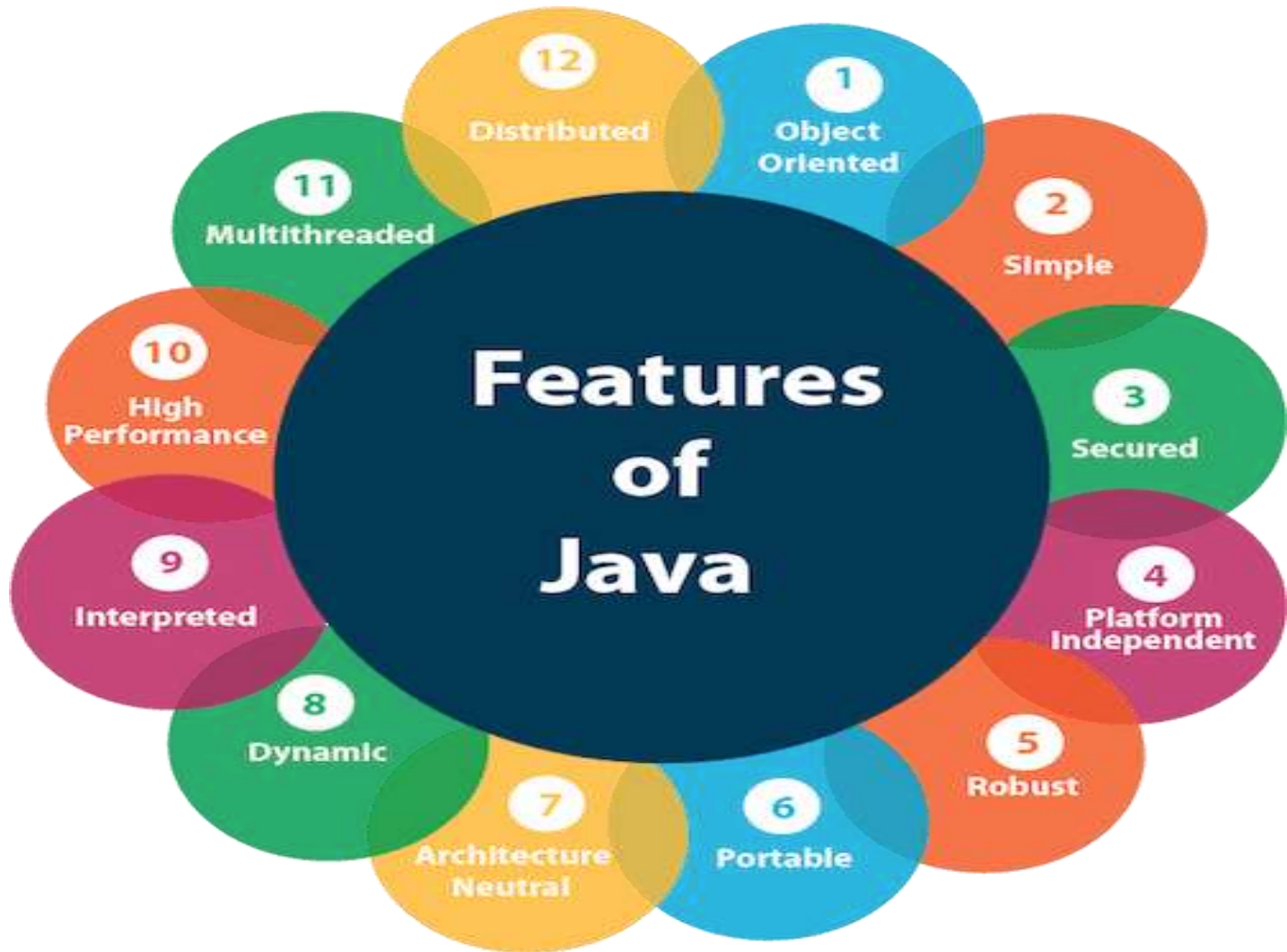


B.Sc (Prog)
Programming in JAVA
V Sem

13-8-2020

Features of JAVA



Simple

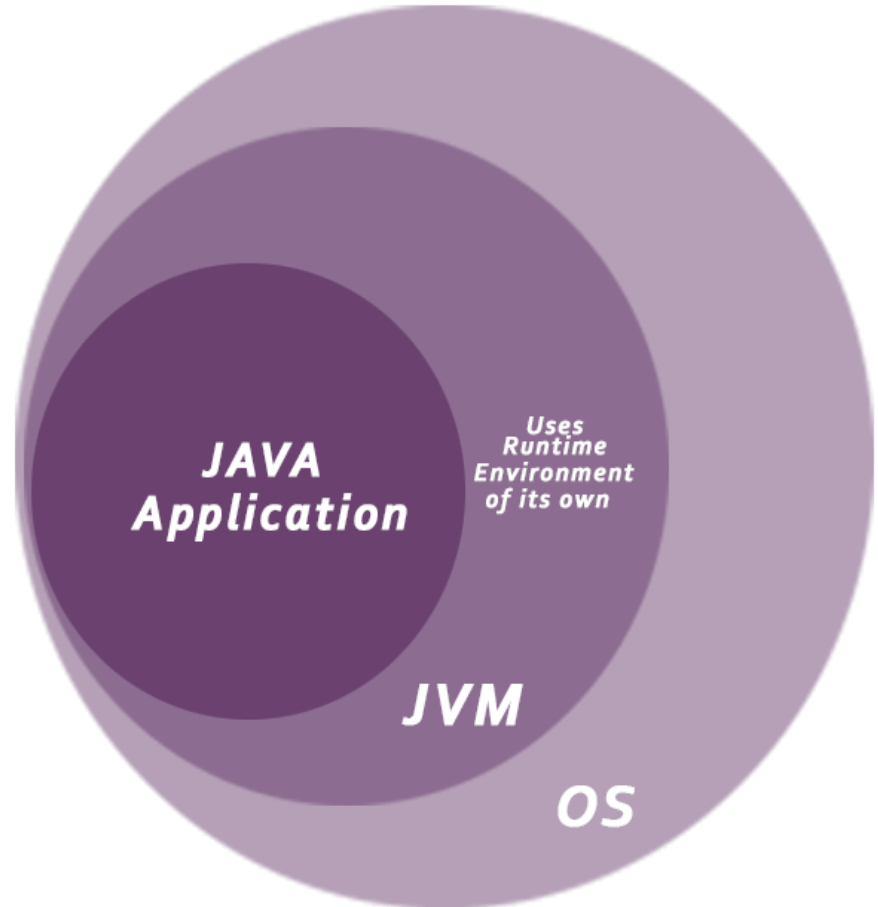
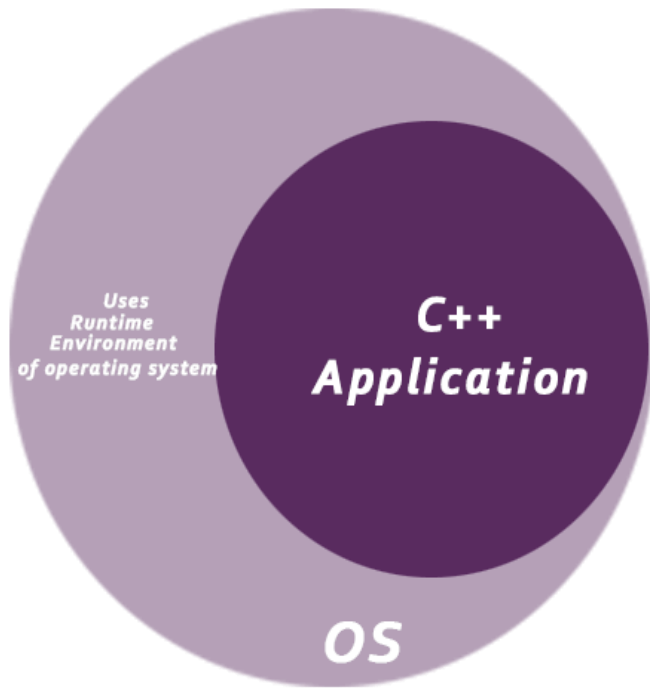
- Java is very easy to learn, and its syntax is simple, clean and easy to understand.
- Java syntax is based on C
- Java has removed many complicated and rarely-used features, for example, explicit pointers, operator overloading, etc.
- There is no need to remove unreferenced objects because there is an Automatic Garbage Collection in Java.

Platform Independent

- Java is platform independent because it is different from other languages like [C](#), [C++](#), etc. which are compiled into platform specific machines while Java is a write once, run anywhere language
- The Java platform differs from most other platforms in the sense that it is a software-based platform that runs on the top of other hardware-based platforms. It has two components:
 - Runtime Environment
 - API(Application Programming Interface)
- Java code can be run on multiple platforms, for example, 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 best known for its security. With Java, we can develop virus-free systems. Java is secured because:
- **No explicit pointer**
- **Java Programs run inside a virtual machine sandbox**



Robust

Java is robust because

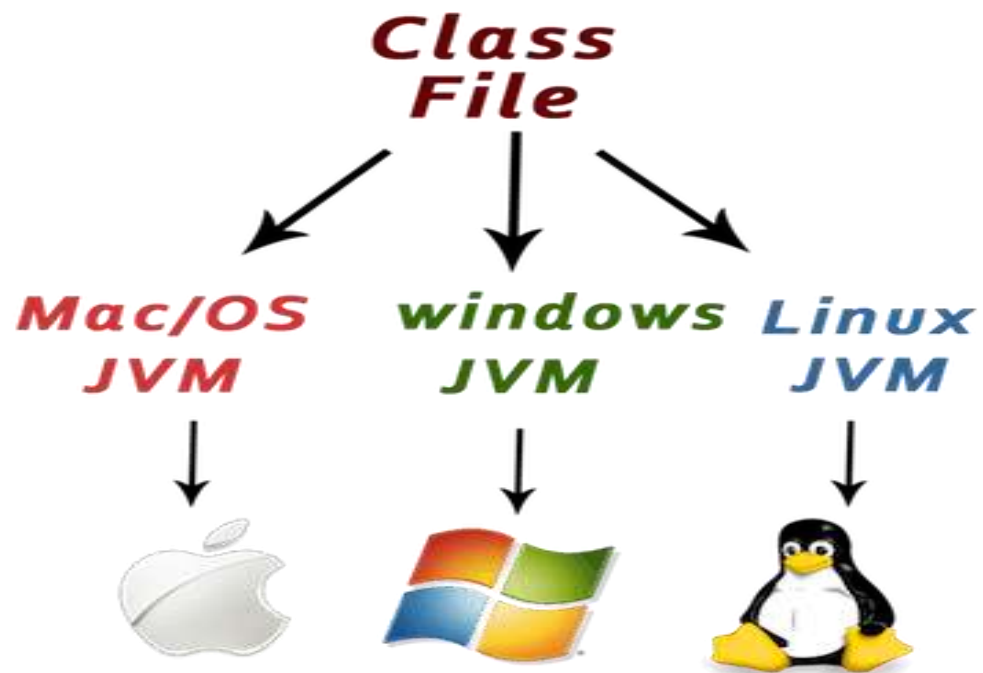
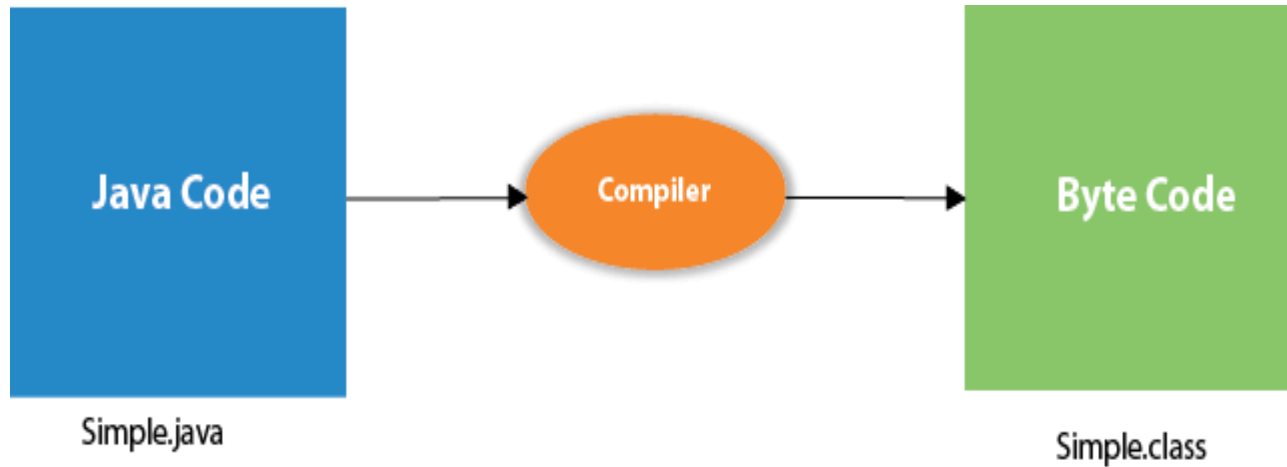
- It uses strong memory management.
- There is a lack of pointers that avoids security problems.
- There is automatic garbage collection in java which runs on the Java Virtual Machine to get rid of objects which are not being used by a Java application anymore.
- There are exception handling and the type checking mechanism in Java

Architecture-neutral

- Java is architecture neutral because there are no implementation dependent features, for example, the size of primitive types is fixed.
- In C programming, int data type occupies 2 bytes of memory for 32-bit architecture and 4 bytes of memory for 64-bit architecture. However, it occupies 4 bytes of memory for both 32 and 64-bit architectures in Java

Portable

- Java is portable because it facilitates you to carry the Java bytecode to any platform. It doesn't require any implementation



High-performance

- Java is faster than other traditional interpreted programming languages because Java bytecode is "close" to native code.
- It is still a little bit slower than a compiled language (e.g., C++). Java is an interpreted language that is why it is slower than compiled languages, e.g., C, C++, etc.

Distributed

- Java is distributed because it facilitates users to create distributed applications in Java.
- RMI and EJB are used for creating distributed applications.
- This feature of Java makes us able to access files by calling the methods from any machine on the internet.

Dynamic

- It supports dynamic loading of classes. It means classes are loaded on demand.
- Java supports dynamic compilation and automatic memory management (garbage collection).

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 doesn't occupy memory for each thread. It shares a common memory area. Threads are important for multi-media, Web applications etc

Object-oriented

- Java is an object-oriented programming language.
Everything in Java is an object

Basic concepts of OOPs are:

- Object
- Class
- Inheritance
- Polymorphism
- Abstraction
- Encapsulation

JVM (Java Virtual Machine)

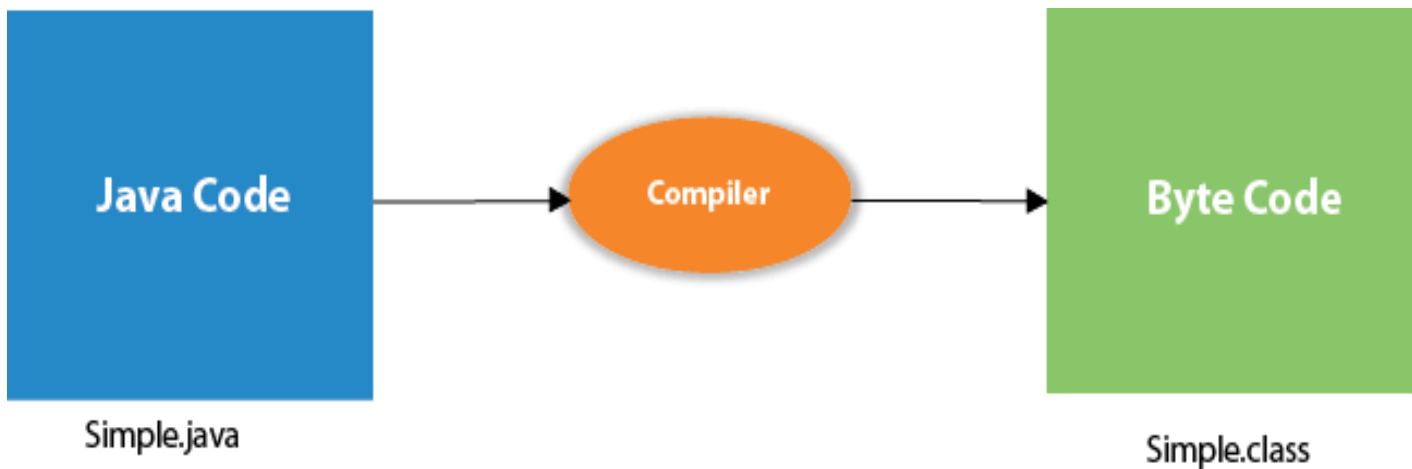
- JVM (Java Virtual Machine) is an abstract machine. It is a specification that provides runtime environment in which java bytecode can be executed
- JVM is platform dependent

JRE

- JRE is an acronym for Java Runtime Environment. It is also written as Java RTE.
- The Java Runtime Environment is a set of software tools which are used for developing Java applications.
- It is used to provide the runtime environment. It is the implementation of JVM.
- It physically exists. It contains a set of libraries + other files that JVM uses at runtime.

Java Development Kit (JDK)

- software development environment which is used to develop Java applications and applets. It physically exists. It contains JRE + development tools.
- JDK is an implementation of any one of the below given Java Platforms released by Oracle Corporation:
 - Standard Edition Java Platform
 - Enterprise Edition Java Platform
 - Micro Edition Java Platform



```
class Simple{  
    public static void main(String args[]){  
        System.out.println("Hello Java");  
    }  
}
```

- **To compile:** javac Simple.java
- **To execute:** java Simple

- **class** keyword is used to declare a class in java.
- **public** keyword is an access modifier which represents visibility. It means it is visible to all.
- **static** is a keyword. If we declare any method as static, it is known as the static method. The core advantage of the static method is that there is no need to create an object to invoke the static method. The main method is executed by the JVM, so it doesn't require to create an object to invoke the main method. So it saves memory.

- **void** is the return type of the method. It means it doesn't return any value.
- **main** represents the starting point of the program.
- **String[] args** is used for command line argument.
- **System.out.println()** is used to print statement. Here, System is a class, out is the object of PrintStream class, println() is the method of PrintStream class.