



**FACULTY OF ENGINEERING**

**Department of Telecommunications and Electronics Engineering**



# VARIABLES

---

**Basic Java Course**

**Advisor: Prof. Thap Thareoun**

---

**Present By : Ngouch HongCheng  
: Saing LymChhun**

# CONTENT

01

INTRODUCTION

---

02

DATATYPES

---

03

TYPE OF VARIABLES

---

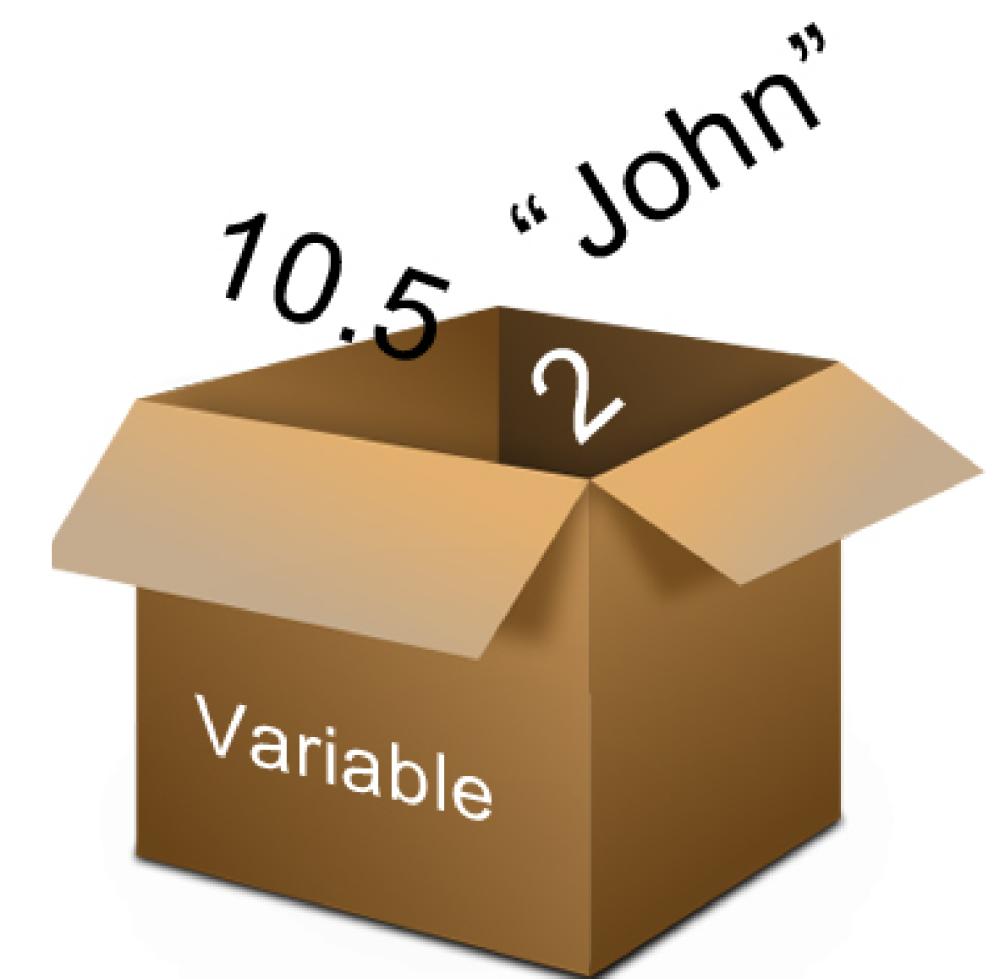
04

PRACTICE

# INTRODUCTION

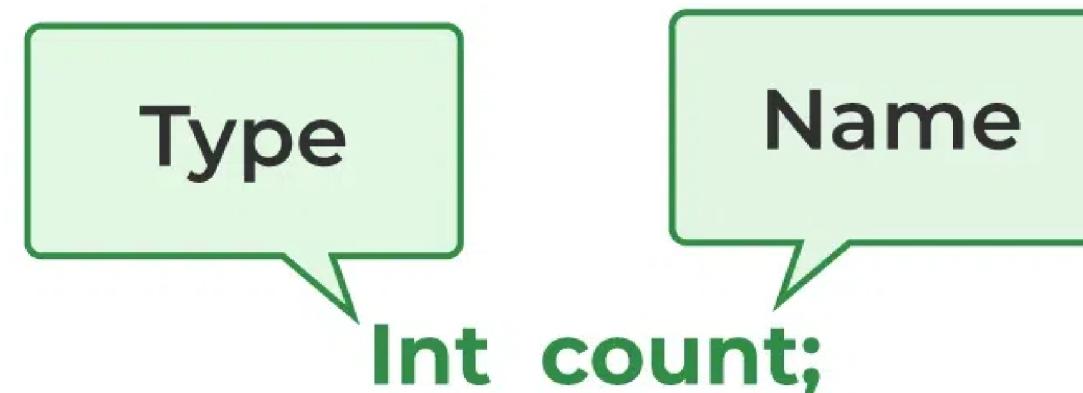
## A. DEFINITION

- **Variables** are the data containers that save the data values during Java program execution.
- Every Variable in Java is assigned a **data type** that designates the type and quantity of value it can hold.
- The value stored in a variable can be changed during program execution.
- Variables in Java are only a name given to a memory location. All the operations done on the variable affect that memory location.

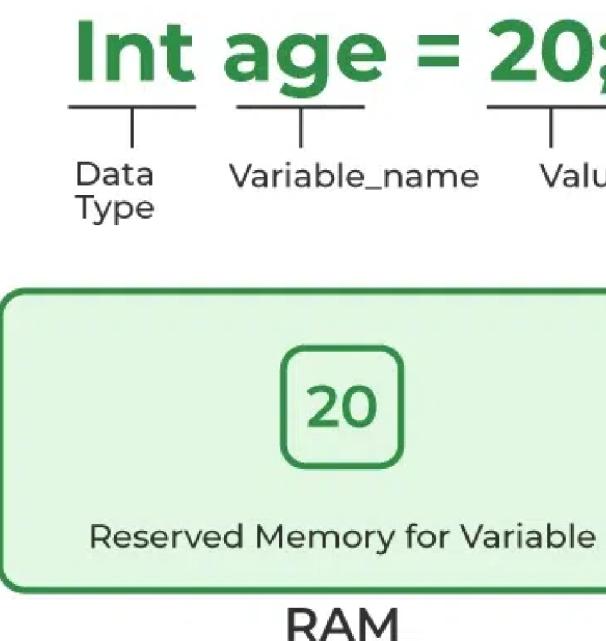


# INTRODUCTION

## B. HOW TO DECLARE VARIABLE



- **datatype:** Type of data that can be stored in this variable.
- **data\_name:** Name was given to the variable.



**Ram(random access memory):** a computer's short-term memory, where the data that the processor is currently using is stored.

# INTRODUCTION

## C. VARIABLE NAME

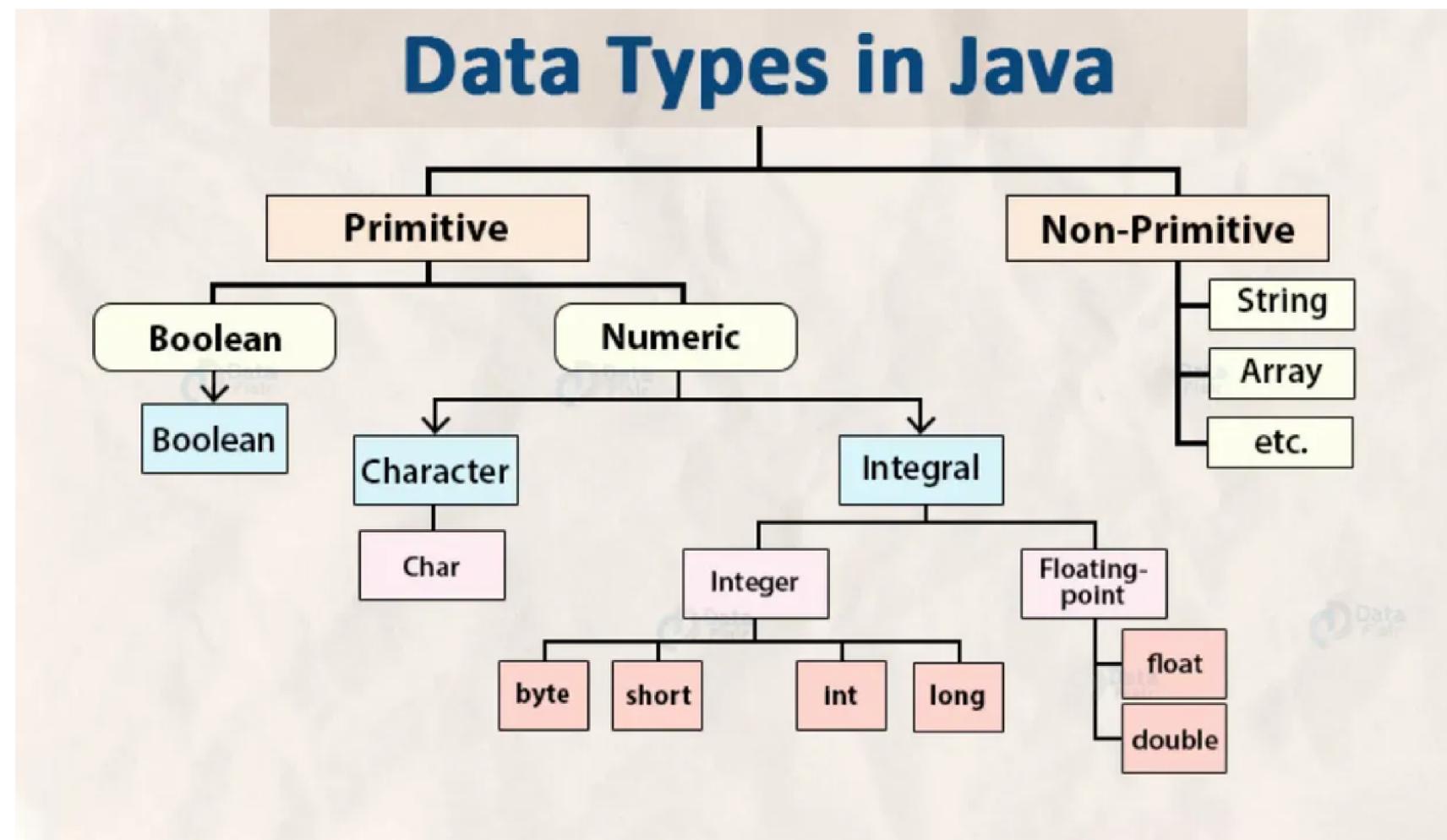
### Rules

- Start from a-z or A-Z and 0-9
- Unable to start with number
- Unable to store symbols like \*, -, /, +, (, ), &, #
- Unable to have the same name as keywords such as **int, float, return,....**
- Unable to leave space, but can use “\_” symbol to represent space



# DATA TYPES

## A. TABLE OF DATA TYPES



<u>data type</u>	<u>size</u>	<u>primitive/ reference</u>	<u>value</u>
★ <b>boolean</b>	1 bit	primitive	true or false
<b>byte</b>	1 byte	primitive	-128 to 127
<b>short</b>	2 bytes	primitive	-32,768 to 32,767
★ <b>int</b>	4 bytes	primitive	-2 billion to 2 billion
<b>long</b>	8 bytes	primitive	-9 quintillion to 9 quintillion
<b>float</b>	4 bytes	primitive	fractional number up to 6-7 digits ex. 3.141592f
★ <b>double</b>	8 bytes	primitive	fractional number up to 15 digits ex. 3.141592653589793
★ <b>char</b>	2 bytes	primitive	single character/letter/ASCII value ex. 'f'
★ <b>String</b>	varies	reference	a sequence of characters ex. "Hello world!"

# DATA TYPES

## B. PRIMITIVE AND NON-PRIMITIVE (REFERENCE)

- **Primitive data types** in Java are fundamental types that directly store simple values like numbers and characters. They are called "primitive" because they hold the actual value directly without involving objects or references.
- **reference data types** don't hold the actual object data but act as pointers to the memory location where the object resides. This allows Java to efficiently manage and manipulate complex objects, arrays, and interfaces.

# DATA TYPES

## B. PRIMITIVE AND NON-PRIMITIVE (REFERENCE)

### Primitive

- 8 types (bool, int, byte, etc.)
- stores data
- can only hold 1 value
- less memory

### Reference

- unlimited (user defined)
- stores an address
- could hold more than 1 value
- more memory

# DATA TYPES

## C. BOOLEAN

- Java has a boolean data type, which can only take the values true or false
- **boolean** data type represents only one bit of information either true or false which is intended to represent the two truth values of logic and Boolean algebra, but the size of the **boolean** data type is virtual machine-dependent.



Syntax

bool good;

# DATA TYPES

## D. BYTE TYPE

- A byte in Java is 8 bits. It is a primitive data type, meaning it comes packaged with Java.
- Bytes can hold values from -128 to 127.
- No special tasks are needed to use it; simply declare a byte variable and you are off to the races.

Syntax

byte byteVar;

# DATA TYPES

## E. INTEGER

- int data type is a 32-bit signed two's complement integer, which has a minimum value of -2<sup>31</sup> and a maximum value of 2<sup>31</sup>
- In Java SE 8 and later, we can use the int data type to represent an unsigned 32-bit integer, which has a value in the range [0, 2<sup>32</sup>-1]. Use the Integer class to use the int data type as an unsigned integer.

### Syntax

```
int intVar;
```

# DATA TYPES

## F. SHORT

- The short data type is a 16-bit signed two's complement integer.
- Similar to byte, use a short to save memory in large arrays, in situations where the memory savings actually matters.

Syntax

short shortVar;

# DATA TYPES

## G. LONG DATA TYPE

- The range of a long is quite large.
- The long data type is a 64-bit two's complement integer and is useful for those occasions where an int type is not large enough to hold the desired value.
- The size of the Long Datatype is 8 bytes (64 bits).

Syntax

long longVar;

# DATA TYPES

## H. FLOAT DATA TYPES

- The float data type is a single-precision 32-bit IEEE 754 floating-point.
- Use a float (instead of double) if you need to save memory in large arrays of floating-point numbers.
- The size of the float data type is 4 bytes (32 bits).

Syntax

float floatVar;

# DATA TYPES

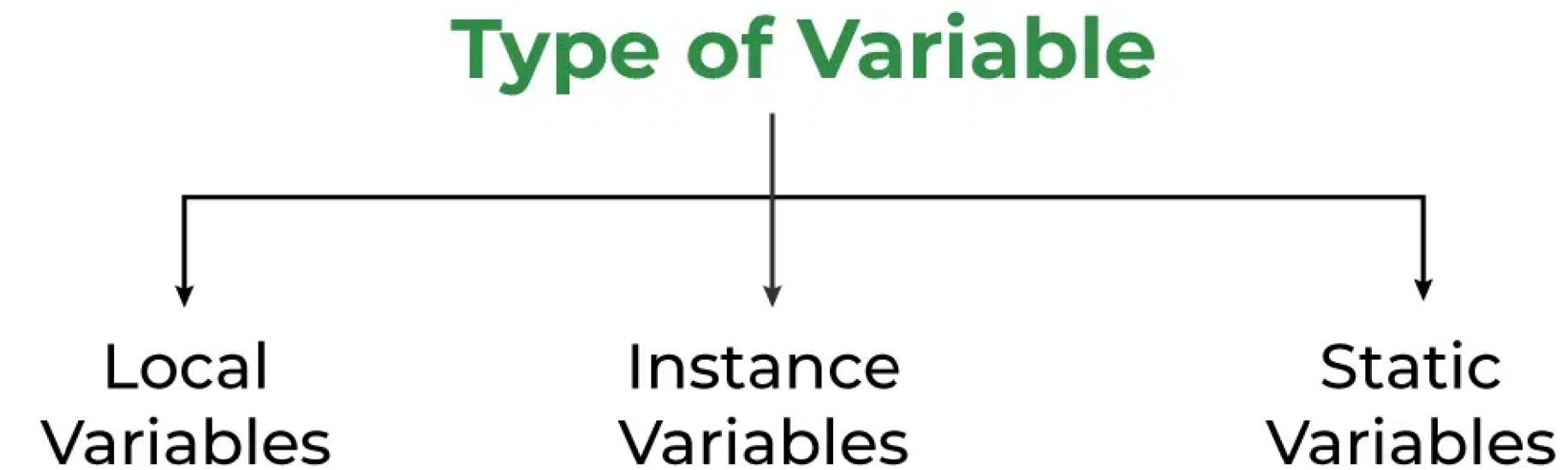
## I. DOUBLE DATA TYPE

- The double data type is a double-precision 64-bit IEEE 754 floating-point.
- For decimal values, this data type is generally the default choice.
- The size of the double data type is 8 bytes or 64 bits.
- Note: Both float and double data types were designed especially for scientific calculations, where approximation errors are acceptable. If accuracy is the most prior concern then, it is recommended not to use these data types and use BigDecimal class instead.

Syntax

double doubleVar;

# TYPE OF VARIABLES



# PRACTICE

---

- Try testing this code

```
public class Simple{  
    public static void main(String[] args){  
        int a=10;  
        float b=10;  
        char c='K';  
    }  
}
```

- Try changing those datatype and add more data type to the program above

**END**