, mayor, menor  
                              mayor o igual, menor o igual

## Operadores Lógicos

**&, |, !, ^, ||, &&**            AND, OR, NOT, XOR, if OR, AND

## Comentarios

```
// resto de la línea
/* comentario multilinea */
/** comentario para documentación */
```

## Compilación y Ejecución

javac NombreDeFichero.java  
java NombreDeFichero  
El CLASSPATH debe estar correctamente configurado  
El nombre del fichero ha de ser igual al nombre de la clase

## Control de Flujo

if (expresión booleana) sentencia1; [ else sentencia2; ]

```
Bucle while
[inicialización]
while (condición de terminación) {
cuerpo;
[iteración;]
}
```

```
Bucle do while
do {
cuerpo;
[iteración;]
} while (condición de terminación);
```

```
Bucle for
for (inicialización; condición de terminación; iteración) {
cuerpo;
}
```

```
Estructura de Programa
class NombreClase {
public static void main (String args[ ]) {
```

```
sentencias;
}
definición de método 1
...
definición de método N
}
```

## Tratamiento de excepciones

```
try
{
Código que puede dar lugar a excepciones
}
catch(TipoDeExcepción1 obj1)
{
Manejador de excepciones de tipo 1
}
catch(TipoDeExcepción2 obj2)
{
Manejador de excepciones de tipo 2
}
finally{
Se ejecuta, haya habido o no excepción, tras terminarse con el try
}
```

## Palabras Clave

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

## Paquetes de Java

**java.applet** Incluye las clases necesarias para crear un applet y las clases y las clases que un applet usa para comunicarse con su contexto.

**java.awt** Contiene todas las clases para crear interfaces de usuario y para dibujar gráficos e imágenes.

**java.awt.color** Incluye clases para tratar el color.

**java.awt.event** Incluye interfaces y clases para tratar los diferentes tipos de eventos.

**java.awt.font** Incluye clases para tratar las fuentes de letra.

**java.awt.image** Incluye clases para crear y modificar imagines.

**java.io** Permite la entrada y salida de datos del sistema a través de “data streams”, serialización y el sistema de ficheros.

**java.lang** Incluye clases fundamentales para el diseño de programas en lenguaje Java.

**java.util** Miscelánea.