

Java Tokens . A Java program is collection of tokens, comments and white space. It includes five types of tokens.

① Reserved keyword → These are an essential part of program. Java language has reserved 50 words as keywords. All keywords are to be written in lower-case letters.

|          |        |         |           |
|----------|--------|---------|-----------|
| abstract | case   | boolean | default   |
| byte     | const  | catch   | enum      |
| class    | double | public  | interface |
| import   | static | this    | package   |

② identifier → They are used for naming classes, methods, variables, objects, package & interfaces in a program.

Rules → ① They can have alphabets, digits, and underscore, \$

② They must not begin with a digit

③ Uppercase & lowercase letters are distinct

④ They can be of any length.

Id → ① average, sum → public method & instance

② total\_student → more than one word

③ Student → class name with capital letter

④ TOTAL; V\_MAX → for constant value

all these are conventions, not rules.

③ Literals → These are sequence of characters (digit, letters & other characters) that represent constant values to be stored in variables. It specifies five major types of literals.

① Integer literals

② floating literals

③ character literals

④ String literals

⑤ Boolean literals

④ operators → which perform some operations to produce result, Arithmetic, Relational, logical etc.

5. Separators  $\rightarrow$   $()$   $\rightarrow$  Parentheses

{ } → Braces

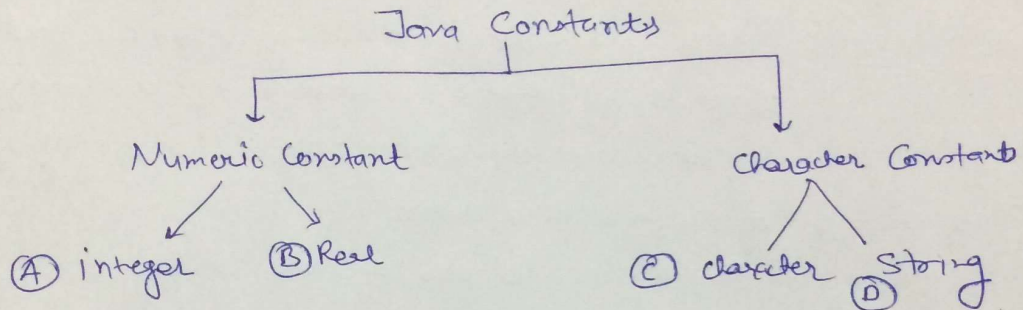
[ ] → Brackets

; → semicolon

→ Comma

→ Period

Constants → refer to fixed values that do not change during execution of program.



① decimal integer —  $0 \rightarrow 9 \Rightarrow 490$

Octal integer  $- 0 \rightarrow 7 \Rightarrow 370$

hexadecimal integer - 0-9, 10-15  $\Rightarrow$  0x2, 0x9F  
A - F

⑧  $215.23, 0.093, -71, \frac{2.15 \times 10^2}{\substack{\downarrow \text{mantissa} \quad \searrow \text{exponent}}} \rightarrow 2.15 \times 10^2$

⑦ Single character enclosed within a pair of single quote  
'5', 'x', 'a'

(d) sequence of character enclosed within double quotes.

Backslash character constants → 'b' — back space

$1/12$  — new line

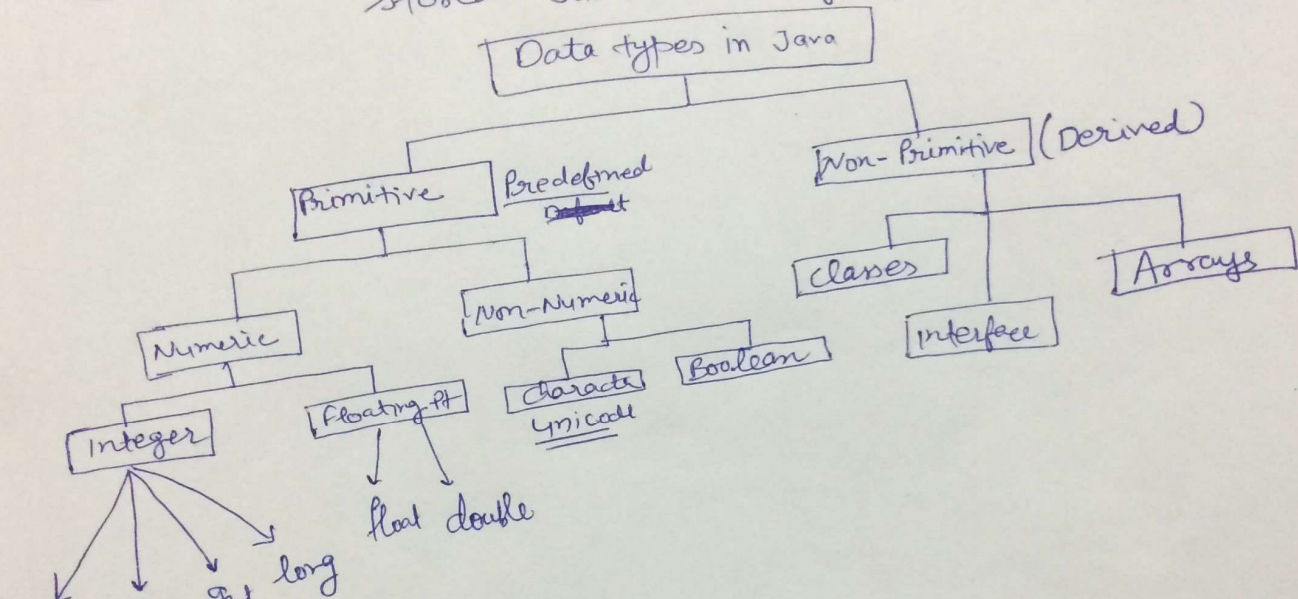
'\t' — horizontal tab



Variables → It is an identifier that denotes a storage location <sup>⑤</sup> used to store a data value. It takes different values at different times during execution of program.

for example, average, average\_of\_number X  
 int X, marks X,  
total & Total → are different

Data types → It specifies the size and type of values that can be stored. Java is strongly typed language.



| ① | Type  | Size | Minimum Value | Max value |
|---|-------|------|---------------|-----------|
|   | Byte  | 1B   | -128          | 127       |
|   | Short | 2B   | -32768        | 32767     |
|   | int   | 4B   | —             | —         |
|   | long  | 8B   | —             | —         |

Range →  $-2^{n-1} \rightarrow 2^{n-1} - 1$

byte →  $-2^{8-1} \rightarrow 2^{8-1} - 1 \rightarrow -2^7 \rightarrow 2^7 - 1$   
 → -128 to 127

| ② | Type   | Size   | Min value              | Max value             |
|---|--------|--------|------------------------|-----------------------|
|   | float  | 4 byte | $3.4 \times 10^{-38}$  | $3.4 \times 10^{38}$  |
|   | double | 8 byte | $1.7 \times 10^{-308}$ | $1.7 \times 10^{308}$ |

character Type - char  $\frac{2B}{\text{unicode}}$   $\rightarrow$  It can hold only a single character. <sup>(6)</sup>  
(in C++, 1B - ASCII)

Boolean Type - boolean 1 bit  $\rightarrow$  It can hold only value True or false  
1 0

\* declaration and assigning a value to variable.

```
int x, y = 10;  
float m = 10.5, n = 12.6;  
double a = 75.36;  
char value = 'a';
```

\* Symbolic Constant  $\rightarrow$  It is used to store the constant value which does not change throughout the program.

final type symbolic-name = value;

```
final int PASS_MARK = 40;
```

```
final int TOTAL_MARK = 100;
```

Note 1  $\rightarrow$  ① It is written in Capital letter to distinguish them from normal variable name. It is convention, not rule.

② After declaration, they should not be assigned any other value within the program. i.e. `PASS_MARK = 33;`

③ Symbolic constants are declared for types. This is not done in C & C++ i.e. `#define PASS_MARK = 40;`

④ They should be used only as class member in the beginning of class.