



Name: Isaac Escobar

NRC: 3682

Primitive Data Types in Java

What are primitive data types in Java?

As we have already mentioned, Java is a static typing language. That is, the data type of the variable is defined when defining it. That is why all variables will have a data type assigned.

The Java language is based on a series of primitive data types.

- byte
- shorts
- int
- long
- float
- double
- boolean
- char

It is important to know that these are language data types and that they do not represent objects. Which is the case with the rest of the elements of the Java language.

byte

Represents a signed 8-bit data type. So you can store numeric values from -128 to 127 (inclusive).

short

Represents a signed 16-bit data type. This way it stores numeric values from -32,768 to 32,767.

int

It is a signed 32-bit data type for storing numeric values. Whose minimum value is -2^{31} and the maximum value $2^{31}-1$.



long

Es un tipo de dato de 64 bits con signo que almacena valores numéricos entre -2^{63} a $2^{63}-1$

float

Es un tipo dato para almacenar números en coma flotante con precisión simple de 32 bits.

double

Es un tipo de dato para almacenar números en coma flotante con doble precisión de 64 bits.

boolean

Sirve para definir tipos de datos booleanos. Es decir, aquellos que tienen un valor de true o false. Ocupa 1 bit de información.

char

Es un tipo de datos que representa a un carácter Unicode sencillo de 16 bits.

Valores por defecto de los tipos de datos primitivos

En el caso de que definamos una variable y no le demos ningún valor, por defecto llevarán los siguientes valores:

Primitive Data	Default value
byte	0
short	0
int	0
long	0L
float	0.0f
double	0.0d
char	'\u0000'
String (or any object)	null



Primitive Data

Default value

boolean

false

There is a String data type for handling strings that is not itself a primitive data type. With the String data type we can handle character strings separated by double quotes.

The String element is an immutable data type. That is, once created, its value cannot be changed.

The String is not a primitive data type of the Java language. But its use is just as important as the data types reviewed here. We will see more in detail the use of the String type.

Simple Primitive Python Data Types

Simple primitive data types

- **Numbers:** A sequence of digits (can include the - for negatives and the. For decimals) that represent numbers.

Example. 0, -1, 3.1415.

- **Strings:** A sequence of alphanumeric characters that represent text. They are enclosed in single or double quotes.

Example. 'Hello Goodbye'.

- **Boolean (boolean):** Contains only two elements True and False that represent the logical values true and false respectively.

This data is immutable, that is, its value is constant and cannot change.

Compound primitive data types (containers)

- **Lists:** Collections of objects that represent ordered sequences of objects of different types. They are represented by square brackets and the elements are separated by commas.
 - **Example.** [1, "two", [3, 4], True].
- **Tuples:** Collections of objects that represent ordered sequences of objects of different types. Unlike lists, they are immutable, that is, they do not change during execution. They are represented by parentheses and items are separated by commas.
 - **Example.** (1, 'two', 3)
- **Dictionaries:** Collections of objects with an associated key. They are represented by braces, the pairs separated by commas and each pair contains



- a key and an associated object separated by a colon.

Example. {'Pi': 3.1416, 'e': 2.718}. Clase de un dato (type())

Numbers (int and float classes)

Sequence of digits (can include the - for negatives and the . for decimals) that represent numbers. They can be integers (int) or reals (float).

```
>>> type(1)
<class 'int'>
>>> type(-2)
<class 'int'>
>>> type(2.3)
<class 'float'>
```

Logical or Boolean data (bool class)

- It contains only two elements True and False that represent the logical values true and false respectively.
 - False has the value 0 associated with it and True has the value 1 associated with it.
 - Operations with logical values
- Logical operators: == (equal to), > (greater), < (less), >= (greater than or equal to), <= (less than or equal to), != (Different from).
 - not b (negation): Returns True if the Boolean data b is False, and False otherwise.
 - b1 and b2: Returns True if the Boolean data b1 and b2 are True, and False otherwise.
 - b1 or b2: Returns True if any of the Boolean data b1 or b2 is True, and False otherwise.

True table

x	y	not x	x and y	x or y
False	False	True	False	False
False	True	True	False	True
True	False	False	False	True
True	True	False	True	True

```
>>> not True
False
>>> False or True
True
```



```
>>> True and False
False
>>> True and True
True
```

Simple primitive data conversion

The following functions convert data of one type to another, as long as the conversion is possible.

- `int ()` converts to integer.
- Example. `int ('12 ')` 12
- `int (True)` 1
- `int ('c')` Error
- `float ()` converts to real.
- Example. `float ('3.14')` 3.14
- `float (True)` 1.0
- `float ('III')` Error
- `str ()` converts to string.
- Example. `str (3.14)` '3.14'
- `str (True)` 'True'
- `bool ()` converts to logical.
- Example. `bool ('0')` False
- `bool ('3.14')` True
- `bool ('')` False
- `bool ('Hello')` True