### Data Types in Java http://zetcode.com/lang/java/datatypes/

|  |
| --- |
| * primitive data types * non-primitive data types |

Any data type built-into a programming language is called ‘primitive’ data type. its not a class. Primitve datatype are not derived from other datatypes. **In Java, all types are signed**

1)Byte- 1

2)Short-2

3)Int-4

4)Long-8

5)Float-4

6)Double-8

7)Boolean-1 bit

8)Char-2

**Local variables are slightly different; the compiler never assigns a default value to an uninitialized local variable. If you cannot initialize your local variable where it is declared, make sure to assign it a value before you attempt to use it. Accessing an uninitialized local variable will result in a compile-time error.**

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Default Value** | **Default size** |
| boolean | false | 1 bit |
| char | '\u0000' | 2 byte |
| byte | 0 | 1 byte- -128 to 127 -**number of children a woman gave birth to** |
| short | 0 | 2 byte- -32,768 to 32,767 -usage- **age of human beings** |
| int | 0 | 4 byte |
| long | 0L | 8 byte |
| float | 0.0f | 4 byte- Real numbers measure continuous quantities, like **weight, height, or speed** |
| double | 0.0d | 8 byte |

### Literals/Constants

You may have noticed that the new keyword isn't used when initializing a variable of a primitive type. **Primitive types are special data types built into the language**; **they are not objects created from a class**. Literals are variables whose values remain constant throughout the program. They are also called Constants

boolean result = true;

char capitalC = 'C';

byte b = 100;

short s = 10000;

int i = 100000;

#### Integer Literals

An integer literal is of type long if it ends with the letter L or l; otherwise it is of type int. It is recommended that you use the upper case letter L because the lower case letterl is hard to distinguish from the digit 1.

 int.parseInt("1") doesn't make sense because int is not a class and therefore doesn't have any methods

**long** b = 23\_482\_345\_629L;

#### Double-precison Literals

*double a = 30.444;*

*double a = 34.33d;*

*double a =345.98D;*

During the declaration of double data type, when the value is being assigned, a 'd' or 'D' can be appended. **But this is optional.**

#### Floating-Point Literals

A floating-point literal is of type float if it ends with the letter F or f; otherwise its type is double and it can optionally end with the letter D or d.

double d1 = 123.4;

float f1 = 123.4f;

**Why char uses 2 byte in java and what is \u0000 ?**

2 bytes because because java uses unicode system rather than ASCII code system. \u0000 is the lowest range of unicode system.To get detail about Unicode see below. **highest value:**\uFFFF

If we deal with even larger numbers, we have to use thejava.math.BigInteger class

In situations where we have to work with precise numbers, we can use the BigDecimal class, The float and double types are not quite accurate. To avoid this margin error, we utilize the BigDecimal class. It is used to hold immutable, arbitrary precision signed decimal numbers.

Rounding of decimals; see in code