

Difference Between compiler & interpreter:

Compiler:

A compiler is a special program that processes statements written in a particular programming language and turns them into machine language or "code" that a computer's [processor](#) uses. Typically, a programmer writes language statements in a language such as [Pascal](#) or [C](#) one line at a time using an *editor*. The file that is created contains what are called the *source statements*. The programmer then runs the appropriate language compiler, specifying the name of the file that contains the source statements.

The [Java](#) programming language, a language used in [object-oriented programming](#), introduced the possibility of compiling output (called [bytecode](#)) that can run on any computer system platform for which a Java [virtual machine](#) or bytecode interpreter is provided to convert the bytecode into instructions that can be executed by the actual hardware processor. Using this virtual machine, the bytecode can optionally be recompiled at the execution platform by a [just-in-time compiler](#).

Interpreter:



An interpreter is a computer program that is used to directly execute program instructions written using one of the many high-level programming languages.

The interpreter transforms the high-level program into an intermediate language that it then executes, or it could parse the high-level source code and then performs the commands directly, which is done line by line or statement by statement.