

Day x - Notes

Java Training



April 1, 2023

Dpoint

[Company address]

# Datatypes:

Java is a strongly typed language, which means that every variable and expression in a Java program has a specific type that determines the range of values it can hold and the operations that can be performed on it. Here are the different data types available in Java:

1. **Primitive data types**: These are the basic data types provided by Java for storing simple values. They are as follows:

* **byte**: an 8-bit signed integer, with a range of -128 to 127.
* **short**: a 16-bit signed integer, with a range of -32,768 to 32,767.
* **int**: a 32-bit signed integer, with a range of -2^31 to 2^31-1. (unsigned = 0 to 2^32-1)
* **long**: a 64-bit signed integer, with a range of -2^63 to 2^63-1.
* **float**: a single-precision 32-bit floating point number.
* **double**: a double-precision 64-bit floating point number.
* **char**: a single Unicode character, represented by a 16-bit integer.
* **boolean**: a boolean value, either **true** or **false**.

Wrapper class = the class/object format of primitive data types.

Instead of long – Long

Instead of int – Integer

Instead of double - Double

**Notes while experimenting:**

var x;

x=10;

x= false;

0101010100

Binary = decimal

Xyz = ( z \* 2^pos.of.z ) + (y \* 2^pos.of.y) + (x \* 2^pos.of.x)

Position starts from right to left, starting from 0.

111 = (1 \* 2^0) + (1 \* 2^1) + (1 \* 2^2) = 1 + 2 + 4 = 7

0 = 0 \* 2^0 = 0 \* 1 = 0

1 = 1 \* 2^0 = 1

01 = (1 \* 2^0) + (0\*2^1) = 1 + 0 = 1

11 = (1\* 2^0) + (1\* 2^1) = 1 + 2 = 3

000 = 0 + 0 + 0 = 0

001

111 = 1\*2^0 + 1\*2^1 + 1\*2^2 = 1+ 2 + 4 = 7 = (2^3 - 1)

UnSigned values:

Max value = 2^no.of.bits – 1

Min value = 0

Signed value:

000 = +0

111 = -3

011 = +3

11111111 = -128

00000000 = +0

01111111 = +127

-------------------------------------------------------------------------

1. **Non-primitive data types**: These are the data types that are **derived from the primitive data** types or created by the programmer. They are as follows:

* **String**: a sequence of characters.
* **Arrays**: a collection of elements of the same type.
* **Class**: a user-defined data type.
* **Interface**: a collection of abstract methods.
* **Enum**: a special data type used to define a set of constants.

It's worth noting that Java also provides some additional data types such as **BigInteger**, **BigDecimal**, **Date**, **Calendar**, and more through its standard libraries. These data types can be used to store and manipulate values that are not represented by the primitive or non-primitive data types in Java.

non-primitive data types and wrapper classes are not exactly the same, although they are related concepts in Java.

Non-primitive data types in Java are data types that are not directly supported by the Java programming language, but are instead defined by the programmer or provided through Java libraries. Examples of non-primitive data types include **String**, **Arrays**, **Class**, **Interface**, and **Enum**.

Wrapper classes, on the other hand, are a set of classes in Java that provide a way to wrap primitive data types in an object. Each primitive data type has a corresponding wrapper class, which allows you to perform operations on primitive values as if they were objects. The wrapper classes are **Byte**, **Short**, **Integer**, **Long**, **Float**, **Double**, **Character**, and **Boolean**.

So, while non-primitive data types and wrapper classes are related, they are not exactly the same. Non-primitive data types are a broader concept that includes all user-defined and library-provided data types, **while wrapper classes are specifically used to wrap primitive data types in objects.**