

# Core Java 8

## Lesson 05: Introduction to Java



# Lesson Objectives

After completing this lesson, participants will be able to -

- Introduction to Java
- Features of Java
- Evolution in Java
- Developing software in Java





## Java's Lineage

C language was result of the need for structured, efficient, high-level language replacing assembly language.

C++, which followed C, became the common (but not the first) language to offer OOP features, winning over procedural languages such as C.

Java, another object oriented language offering OOP features, followed the syntax of C++ at most places. However, it offered many more features.



## What is Java?

Java is an Object-Oriented programming language – most of it is free and open source!

- It is developed in the early 1990s, by James Gosling of Sun Microsystems
- It allows development of software applications.
- It is amongst the preferred choice for developing internet-based applications



## Java Language Features

Java has advantages due to the following features:

- Completely Object-Oriented
- Simple
- Robust: Strongly typed language
- Security
  - Byte code Verifier
  - Class Loader
  - Security Manager
- Architecture Neutral: Platform independent
- Interpreted and Compiled
- Multithreaded: Concurrent running tasks
- Dynamic
- Memory Management and Garbage Collection



## 5.3: Writing Sample Java Program

### A Sample Program

```
// Lets see a simple java program
public class HelloWorld {
    /* The execution starts here */
    public static void main(String args[])
    {
        System.out.println("Hello World!");
    } //end of main()
} //end of class
```

Single line comment

Multi-line comment

entry point for your application

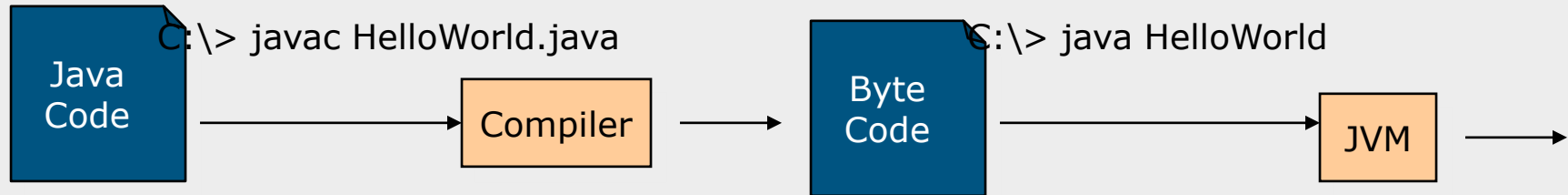
Prints "Hello World!" message to standard output

Type all code, commands and file names exactly as shown. Java is highly *case-sensitive*



## 5.3: Writing Sample Java Program

# Java Development Process



Compiling Java program

```
C:\WINNT\system32\cmd.exe

F:\JEE-Demos>javac HelloWorld.java

F:\JEE-Demos>java HelloWorld
Hello World!

F:\JEE-Demos>_
```

The screenshot shows a Windows command prompt window with the title 'C:\WINNT\system32\cmd.exe'. The prompt is 'F:\JEE-Demos>'. The first command entered is 'javac HelloWorld.java'. The second command is 'java HelloWorld', which produces the output 'Hello World!'. The prompt then shows a trailing underscore, indicating the program has finished execution.

Running Java program



## 5.3: Writing Sample Java Program

### Demo

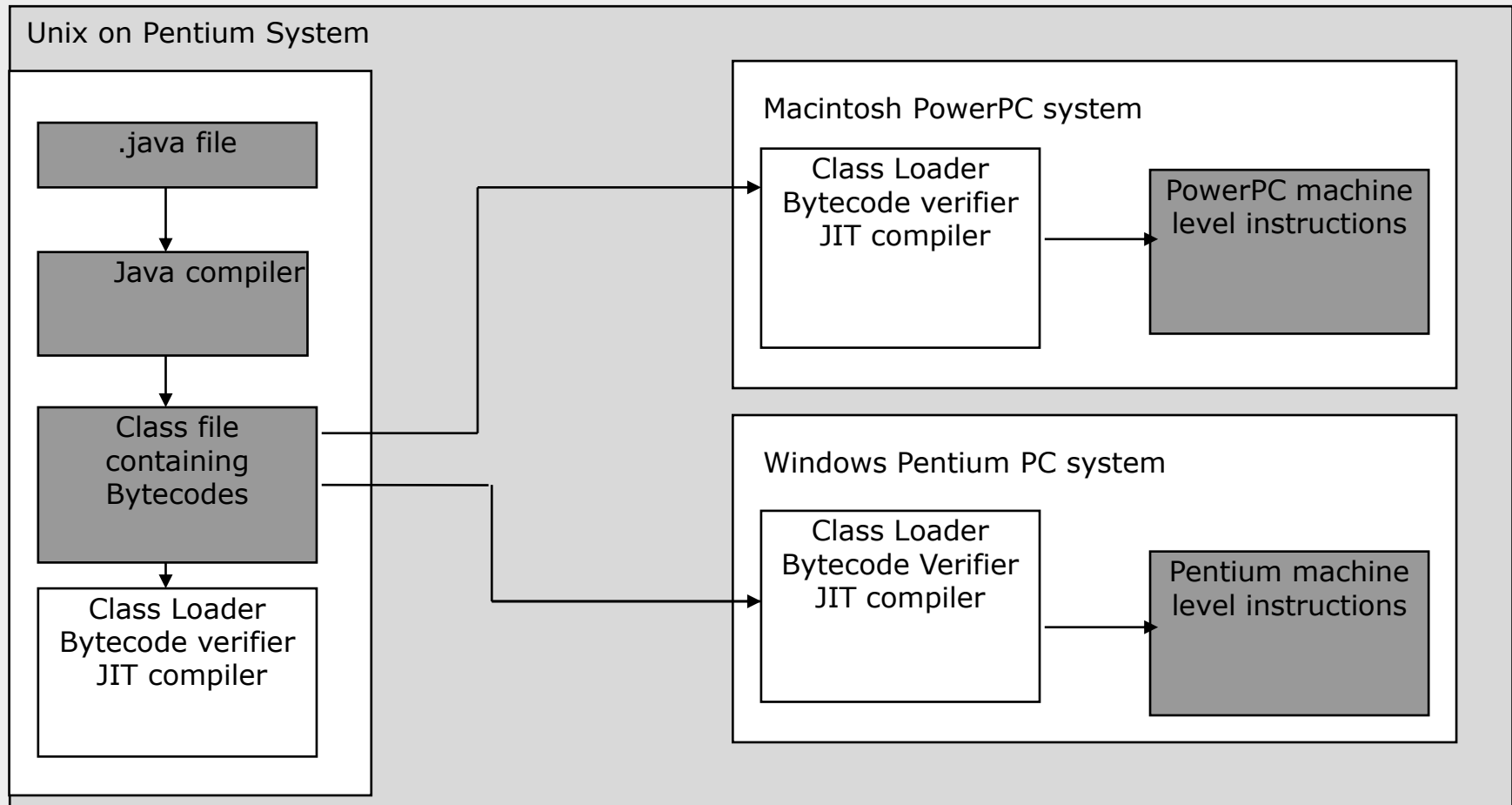
Creating and executing the First Java application







## Platform Independence feature of Java





## JRE versus JDK

JRE is the “Java Runtime Environment”. It is responsible for creating a Java Virtual Machine to execute Java class files (that is, run Java programs).

JDK is the “Java Development Kit”. It contains tools for Development of Java code (for example: Java Compiler) and execution of Java code (for example: JRE)

JDK is a superset of JRE. It allows you to do both – write and run programs.

# Summary



In this lesson, you have learnt:

- Features of Java and its different versions
- How Java is platform Independent
- Difference between JRE and JDK
- Writing, Compiling, and Executing a simple program





## Review Question

Question 1: A program written in the Java programming language can run on any platform because...

- **Option 1:** The JIT Compiler converts the Java program into machine equivalent
- **Option 2:** The Java Virtual Machine(JVM) interprets the program for the native operating system
- **Option 3:** The compiler is identical to a C++ compiler
- **Option 4:** The APIs do all the work



Question 2: Java Compiler compiles the source code into \_\_\_\_ code, which is interpreted by \_\_\_\_ to produce Native Executable code.



## Review Question

Question 3: Which of the following are true about JVM?

- **Option 1:** JVM is an interpreter for byte code
- **Option 2:** JVM is platform dependent
- **Option 3:** Java programs are executed by the JVM
- **Option 4:** All the above is true

Question 4 : \_\_\_\_\_ allows a Java program to perform multiple activities in parallel.

- **Option 1:** Java Beans
- **Option 2:** Swing
- **Option 3:** Multithreading
- **Option 4:** None of the above

