[Naming Conventions and data types]

naming conventions:- Rules to be followed by a Java Programmer while writing names of <u>Packages</u>, <u>Classes</u> and <u>methods</u>.

① Packages:- represents a sub directory that contain group of class and interface.

- always written in <u>small letters</u>, as
  → java.awt
  → java.io
  → java.swing

(ii) class:- It is the model for creating an object. It specifies the Properties and actions of objects. (also interface too)

- start with <u>Capital letter</u>
  → String
  → <u>DataInputstream</u>
  → <u>ActionListener</u>

(iii) method :- first word is <u>small letter</u> then from next word <u>Capital letter</u> as,
  → Print<u>Ln</u>()
  → read<u>Line</u>()
  → get<u>Number</u><u>Instance</u>()

(also same for variable ↑ -too)

① keyword should always be in <u>small letter</u>

as → <u>Public</u>
       <u>void</u>
       <u>static</u>

Data-types (in java)

① **Integer data types:-** used to represent any integer no.

  ⓘ byte → 1 byte ⟶ -128 to 127

  ⓘⓘ short → 2 byte ⟶ -32,768 to 32,767

  ⓘⓘⓘ int → 4 byte ⟶ -2147483648 to 2147483647

  ⓘⱽ long → 8 byte ⟶ -9223372036854775808 to . . . . .

② **Float data type :-** used to represent numbers using **decimal point**.

  ⓘ float → 4 byte (7 digit accuracy after decimal point)

  ⓘⓘ double → 8 byte (15 digit ")

    float Pi = 3.142 f

(here f is used as Jvm would allot 8 byte assuming it the be a double) (default value is taken as double so).

③ **character data type:-** Single character representation.

  char → **2 byte** (**0** to **65535**)

why 2 byte?

⇒ as Javasoft People want to use any character not only english but all other human language, it will be advantage on internet.

unicode?

⇒ System uses an encoding standard that provide a unique no for every character, no matter what the platform, program, or language. uses 2 byte.

(iv) **String data type :-** represent a group of character.

eg→ String str = "new delhi";

① String is a class, why it is taken as Data type?
⇒ Every class is a data-type and is called as user-defined data-type.

(v) **Boolean data-type :-** Represent two value-either true or false (uses 1 bit)

eg→ boolean response = true;

① **literals:** value that is stored into a variable directly in the Program.

eg→ int num = 10;

data ⌞ᴛ⌟ ⌞ᴛ⌟ literals
type variable