1. Difference between Java and C language (geeksforgeeks)

C is much faster than Java

Java is slower than C due to overhead.

|  |  |
| --- | --- |
| C | JAVA |
| C was developed by Dennis M. Ritchie between 1969 and 1973. | Java was developed by James Gosling in 1995. |
| C is a Procedural Programming Language. | Java is Object-Oriented language. |
| C is more procedure-oriented. | Java is more data-oriented. |
| C is a middle-level language because binding of the gaps takes place between machine level language and high-level languages. | Java is a high-level language because translation of code takes place into machine language using compiler or interpreter. |
| C is a compiled language that is it converts the code into machine language so that it could be understood by the machine or system. | Java is an Interpreted language that is in Java, the code is first transformed into bytecode and that bytecode is then executed by the JVM (Java Virtual Machine). |
| C generally breaks down to functions. | Java breaks down to Objects. |
| C programming language can be used for system programming as well as Application programming. | This is not the case in Java. |
| C does not contain the property called Inheritance because it does not support OOPS, which is very useful for code reusability. Thus C is not suited when one has to relate the things according to the real world. | Java contains the property of Inheritance which is very useful in code reuseability. |
| Memory allocation can be done by malloc in C | Memory allocation can be done by a new keyword in Java. |
| C is a low-level language. It has difficult interpretation for the user but it has a closer significance to the machine-level code. | Java is a high-level language because translation of code takes place into machine language using compiler or interpreter. |
| In C89 declaration of variables is at the beginning of the block but in the latest version of C that is C99 we can also declare variables anywhere. | We can declare variables anywhere. |
| free is used for freeing the memory in C. | A compiler will free up the memory internally by calling the garbage collector. |
| C does not supports Threading. | Java supports the concept of threading. |
| C supports pointers. | Java does not supports pointers. |
| It is not portable. | It is portable. |
| Call by value and call by reference is supported in C. | It only supports a call by value. |
| C is platform dependent. | Java is a platform independent. |
| It supports user-based memory management. | It internally manages the memory. |
| C is not robust that is strict type checking does not takes place while compile and run time. | Java is robust. |
| Exception handling cannot be directly achieved in C and thus it lacks the maintenance of normal flow of the program. | Exception Handling is supported in Java. |
| It follows a top-down approach. | Java follows a bottom-up approach. |
| Overloading functionality is not supported by C. | Java supports method overloading which helps in code readability. |
| C supports Preprocessors. | Java does not support Preprocessors. |
| C does not supports OOPS concept. | Java supports OOPS concept. |
| Union and structure datatypes are supported by C. | Java does not supports union and structures. |
| C supports the storage classes. | Whereas Java does not suport the storage classes. |
| It has 32 keywords. | It has 50 keywords. |
| Go-to statements are supported in C language. | Java does not supports go-to statements. |
| Virtual keywords are supported by C. | Virtual keywords are not supported by Java. |
| Overloading functionality is not supported by C. | Java supports method overloading which helps in code readability. |
| Default members of C are public. | Default members of Java are private. |
| Data hiding is done by using static in C. | Data hiding is done by using private in Java. |

1. Compiler vs Interpreter (geeksforgeeks)

Compiler and Interpreter are two different ways to translate a program from programming or scripting language to machine language.

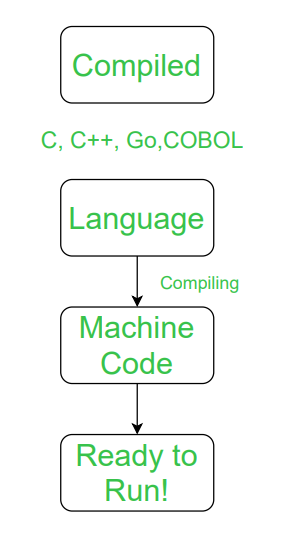
A [**compiler**](http://en.wikipedia.org/wiki/Compiler)takes entire program and converts it into object code which is typically stored in a file. The object code is also refereed as binary code and can be directly executed by the machine after linking. Examples of compiled programming languages are [C](https://www.geeksforgeeks.org/c/) and [C++](https://www.geeksforgeeks.org/c-plus-plus/).

An [**Interpreter**](http://en.wikipedia.org/wiki/Interpreter_%28computing%29)directly executes instructions written in a programming or scripting language without previously converting them to an object code or machine code. Examples of interpreted languages are Perl, Python and Matlab.

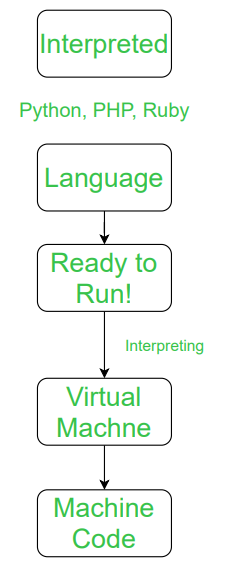
Following are some interesting facts about interpreters and compilers.

1. Both compilers and interpreters convert source code (text files) into tokens, both may generate a parse tree, and both may generate immediate instructions. The basic difference is that a compiler system, including a (built in or separate) linker, generates a stand alone machine code program, while an interpreter system instead performs the actions described by the high level program.
2. Once a program is compiled, its source code is not useful for running the code. For interpreted programs, the source code is needed to run the program every time.
3. In general, interpreted programs run slower than the compiled programs.
4. [Java](https://www.geeksforgeeks.org/java/)programs are first compiled to an intermediate form, then interpreted by the interpreter.
5. Difference between Compiled and Interpreted Language (geeksforgeeks)

**Compiled Language:**  
A compiled language is a programming language which are generally compiled and not interpreted. It is one where the program, once compiled, is expressed in the instructions of the target machine; this machine code is undecipherable by humans. Types of compiled language – C, C++, C#, CLEO, COBOL, etc.



**Interpreted Language:**  
An interpreted language is a programming language which are generally interpreted, without compiling a program into machine instructions. It is one where the instructions are not directly executed by the target machine, but instead read and executed by some other program. Interpreted language ranges – JavaScript, Perl, Python, BASIC, etc.



Let’s see the difference between Compiled and Interpreted Language:

|  |  |  |
| --- | --- | --- |
| S.NO. | COMPILED LANGUAGE | INTERPRETED LANGUAGE |
| 1 | A compiled language is a programming language whose implementations are typically compilers and not interpreters. | An interpreted language is a programming language whose implementations execute instructions directly and freely, without previously compiling a program into machine-language instructions. |
| 2 | In this language, once the program is compiled it is expressed in the instructions of the target machine. | While in this language, the instructions are not directly executed by the target machine. |
| 3 | There are at least two steps to get from source code to execution. | There is only one steps to get from source code to execution. |
| 4 | In this language, compiled programs run faster than interpreted programs. | While in this language, interpreted programs can be modified while the program is running. |
| 5 | In this language, compilation errors prevent the code from compiling. | In this languages, all the debugging occurs at run-time. |
| 6 | The code of compiled language can be executed directly by the computer’s CPU. | A program written in an interpreted language is not compiled, it is interpreted. |
| 7 | This language delivers better performance. | This languages delivers relatively slower performance. |
| 8 | Example of compiled language – C, C++, C#, CLEO, COBOL, etc. | Example of Interpreted language – JavaScript, Perl, Python, BASIC, etc. |

1. JAVA and .NET (tutorialPoint)

Java is a high-level programming language originally developed by Sun Microsystems and released in 1995. Java runs on a variety of platforms, such as Windows, Mac OS, and the various versions of UNIX.

.NET framework is a computer software framework invented by Microsoft. It runs on Microsoft Windows OS (Operating Systems). It provides user interface, data access, database connectivity, cryptography, web application development, etc.

## Languages

* Java supports only Java patterns.
* .NET supports multiple languages such as VB.NET, C#, F#, etc.

## Platforms

* Java is platform independent and it can run on Windows, Linux and Mac OS.
* .NET works on Windows.

## Runtime

* Java has Java Virtual Machine, ByteCode, JDK, JRE, etc.
* .NET has Common Language Runtime

## GUI Components

* Java has Java Beans, whereas .NET has its classes.

## Database

* JDBC is used for connecting Java with database, whereas for database in .NET, use ADO.NET.