Day 1 - assignment.

Difference between compiler and Interpreter

Difference types	compiler	Interpreter
Programming steps	* write a program in source code * compile will analyze youre program statements and check their correctness - If an error is found in a program, it throws an error message * If the program contains no error, then the compiler will convert the source code program into machine code. * The compiler links all the code files into a single runnable program, which is known as the exe file. * Finally it runs the program and generates output.	* write a program in source code * No linking of files happens, or no machine code will generate separately * The source code programming statements are executed line-by-line during the excecution. If an error is found at any specific statement interpreter, it stops further execution until the error gets removed.
Translation type.	A compiler translates complete high-level programming code into machine code at once	An interpreter translates one statement of programming code at a time into machine code.

Difference Types	Compiler	Interpreter
Machine code.	It stores converted machine code from your source code program on the disk.	It never stores the machine code at all on the disk.
Running time	A compiler takes an enormous time to analyze source code. However, overall compiled programming code runs faster as compression to an interpreter.	An interpreter takes less time to analyze source code as compared to compiler. However, overall interpreted programming code runs slower as compression to an compiler.
Program. generation.	The compiler generates an output of a program (in the form of exe-file) that can run separately from the source program	The Interpreter doesn't generate a separate machine code as an output program. so it checks the source code every time during the execution.
Memory requirement	A compiled program is generated into an intermediate object code, and it further required linking. so there is required for memory	An interpreted program does not generate an intermediate code. So there is no requirement of extra memory.