

OOPJ

Assignment-1

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

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

```
D:\Practice\OOPJ>java HelloWorld.java  
Hello World!!
```

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

When we compile a Java file using the “`-verbose`” option with the `javac` command, it provides detailed information about the compilation process, including the classes that are being loaded, the files being read, and other internal details.

```
D:\Practice\OOPJ>javac -verbose HelloWorld.java  
[parsing started SimpleFileObject[D:\Practice\OOPJ\HelloWorld.java]]  
[parsing completed 19ms]  
[loading /modules/jdk.jstatd/module-info.class]  
[loading /modules/java.instrument/module-info.class]  
[loading /modules/jdk.scripting.nashorn.shell/module-info.class]  
[loading /modules/jdk.naming.dns/module-info.class]  
[loading /modules/jdk.jfr/module-info.class]  
[loading /modules/jdk.zipfs/module-info.class]  
[loading /modules/jdk.jconsole/module-info.class]  
[loading /modules/java.security.jgss/module-info.class]  
[loading /modules/java.sql/module-info.class]  
[loading /modules/jdk.security.jgss/module-info.class]  
[loading /modules/jdk.internal.jvmstat/module-info.class]  
[loading /modules/jdk.crypto.ec/module-info.class]  
[loading /modules/jdk.compiler/module-info.class]  
[loading /modules/jdk.management/module-info.class]  
[loading /modules/jdk.charsets/module-info.class]  
[loading /modules/jdk.net/module-info.class]  
[loading /modules/jdk.jdi/module-info.class]  
[loading /modules/java.desktop/module-info.class]  
[loading /modules/jdk.scripting.nashorn/module-info.class]
```

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

```
D:\Practice\OOPJ>javap -c HelloWorld
Compiled from "HelloWorld.java"
class HelloWorld {
    HelloWorld();
        Code:
            0: aload_0
            1: invokespecial #1                  // Method java/lang/Object."<init>":()V
            4: return

    public static void main(java.lang.String[]);
        Code:
            0: getstatic     #2                  // Field java/lang/System.out:Ljava/io/PrintStream;
            3: ldc           #3                  // String Hello World!!
            5: invokevirtual #4                  // Method java/io/PrintStream.println:(Ljava/lang/String;)V
            8: return
}
```