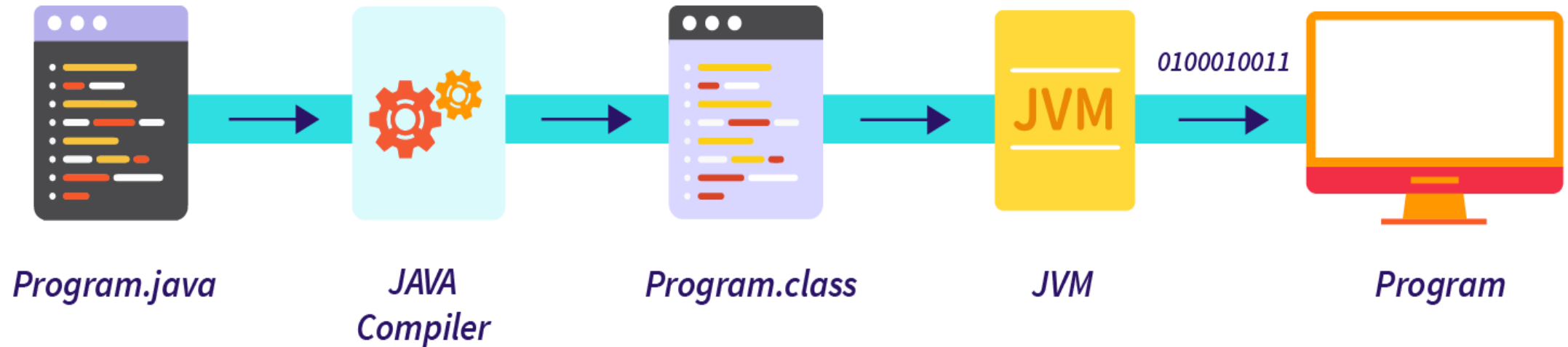




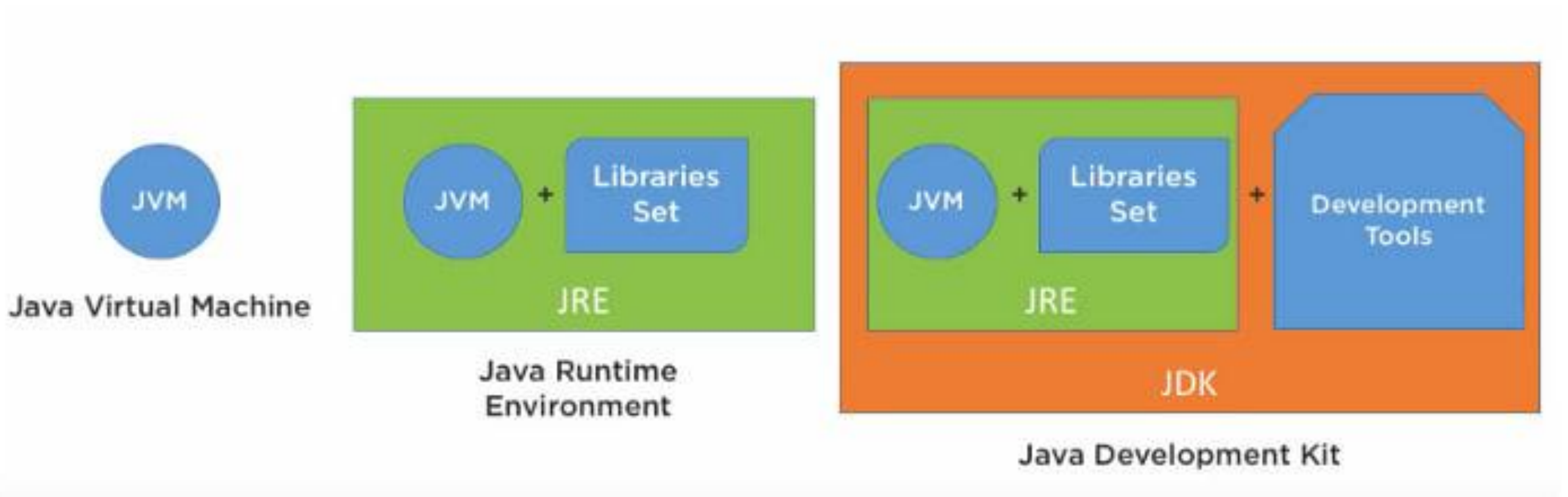
# DIFFERENCE

JDK vs JRE vs JVM vs JIT

# How Java Program Run ?



# JVM vs JRE vs JDK



# What is JVM ?

JVM → Java Virtual Machine

Provides Runtime Environment in which Java bytecode can be executed

## Tasks of JVM

- Loads Code
- Verifies Code
- Executes Code
- Provides Runtime Environment

JVM is platform dependent i.e. for each software and hardware we have different JVM configuration.

JVM does not exist physically. It is abstract in nature.



# What is JRE ?

JRE → JVM + Set of Libraries

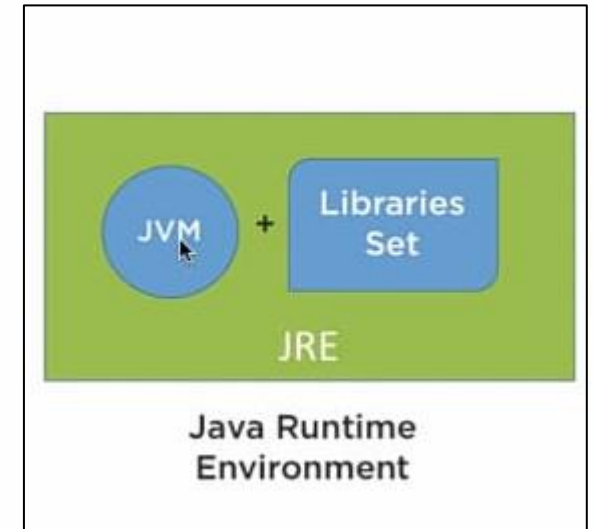
It is the implementation of JVM

To run any Java code JRE is minimum required.

JRE contains set of libraries that JVM uses at runtime.

JRE physically exists.

JRE is platform dependent.



# What is JDK?

JDK → JRE + Development Tools

It is a full featured Software Development Kit.

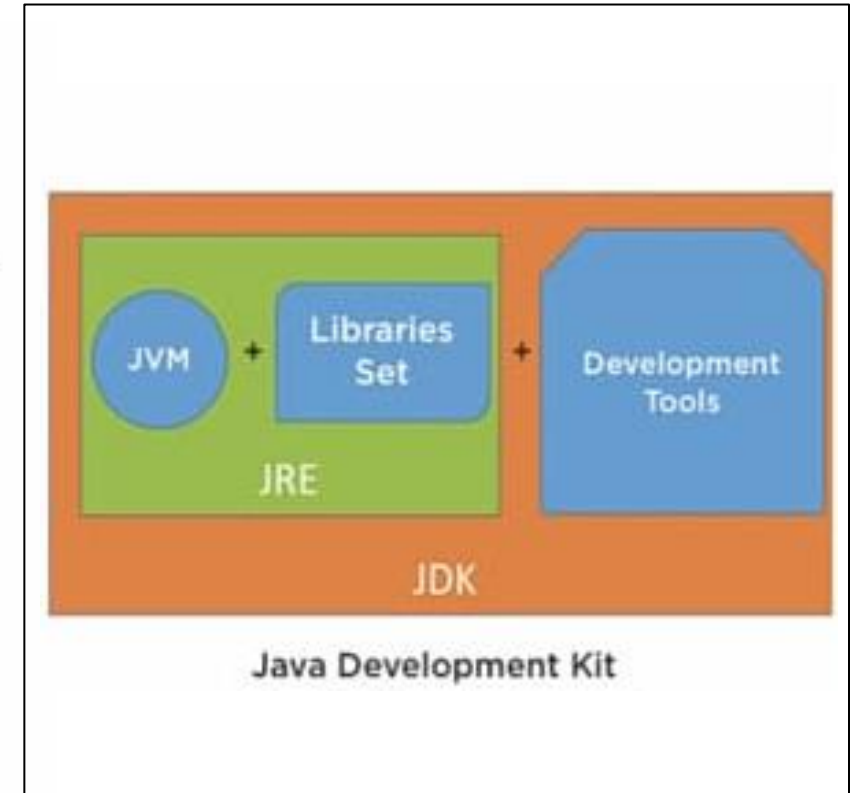
It contains JRE + Development Tools

→ JRE

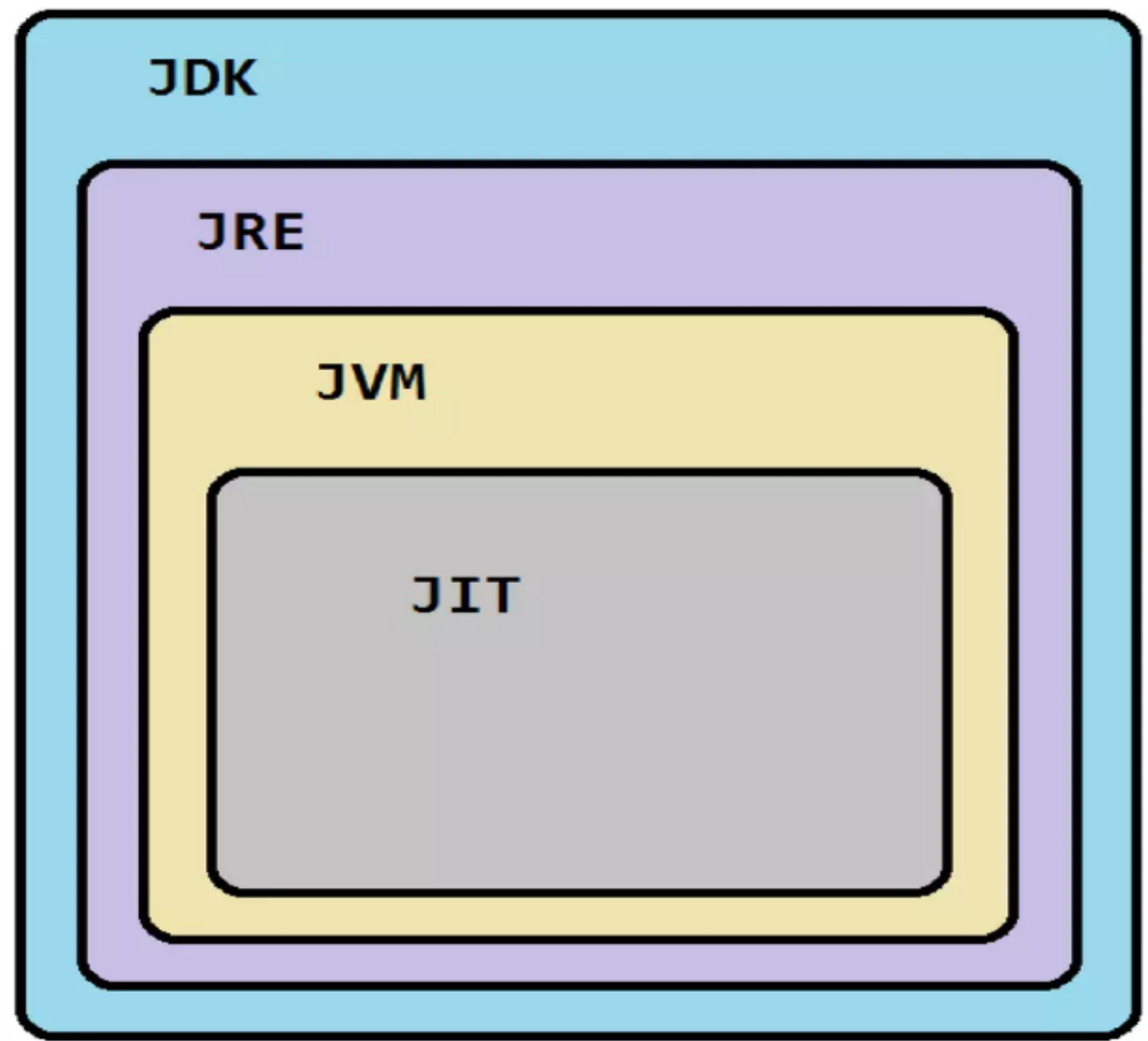
- JVM + Libraries

→ Development Tools

- Debugger + Compiler + JavaDoc

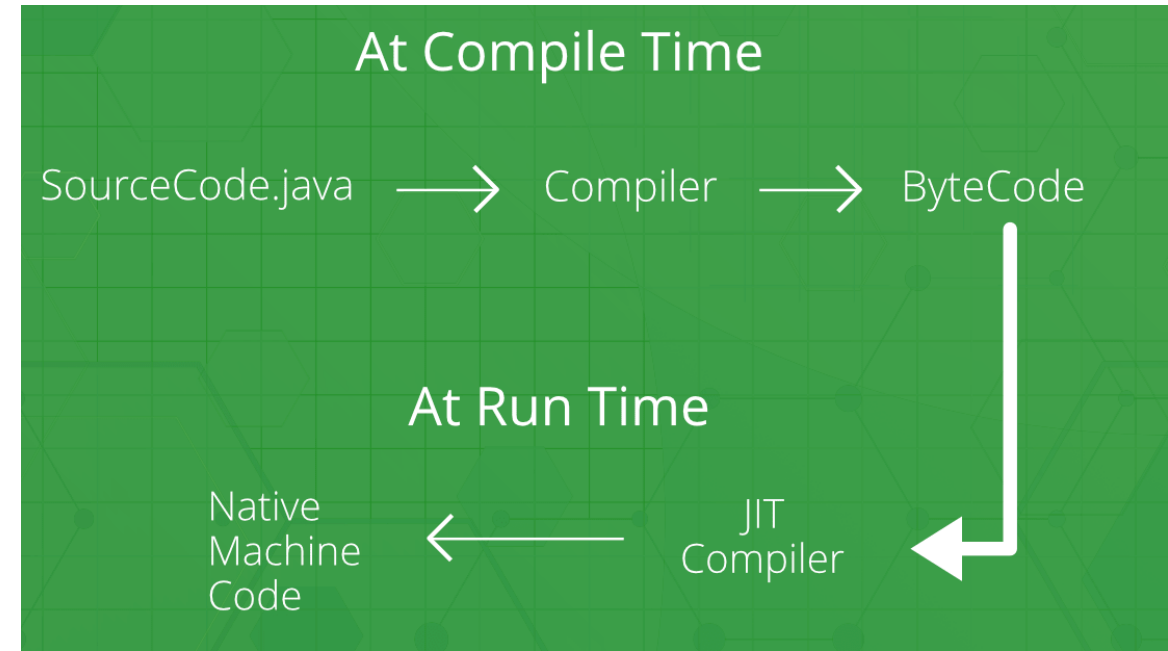


JDK contains JRE, JRE contains JVM, JVM contains JIT.  
Hence once if we install JDK all the other components gets installed.



# What is JIT ?

- **JIT** stands for **Just-In-Time Compiler**.
- JIT was introduced from 1.2 JDK
- It transforms the **bytecode** of a VM into the machine code.
- **JIT** in Java is an integral part of the **JVM**. It accelerates execution performance many times over the previous level.
- It optimizes the performance of the Java application at compile or run time.





## **Advantages of JIT Compiler**

- It requires less memory usages.
- The code optimization is done at run time.
- It uses different levels of optimization.

## **Disadvantages of JIT Compiler**

- The program with less line of code does not take the benefit of the JIT compilation.
-

# JVM vs JRE vs JDK Conclusion



# THANKS FOR WATCHING



DON'T FORGET **LIKE**♥ **COMMENT**□ **SUBSCRIBE**□