

Java Programming Language SE - 6

Module 1: Getting Started

Team Emertxe

ORACLE®

Certified Professional

Java SE 6 Programmer



Objectives

Upon completion of this module, you should be able to:

- Describe the key features of Java technology
- Write, compile and run simple java application
- Describe the function of the Java virtual machine
- Define garbage collection
- List the three task perform by Java that handle code security



Relevance

- Is the Java programming language a complete language or is it useful only for writing programs for the Web?
- Why do you need another programming language?
- How does the Java technology platform improve on other language platforms?

This material also serves as a guide for Oracle Certified Java Professional (OCJP), by making the flow in alignment with prescribed syllabus

What is Java Technology?

- Java technology is:
 - A programming language
 - A development environment
 - An application environment
 - A deployment environment
- It is similar in syntax to C++.
- It is used for developing both applets and applications.

Thanks to its scalable nature, Java has emerged as the most preferred language in Web and Mobile application development

Primary Goals of the Java Technology



- Provides an easy-to-use language by:
 - Avoiding many pitfalls of other languages
 - Being object-oriented
 - Enabling users to create streamlined and clear code
- Provides an interpreted environment for:
 - Improved speed of development
 - Code portability



Primary Goals of the Java Technology



- Enables users to run more than one thread of activity
- Loads classes dynamically; that is, at the time they are actually needed
- Supports changing programs dynamically during runtime by loading classes from disparate sources
- Furnishes better security

Primary Goals of the Java Technology



The following features fulfill these goals:

- The Java Virtual Machine (JVM) 1
- Garbage collection
- The Java Runtime Environment (JRE)
- JVM tool interface

The Java Virtual Machine

- Provides hardware platform specifications
- Reads compiled byte codes that are platform-independent
- Is implemented as software or hardware
- Is implemented in a Java technology development tool or a Web browser



The Java Virtual Machine(JVM) make the code run in any platform, thereby making it platform independent

The Java Virtual Machine

JVM provides definitions for the:

- Instruction set (central processing unit [CPU])
- Register set
- Class file format
- Stack
- Garbage-collected heap
- Memory area
- Fatal error reporting
- High-precision timing support

The Java Virtual Machine



- The majority of type checking is done when the code is compiled.
- Implementation of the JVM approved by Sun Microsystems must be able to run any compliant class file.
- The JVM executes on multiple operating environments.

Garbage Collection

- Allocated memory that is no longer needed should be deallocated.
- In other languages, deallocation is the programmer's responsibility.
- The Java programming language provides a system-level thread to track memory allocation.

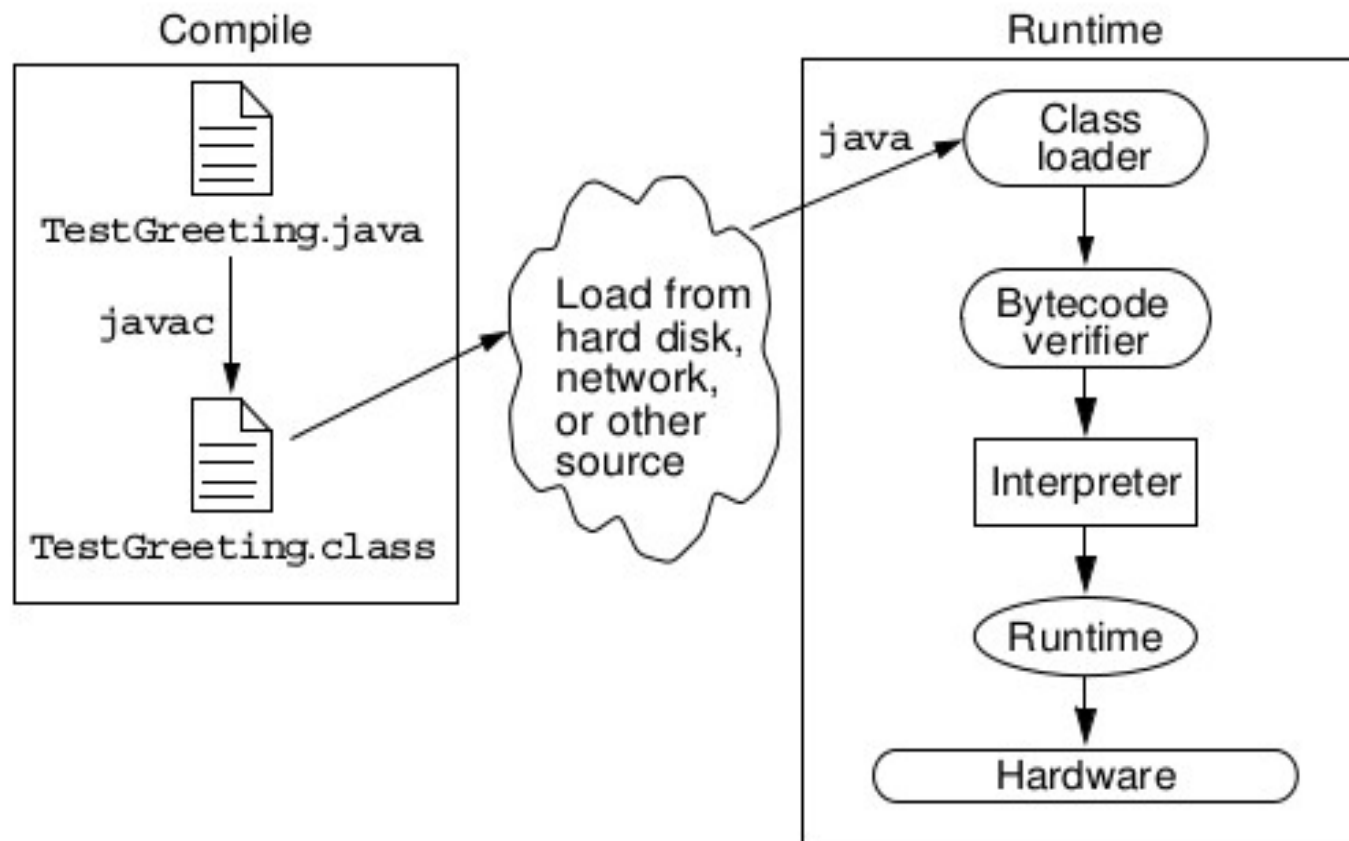


Garbage Collection

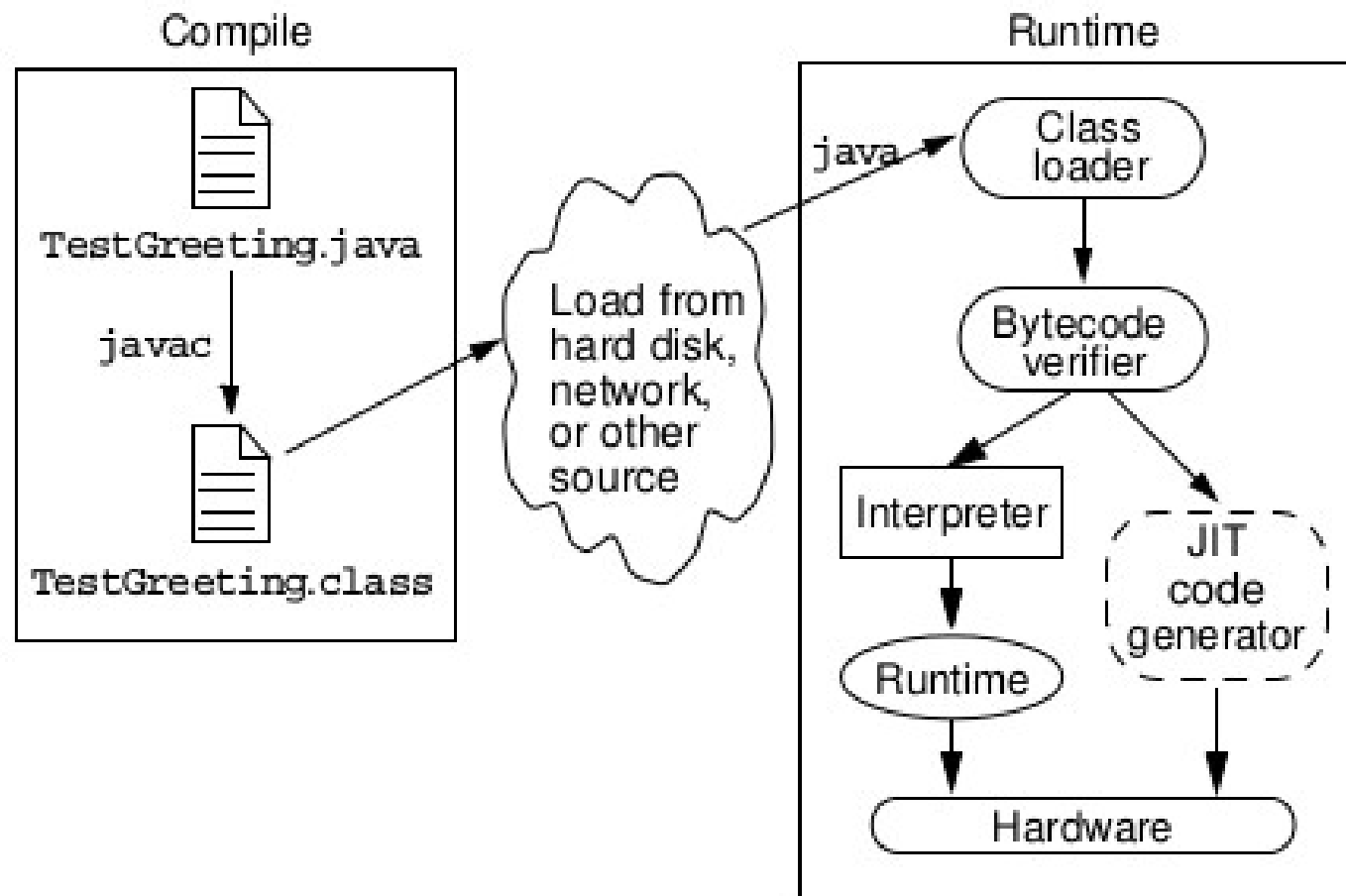
Garbage collection has the following characteristics:

- Checks for and frees memory no longer needed
- Is done automatically
- Can vary dramatically across JVM implementations

The Java Runtime Environment



JRE With a Just-In-Time (JIT) Compiler



JVM Tasks

The JVM performs three main tasks:

- Loads code
- Verifies code
- Executes code



The Class Loader



- Loads all classes necessary for the execution of a program
- Maintains classes of the local file system in separate namespaces
- Prevents spoofing

The Bytecode Verifier

- The code adheres to the JVM specification.
- The code does not violate system integrity.
- The code causes no operand stack overflows or underflows.
- The parameter types for all operational code are correct.
- No illegal data conversions (the conversion of integers to pointers) have occurred.

A Simple Java Application

The TestGreeting.java Application

```
//  
// Sample "Hello World" application  
//  
public class TestGreeting{  
    public static void main (String[] args) {  
        Greeting hello = new Greeting();  
        hello.greet();  
    }  
}
```

A Simple Java Application

The Greeting.java Class

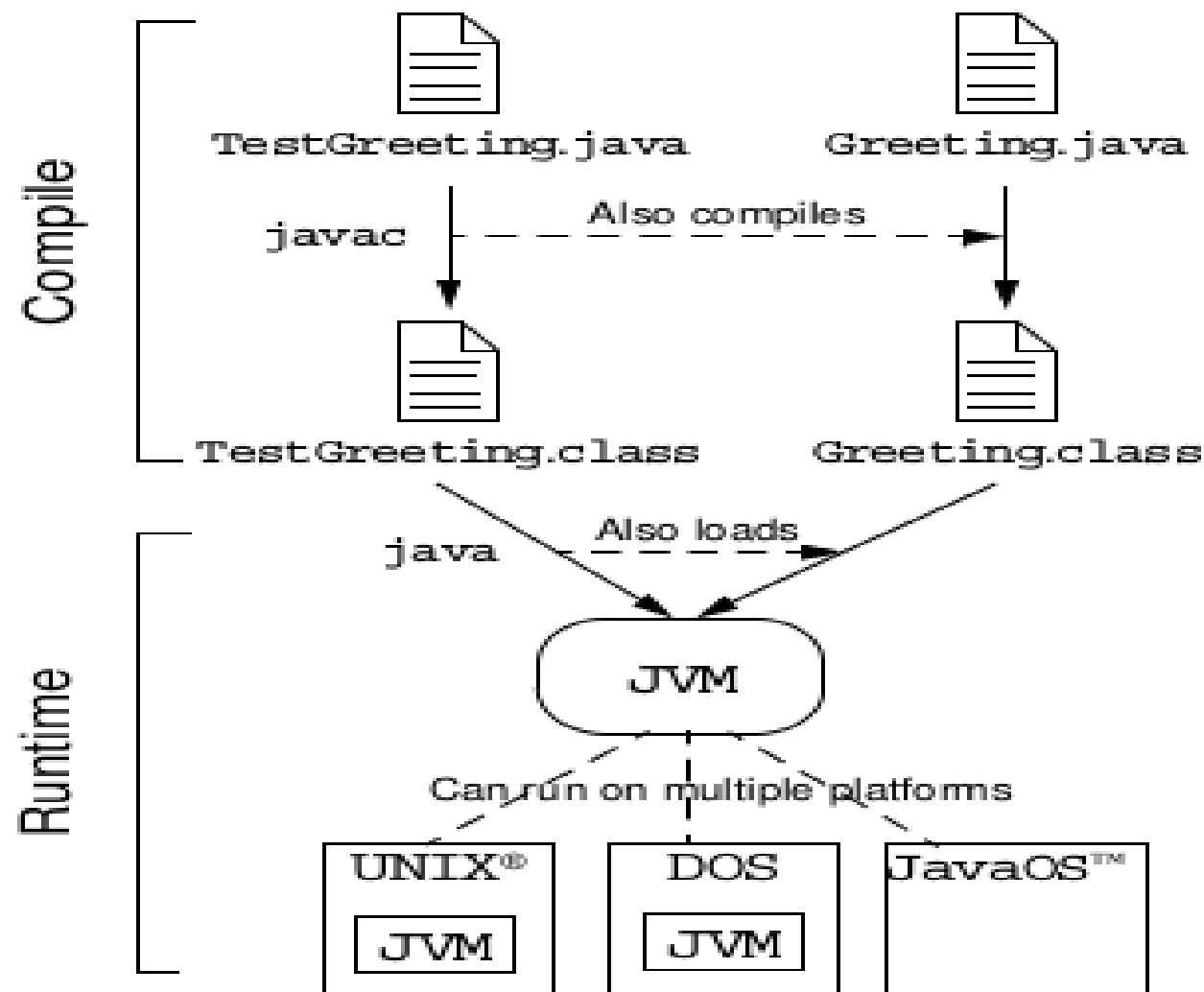
```
public class Greeting {  
    public void greet() {  
        System.out.println("hi");  
    }  
}
```

Compiling and Running the TestGreeting Program



- Compile TestGreeting.java:
`javac TestGreeting.java`
- The Greeting.java is compiled automatically.
- Run the application by using the following command:
`java TestGreeting`
- Locate common compile and runtime errors.

Java Technology Runtime Environment



Stay connected

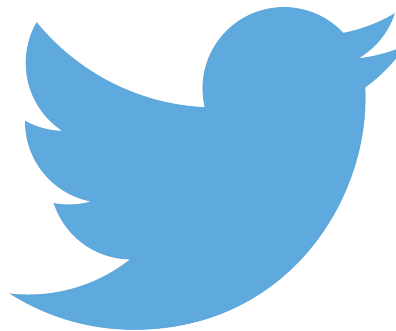


About us: Emertxe is India's one of the top IT finishing schools & self learning kits provider. Our primary focus is on Embedded with diversification focus on Java, Oracle and Android areas

Emertxe Information Technologies,
No-1, 9th Cross, 5th Main,
Jayamahal Extension,
Bangalore, Karnataka 560046
T: +91 80 6562 9666
E: training@emertxe.com



<https://www.facebook.com/Emertxe>



<https://twitter.com/EmertxeTweet>



slideshare
Present Yourself

<https://www.slideshare.net/EmertxeSlides>



Thank You