

Understanding Java Architecture

An Overview of the Java Platform

Presented by: Manikandan C

Date: October 22, 2024

Introduction to Java

- Java is a high-level, object-oriented programming language.
- It is platform-independent due to the Java Virtual Machine (JVM).
- Key Features:
 - Write Once, Run Anywhere (WORA)
 - Strongly typed language
 - Automatic memory management with Garbage Collection

Java Architecture Overview

- Main Components:
 - Java Development Kit (JDK)
 - Java Runtime Environment (JRE)
 - Java Virtual Machine (JVM)
- Layered Architecture:
 - Application Layer
 - Java API Layer

Java Development Kit (JDK)

- JDK is a software development kit used to develop Java applications.
- Components:
 - Compiler (javac)
 - Java Runtime Environment (JRE)
 - Development Tools (e.g., javadoc, jar, etc.)
- Used for writing, compiling, and debugging Java programs.

Java Runtime Environment (JRE)

- JRE provides the environment to run Java applications.
- Components:
 - JVM (Java Virtual Machine)
 - Core Libraries (Java API)
- JRE is for running Java programs, without development tools.

Java Virtual Machine (JVM)

- JVM is the runtime engine that executes Java bytecode.
- Key Functions:
 - Converts bytecode to machine code
 - Provides memory management
 - Handles garbage collection
- Enables platform independence for Java applications.

Java API

- A collection of classes and interfaces providing functionality for Java applications.- Key Packages :
- java.lang (core classes)
- java.util (utilities)
- java.io (input/output)
- java.net (networking)
- Simplifies development with ready-to-use functionality.

Java Architecture Diagram

- Visual representation of the relationship between JDK, JRE, and JVM.
- Layers: Application Layer, API Layer, JVM Layer.

Compilation and Execution Process

- Steps:
 1. Write Java code (source file).
 2. Compile to bytecode using javac.
 3. Execute bytecode on the JVM.
- Flowchart illustrating the compilation and execution process.

Conclusion

- Java's architecture enables platform independence and ease of development.
- Understanding JDK, JRE, and JVM is crucial for Java developers.
- Java continues to evolve with regular updates and strong community and support.

