

Python Interpreter



Python source files (.py) are plain text files stored on the computer's hard drive. When executing a Python script, the operating system loads the Python interpreter into RAM. The interpreter then reads the source file from disk into memory as needed, processing it line by line.



source.py

```
import math
x = math.cos(math.pi)
print(x)
# Output equals -1
```



Machine

Machine code instructions for the CPU are executed.

EXE

Python Interpreter

1) Lexical Analysis

The raw source code is read line by line converting it into a tokenized sequence recognizable elements like keywords, identifiers, operators, and punctuation marks.

2) Parsing

The tokenized code is now parsed into an Abstract Syntax Tree (AST), checking for structural errors.

3) Bytecode Compilation

The AST is now converted into lower level byte code. The bytecode defines the instructions for the Python Virtual Machine (PVM).

4) Interpretation (Run Time)

The Python Virtual Machine (PVM) now executes the bytecode, managing memory and performing calculations.

At this stage, additional operations occur, such as Garbage Collection (GC) and module imports.

When the running code encounters an import statement, Python searches for the module, compiles it if necessary, and loads its components. If the module has already been imported, Python retrieves it from memory instead of reloading it.

IN

OUT