**Differentiate JVM JRE JDK JIT**

Java Virtual Machine (JVM) is an abstract computing machine. Java Runtime Environment (JRE) is an implementation of the JVM. Java Development Kit (JDK) contains JRE along with various development tools like Java libraries, Java source compilers, Java debuggers, bundling and deployment tools.

JVM becomes an instance of JRE at runtime of a java program. It is widely known as a runtime interpreter. The Java virtual machine (JVM) is the cornerstone on top of which the Java technology is built upon. It is the component of the Java technology responsible for its hardware and platform independence. JVM largely helps in the abstraction of inner implementation from the programmers who make use of libraries for their programmes from JDK.

## JVM Internals

Like a real computing machine, JVM has an instruction set and manipulates various memory areas at run time. Thus for different hardware platforms one has corresponding implementation of JVM available as vendor supplied JREs. It is common to implement a programming language using a virtual machine. Historicaly the best-known virtual machine may be the P-Code machine of UCSD Pascal.

A Java virtual machine instruction consists of an opcode specifying the operation to be performed, followed by zero or more operands embodying values to be operated upon. From the point of view of a compiler, the Java Virtual Machine (JVM)is just another processor with an instruction set, Java bytecode, for which code can be generated. Life cycle is as follows, source code to byte code to be interpreted by the JRE and gets converted to the platform specific executable ones.

## Sun’s JVM

Sun’s implementations of the Java virtual machine (JVM) is itself called as JRE. Sun’s JRE is availabe as a separate application and also available as part of JDK. Sun’s Java Development Tool Kit (JDK) comes with utility tools for byte code compilation “javac”. Then execution of the byte codes through java programmes using “java” and many more utilities found in the binary directory of JDK. ‘java’ tools forks the JRE. Implementation of JVMs are also actively released by other companies besides Sun Micro Systems.



## JVM for other languages

A JVM can also be used to implement programming languages other than Java. For example, Ada source code can be compiled to Java bytecode, which may then be executed by a Java virtual machine (JVM). That is, any language with functionality that can be expressed in terms of a valid class file can be hosted by the Java virtual machine (JVM). Attracted by a generally available, machine-independent platform, implementors of other languages are turning to the Java virtual machine (JVM) as a delivery vehicle for their languages. PHP with Quercus is such an example.

## Just-in-time Compiler (JIT)

JIT is the part of the Java Virtual Machine (JVM) that is used to speed up the execution time. JIT compiles parts of the byte code that have similar functionality at the same time, and hence reduces the amount of time needed for compilation. Here the term “compiler” refers to a translator from the instruction set of a Java virtual machine (JVM) to the instruction set of a specific CPU.