

# **Programming Language:**

* Language is a mode of communication, used to share ideas, opinions with each other…
* Programming language is a computer language, using by programmers to communicate with computers.
* It is a set of rules for instructing a computer to perform specific tasks(which tells the computer what to do).

# **Machine Language:**

* Machine Code or Language is a low-level language made up of binary numbers or bits that computers can understand.
* Machine code is a set of instructions in machine language. The CPU can directly execute it
* The Programming language must be translated to machine code by a compiler or interpreter.because binary code is the only language that computer hardware can understand

### **Compiler:**

A compiler converts the entire source code into an equivalent machine code at once.

### **Interpreter:**

An interpreter converts the source code line by line into the equivalent machine code.

### **Byte Code:**

Bytecode is an intermediate code generated from compiling a source code which can be executed by a virtual machine.

# **JAVA**

* JAVA is a Open Source class based, high level object-oriented programming language
* It is a platform independent language. It is a general-purpose programming language intended to let programmers write once, run anywhere (WORA)
* The compiled Java code can run on all platforms that support Java without the need to recompile
* Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM)

## **Basic Java Features:**

1. **Platform Independent** - we can run the application to any platform such as windows, mac, linux,etc
2. **Open Source** - all the codes are designed to be publicly accessible
3. **Multithreading** - It enables a program to perform several task simultaneously
4. **More Secure -**  type-safe and provides automatic garbage collection, enhancing the robustness of application code
5. **Portable** - "Write once run anywhere”

## **Principles:**

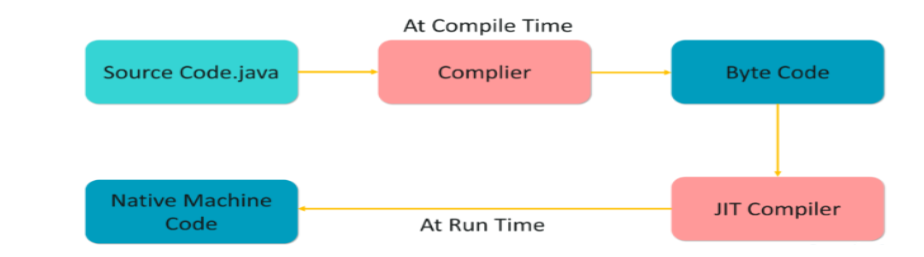
* It must be simple, object-oriented, and familiar.
* It must be robust and secure.
* It must be architecture-neutral and portable.
* It must execute with high performance.
* It must be interpreted, threaded, and dynamic.

## **JAVA Virtual Machine:**

* JVM acts as a run-time engine to run java applications.
* JVM is the one that actually calls the main method present in Java Code. It is a part of JRE
* It is specifically responsible for converting bytecode to machine-specific code
* The JVM **Loads** the Code, **Verifies** the Code, **Executes** the Code and **provide Runtime Environment**

## **JIT in java (Just In Time)**

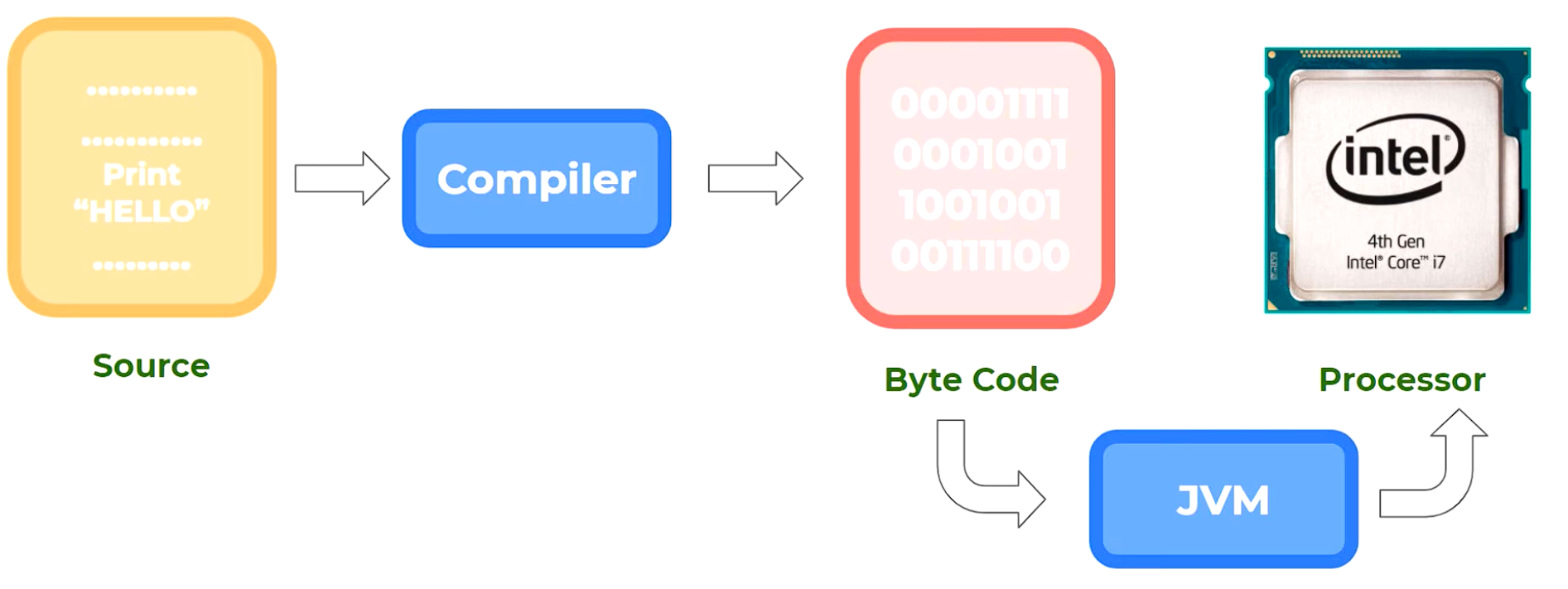
* JIT is a integral part of JVM
* Java Runtime Environment, that is responsible for performance optimization of java based applications at run time
* Bytecode is one of the most important features of java that aids in cross-platform execution.
* These Bytecodes have to be interpreted or compiled to proper machine instructions.
* Interpreting the bytecode affects the speed of execution.
* In order to improve performance, JIT compilers interact with the Java Virtual Machine (JVM) at run time to compile byte code sequences into native machine code.
* The frequently used methods are compiled once and kept into memory



## 

## **How source codes are executed in java**

* In java, Programs are compiled into bytecode, which JVM then executes at runtime.



## **OOPs in JAVA**

* Object-oriented programming is a core of Java Programming, which is used for designing a program using classes and objects
* OOps in java is to improve code readability and reusability by defining a Java program efficiently
* The main principles of object-oriented programming are abstraction, encapsulation, inheritance, and polymorphism. These concepts aim to implement real-world entities in programs.

## **List of OOPs Concepts in Java**

* Class
* Method
* Object
* Abstraction
* Encapsulation
* Inheritance
* Polymorphism

## 