how java complies code with example and process

How Java Compiles Code (With Example and Process)

Java follows a two-step compilation and execution process using a compiler and an interpreter. This makes Java platform-independent, meaning you can write Java code once and run it anywhere.

Java Compilation & Execution Process

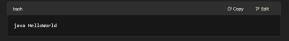
Step-by-Step:

1. Write Source Code

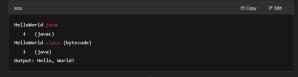
You write code in a file with the . java extension. Example:

```
public class HelloWorld {
   public static void main(String[] args) {
        System.out.println("Hello, Morld!");
}
```

- 2. Compile the Code (javac HelloWorld.java)
 - The Java Compiler (javac) checks for syntax errors.
 - Converts the .java file to a .class file.
 - THE LUMS THE CONTAINS DYTECODE, WHICH IS NOT HACHINE SPECIFIC. javac HelloWorld.java
- 3. Run the Bytecode (java HelloWorld)
 - The Java Virtual Machine (JVM) reads the .class file.
 - $\bullet \quad \mbox{JVM interprets or JIT compiles the bytecode into machine code}.$
 - Executes the program on your current platform (Windows, Linux, etc.).



Flow Diagram



Summary			
Step	Tool Used	Output File	Description
Write Code	Text Editor	HelloWorld.java	Java source file
Compile Code	javac	HelloWorld.class	Bytecode (platform-independent)
Run Bytecode	java	Terminal Output	JVM interprets & executes the bytecode

→ Bonus: Why Bytecode?

- Bytecode makes Java platform-independent.
- JVM on each OS translates bytecode to native machine code.

Let me know if you want the same process for multiple . java files or class with packages.

iava idk explanation components

☑ Java JDK (Java Development Kit) – Eالبدا lanation & Components

JDK (Java Development Kit) is a complete software development kit required to develop Java applications. It includes tools for compiling, debugging, and running Java programs

- It is used by Java developers to build, compile, and test Java applications.
 It includes JRE (Java Runtime Environment) + development tools.





Example (How JDK Works):

- 1. You write Hello.java
- 2. Compile using: javac Hello.java → Hello.class (bytecode)
- 3. Run using: java Hello → Output shown by JVM

All these tools (javac , java , etc.) are part of the JDK.

Summary (Easy to Remember)

- JDK = JRE + Dev Tools
- It is used for writing, compiling, and running Java programs.
- Includes important tools: javac , java , javadoc , jar , jdb , etc.

Let me know if you'd like a diagram or need the JDK setup steps.

ច្ចឲ្យ 🌣 ជ 🗸

