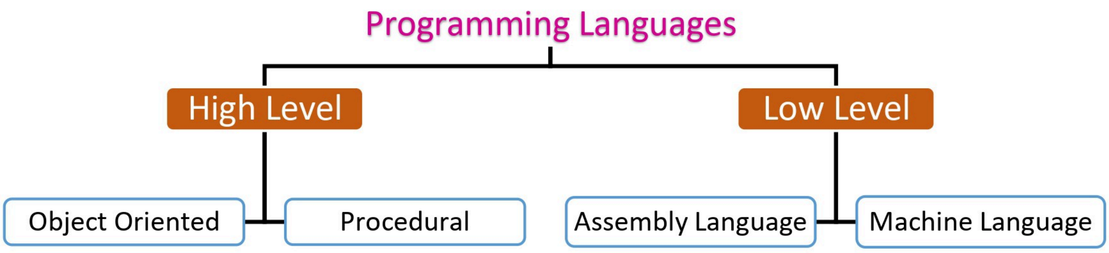
Programming

- The process of creating a set of instructions that tells a computer or a machine how to perform a task.
- The person who creates those set of instructions is known as a "programmer".
- A programmer creates a computer program to solve a particular program.
- The programmer uses a programming language to create a computer program.

Programming Languages

- Language is a mode of communication that we use to share and express our ideas to others.
- When we want to communicate with a person we need a specific language, similarly programmers need a language to communicate with the computer or a machine.
- A programming language is a computer language that is used by programmers to create computer programs.
- It is a set of instructions written in any specific language (computer language) to perform a specific task.

Types of Programming Languages

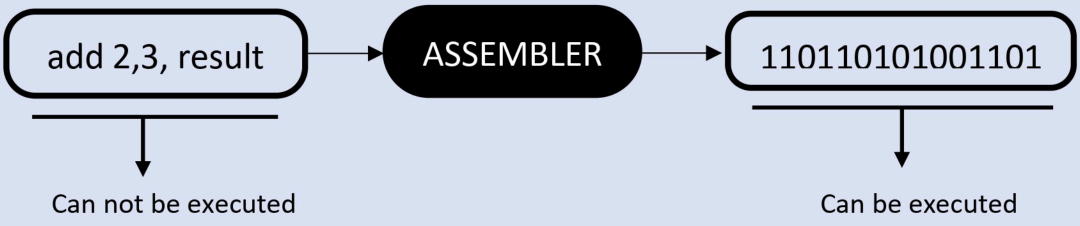


1. High Level Language: A new generation of programming languages, high-level languages, utilize English words, making them easy to learn and use. These languages are machine-independent, meaning that programs written in them can run on different machines without modification. Instructions in high-level languages are called "statements". A program written in a high-level language is referred to as a "source program" or "source code". For example, to add 2 and 3, the statement would be written as "result = 2 + 3;". To execute a high-level language program, "compiler" or "interpreter" is used to translate the source code into machine code.

- **Object Oriented:** Object-Oriented Programming refers to a concept of programming based on "objects" that can contain data and code. The data can exist in the form of fields the code exists in the form of procedures. Example: Java and Python etc.
- **Procedural:** Programming languages considered procedural has a series of instructions written in a sequence from the beginning to the end. Example: BASIC, COBOL, PASCAL, C, PL1.

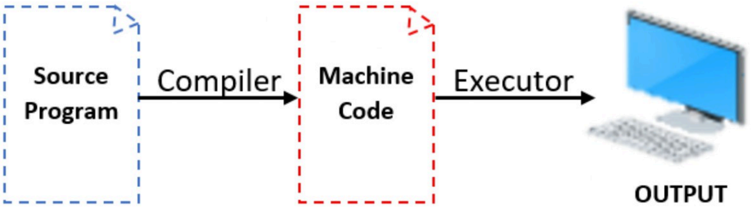
2. Low Level Language: Low-level languages are designed to be easily understood by computers. It includes machine codes and assembly language, both of which instruct computer hardware components to carry out instructions directly. A computer's native language uses zeros and ones (0|1), also known as binary language. For example to add 2 and 3, the result would be represented as 1101101010011010 in binary. However, this language is difficult for humans to use.

- **Assembly:** To make programming easier, assembly language was developed. Assembly language helps in understanding the programming language to machine code. Although still machine-dependent, assembly language introduced keywords such as "add" and "sub" to represent instructions. A program called assembler translates assembly code into machine code, making it easier for programmers to write instructions that can be executed by the computers.
  Example: SUB AX, BX, MOV CX, AX.

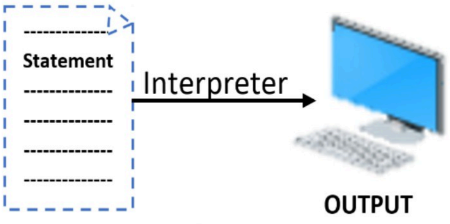


- **Machine Language:** Machine language is the language understood by a computer. All programs and programming language eventually generate or run programs in machine language. Machine language is made up of instructions and date that are all binary numbers.
  For example, the ASCII value for the letter "A" is 01000001 in machine code, but this data is displayed as "A" on the screen.

Compiler: A compiler translates all the source code into machine code at once, generating an executable file.



Interpreter: An interpreter, on the other hand, translates each statement into machine code and executes it immediately.


Statement
Interpreter
OUTPUT