Features of Java

There is given many features of java. They are also known as java buzzwords. The Java Features given below are simple and easy to understand.

1. Simple
2. Object-Oriented
3. Platform independent
4. Secured
5. Robust
6. Architecture neutral
7. Portable
8. Dynamic
9. Interpreted
10. High Performance
11. Multithreaded
12. Distributed

Simple

According to Sun, Java language is simple because:

syntax is based on C++ (so easier for programmers to learn it after C++).

removed many confusing and/or rarely-used features e.g., explicit pointers, operator overloading etc.

No need to remove unreferenced objects because there is Automatic Garbage Collection in java.

Object-oriented

Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behaviour.

Object-oriented programming(OOPs) is a methodology that simplify software development and maintenance by providing some rules.

Basic concepts of OOPs are:

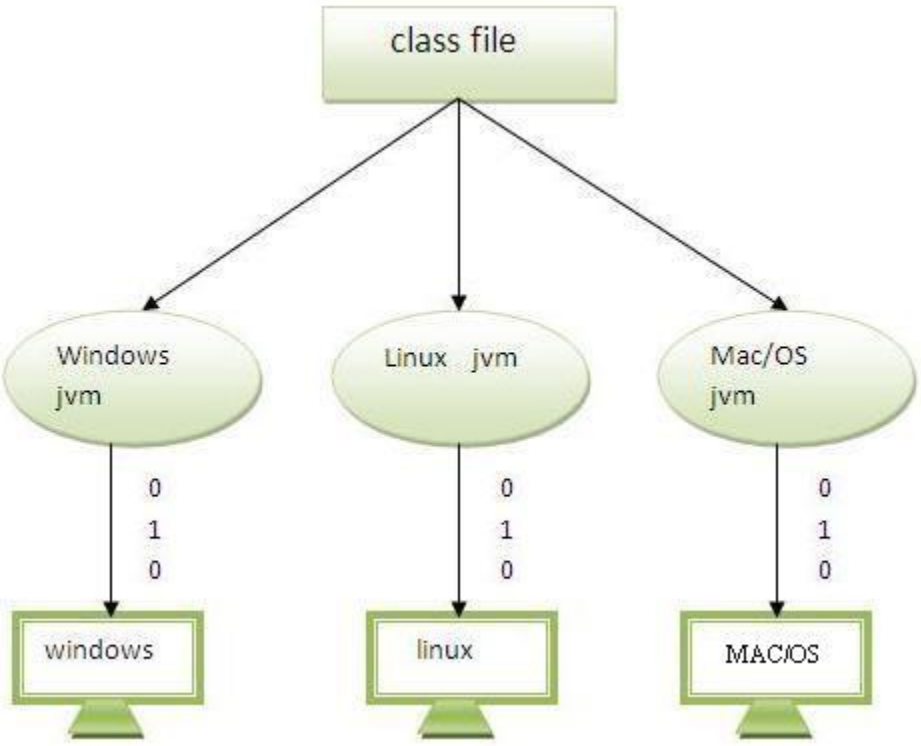
1. Object
2. Class

3. Inheritance
4. Polymorphism
5. Abstraction
6. Encapsulation

Platform Independent

A platform is the hardware or software environment in which a program runs. There are two types of platforms software-based and hardware-based. Java provides software-based platform. The Java platform differs from most other platforms in the sense that it's a software-based platform that runs on top of other hardware-based platforms.It has two components:

1. Runtime Environment
2. API(Application Programming Interface)

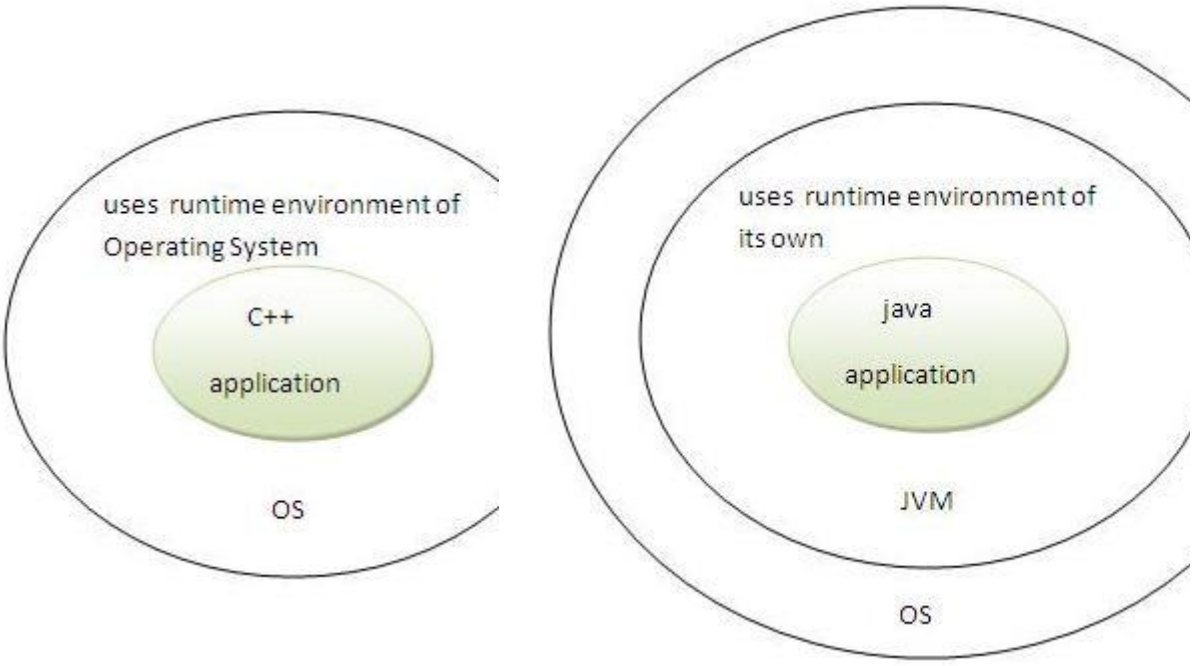


Java code can be run on multiple platforms e.g.Windows,Linux,Sun Solaris,Mac/OS etc. Java code is compiled by the compiler and converted into bytecode.This bytecode is a platform independent code because it can be run on multiple platforms i.e. Write Once and Run Anywhere(WORA).

Secured

Java is secured because:

- No explicit pointer
- Programs run inside virtual machine sandbox.



- **Classloader-** adds security by separating the package for the classes of the local file system from those that are imported from network sources.
- **Bytecode Verifier-** checks the code fragments for illegal code that can violate access right to objects.
- **Security Manager-** determines what resources a class can access such as reading and writing to the local disk.

These security are provided by java language. Some security can also be provided by application developer through SSL,JAAS,cryptography etc.

Robust

Robust simply means strong. Java uses strong memory management. There are lack of pointers that avoids security problem. There is automatic garbage collection in java. There

is exception handling and type checking mechanism in java. All these points makes java robust.

Architecture-neutral

There is no implementation dependent features e.g. size of primitive types is set.

Portable

We may carry the java bytecode to any platform.

High-performance

Java is faster than traditional interpretation since byte code is "close" to native code still somewhat slower than a compiled language (e.g., C++)

Distributed

We can create distributed applications in java. RMI and EJB are used for creating distributed applications. We may access files by calling the methods from any machine on the internet.

Multi-threaded

A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it shares the same memory. Threads are important for multi-media, Web applications etc.

11 Features of Java Programming Language

Simple :

- Java is Easy to write and more readable and eye catching.

- Java has a concise, cohesive set of features that makes it easy to learn and use.
- Most of the concepts are drew from C++ thus making Java learning simpler.

Secure :

- Java program cannot harm other system thus making it secure.
- Java provides a secure means of creating Internet applications.
- Java provides secure way to access web applications.

Portable :

- Java programs can execute in any environment for which there is a Java run-time system.(JVM)
- Java programs can be run on any platform (Linux,Window,Mac)
- Java programs can be transferred over world wide web (e.g applets)

Object-oriented :

- Java programming is object-oriented programming language.
- Like C++ java provides most of the object oriented features.
- Java is pure OOP. Language. (while C++ is semi object oriented)

Robust :

- Java encourages error-free programming by being strictly typed and performing run-time checks.

Multithreaded :

- Java provides integrated support for multithreaded programming.

Architecture-neutral :

- Java is not tied to a specific machine or operating system architecture.
- Machine Independent i.e Java is independent of hardware .

Interpreted :

- Java supports cross-platform code through the use of Java bytecode.

- Bytecode can be interpreted on any platform by JVM.

High performance :

- Bytecodes are highly optimized.
- JVM can executed them much faster .

Distributed :

- Java was designed with the distributed environment.
- Java can be transmit,run over internet.

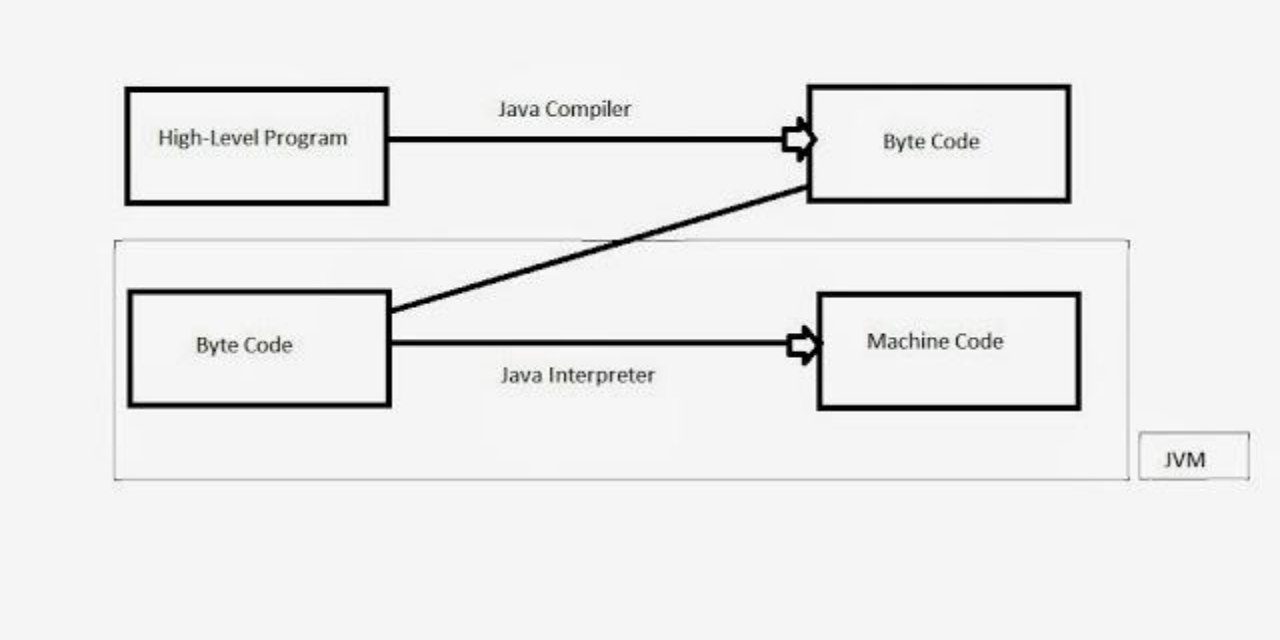
Dynamic :

- Java programs carry with them substantial amounts of run-time type information that is used to verify and resolve accesses to objects at run time.

Why is Java known as platform-neutral language?

Java is known as platform independent because it uses the concept of generating the byte code of the high level program ,and then run that byte code(intermediate code) on JVM(Java Virtual Machine),now what is JVM,actual JVM is a virtual computer system that run on your original computer system .

JVM converts the byte code to machine code according to the your original computer's machine architecture(your computer system architecture like x86,ARM etc.).Here is flow how the Java handle your high level program.



So java compilation is done only once ,after that the byte code can be interpreted on any machine that have JVM.JVM is of different type according to computer system architecture,means for x86 JVM will be different for ARM JVM will be different etc.These are developed by the java vender that isOracle.This JVM come as a part of JDK(Java Development Kit).
So this compiler and interpreter has make it platform-neutral language.

Why main method is static in Java

Now come to the main point *"Why the main method is static in Java"*, there are quite a few reasons around but here are few reasons which make sense to me:

1. Since the main method is static Java virtual Machine can call it without creating any instance of a class which contains the main method.

2. Since C and C++ also have similar main method which serves as entry point for program execution, following that convention will only help Java.

3. If main method were not declared static than JVM has to create instance of main Class and since constructor can be overloaded and can have arguments there would not be any certain and consistent way for**JVM to find main method in Java**.

4. Anything which is declared in class in Java comes under reference type and requires object to be created before using them but static method and static data are loaded into separate memory inside JVM called context which is created when a class is loaded. If main method is static than it will be loaded in JVM context and are available to execution.

Why main method is public in Java

Java specifies several access modifiers e.g. private, protected and public. Any method or variable which is declared public in Java can be accessible from outside of that class. Since the main method is public in

Java, JVM can easily access and execute it.

Why the main method is void in Java

Since the main method in Java is not supposed to return any value, it's made void which simply means main is not returning anything.