**Bytecode in Java | Bytecode vs Machine code**

**Bytecode in Java** is a highly optimized set of instructions for the [*Java Virtual Machine*](https://www.scientecheasy.com/2021/03/what-is-jvm.html/) (JVM) that reads and interprets to run the java program.

A bytecode is a binary program code that can only run on JVM. In other words, it is a machine language (code) for JVM in the form of .class file, but it is not machine specific because it is not a native code.

In simple words, it is not machine language (machine instructions) for any specific hardware processor.

Byte code acts as an intermediate language that is platform (machine) independent. It is generated by Java interpreter that can be directly run by a real machine.

**Characteristics of Java Bytecode**

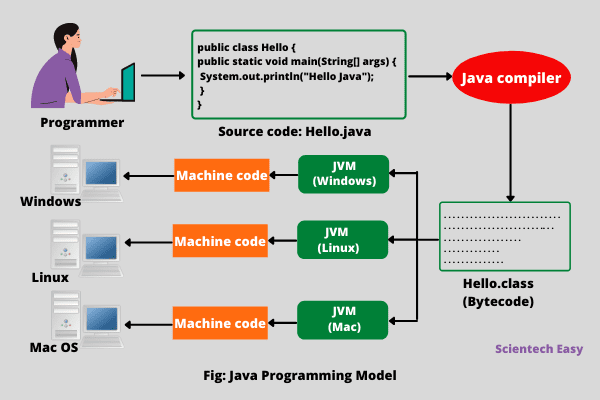
Java bytecode has two most important characteristics that are as follows:

* Byte code is independent of processor, i.e., Java program can be executed on any processor architecture.
* It does not depend on operating systems such as Windows, Linux, and Mac OS.

**How does Bytecode work in Java?**

When we write a java program, the source code (in the form of .java file) is compiled by [*Java compiler*](https://www.scientecheasy.com/2021/03/java-compiler.html/) and converted into byte code in the form of a .class file.

Look at the below figure to understand better.

[](https://www.scientecheasy.com/2021/03/java-bytecode.html/)

This compiled byte code is platform-independent code that can be run on any different computer machine on which JVM [*interpreter*](https://www.scientecheasy.com/2021/04/interpreter-in-java.html/) is installed. In simple words, write once, compile and run anywhere (WOCRA).

These bytecodes are not machine instructions. Therefore, in the second stage, JVM interpreter takes the compiled byte code and converts it into machine code that can be directly executed by any computer system that is running java program source code.

Resources required to execute byte code are made available by the JVM that calls the microprocessor to allocate the required resources.

Thus, we can say that JVM plays an important role in the execution of java program.

Hence, Java is both a compiled and an interpreted language that helps to move java programs easily from one computer system to another computer system.

Any changes and upgrades of operating systems, processors, and system resources do not affect the java program.

This is the reason why Java has become one of the most popular programming languages in the world that interconnects different kinds of systems worldwide on the internet.

**Advantages of Java Bytecode**

Java Bytecode has mainly two advantages that are as follows:

1. Byte code makes the java program portable across the different hardware (processors) and operating system platforms. The poly requirement is that Java Virtual Machine must be installed on them.

2. The second advantage is that it increases security of code because of control of JVM over the execution of byte code file.

**Difference between Bytecode vs Machine code**

The main difference between the byte code and machine code is that byte code can be run only on JVM whereas machine code is a set of instructions in machine language that can be directly run by the CPU.

**Difference between .java and .class**

Java source code file (with a .java extension) are compiled into bytecode (with a .class extension), which is then interpreted and executed by a Java processor called JVM.

**Key points:**

1. A bytecode in Java is a set of byte-long instructions that Java compiler produces and Java interpreter (JVM) executes.

2. When Java compiler compiles .java file, it generates a series of bytecode (machine independent code) and stores them in a .class file.

3. JVM then interprets and executes the byte code stored in the .class file and converts them into machine code.

4. The byte code remains the same on different platforms such as Windows, Linux, and Mac OS.