1. JDK (Java Development Kit) is a Kit that provides the environment to **develop and execute(run)** the Java program. JDK is a kit(or package) that includes two things

- Development Tools (to provide an environment to develop your java programs)
- JRE (to execute your java program).

2. JRE (Java Runtime Environment) is an installation package that provides an environment to **only run (not develop)** the java program (or application) onto your machine. JRE is only used by those who only want to run Java programs that are end-users of your system.

3. JVM (Java Virtual Machine) is a very important part of both JDK and JRE because it is contained or inbuilt in both. Whatever Java program you run using JRE or JDK goes into JVM and JVM is responsible for executing the java program line by line, hence it is also known as an **interpreter.**

JSE - Java Standard Edition

Java Platform, Standard Edition or Java SE is a widely used platform for programming in the Java language. It is the Java Platform used to deploy portable applications for general use. In practical terms, Java SE consists of a virtual machine, which must be used to run Java programs, together with a set of libraries (or "packages") needed to allow the use of file systems, networks, graphical interfaces, and so on, from within those programs.

JME - Java Micro Edition

Java Platform, Micro Edition, or Java ME, is a Java platform designed for embedded systems (mobile devices are one kind of such systems) . Target devices range from industrial controls to mobile phones (especially feature phones) and set-top boxes.

JEE - Java Enterprise Edition

The Java platform Enterprise Edition differs from the Java Standard Edition Platform (JSE) in that it adds libraries which provide functionality to deploy faulttolerant, distributed, multi-tier Java software, based largely on modular components running on an application server.

How does java achieve platform independency?

Java is platform-independent because it uses a virtual machine. The Java programming language and all APIs are compiled into bytecodes. Bytecodes are effectively platform-independent. The virtual machine takes care of the differences between the bytecodes for the different platforms. The run-time requirements for Java are therefore very small. The Java virtual machine takes care of all hardware-related issues, so that no code has to be compiled for different hardware.

features of Java

Following are the notable features of Java:

Object Oriented

In Java, everything is an Object. Java can be easily extended since it is based on the Object model.

Platform Independent

Unlike many other programming languages including C and C++, when Java is compiled, it is not compiled into platform specific machine, rather into platform-independent byte code. This byte code is distributed over the web and interpreted by the Virtual Machine (JVM) on whichever platform it is being run on.

Simple

Java is designed to be easy to learn. If you understand the basic concept of OOP Java, it would be easy to master.

Secure

With Java's secure feature it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption.

Architecture-neutral

Java compiler generates an architecture-neutral object file format, which makes the compiled code executable on many processors, with the presence of Java runtime system.

Portable

Being architecture-neutral and having no implementation dependent aspects of the specification makes Java portable. The compiler in Java is written in ANSI C with a clean portability boundary, which is a POSIX subset.

Robust

Java makes an effort to eliminate error-prone situations by emphasizing mainly on compile time error checking and runtime checking.

Multithreaded

With Java's multithreaded feature it is possible to write programs that can perform many tasks simultaneously. This design feature allows the developers to construct interactive applications that can run smoothly.

Interpreted

Java byte code is translated on the fly to native machine instructions and is not stored anywhere. The development process is more rapid and analytical since the linking is an incremental and light-weight process.

High Performance

With the use of Just-In-Time compilers, Java enables high performance.

Distributed

Java is designed for the distributed environment of the internet.

Dynamic

Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry an extensive amount of run-time information that can be used to verify and resolve accesses to objects at run-time.

```
package com.masaijava111;

public class Assign_1 {
	public static void main(String[] args) {
		System.out.println("Nmae : Vikas Kumar Gautam");
		System.out.println("Father’s Nmae : Mr. Ram Adhar");
		System.out.println("Mother ‘s Nmae : Mrs. Rama Devi");
		System.out.println("Age : 26 years");
		System.out.println("Gender  : Male");
		System.out.println("Address   : Uttar Pradesh");
		System.out.println("Mobile No.  : 8957411197");
	}
}
```