2   Introduction to Java

Objectives

At the end of the lesson, the student should be able to:

- Describe the features of Java technology such as the Java virtual machine, garbage collection and code security
- Describe the different phases of a Java program

Java Background: History

- Java
    - was created in 1991
    - by James Gosling et al. of Sun Microsystems.
    - Initially called Oak, in honor of the tree outside Gosling's window, its name was changed to Java because there was already a language called Oak.

Java Background: History

- Java

  - The original motivation for Java

    - The need for platform independent language that could be embedded in various consumer electronic products like toasters and refrigerators.

  - One of the first projects developed using Java

    - a personal hand-held remote control named Star 7.

  - At about the same time, the World Wide Web and the Internet were gaining popularity. Gosling et. al. realized that Java could be used for Internet programming.

Java Background: What is Java Technology?

- The Java technology is:
    - A programming language
    - A development environment
    - An application environment
    - A deployment environment

JEDI

Java Technology: Programming Language

- As a programming language, Java can create all kinds of applications that you could create using any conventional programming language.

Java Technology: A Development Environment

- As a development environment, Java technology provides you with a large suite of tools:

    - A compiler

    - An interpreter

    - A documentation generator

    - A class file packaging tool
      and so on...

Java Technology:
An Application and Runtime Environment

- Java technology applications are typically general-purpose programs that run on any machine where the Java runtime environment (JRE) is installed.

- There are two main deployment environments:

    1. The JRE supplied by the Java 2 Software Development Kit (SDK) contains the complete set of class files for all the Java technology packages, which includes basic language classes, GUI component classes, and so on.

    2. The other main deployment environment is on your web browser. Most commercial browsers supply a Java technology interpreter and runtime environment.

Java Features

- Some features of Java:
    - The Java Virtual Machine
    - Garbage Collection
    - Code Security

Java Features:
The Java Virtual Machine

- Java Virtual Machine (JVM)

    - an imaginary machine that is implemented by emulating software on a real machine

    - provides the hardware platform specifications to which you compile all Java technology code

- Bytecode

    - a special machine language that can be understood by the Java Virtual Machine (JVM)

    - independent of any particular computer hardware, so any computer with a Java interpreter can execute the compiled Java program, no matter what type of computer the program was compiled on

Java Features:
Garbage Collection

- Garbage collection thread

  - responsible for freeing any memory that can be freed. This happens automatically during the lifetime of the Java program.

  - programmer is freed from the burden of having to deallocate that memory themselves

Java Features:
Code Security

- Code security is attained in Java through the implementation of its Java Runtime Environment (JRE).

- JRE

  - runs code compiled for a JVM and performs class loading (through the class loader), code verification (through the bytecode verifier) and finally code execution

Java Features:
Code Security

- Class Loader

  - responsible for loading all classes needed for the Java program

  - adds security by separating the namespaces for the classes of the local file system from those that are imported from network sources

  - After loading all the classes, the memory layout of the executable is then determined. This adds protection against unauthorized access to restricted areas of the code since the memory layout is determined during runtime
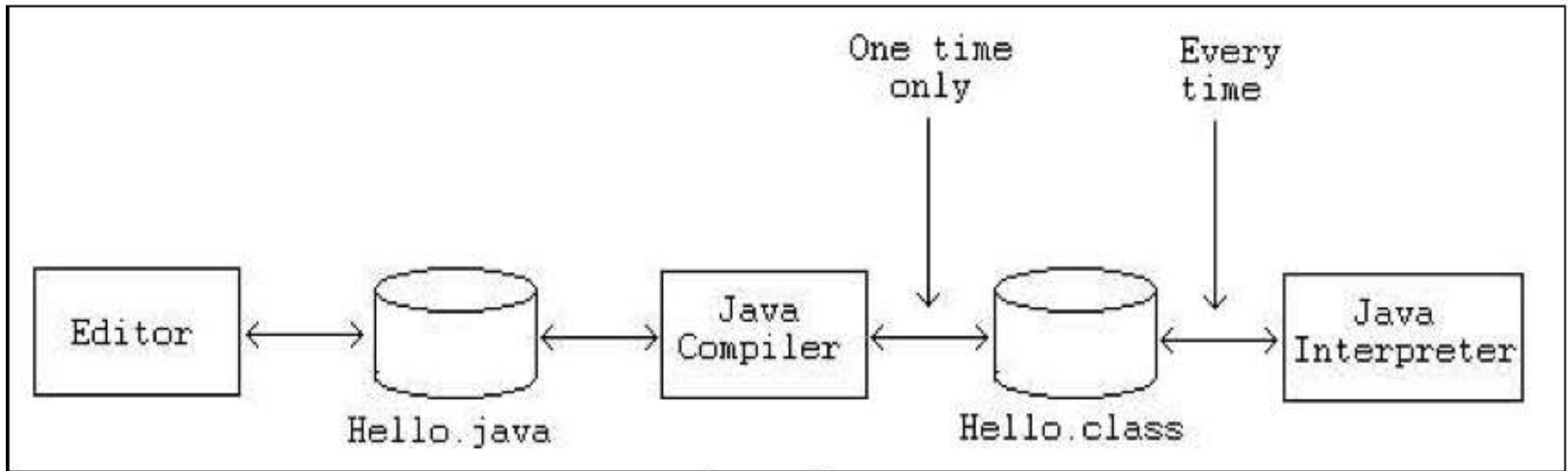
Java Features:
Code Security

- Bytecode verifier

  - tests the format of the code fragments and checks the code fragments for illegal code that can violate access rights to objects

Phases of a Java Program

- The following figure describes the process of compiling and executing a Java program

Phases of a Java Program

Task	Tool to use	Output
Write the program	Any text editor	File with .java extension
Compile the program	Java Compiler	File with .class extension (Java bytecodes)
Run the program	Java Interpreter	Program Output

Summary

- Java Background

  - History

  - Java Technology

    - A programming language, development environment, application environment and deployment environment

  - Java Features

    - Java Virtual machine, garbage collection and code security

- Phases of a Java Program

  - Write, compile, run