JAVA Basics

Topics

- Two Basic Programming Model
  - Process / Procedure Oriented Programming
  - Object Oriented Programming
- Features of Object Oriented Programming
  - Class & Objects
  - Abstraction
  - Encapsulation
  - Inheritance
  - Polymorphism
- Introduction to Java

Topics

- Features Of Java
- Java Architecture
- Editions of Java
- Running Java - First Example

Programming Models

Process Oriented Programming

- Program is written around “What is happening”.
- Focus on procedure / algorithms rather than data.
- Follows top down approach
- Program is divided into small parts called functions and then it follows a series of defined calculational steps to perform the task.
- Communication through functions

Programming Models

Process Oriented Programming

- Used mainly for applications where data security is not a major concern.
- Example embedded system like calculator, freeze, washing machine, Car's temperature controller etc.
- Ex. C, Fortran, Pascal, COBOL, BASIC

Programming Models

Object Oriented Programming

- Program is written around “who is being affected”.
- Focus on data security
- Functions that operate on the data of an object are tied together in the data type.
- Program is divided into small parts called Objects.
- Communication through objects
- Ex. C++, Java, Python, Ruby, Java Script, PHP, Perl

Features Of OOP

- Objects – Instance of class which contains states and behavior.
- Class - A blueprint from which instance of class is created.
- Abstraction – Hiding complexity and showing only necessary details.
- Encapsulation – Binding of data and functions together inside class.
- Inheritance – One object acquires the properties of another object
- Polymorphism – One interface multiple methods or One interface to be used for a general class of actions.

Introduction To Java

Java is a cross platform Object oriented programming language.

History of Java

- Developed in 1991 by James Gosling and Patric Naughton of Sun Microsystem Inc , named as 'Oak'.
- In 1995, name changed to Java.
- In 2010, acquired by Oracle Corportation.
- Design goal of Java is “WORA (Write Once Run Anywhere”).
- First publicly available version of Java (Java 1.0) was released in 1995
- The latest version of Java is Java SE 18.0.2.1, released in March, 2021.

Features of Java

- Simple
- Compiled & Interpreted
- Platform Independent
- Portable
- Object Oriented
- Robust
- Secure

Features of Java

- Multithreading
- Distributed
- Dynamic
- High Performance

Features of Java

Simple

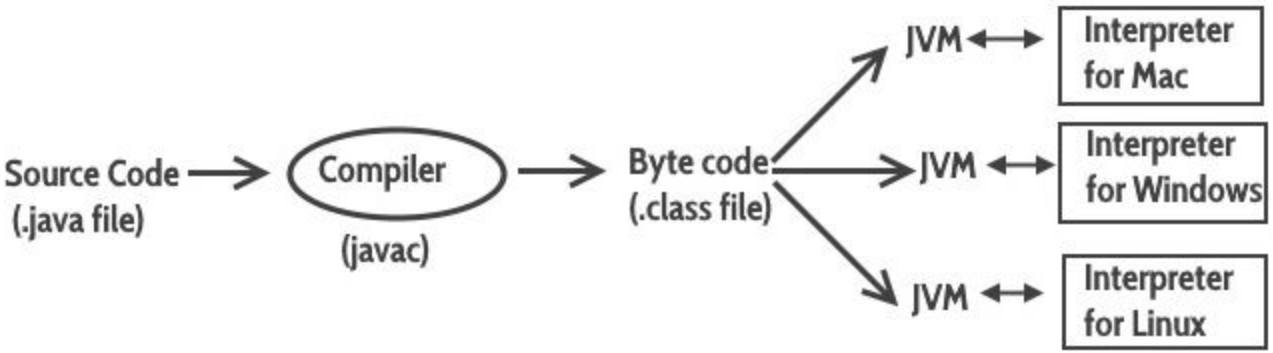
- Easy to learn and use
- No header files, pointers, operator overloading

Compiled & Interpreted

- Some interpreted languages: JavaScript, Perl, Python, BASIC, PHP, Ruby

Platform Independent / Architecture Neutral

- Program compiled on one machine can be executed on any other machine by using the



(Courtesy: Google Image)

Features of Java

Portable

- The size of primitive data type is machine independent.
- The same Java program running on different machines yields the same results. Ex: Applet.

Object Oriented

- Java is an object oriented language as it uses classes and objects to develop various applications and supports various object oriented features like Encapsulation, Abstraction, Inheritance and Polymorphism.

Robust

- The Java program executes reliably in a variety of systems.
- Java force you to find your mistakes early in program development.

Features of Java

- Java handles properly two main reasons of program failure : memory management & exception handling. Java provides automatic garbage collection for unused objects

Secure

- Java confines a java program to the Java execution environment and not allowing access to other parts of the computer.

Multihthreded

- Java allows to write programs that do many things simultaneously.

Features of Java

Distributed

- Java comes with extensive libraries for the development of applications for the distributed environment.
- Java supports RMI (Remote Method Invocation)

Features of Java

Dynamic

- Java has run-time type information that is used to verify and resolve access to objects at run time. It is safe to link code in safe manner. It also provides dynamic memory allocation.

High Performance

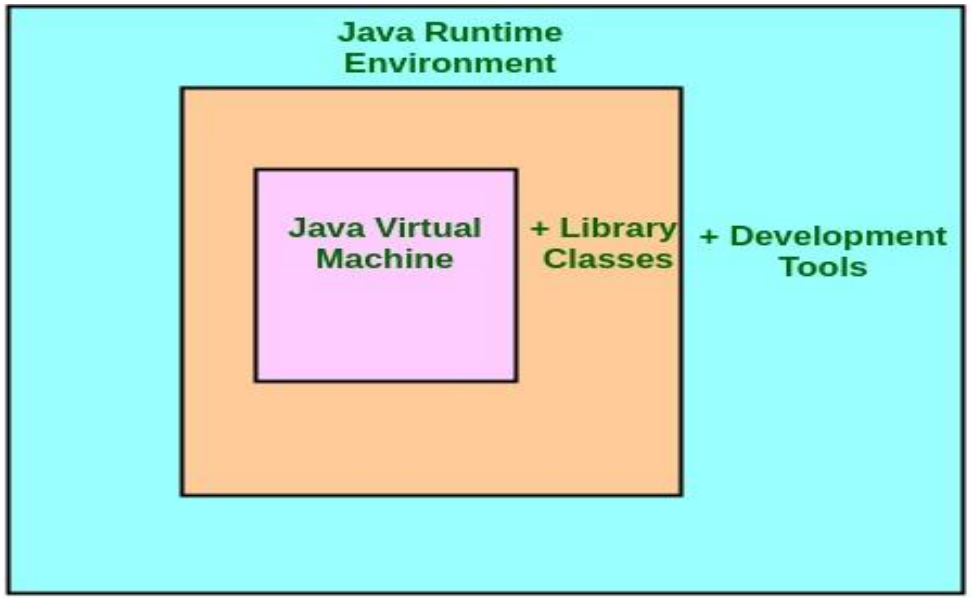
- Byte codes generated for JVM are highly optimized, so execution of byte code is fast, thus performance of Java applications is very high. It is part of the code as on demand and thus improving the performance of Java applications.

Components of Java Software

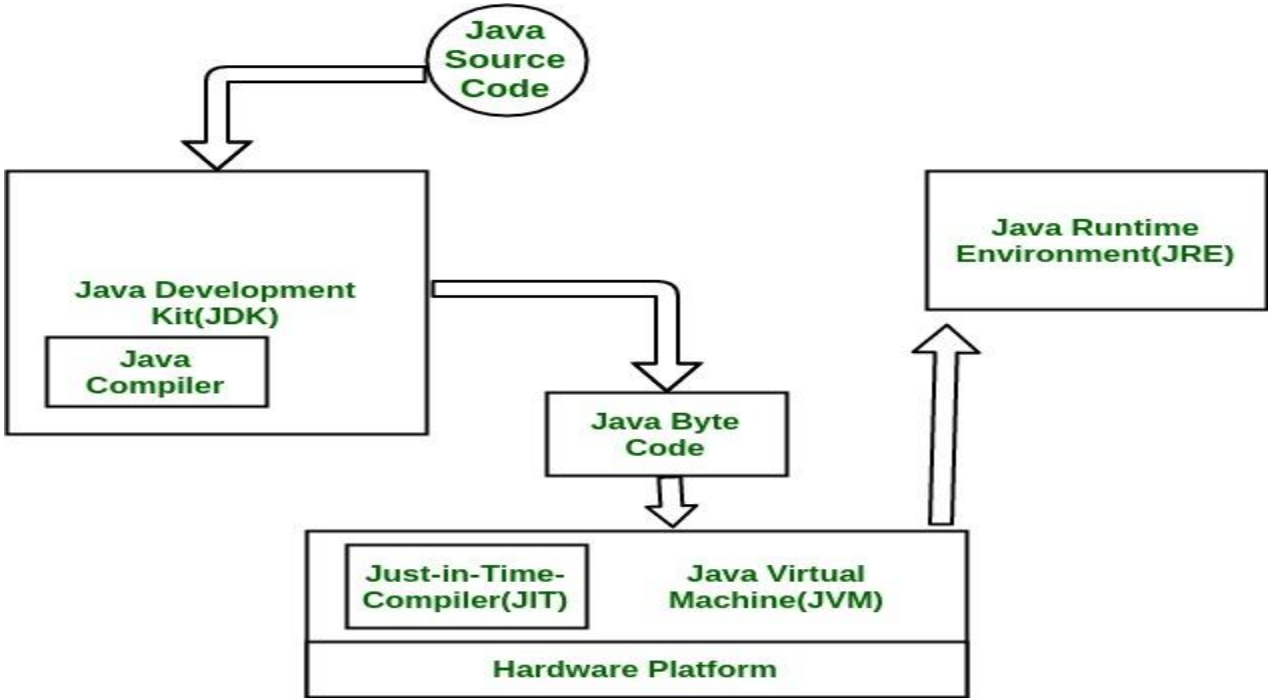
Components of Java Software

- JDK (Java Development Kit)
- JRE (Java Runtime Environment)
- JVM (Just in time compiler)

What is hotspot?
Block of code given to JIT compiler.



JDK = JRE + Development Tool
JRE = JVM + Library Classes


Java
Source
Code
Java Development
Kit(JDK)
Java
Compiler
Java Byte
Code
Java Runtime
Environment(JRE)
Just-in-Time-
Compiler(JIT)
Java Virtual
Machine(JVM)
Hardware Platform

Editions Of Java

Editions of Java

- Java SE (Java Standard / Software Edition) - Used for desktop environment applications.
- Java ME (Java Micro Edition) - Used for embedded, mobiles, wireless devices, set top box.

Writing and Executing a Simple Java Program

Writing and Executing a Simple Java Program

1. Create a class
2. Write main method inside class
3. Write logic
4. Compile using 'javac'
5. Execute using 'java'

Comments

1. // Single line comment
2. /* Multiple
   Line
   Comment */
3. /** This is for documentation **/

Myths About Java File Name

Myths …

1. Class name and file name should be same
2. Class name should be same as the class name which contains the main method.

Reality …

1. If the file contains any public class then file name must be same as that public class name.

But it is better to follow the myths :-)