Compiler

+2
Published Sep 7, 2022•**Updated Apr 2, 2023**

Contribute to Docs

Java **compilers** are programs that take source code and produce class files containing platform-neutral Java bytecode that can be executed by the Java Virtual Machine (JVM).

Rather than interpret high-level Java source code, the JVM interprets low-level Java bytecode (somewhere between human-readable code and machine code that is specific to a particular computer). Bytecode is platform-neutral and, therefore, can be interpreted by any JVM running on any computer system. This is what makes compiled Java programs portable.

Most Java compilers do little to no optimization of the code, leaving that task for the JVM to do at run time. The JVM loads the bytecode and either interprets it, or just-in-time (JIT) compiles it to machine code, and then optimizes it.

Compiling a Java File

The Java compiler is executed on the command line in Unix or DOS shells as follows:

```
javac ProgramName.java
```

```
javac options ProgramName.java
```

Where `ProgramName` is the name of a given `.java` file to be compiled.

Part of the configuration of Java requires setting up the `CLASSPATH`, so the Java compiler can find `ProgramName.java` in the file system. This can be done by using the -classpath option with the javac compiler command, or by setting the CLASSPATH environment variable, shown below.

Linux or Unix:

```
export CLASSPATH=path/to/file
```

Windows:

```
set CLASSPATH=path\to\file
```

Online Java Compilers

Several sites offer online Java compilers, allowing the entry of small Java programs in a web-based IDE and running them on the fly. JDoodle is one such site that offers this ability.