

**Java Compiler (javac):**

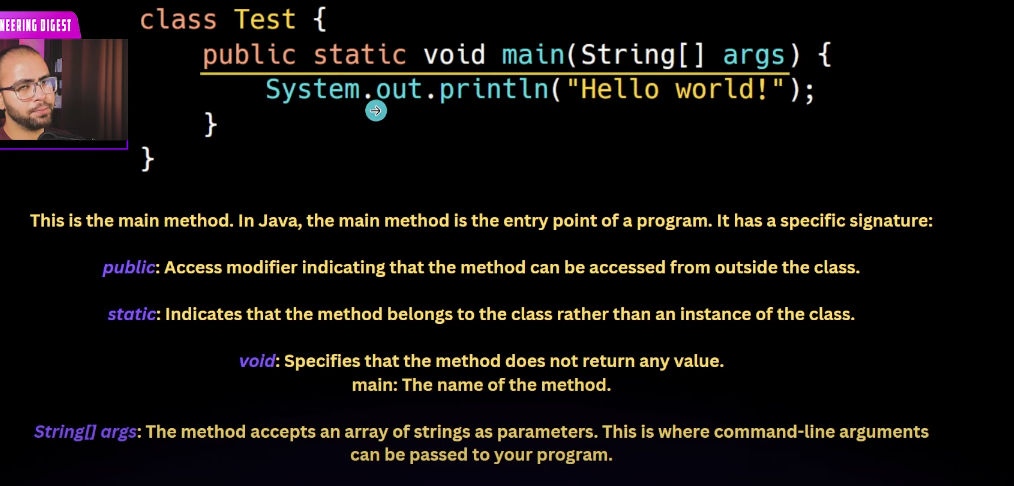
1. **Compilation to Bytecode**:
   * Java source code (.java files) is compiled into bytecode (.class files) by the Java Compiler (javac). Bytecode is an intermediate, platform-independent code that can be executed by the Java Virtual Machine (JVM).
2. **Error Detection**:
   * The compiler checks for syntax errors, type checks, and other compile-time errors before generating the bytecode. If there are errors, the compilation process stops, and the errors are reported.

javac MyProgram.java

* + This command compiles the MyProgram.java file into a MyProgram.class file containing bytecode.

**Java Virtual Machine (JVM):**

1. **Interpretation and Just-In-Time (JIT) Compilation**:
   * The JVM interprets the bytecode and executes it. The JVM translates the bytecode into machine code for the host system on the fly, which is an interpretive process.
2. **Just-In-Time (JIT) Compiler**:
   * To improve performance, the JVM includes a Just-In-Time (JIT) compiler that compiles frequently executed bytecode into native machine code at runtime. This combination of interpretation and JIT compilation balances portability and execution speed.



A black background with yellow text

Description automatically generated

Command Line Arguments

 System.out.println(args[0]);

 System.out.println(args.length);

Java Test “hi” “hello” (hi hello 🡪 This will be treated as 2 argument)

O/P: hi

2

print: Prints text without a newline.

println: Prints text with a newline.

printf: Prints formatted text without a newline (unless specified).

System.out.printf("Number: %d, Pi: %.2f", number, pi);

package demo;

Import test.cat;

Import package\_name.class\_name;

Cat c = new cat(); //this is from test folder

demo.Cat c1 = new demo.cat() //this is from demo package which is the current package the file is in.

package test.try;

Both test and try are pakages and try is the sub package of test