# JAC444 - Lecture1

Introduction to

Java Programming Language Segment 1

# Objectives

**Upon completion of this lecture, you should be able to:**

* Understand Java Technology
* Use Java Platforms
* Know Basic Elements of Java Programming Language
* Write your First Java Program

# Understanding Java Technology

Java Technology is defined by two elements:

* Java Platform
* Java Programming Language

**Java Platforms**

**In this segment you will be learning about:**

* What is a Java Platform
* Types of Java Platforms: Java ME, Java SE, Java EE
* History of Java Platform Standard Edition (Java SE)
* How to use Java SE

**Java Platforms**

* Java Platform has two components:
  1. Java Virtual Machine.
  2. Java Application Programming Interface (API)

1. Java Virtual Machine (JVM).

•The Java virtual machine is an abstract computing machine.

1. Java API – Libraries of Related Units of Functionality

•Grouping of related interfaces and classes.

These libraries are known in Java as packages

**How to Use Java Platform**

* Learn Java Programming Language Concepts, Vocabulary, Syntax, Semantics and Patterns.
* Write program and save your source code in files according with the Java Programming Language rules.
* Learn the structure of Java API packages that come with platform.
* Use classes already defined in the packages according with your program design. Do not invent the wheel!
* Develop your software using tools from Java SE.
* Compile using java compiler: ***javac MyProgramName.java***
* Run your program using JVM: ***java MyProgrmName***

**Steps to Use Java SE**

* **Coding***:* Java code is produced by the programmer**.**
* **Compiling***:* Build the Java program into bytecode – result is a ".class" file.

COMPILER: **javac**

* **Running**: The class file is loaded by the JVM with its attached digital signature. JAVA VIRTUAL MACHINE (JVM): **java**
* **Bytecode Verification***:* The JVM verifies the class file digital signature. The JVM is simply an interpreter.
* **Internal Integrity Check***:* It checks if loaded Java program is well formed. Data types are verified along with other syntax structure.
* **Execution***:* Program execution begins from the *main* entry point.