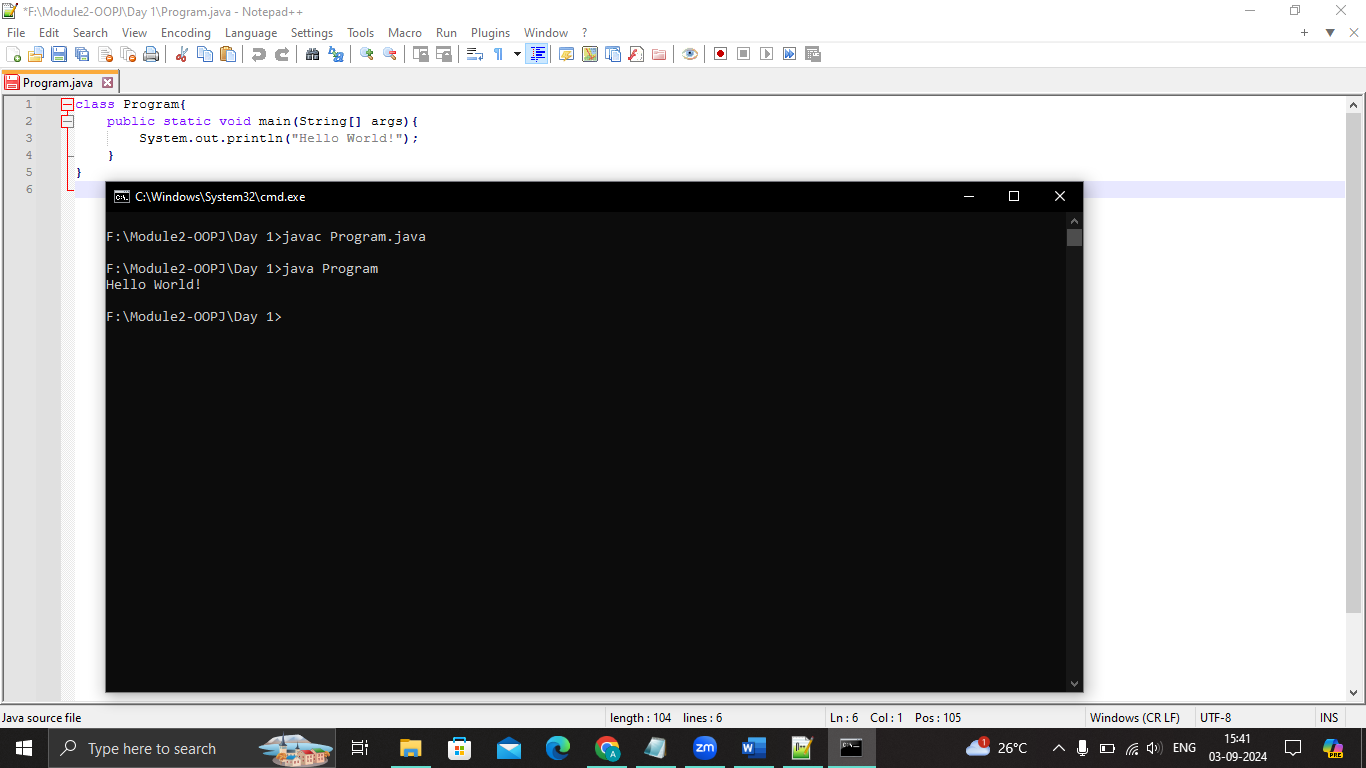
**Assignment 1**

**Coding Assignments**

1. **Hello World Program**: Write a Java program that prints "Hello World!!" to the console.



1. **Compile with Verbose Option**: Compile your Java file using the -verbose option with javac. Check the output.

* When you compile a Java file using the -verbose option with javac, the compiler provides detailed information about the compilation process. This includes messages about which classes and resources are being loaded, what files are being written, and more.

F:\Module2-OOPJ\Day 1>javac -verbose Program.java

[parsing started SimpleFileObject[F:\Module2-OOPJ\Day 1\Program.java]]

parsing completed 33ms]

[loading /modules/jdk.crypto.cryptoki/module-info.class]

[loading /modules/jdk.nio.mapmode/module-info.class]

[loading /modules/java.rmi/module-info.class]

[loading /modules/java.xml/module-info.class]

[loading /modules/jdk.jcmd/module-info.class]

[loading /modules/java.logging/module-info.class]

.

.

.

[loading /modules/java.base/java/io/Flushable.class]

[loading /modules/java.base/java/lang/Comparable.class]

[loading /modules/java.base/java/lang/CharSequence.class]

[loading /modules/java.base/java/lang/constant/Constable.class]

[loading /modules/java.base/java/lang/constant/ConstantDesc.class]

[wrote Program.class]

[total 688ms]

1. **Inspect Bytecode**: Use the javap tool to examine the bytecode of the compiled .class file. Observe the output.

* The javap tool is used to disassemble Java bytecode and display information about the classes and methods in a .class file. This tool can be particularly useful for inspecting the bytecode generated by the Java compiler.

