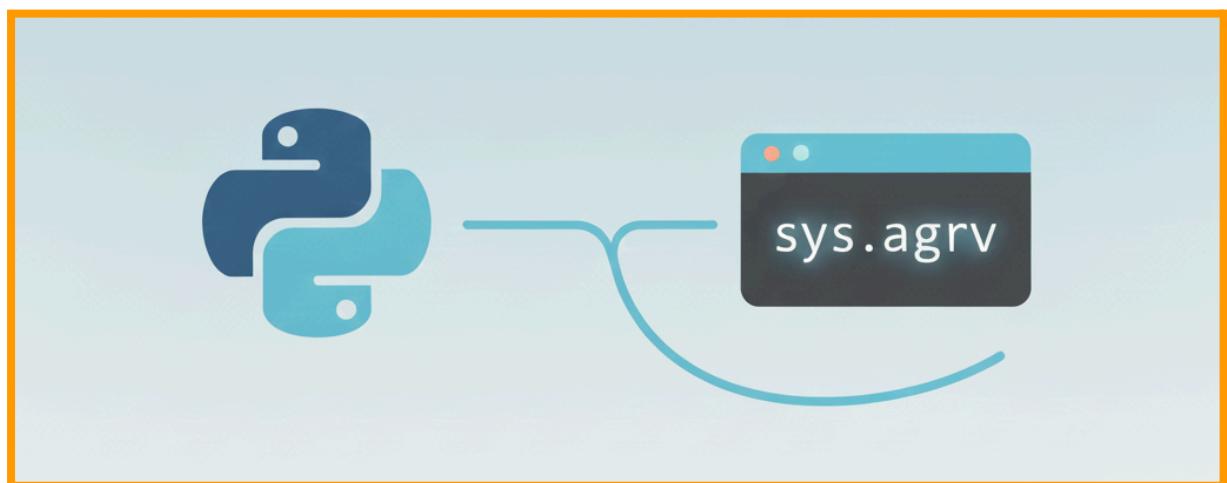


Taking Command Line Arguments in Python



In Python, command line arguments are taken using the **sys module**, which provides access to arguments passed while running a program from the terminal. These arguments are stored in a list called **sys.argv**, where:

- `sys.argv[0]` → contains the script name
- `sys.argv[1]` onward → contain the actual arguments given by the user

To use command line arguments, we first import the module using:

```
import sys
```

Since all arguments are taken as **strings**, we must convert them to the required data type (like `int` or `float`). A `for` loop or list comprehension can be used for conversion.

Example:

```
import sys

def average():
    l = [int(x) for x in sys.argv[1::]] # reading arguments from
index 1
    print(sum(l) / len(l))

average()
```

In this example, the user passes numbers while running the file (e.g., `python program.py 10 20 30`). The program reads the arguments, converts them into integers, and finds their average.