Prerequisite:

- **High level language**
- **Low level language**
- **Different type of High-level language (procedural, functional, OOPS)**

JAVA 🍵

Ok here we learn about the JAVA programming **but why are you used coffee symbol 🤔 ?**

History of JAVA:

1. **Green Team and the Birth of Java**:
    - In **June 1991**, a small team of Sun engineers, known as the **Green Team**, initiated the **Java language project**. Their goal was to create a language for digital devices like set-top boxes and televisions.
    - Initially, Java was designed for **embedded systems** in electronic appliances.
    - The project started with the name **“Greentalk”**, and the file extension was .gt.
    - Later, it was renamed to **“Oak”** as part of the **Green** project.
2. **Why “Oak”?**:
    - The name **“Oak”** symbolized strength and resilience, much like the oak tree, which is a national symbol in several countries.
    - However, in 1995, it was renamed to **“Java”** due to trademark conflicts with Oak Technologies.
3. **Choosing the Name “Java”**:
    - The team brainstormed various names, including “dynamic,” “revolutionary,” “Silk,” “jolt,” and “DNA.”
    - They sought a name that reflected the essence of the technology: **revolutionary, dynamic, lively, cool, unique, and easy to spell**.
    - **James Gosling**, the father of Java, chose the name **“Java”** while sipping coffee near his office.

- Interestingly, **Java** is an island in Indonesia where the first coffee was produced (known as **Java coffee**).

<u>This is the reason why I'm used this coffee symbol</u>**

What is JAVA?

JAVA is a class based, heigh-level, object-oriented programming language developed by "JAMES GOSLING" and his friend in the year 1995.

NOTE:

- The first version of java (JDK 1.0) was released on the year JAN-23rd 1996 by "**SUN MICROSYSTEM**"
- Latest version of java (JDK 22) 2024 by "oracle"

But the question is we are already done C++ and but why we learn JAVA??

Java is one of the most popular programming languages in the world, and for good reason. It is a powerful, versatile, and secure language that can be used to develop a wide variety of applications.

** Features of JAVA

1. **Platform Independence (Write Once, Run Anywhere)**:
   - Java code is compiled into an **intermediate form called bytecode**. This bytecode can run on any platform with a **Java Virtual Machine (JVM)**.
   - JVM interprets the bytecode, making Java applications **platform-independent**.
2. **Object-Oriented Programming (OOP)**:
   - Java follows the principles of OOP, including **encapsulation, inheritance, and polymorphism**.
   - It allows developers to create modular, reusable, and maintainable code.
3. **Strongly Typed Language**:

  - Java enforces strict type checking during compilation.
  - This helps catch errors early and ensures type safety.

4. **Automatic Memory Management (Garbage Collection)**:
   - Java manages memory automatically through **garbage collection**.
   - Developers don’t need to explicitly free memory; the JVM handles it.
5. **Rich Standard Library (Java API)**:
   - Java provides a comprehensive set of classes and methods in its **Standard Library**.
   - It covers areas like I/O, networking, data structures, and more.
6. **Exception Handling**:
   - Java has a robust exception handling mechanism.
   - Developers can catch and handle exceptions gracefully.
7. **Multithreading and Concurrency**:
   - Java supports multithreading, allowing developers to create **concurrent** applications.
   - The java.util.concurrent package provides tools for managing threads.
8. **Security and Safety**:
   - Java emphasizes security.
   - The JVM ensures that untrusted code doesn’t harm the system.

First program in java

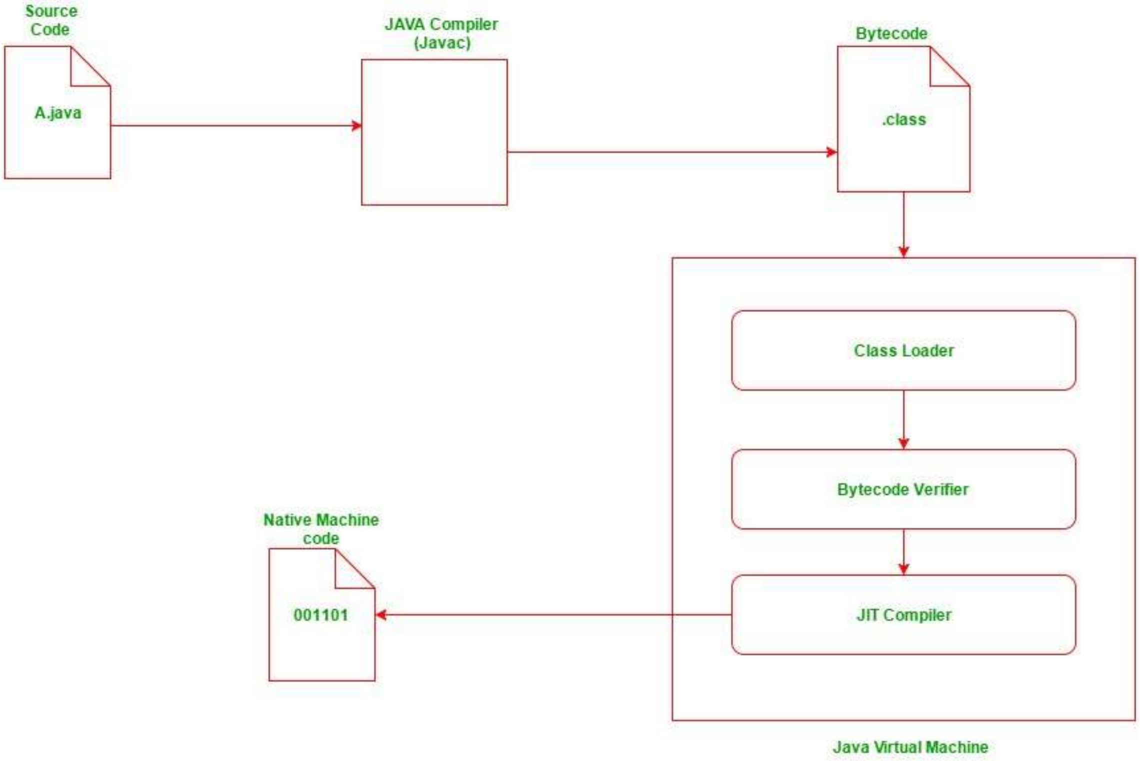
Now how to execute this line of code??

```
class Test
{
    public static void main(String []args)
    {
        System.out.println("My First Java Program.");
    }
}
```

Execution process:

1. Save class name (Test).java →save
2. javac Test.java →
3. java class name (Test)

Now, how to execute .java file internally 🤔 🤔?



Important Terms: JDK, JRE, JVM and JIT

1. **JDK (Java Development Kit)**:
   - The **JDK** is a **software development environment** used for creating and running Java applications.
   - It includes:
     - **Development Tools**: These tools provide an environment for writing and compiling Java programs.
     - **JRE (Java Runtime Environment)**: Necessary for executing Java programs.
   - Developers use the JDK to write, compile, and package Java code.
2. **JRE (Java Runtime Environment)**:
   - The **JRE** is an **installation package** that allows you to **run** Java programs on your machine.
   - It includes:
     - **Java Virtual Machine (JVM)**: Responsible for executing Java bytecode.
     - **Core classes** required for running Java applications.
     - **Supporting files** needed for proper execution.
   - End-users who only want to run Java programs use the JRE.

3. **JVM (Java Virtual Machine)**:
    - The **JVM** is a crucial part of both the JDK and JRE.
    - It serves as an **interpreter** for Java programs.
    - Key points about JVM:
        - It executes Java programs **line by line**.
        - JVM is responsible for converting Java bytecode into machine-specific instructions.
        - Whenever you run a Java program, an instance of JVM is created.
4. **JIT (Just-In-Time Compiler)**:
    - The **JIT** is a dynamic component within the JVM.
    - It **optimizes** the execution of Java programs.
    - How it works:
        - When a Java program starts executing, the JIT compiles parts of the bytecode into native machine code.
        - This compilation happens **on the fly**, improving performance during runtime.