

# Lecture One

# Introduction to Java

# History of Java

- ✓ Java is a general purpose object oriented programming language.
- ✓ Developed by Sun Microsystems. (James Gosling)
- ✓ Initially called “Oak” but was renamed as “Java” in 1995.
- ✓ Initial motivation is to develop a platform independent language to create software to be embedded in various consumer electronics devices.
- ✓ Become the language of internet. (portability and security).

# Features of Java

1. Simple, Small and Familiar
2. Compiled and Interpreted
3. Object Oriented
4. Platform Independent and portable
5. Robust and Secure
6. Distributed / Network Oriented
7. Multithreaded and Interactive
8. High Performance
9. Dynamic

# Simple, Small and Familiar

- ❑ Similar to C/C++ in syntax
- ❑ But eliminates several complexities of
  - ❑ No operator overloading
  - ❑ No direct pointer manipulation or pointer arithmetic
  - ❑ No multiple inheritance
  - ❑ No malloc() and free() – handles memory automatically

# Compiled and Interpreted

❏ Java works in two stage

❏ Java compiler translate the source code into byte code.

❏ Java interpreter converts the byte code into machine level representation.

## Byte Code:

-A highly optimized set of instructions to be executed by tehe java runtime system, known as java virtual machine (JVM).

-Not executable code.

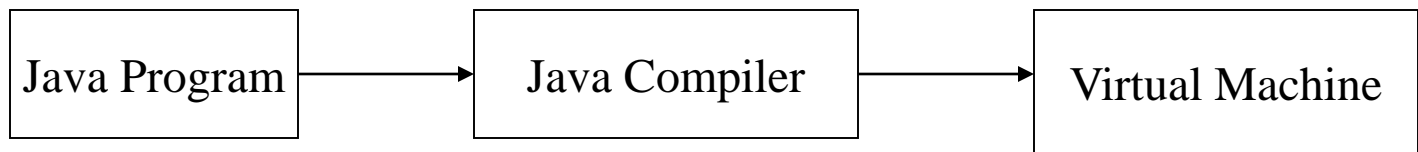
## Java Virtual Machine (JVM):

- Need to be implemented for each platform.

- Although the details vary from machine to machine, all JVM understand the same byte code.

# Java Virtual Machine

- ✓ Java compiler produces an intermediate code known as byte code for a machine, known as JVM.
- ✓ It exists only inside the computer memory.



- ✓ Machine code is generated by the java interpreter by acting as an intermediary between the virtual machine and real machine.



# Object Oriented

- ❑ Fundamentally based on OOP

- ❑ Classes and Objects

- ❑ Efficient re-use of packages such that the programmer only cares about the interface and not the implementation

- ❑ The object model in java is simple and easy to extend.

# Platform Independent and Portable

- ❏ “Write-Once Run-Anywhere”
- ❏ Changes in system resources will not force any change in the program.
- ❏ The Java Virtual Machine (JVM) hides the complexity of working on a particular platform
  - ❏ Convert byte code into machine level representation.



# Robust and Secure

- ☐ Designed with the intention of being secure
  - ☐ No pointer arithmetic or memory management!
  - ☐ Strict compile time and run time checking of data type.
  - ☐ Exception handling
  - ☐ It verify all memory access
  - ☐ Ensure that no viruses are communicated with an applet.

# Distributed and Network Oriented

- ▶ Java grew up in the days of the Internet
  - Inherently network friendly
  - Original release of Java came with Networking libraries
  - Newer releases contain even more for handling distributed applications
    - RMI, Transactions

# Multithreaded and Interactive

- ❑ Handles multiple tasks simultaneously.
- ❑ Java runtime system contains tools to support multiprocess synchronization and construct smoothly running interactive systems.

# High Performance

- ❑ Java performance is slower than C
- ❑ Provisions are added to reduce overhead at runtime.
- ❑ Incorporation of multithreading enhance the overall execution speed.
- ❑ Just-in-Time (JIT) can compile the byte code into machine code.
- ▶ Can sometimes be even faster than compiled C code!

# Dynamic

- ❏ Capable of dynamically linking a new class libraries, methods and objects.
- ❏ Java can use efficient functions available in C/C++.
- ❏ Installing new version of library automatically updates all programs

# Language of Internet Programming

- Java Applets
- Security
- Portability

## 1. Applets:

Special java program that can transmitted over the network and automatically executed by a java-compatible web browser.

## 2. Security:

Java compatible web browser can download java applets without fear of viral infection and malicious agent.

## 3. Portable:

Java applets can be dynamically downloaded to all the various types of platforms connected to the internet

# Why portable and Secure?

- - ❑ The output of java compiler is not executable code.
  - ❑ Once JVM exists for a platform, any java program can run on it.
  - ❑ The execution of byte code by the JVM makes java programs portable.
  - ❑ Java program is confined within the java execution environment and cannot access the other part of the computer.

# Basics of Java Environments

- ❏ Java programs normally undergo five phases
  - ❏ Edit
    - ❏ Programmer writes program (and stores program on disk)
  - ❏ Compile
    - ❏ Compiler creates *bytecodes* from program
  - ❏ Load
    - ❏ Class loader stores bytecodes in memory
  - ❏ Verify
    - ❏ Verifier ensures bytecodes do not violate security requirements
  - ❏ Execute
    - ❏ Interpreter translates bytecodes into machine language



