**NOTED: KEBINGUNGAN JAVA**

**Variables, Data Types and structure:**

**Fixed Values** in our program→called **Literals**

contoh:

final int test = 0;

char testDua = ‘2’;

The eight **primitive data types** are: **byte, short, int, long, float, double, boolean, and char**. The [java.lang.String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html) class represents character strings.

There are **4 types of variables** in Java programming language:

* **Instance Variables** (Non-Static Fields)
* **Class Variables** (Static Fields)
* **Local Variables:**

**yaitu variable yang ada didalam method. Yang tidak memiliki modifier.**

Contoh yang berwarna merah:

public static void main(String[] args→Parameter){

int dataMahasiswaITB = 120; // siswa→local V.

}

* **Parameters** : variable in the fileds/method

### Integer Literals

An integer literal is a numeric value(associated with numbers) without any fractional or exponential part. There are 4 types of integer literals in Java:

1. binary (base 2)
2. decimal (base 10)
3. octal (base 8)
4. hexadecimal (base 16)

In Java, binary starts with 0b, octal starts with 0, and hexadecimal starts with 0x.

Note: **Integer literals** are used to initialize variables of integer types like **byte**, **short**, **int**, and **long**.

// binary

int binaryNumber = 0b10010;

// octal

int octalNumber = 027;

// decimal

int decNumber = 34;

// hexadecimal

int hexNumber = 0x2F; // 0x represents hexadecimal

// binary

int binNumber = 0b10010; // 0b represents binary

**Array**

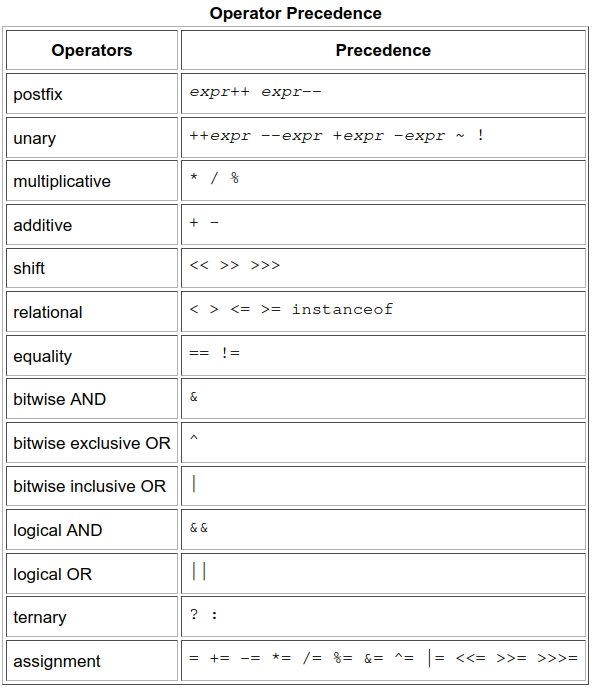
Array adalah tempat penyimpanan object yang berisikan nilai-nilai angka yang sudah pasti dari satu tipe.

Array adalah **himpunan**,**Kumpulan data.**

Di java array adalah class.

Menurut wikipedia: “array/larik adalah suatu **tipe data terstruktur** yang dapat **menyimpan banyak data** dengan suatu nama yang sama dan **menempati tempat di memori** yang berurutan serta bertipe data sama pula.”

**Operators**

****

source: oracle.com

**Unary Operator**

The increment/decrement operators can be applied before (prefix) or after (postfix) the operand. Prefix: ++1, Postfix: 1++, PreFix tidak memerlukan untuk diprint.out ulang untuk menambah suatu angka. Sedangkan PostFix Sebaliknya.

**Bitwise Operator**

instanceOf → salah satu dari operator bitwise. Operator bitwise merupakan operator yang digunakan untuk memanipulasi bilangan bit.

1 byte = 8 bit → 00000000 = 0

→ 00000001 = 1

→ 00000010 = 2

→ 00000011 = 3, dst.

Operator logika dalam bitwise: &, | , ^, ~(NOT)