

# → Java Programming 2025 :

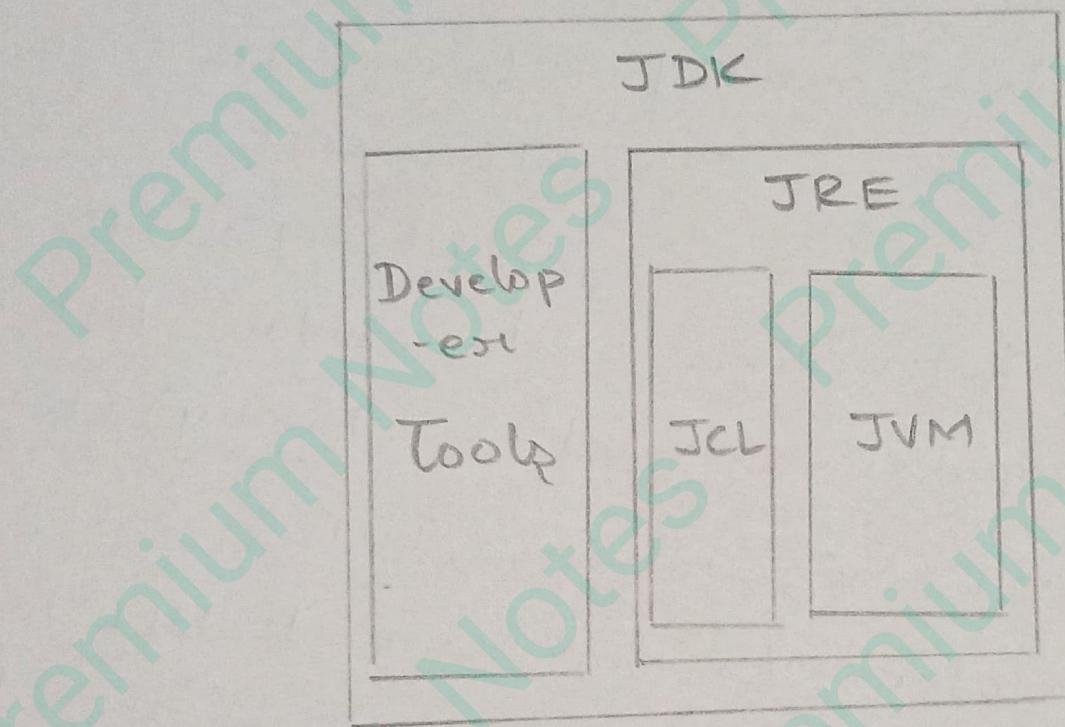
## Module 01 → Java Basics:

- \* I/O device - gets input (eg - keyboard, mouse etc...)
- \* O/P device - gives output (eg - monitor, speaker etc...)
- \* memory - cache, main memory and hard disk.
- \* CPU - central processing unit (also has cache)
- \* RAM - random access memory (main memory)
- \* Secondary storage device - hard disk, CD, DVD, USB etc...
- \* volatile memory - main memory, cache
- \* OS - controller of system (software) [eg: windows, linux...]
- \* desirable features of memory -
  - ① low access time
  - ② low cost
  - ③ high capacity
- \* program - set of instructions to do certain tasks.
- \* language of computer - 0's and 1's (binary) (low level)
- \* programming lang. - C, C++, Java, Python... (high level)
- \* compiler/interpreter - s/w that converts HLL to LLL and
  - compiler → takes whole code whereas
  - interpreter → line by line conversion.
- \* platform - OS + architecture (eg - windows + AMD / windows + intel)
- \* Java is platform independent  $\Rightarrow$  it generates an intermediate bytecode which can run on any platform that has machine specific JVM installed in it.
- \* JDK - Java Development Kit
- \* JRE - Java Runtime Environment
- \* JCL - Java Class Library
- \* other platform independent prog. lang. →
  - ① python ✓
  - ② C# ✓
  - ③ Scala
- \* prog. lang. of Android → Java

- \* Java is simple and have clear rules.
- \* Java is highly secure → JVM acts as a monitor.
- \* Java is statically typed → datatype of variables are declared in program itself & doesn't get designated at run-time → runs faster.

\* Java has out of Bound index of Bound checking, bigger garbage collection, better multi-thread reading and so on....

### \* Java Environment -



\* Comments - used for code readability & understanding by other users - ① single line - //...  
② multi line - /\* ... \*/

→ **public**      **static**      **void**      **main(String[] args)**  
 access      special      return      array of  
 specifier      methods      type of      strings that  
 that can be      main()      main()      are arguments  
 called      (decided      for the main  
 without      by Java)      function  
 creating  
 objects

*/\* Refer GitHub repository for theoretical notes for basics, features, comparison, use cases etc... of Java. \*/*

Program 001 → HelloWorld Program -

i/p → class HelloWorld{  
    public static void main(String [] args){  
        System.out.println("Hello World");  
    }  
}

o/p → Hello World [NOTE - underlined characters must be in capitals, as they represent class names.]