

Java Architecture

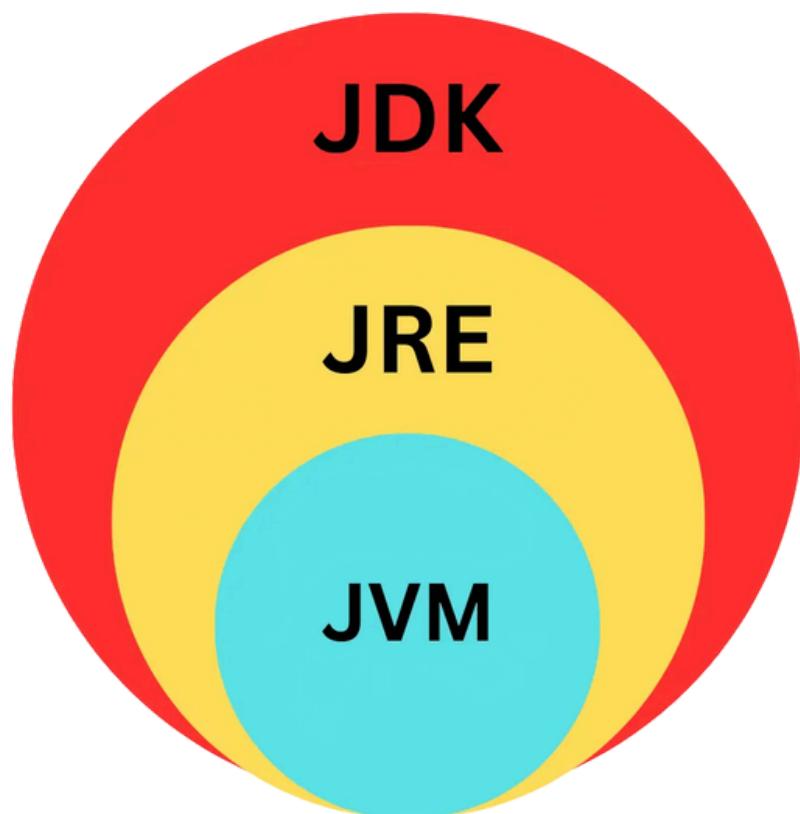
Java architecture defines how Java programs are executed – from source code to output.

It includes components like:

JDK (Java Development Kit)

JRE (Java Runtime Environment)

JVM (Java Virtual Machine)



JDK : To develop and run java applications the required environment is JDK.

JRE : To run java application the required environment is JRE.

JVM : To execute java application the required virtual machine is JVM.

JDK (Java Development Kit)

It's the complete package required to develop and Execute Java Applications..

It contains:

JRE (to execute Java programs)

Development tools (compiler, debugger, jar, etc.)

Main Tools in JDK:

- **javac → Java Compiler (Converts Java source code into bytecode)**
- **java → Launches JVM**
- **javadoc → Generates documentation**
- **jar → Packages files into JARs**

JRE (Java Runtime Environment)

- It provides libraries, JVM, and other files necessary to run Java applications.
- It does not include development tools like compiler (javac).
- In short: JRE = JVM + Java Class Libraries

It Consists of :

1. **Java Virtual Machine (JVM):** The execution engine
2. **Java class libraries:** Standard libraries for common functionality
3. **Integration libraries:** For database connectivity, XML processing, etc.

JVM (Java Virtual Machine)

The heart of Java – it executes Java bytecode.

JVM makes Java platform-independent because the JVM on each OS knows how to convert bytecode into machine-specific instructions.

Java is platform-independent and JVM is platform-dependent

JVM Responsibilities:

- 1. Loading:** Loads class files into memory.
- 2. Verification:** Checks bytecode for security and errors.
- 3. Execution:** Uses Interpreter and JIT Compiler to execute bytecode efficiently.
- 4. Memory Management:** Uses Garbage Collector to free unused objects automatically.

JVM Architecture

