# Java for Programmers

## P.J.DEITEL

## H.M.DEITEL

**Chapter 1: Introduction**
**Lecture 1**

**Tasnim Sharmin Alin**
Lecturer, Department of Computer Science & Engineering,
Leading University

# History

- A research project began at Sun Microsystems in 1991.
- Research efforts birthed a new language, OAK created by James Gosling-”The father of JAVA”.
- OAK was renamed java in 1994.

- Java was publicly released in MAY 1995.

## *History*

- Language was created with five goals:
- ✓ It should be object oriented.
- ✓ A single representation of a program could be executed on multiple operating systems.
- ✓ It should fully support network programming.
- ✓ It should be easy to use.
- ✓ It should execute code from remote sources securely.

# *What is JAVA*

- ***Java (with a capital J) is a high-level, third generation programming language,***
- Java Program consist of pieces called *classes.*
- *Classes* include pieces called *methods* that perform tasks and return information when they complete them.
- Programmers can create each piece they need to form java program.
- Most programmers take advantages of the rich collections of existing classes in the java class libraries, which are also known as the Java API(Application Programming Interfaces.)

# *Key features Of Java*

- **Object-oriented**
- **Architecture-neutral**
- **Portable**
- **Somewhat Interpreted**
- **Simple and Familiar**
- **Distributed**
- **Robust**
- **Secure**
- **High performance**
- **Multi Threaded**
- **Dynamic**

# *Features of Java*

- **Simple**
  - fixes some clumsy features of C++
  - no pointers
  - automatic garbage collection
  - rich pre-defined class library
- **Object oriented**
  - focus on the data (objects) and methods manipulating the data
  - all functions are associated with objects
  - almost all data types are objects (files, strings, etc.)
  - potentially better code organization and reuse
- **Robust:**

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

# *Features of JAVA*

- **Interpreted**
  - java compiler generate byte-codes, not native machine code
  - the compiled byte-codes are platform-independent
  - java bytecodes are translated on the fly to machine readable instructions in runtime (Java Virtual Machine)
- **Portable**
  - same application runs on all platforms
  - the sizes of the primitive data types are always the same
  - the libraries define portable interfaces
- **Reliable**
  - extensive compile-time and runtime error checking
  - no pointers but real arrays. Memory corruptions or unauthorized memory accesses are impossible
  - automatic garbage collection tracks objects usage over time
- **Secure**
  - usage in networked environments requires more security
  - memory allocation model is a major defense
  - access restrictions are forced (private, public)

# Features of JAVA

- **Multithreaded**
  - multiple concurrent threads of executions can run simultaneously
  - utilizes a sophisticated set of synchronization primitives (based on monitors and condition variables paradigm) to achieve this
- **Dynamic**
  - java is designed to adapt to evolving environment
  - libraries can freely add new methods and instance variables without any effect on their clients
  - interfaces promote flexibility and reusability in code by specifying a set of methods an object can perform, but leaves open how these methods should be implemented
  - can check the class type in runtime

- **Architectural-neutral :**

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

# Java Platforms

- The *Java platform* is the name for a bundle of related programs from Sun that allow for developing and running programs written in the Java programming language
- Java has different Platform Editions:
- Java Standard Edition (Java SE ,J2SE)
- Java Enterprise Edition (Java EE, J2EE)
- Java Micro Edition (Java ME, J2ME)-

## Java Platforms

- Java ME (Micro Edition):

  Specifies several different sets of libraries (known as profiles) for devices with limited storage, display, and power capacities. Often used to develop applications for mobile devices, PDAs, TV set-top boxes, and printers.

- Java SE (Standard Edition):

  For general-purpose use on desktop PCs, servers and similar devices.

- Java EE (Enterprise Edition):

  Java SE plus various APIs useful for multi-tier client–server enterprise applications.

## Java Platform Softwares & IDEs

- **Softwares**
  - **Java Runtime Environment (JRE)**
  - **Java Development Kit (JDK)**
- **IDEs:** A Java IDE (Integrated Development Environment) is a software application which enables users to more easily write and debug Java programs.
- **Some popular IDEs** : Eclipse, NetBeans, Jbuilder, Jcreator, Jedit etc.

# Java Virtual Machine

- A **Java virtual machine** (**JVM**) is a process virtual machine that can execute Java bytecode. It is the code execution component of the Java platform.

- This bytecode is the same no matter what hardware or operating system the program is running under.

- There is a JIT (Just In Time) compiler within the *Java Virtual Machine*, or JVM. The JIT compiler translates the Java bytecode into native processor instructions at run-time and caches the native code in memory during execution.

- Sun Microsystems has stated that there are over 5.5 billion JVM-enabled devices.