BITS Pilani
Pilani Campus

Object Oriented Programming
CS F213

J. Jennifer Ranjani
email: jennifer.ranjani@pilani.bits-pilani.ac.in
Chamber: 6121 B, NAB
Consultation: Friday 4.00 – 5.00 p.m.

Java – an Introduction

BITS Pilani
Pilani Campus

Prescribed Books

Text Books:

T1. Java: The Complete Reference, Herbert Schildt, McGraw Hill Education, Tenth Edition, 2017.

T2. Object Oriented Design & Patterns, Cay Horstmann, John Wiley & Sons, Second Edition, 2005.

Reference Book:

R1. Java^{TM} Design Patterns – A Tutorial, James W. Cooper, Addison-Wesley, 2000.

Evaluation Scheme

Component	Duration	Weightage (%)	Nature of component (Close Book/ Open Book)
Mid-Semester Test	90 Min.	25	Closed Book
Comprehensive Examination	3 h	35	Partly Open
Online Test	90 Min.	20	Open Book
Quizzes (Take home, Nalanda)	30 Min	6	Open Book
Online Quiz	2 h	9	Open Book
♣Lab Attendance	2 h	5	NA

What is Java?

- Programming language and a platform

- **Platform:** Any hardware or software environment in which a program runs
  - Java has its own runtime environment (JRE) and API

How Java Impacted the Internet?

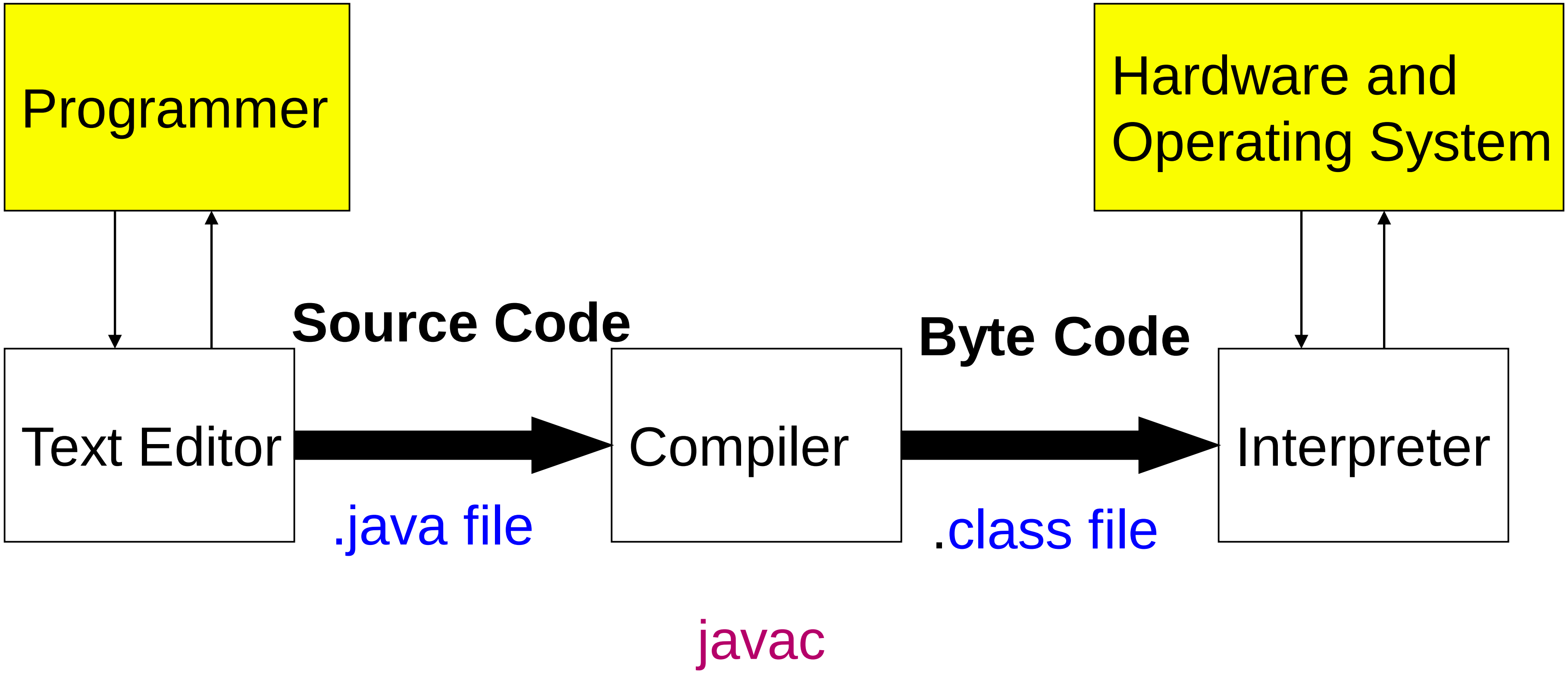
- **Security:** Java confines an application to the Java execution environment and prevent it from accessing other parts of the computer.

- **Portability:** Allows same application to be downloaded and executed by a variety of CPU, OS and Browser without the necessity to have different versions for different computers.

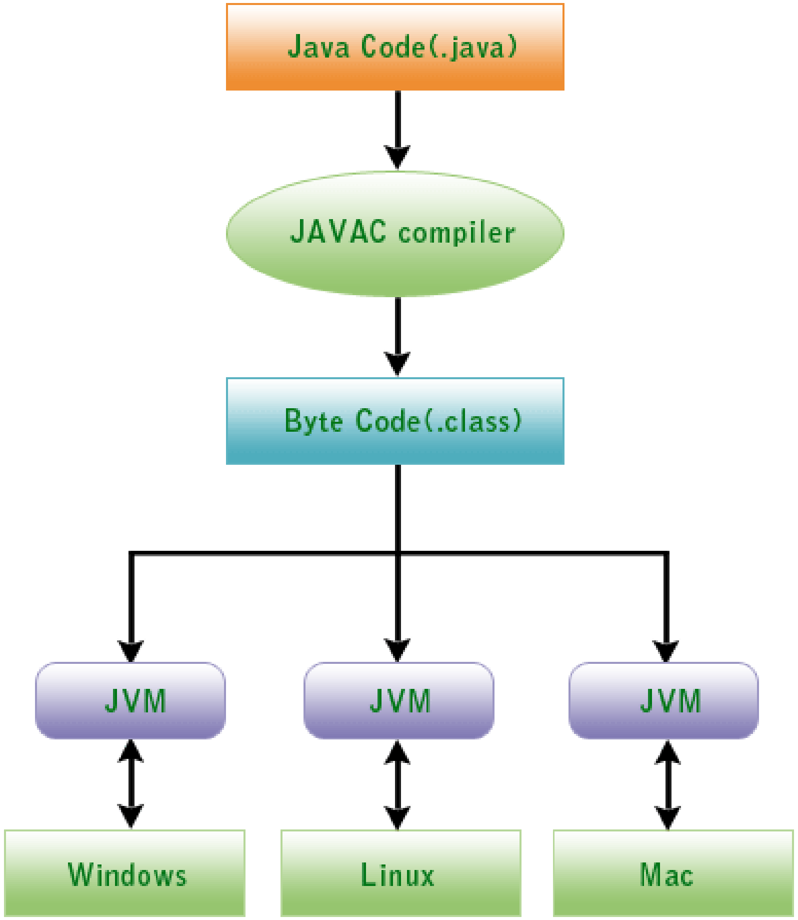
Bytecode: Java’s Magic

- Output of a Java compiler is not an executable code but a bytecode.
- Bytecode – highly optimized set of instructions to be executed by the Java’s run time system – JVM.
- JVM is the interpreter for bytecode
- JVM helps to solve the security and portability issues.

Java is compiled and interpreted



Portability



Where is Java used?

Acc. To Sun, 3 billion devices run Java

- Desktop applications
  - Acrobat reader, media player, antivirus etc
- Web applications
- Enterprise applications
- Mobile
- Embedded System
- Smart card
- Robotics
- Games etc.

Java Platforms / Editions

- Java SE (Standard Edition)
  - Programming platform
- Java EE (Enterprise Edition)
  - Web and enterprise applications
- Java ME (Micro Edition)
  - Mobile applications
- JavaFx
  - Rich internet applications. Uses light weight user interface APIs.

History of Java

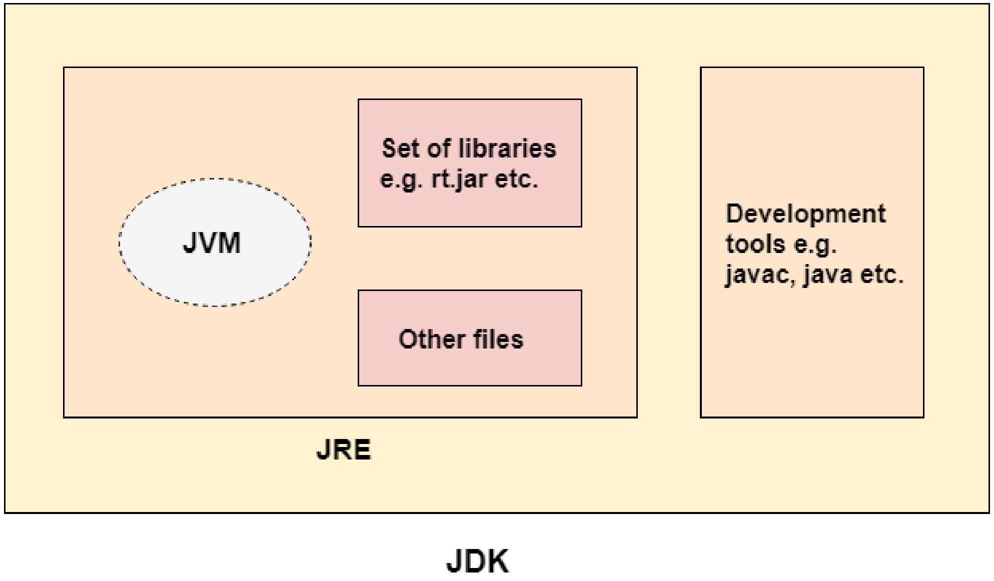
- James Gosling, Mike Sheridan, Patrick Naughton initiated the project in June 1991 (Green Team).
- Originally designed for small embedded systems
- “Greentalk” with file extension .gt
- Oak – symbol of strength and national tree of countries like U.S., France, Germany, Romania etc.
- Suggested names: Dynamic, Revolutionary, Silk, Jolt, DNA etc
  - Java is named after an island in Indonesia where first coffee was produced
  - Java is a name not an acronym
- JDK 1.0 was released in Jan 23, 1996.

Java Version History

- JDK Alpha and Beta (1995)
- JDK 1.0 (23rd Jan, 1996)
- JDK 1.1 (19th Feb, 1997)
- J2SE 1.2 (8th Dec, 1998)
- J2SE 1.3 (8th May, 2000)
- J2SE 1.4 (6th Feb, 2002)
- J2SE 5.0 (30th Sep, 2004)
- Java SE 6 (11th Dec, 2006)
- Java SE 7 (28th July, 2011)
- Java SE 8 (18th March, 2014)
- Java SE 9 (21st Sep, 2017)
- Java SE 10 (20th March, 2018)

JDK

- JVM – provides runtime environment for bytecode execution; loads, verifies, executes code
- JRE – contains libraries and files used by JVM
- JDK – JVM, java, javac, jar, Javadoc etc. for complete java application development.

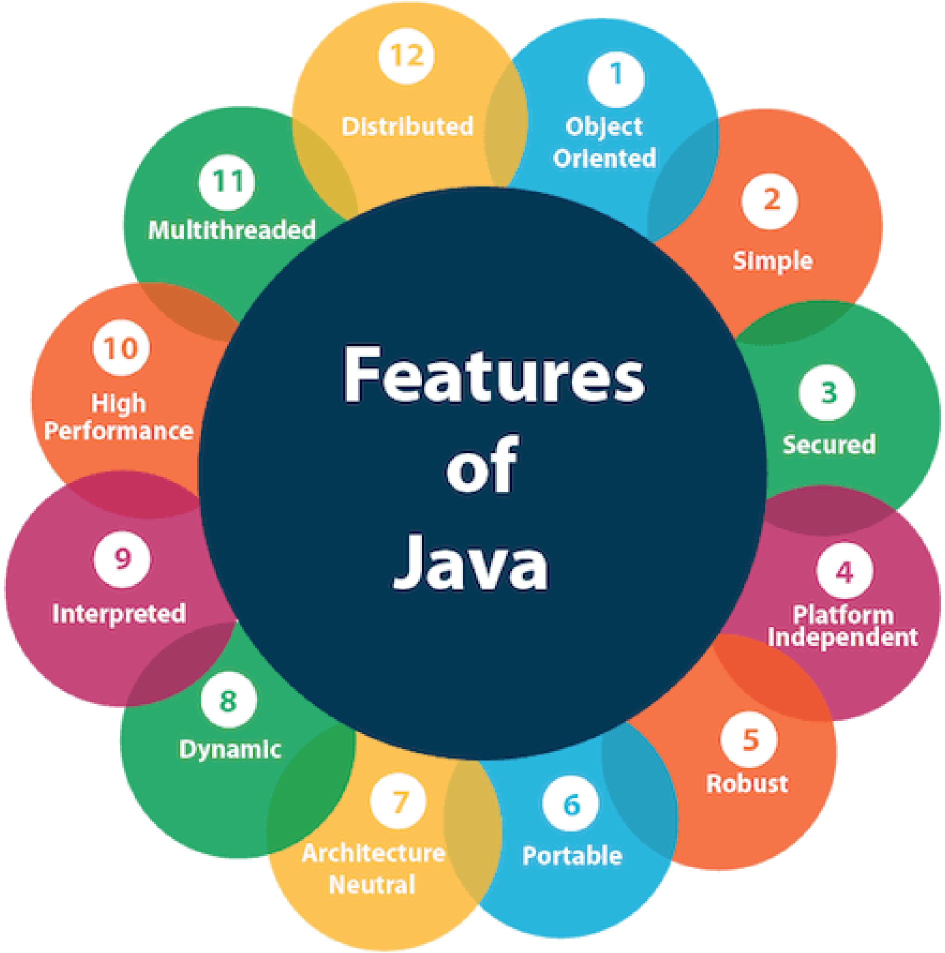

Set of libraries e.g. rt.jar etc.
JVM
Other files
Development tools e.g. javac, java etc.
JRE
JDK

innovate achieve lead

BITS Pilani
Pilani Campus

Features of Java

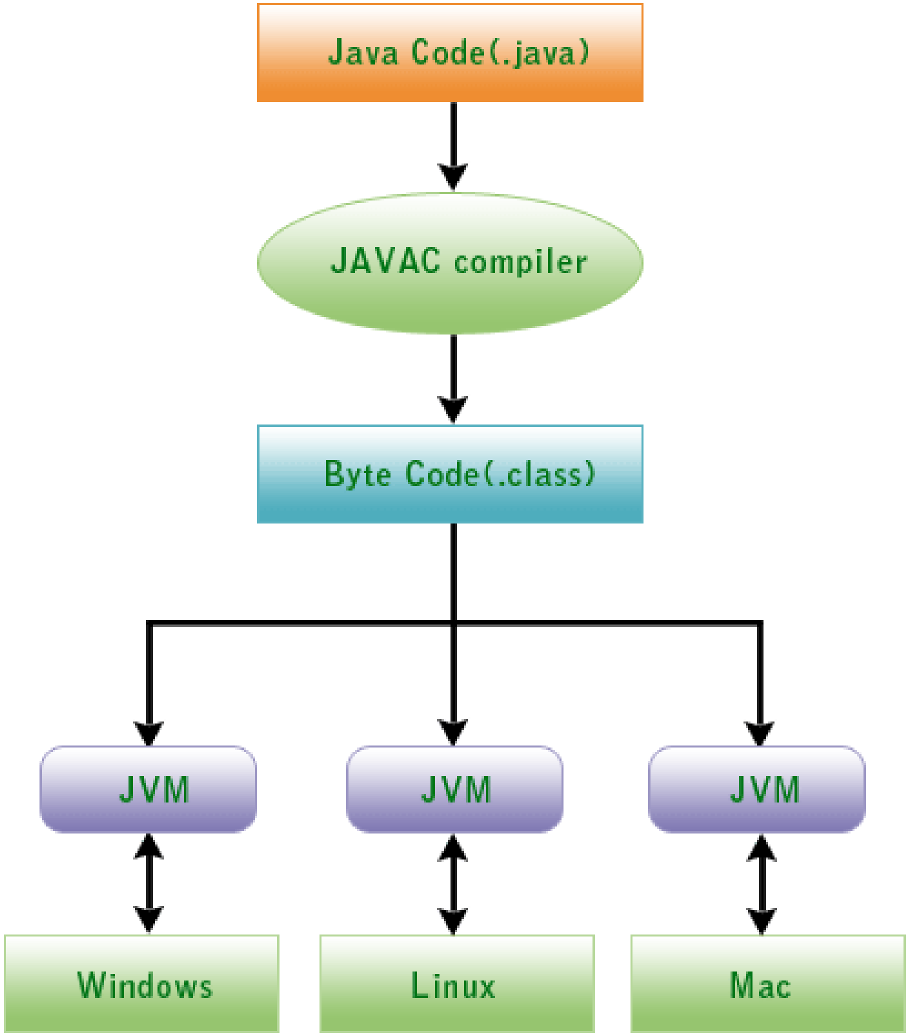
Java Buzzwords



Features of Java

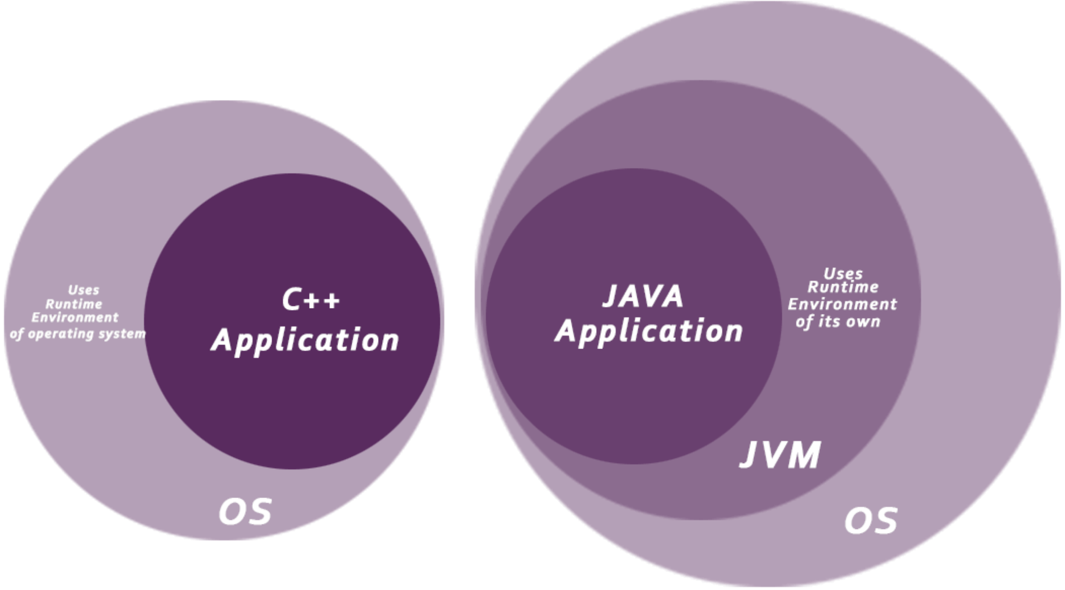
- Simple
  - Syntax based on C++
  - Removed confusing and rarely used features like pointers, operator overloading etc.,
  - Automatic garbage collection
- Object Oriented
  - Object, Class, Inheritance, Polymorphism, Abstraction, Encapsulation
- Platform Independent
  - Compiler converts Java code to bytecode
  - Bytecode is platform independent
  - Write Once and Run Anywhere

Platform Independence (Recall)



Features of Java

- Secured



- Robust
  - Strong memory management, secure due to lack of pointers, automatic garbage collection, exception handling and type checking

Features of Java

- Architecture-neutral and Portable
  - Size of primitive types is fixed i.e., 4 bytes for both 32 and 64 bit architectures
  - Porting the java system to any new platform involves writing an interpreter.
  - The interpreter will figure out what the equivalent machine dependent code to run

Features of Java

- High Performance
  - Bytecode is close to native code
  - It is an interpreted language hence slower than C, C++
- Distributed
  - Enables access to files by calling methods from any machine on the internet
  - RMI, EJB
- Multi-threaded
  - Thread is like a separate program executing concurrently
  - Doesn’t occupy memory for each thread
  - Multimedia, Web applications etc
- Dynamic
  - Small fragments of bytecode may be dynamically updated at run time.
  - Also supports functions from native languages i.e. C and C++

Comparison with C++

Comparison Index	C++	Java
Platform-independent	Platform-dependent.	Platform-independent.
Mainly used for	System programming.	Application programming.
Goto	Yes	No
Multiple inheritance	C++ supports multiple inheritance.	Java doesn't support multiple inheritance through class. It can be achieved by interfaces in java.
Operator Overloading	Yes	No
Pointers	C++ supports pointers. You can write pointer program in C++.	Java supports pointer internally. But you can't write the pointer program in java.

Comparison Index	C++	Java
Compiler and Interpreter	C++ uses compiler only.	Java uses compiler and interpreter both. Java source code is converted into byte code at compilation time. The interpreter executes this byte code at run time and produces output.
Call by Value and Call by reference	C++ supports both call by value and call by reference.	Java supports call by value only.
Structure and Union	C++ supports structures and unions.	Java doesn't support structures and unions.

Comparison Index	C++	Java
Thread Support	C++ doesn't have built-in support for threads. It relies on third-party libraries for thread support.	Java has built-in thread support.
Virtual Keyword	C++ supports virtual keyword so that we can decide whether or not override a function.	Java has no virtual keyword. Non-static methods are virtual by default.
unsigned right shift >>>	C++ doesn't support >>> operator.	Supports unsigned right shift >>> operator that fills zero at the top for the negative numbers. For positive numbers, it works same like >> operator.