



APRENDERAPROGRAMAR.COM

TIPOS DE DATOS
(VARIABLES) EN JAVA.
TIPOS PRIMITIVOS (INT,
BOOLEAN, ETC.) Y OBJETO
(STRING, ARRAY, ETC.)
(CU00621B)

Sección: Cursos

Categoría: Curso “Aprender programación Java desde cero”

Fecha revisión: 2029

Resumen: Entrega nº21 curso Aprender programación Java desde cero.

Autor: Alex Rodríguez

TIPOS DE DATOS (VARIABLES) EN JAVA. TIPOS PRIMITIVOS (INT, ETC.) Y OBJETO.

Los primeros lenguajes de programación no usaban objetos, solo variables. Una variable podríamos decir que es **un espacio de la memoria del ordenador a la que asignamos un contenido** que puede ser un valor numérico (sólo números, con su valor de cálculo) o de tipo carácter o cadena de caracteres (valor alfanumérico que constará sólo de texto o de texto mezclado con números).



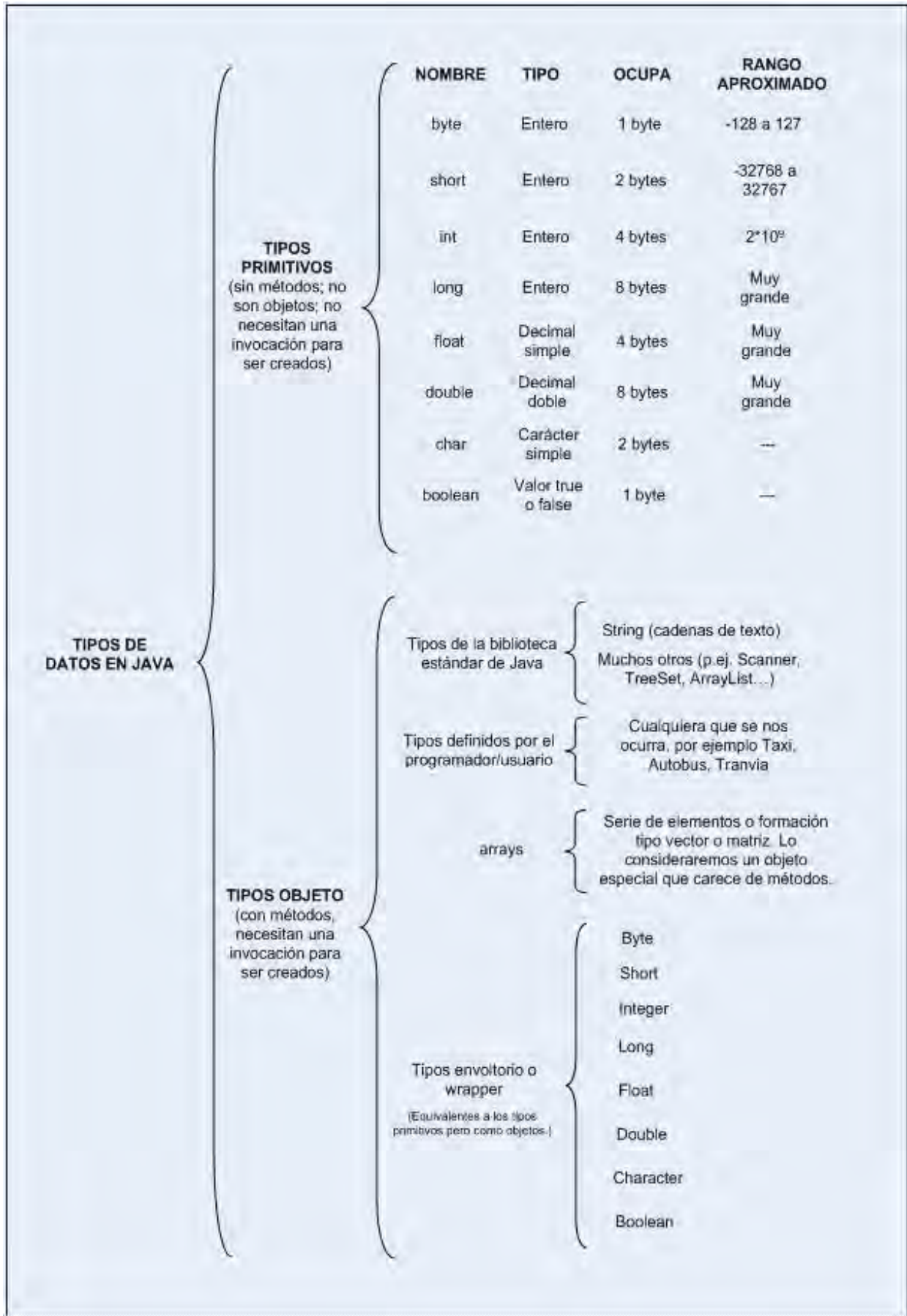
Como ejemplo podemos definir una variable a que contenga 32 y esto lo escribimos como `a = 32`. Posteriormente podemos cambiar el valor de a y hacer `a = 78`. O hacer “a” equivalente al valor de otra variable “b” así: `a = b`.

Dado que antes hemos dicho que un objeto también ocupa un espacio de memoria: **¿en qué se parecen y en qué se diferencia un objeto de una variable?** Consideraremos que las variables son entidades elementales: un número, un carácter, un valor verdadero o falso... mientras que los objetos son entidades complejas que pueden estar formadas por la agrupación de muchas variables y métodos. Pero ambas cosas ocupan lo mismo: un espacio de memoria (que puede ser más o menos grande).

En los programas en Java puede ser necesario tanto el uso de datos elementales como de datos complejos. Por eso en Java se usa el término “Tipos de datos” para englobar a cualquier cosa que ocupa un espacio de memoria y que puede ir tomando distintos valores o características durante la ejecución del programa. Es decir, en vez de hablar de tipos de variables o de tipos de objetos, hablaremos simplemente de tipos de datos. Sin embargo, a veces “coloquialmente” no se utiliza la terminología de forma estricta: puedes encontrarte textos o páginas web donde se habla de una variable en alusión a un objeto.

En Java diferenciamos dos tipos de datos: por un lado, los tipos primitivos, que se corresponden con los tipos de variables en lenguajes como C y que son los datos elementales que hemos citado. Por otro lado, los tipos objeto (que normalmente incluyen métodos).

Veamos los tipos de datos en Java sobre un esquema de síntesis:



Esquema de síntesis de tipos de datos en Java

Este esquema no es necesario aprendérselo de memoria en todos sus detalles, aunque sí lo iremos memorizando poco a poco a medida que lo utilicemos, por lo menos hasta tener en nuestra cabeza los nombres de todos los tipos primitivos y envoltorio y sus características (si son objetos o no y su rango aproximado). Vamos a comentar distintas cuestiones:

1. Un objeto es una cosa distinta a un tipo primitivo, aunque “porten” la misma información. Tener siempre presente que los objetos en Java tienen un tipo de tratamiento y los tipos primitivos, otro. Que en un momento dado contengan la misma información no significa en ningún caso que sean lo mismo. Iremos viendo las diferencias entre ambos poco a poco. De momento, recuerda que el tipo primitivo es algo elemental y el objeto algo complejo. Supón una cesta de manzanas en la calle: algo elemental. Supón una cesta de manzanas dentro de una nave espacial (considerando el conjunto nave + cesta): algo complejo. La información que portan puede ser la misma, pero no son lo mismo.

2. ¿Para qué tener esa aparente duplicidad entre tipos primitivos y tipos envoltorio? Esto es una cuestión que atañe a la concepción del lenguaje de programación. Tener en cuenta una cosa: un tipo primitivo es un dato elemental y carece de métodos, mientras que un objeto es una entidad compleja y dispone de métodos. Por otro lado, de acuerdo con la especificación de Java, es posible que necesitemos utilizar dentro de un programa un objeto que “porte” como contenido un número entero. Desde el momento en que sea necesario un objeto habremos de pensar en un envoltorio, por ejemplo Integer. Inicialmente nos puede costar un poco distinguir cuándo usar un tipo primitivo y cuándo un envoltorio en situaciones en las que ambos sean válidos. Seguiremos esta regla: usaremos por norma general tipos primitivos. Cuando para la estructura de datos o el proceso a realizar sea necesario un objeto, usaremos un envoltorio.

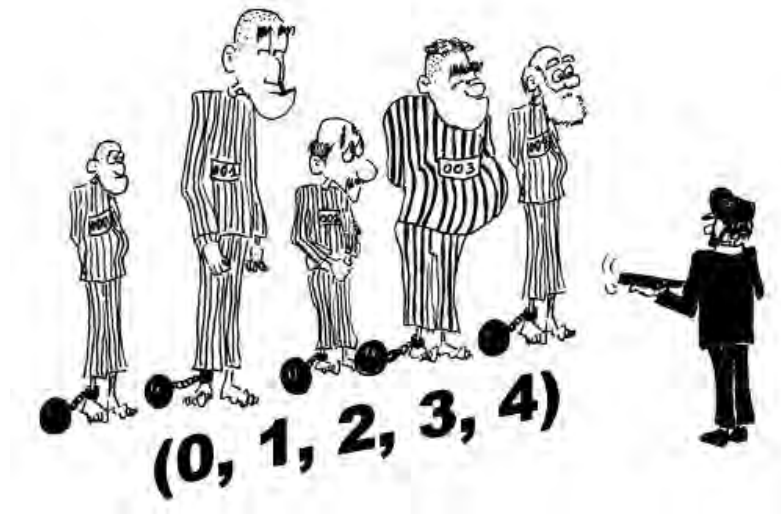
3. Los nombres de tipos primitivos y envoltorio se parecen mucho. En realidad, excepto entre int e Integer y char y Character, la diferencia se limita a que en un caso la inicial es minúscula (por ejemplo double) y en el otro es mayúscula (Double). Esa similitud puede confundirnos inicialmente, pero hemos de tener muy claro qué es cada tipo y cuándo utilizar cada tipo.

4. Una cadena de caracteres es un objeto. El tipo String en Java nos permite crear objetos que contienen texto (palabras, frases, etc.). El texto debe ir siempre entre comillas. Muchas veces se cree erróneamente que el tipo String es un tipo primitivo por analogía con otros lenguajes donde String funciona como una variable elemental. En Java no es así.

5. Hay distintos tipos primitivos enteros. ¿Cuál usar? Por norma general usaremos el tipo int. Para casos en los que el entero pueda ser muy grande usaremos el tipo long. Los tipos byte y short los usaremos cuando tengamos un mayor dominio del lenguaje.

6. ¿Cuántos tipos de la biblioteca estándar de Java hay? Cientos o miles. Es imposible conocerlos todos.

7. **¿Un array es un objeto?** Los arrays los consideraremos objetos especiales, los únicos objetos en Java que carecen de métodos.



Concepto de array: serie de elementos, cada uno de los cuales lleva asociado un índice numérico 0, 1, 2, 3, ... , n-1

Próxima entrega: CU00622B

Acceso al curso completo en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente:

http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=68&Itemid=188