

**The eight primitive data types in Java are:**

* boolean, the type whose values are either true or false.
* char, the character type whose values are 16-bit Unicode characters.
* the arithmetic types: the integral types: byte. short. int. long. the floating-point types: float. double.

What are the different data types in Java programming?

The platform independent feature of Java is achieved through bytecode.

The eight primitive data types are: byte, short, int, long, float, double, boolean, and char. The java.lang.**String** class represents character strings. Instance variables (non-static fields) are unique to each instance of a class.

**Java: Primitive data types**

The eight primitive data types in Java are:

* boolean, the type whose values are either true or false
* char, the character type whose values are 16-bit Unicode characters
* the arithmetic types:
  + the integral types:
    - byte
    - short
    - int
    - long
  + the floating-point types:
    - float
    - double

Values of class type are references. Strings are references to an instance of class String.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Primitive data types in Java** | | | | | | | | |
| **Type** | | | **Description** | | **Default** | **Size** | **Example Literals** | |
| boolean | | | true or false | | false | 1 bit | true, false | |
| byte | | | twos complement integer | | 0 | 8 bits | (none) | |
| char | | | Unicode character | | \u0000 | 16 bits | 'a', '\u0041', '\101', '\\', '\'', '\n', 'ß' | |
| short | | | twos complement integer | | 0 | 16 bits | (none) | |
| int | | | twos complement integer | | 0 | 32 bits | -2, -1, 0, 1, 2 | |
| long | | | twos complement integer | | 0 | 64 bits | -2L, -1L, 0L, 1L, 2L | |
| float | | | IEEE 754 floating point | | 0.0 | 32 bits | 1.23e100f, -1.23e-100f, .3f, 3.14F | |
| double | | | IEEE 754 floating point | | 0.0 | 64 bits | 1.23456e300d, -1.23456e-300d, 1e1d | |
| **Range of numeric data types in Java** | | | | | | |
| **Type** | **Size** | | **Range** | | | |
| byte | 8 bits | | -128 .. 127 | | | |
| short | 16 bits | | -32,768 .. 32,767 | | | |
| int | 32 bits | | -2,147,483,648 .. 2,147,483,647 | | | |
| long | 64 bits | | -9,223,372,036,854,775,808 .. 9,223,372,036,854,775,807 | | | |
| float | 32 bits | | 3.40282347 x 1038, 1.40239846 x 10-45 | | | |
| double | 64 bits | | 1.7976931348623157 x 10308, 4.9406564584124654 x 10-324 | | | |

### Primitives vs. References

* **primitive types** are the basic types of data
  + byte, short, int, long, float, double, boolean, char
  + primitive variables store primitive values
* **reference types** are any instantiable class as well as arrays
  + String, Scanner, Random, Die, int[], String[], etc.
  + reference variables store addresses

### Assignment

* copies the contents of RHS variable into LHS variable
  + primitives: the primitive value is copied
  + references: the address is copied
* **implications**: for references the object is not copied, it is shared (reference variables are *aliases*)

### Comparisons (e.g. ==)

* compares the contents of the variables
  + primitives: the primitive values are compared
  + references: the addresses are compared
* **implications**: for references the contents of the objects are not compared