



# Command Line Arguments in Python

[Read](#)[Courses](#)[Practice](#)[Jobs](#)

The arguments that are given after the name of the program in the command line shell of the operating system are known as **Command Line Arguments**. Python provides various ways of dealing with these types of arguments. The three most common are:

- [Using sys.argv](#)
- [Using getopt module](#)
- [Using argparse module](#)

## Using sys.argv

The sys module provides functions and variables used to manipulate different parts of the Python runtime environment. This module provides access to some variables used or maintained by the interpreter and to functions that interact strongly with the interpreter.

One such variable is sys.argv which is a simple list structure. It's main purpose are:

- It is a list of command line arguments.
- len(sys.argv) provides the number of command line arguments.
- sys.argv[0] is the name of the current Python script.

**Example:** Let's suppose there is a Python script for adding two numbers and the numbers are passed as command-line arguments.

### Python3



```
import sys

# total arguments
n = len(sys.argv)
print("Total arguments passed:", n)

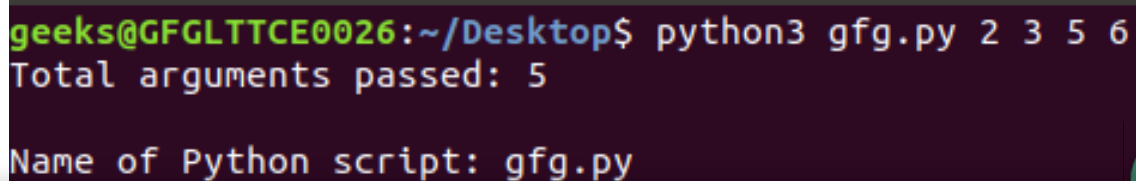
# Arguments passed
print("\nName of Python script:", sys.argv[0])

print("\nArguments passed:", end = " ")
for i in range(1, n):
    print(sys.argv[i], end = " ")

# Addition of numbers
Sum = 0
# Using argparse module
for i in range(1, n):
    Sum += int(sys.argv[i])

print("\n\nResult:", Sum)
```

**Output:**



```
geeks@GFGLTTCE0026:~/Desktop$ python3 gfg.py 2 3 5 6
Total arguments passed: 5

Name of Python script: gfg.py
```

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

**Got It !**

```
geeks@GFGLTTCE0026:~/Desktop$
```

## Using getopt module

Python **getopt module** is similar to the `getopt()` function of C. Unlike `sys` module `getopt` module extends the separation of the input string by parameter validation. It allows both short, and long options including a value assignment. However, this module requires the use of the `sys` module to process input data properly. To use `getopt` module, it is required to remove the first element from the list of command-line arguments.

**Syntax:** `getopt.getopt(args, options, [long_options])`

**Parameters:**

**args:** List of arguments to be passed.

**options:** String of option letters that the script want to recognize.

Options that require an argument should be followed by a colon (:).

**long\_options:** List of string with the name of long options. Options that require arguments should be followed by an equal sign (=).

**Return Type:** Returns value consisting of two elements: the first is a list of (option, value) pairs. The second is the list of program arguments left after the option list was stripped.

**Example:**

### Python3

```
# Python program to demonstrate  
# command line arguments
```



```
# list of command line arguments
argumentList = sys.argv[1:]

# Options
options = "hmo:"

# Long options
long_options = ["Help", "My_file", "Output="]

try:
    # Parsing argument
    arguments, values = getopt.getopt(argumentList, options, long_opti

    # checking each argument
    for currentArgument, currentValue in arguments:

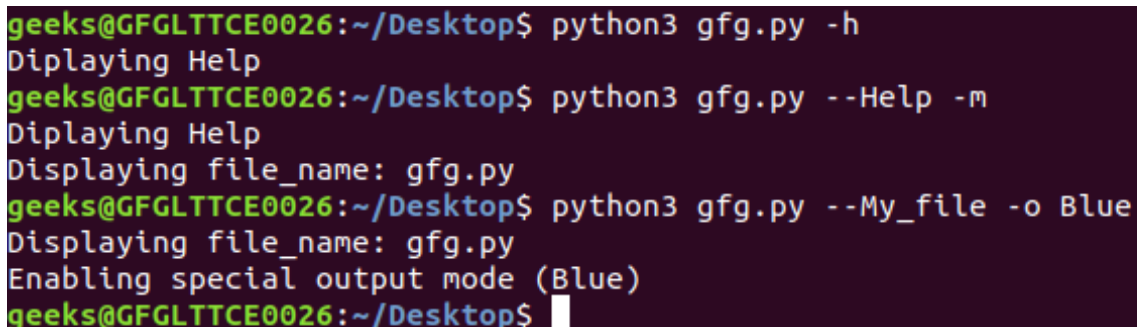
        if currentArgument in ("-h", "--Help"):
            print ("Displaying Help")

        elif currentArgument in ("-m", "--My_file"):
            print ("Displaying file_name:", sys.argv[0])

        elif currentArgument in ("-o", "--Output"):
            print (("Enabling special output mode (% s)") % (currentVa

except getopt.error as err:
    # output error, and return with an error code
    print (str(err))
```

## Output:



```
geeks@GFGLTTCE0026:~/Desktop$ python3 gfg.py -h
Displaying Help
geeks@GFGLTTCE0026:~/Desktop$ python3 gfg.py --Help -m
Displaying Help
Displaying file_name: gfg.py
geeks@GFGLTTCE0026:~/Desktop$ python3 gfg.py --My_file -o Blue
Displaying file_name: gfg.py
Enabling special output mode (Blue)
geeks@GFGLTTCE0026:~/Desktop$
```

## Using argparse module

Using argparse module is a better option than the above two options as it provides a lot of options such as positional arguments, default value for arguments, help message, specifying data type of argument etc.

**Note:** As a default optional argument, it includes -h, along with its long version -help.

**Example 1:** Basic use of argparse module.

---

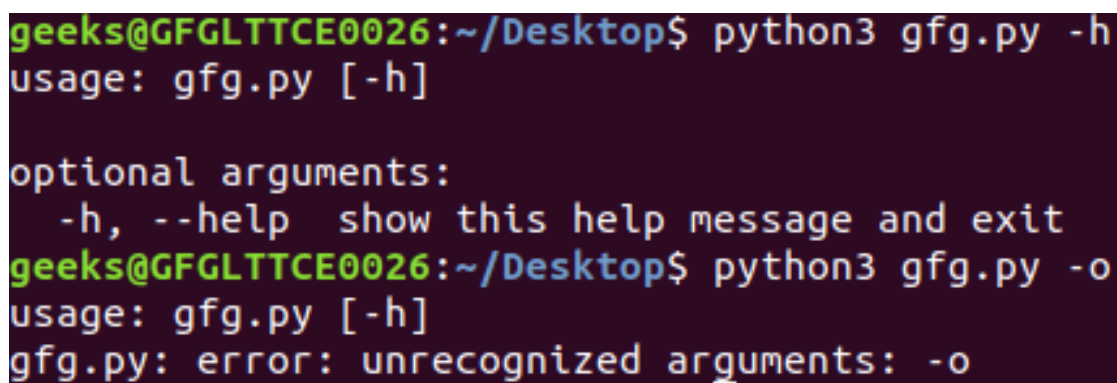
### Python3

```
# Python program to demonstrate
# command line arguments

import argparse

# Initialize parser
parser = argparse.ArgumentParser()
parser.parse_args()
```

**Output:**



```
geeks@GFGLTTCE0026:~/Desktop$ python3 gfg.py -h
usage: gfg.py [-h]

optional arguments:
  -h, --help  show this help message and exit
geeks@GFGLTTCE0026:~/Desktop$ python3 gfg.py -o
usage: gfg.py [-h]
gfg.py: error: unrecognized arguments: -o
```

**Example 2:** Adding description to the help message.

---

## Python3

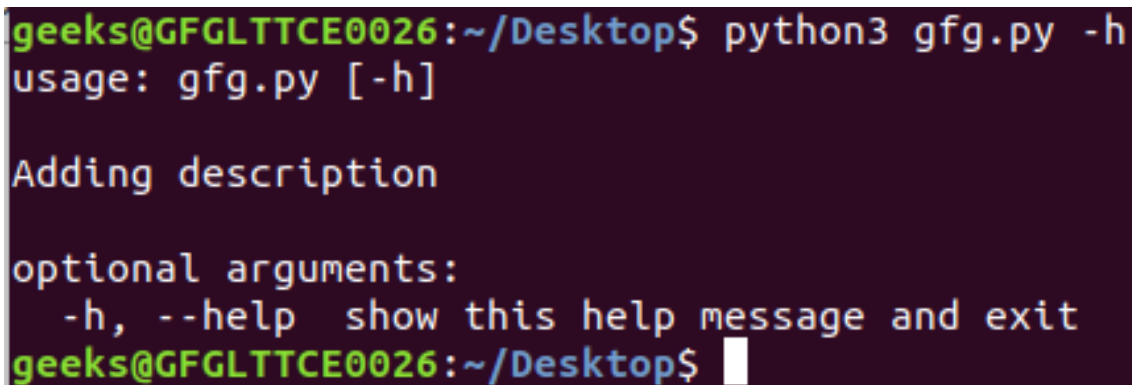
```
# Python program to demonstrate
# command line arguments

import argparse

msg = "Adding description"

# Initialize parser
parser = argparse.ArgumentParser(description = msg)
parser.parse_args()
```

**Output:**



```
geeks@GFGLTTCE0026:~/Desktop$ python3 gfg.py -h
usage: gfg.py [-h]

Adding description

optional arguments:
  -h, --help  show this help message and exit
geeks@GFGLTTCE0026:~/Desktop$
```

**Example 3:** Defining optional value

---

## Python3



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

**Got It !**

```
import argparse

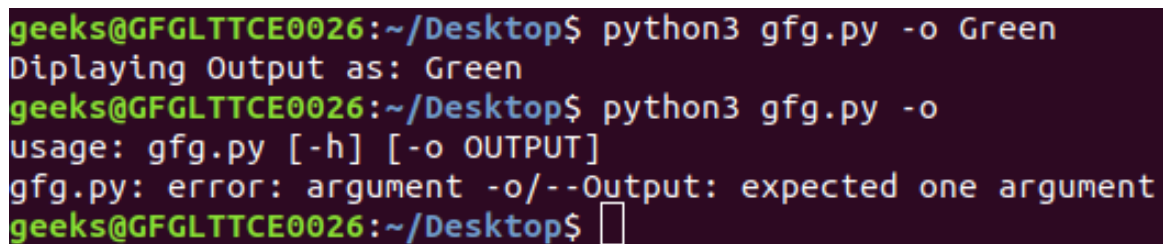
# Initialize parser
parser = argparse.ArgumentParser()

# Adding optional argument
parser.add_argument("-o", "--Output", help = "Show Output")

# Read arguments from command line
args = parser.parse_args()

if args.Output:
    print("Displaying Output as: % s" % args.Output)
```

### Output:



```
geeks@GFGLTTCE0026:~/Desktop$ python3 gfg.py -o Green
Displaying Output as: Green
geeks@GFGLTTCE0026:~/Desktop$ python3 gfg.py -o
usage: gfg.py [-h] [-o OUTPUT]
gfg.py: error: argument -o/--Output: expected one argument
geeks@GFGLTTCE0026:~/Desktop$
```

Don't miss your chance to ride the wave of the data revolution! Every industry is scaling new heights by tapping into the power of data. Sharpen your skills and become a part of the hottest trend in the 21st century.

Dive into the future of technology - explore the [Complete Machine Learning and Data Science Program](#) by GeeksforGeeks and stay ahead of the curve.



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

Three 90 Challenge ending on 31st Jan! Last chance to get 90% refund by completing 90% course in 90 days. [Explore](#) offer now.

Last Updated : 12 Apr, 2022

29

[Previous](#)

[Next](#)

**How to find Gradient of a Function using Python?**

**Access Modifiers in Python : Public, Private and Protected**

[Share your thoughts in the comments](#)

[Add Your Comment](#)

## Similar Reads

Python | Set 6 (Command Line and Variable Arguments)

How to Pass Arguments to Tkinter Button Command?

Pass list as command line argument in Python

Click Module in Python | Making awesome Command Line Utilities

Command Line Scripts | Python Packaging

Python | How to Parse Command-Line Options

Argparse VS Docopt VS Click - Comparing Python Command-Line Parsing Libraries

Command Line File Downloader in Python

Command-Line Option and Argument Parsing using argparse in Python

Run function from the command line In Python

## Complete Tutorials

Python Crash Course

Python API Tutorial: Getting Started with APIs



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

**Got It !**



OpenAI Python API - Complete Guide

N [nikhilagg...](#)

Article Tags : [python-utility](#) , [Python](#)

Practice Tags : [python](#)

### Additional Information



A-143, 9th Floor, Sovereign Corporate  
Tower, Sector-136, Noida, Uttar Pradesh -  
201305



### Company

[About Us](#)

[Legal](#)

[Careers](#)

### Explore

[Job-A-Thon Hiring Challenge](#)

[Hack-A-Thon](#)

[GfG Weekly Contest](#)



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

Got It !

[Advertise with us](#)[GFG Corporate Solution](#)[Placement Training Program](#)[Apply for Mentor](#)

## Languages

[Python](#)[Java](#)[C++](#)[PHP](#)[GoLang](#)[SQL](#)[R Language](#)[Android Tutorial](#)[Tutorials Archive](#)

## Data Science & ML

[Data Science With Python](#)[Data Science For Beginner](#)[Machine Learning Tutorial](#)[ML Maths](#)[Data Visualisation Tutorial](#)[Pandas Tutorial](#)[NumPy Tutorial](#)[NLP Tutorial](#)[Deep Learning Tutorial](#)

## Python

[Python Programming Examples](#)[Django Tutorial](#)[Python Projects](#)[Python Tkinter](#)[Web Scraping](#)[OpenCV Python Tutorial](#)[Python Interview Questions](#)[Master System Design](#)[Master CP](#)[GeeksforGeeks Videos](#)[Geeks Community](#)

## DSA

[Data Structures](#)[Algorithms](#)[DSA for Beginners](#)[Basic DSA Problems](#)[DSA Roadmap](#)[Top 100 DSA Interview Problems](#)[DSA Roadmap by Sandeep Jain](#)[All Cheat Sheets](#)

## HTML & CSS

[HTML](#)[CSS](#)[Web Templates](#)[CSS Frameworks](#)[Bootstrap](#)[Tailwind CSS](#)[SASS](#)[LESS](#)[Web Design](#)

## Computer Science

[GATE CS Notes](#)[Operating Systems](#)[Computer Network](#)[Database Management System](#)[Software Engineering](#)[Digital Logic Design](#)[Engineering Maths](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

**Got It !**

[Git](#)[AWS](#)[Docker](#)[Kubernetes](#)[Azure](#)[GCP](#)[DevOps Roadmap](#)

## System Design

[High Level Design](#)[Low Level Design](#)[UML Diagrams](#)[Interview Guide](#)[Design Patterns](#)[OOAD](#)[System Design Bootcamp](#)[Interview Questions](#)

## NCERT Solutions

[Class 12](#)[Class 11](#)[Class 10](#)[Class 9](#)[Class 8](#)[Complete Study Material](#)

## Commerce

[Accountancy](#)[Business Studies](#)[Indian Economics](#)[Macroeconomics](#)[Microeconomics](#)[Statistics for Economics](#)

## UPSC Study Material

[Top DS or Algo for CP](#)[Top 50 Tree](#)[Top 50 Graph](#)[Top 50 Array](#)[Top 50 String](#)[Top 50 DP](#)[Top 15 Websites for CP](#)

## JavaScript

[JavaScript Examples](#)[TypeScript](#)[ReactJS](#)[NextJS](#)[AngularJS](#)[NodeJS](#)[Lodash](#)[Web Browser](#)

## School Subjects

[Mathematics](#)[Physics](#)[Chemistry](#)[Biology](#)[Social Science](#)[English Grammar](#)

## Management & Finance

[Management](#)[HR Management](#)[Income Tax](#)[Finance](#)[Economics](#)

Science and Technology Notes

Economy Notes

Ethics Notes

Previous Year Papers

## Colleges

Indian Colleges Admission & Campus Experiences

List of Central Universities - In India

Colleges in Delhi University

IIT Colleges

NIT Colleges

IIIT Colleges

## Preparation Corner

Company Wise Preparation

Preparation for SDE

Experienced Interviews

Internship Interviews

Competitive Programming

Aptitude Preparation

Puzzles

## More Tutorials

Software Development

Software Testing

Product Management

SAP

SEO - Search Engine Optimization

Linux

Excel

IBPS PO Syllabus

IBPS Clerk Syllabus

SSC CGL Practice Papers

## Companies

META Owned Companies

Alphabet Owned Companies

TATA Group Owned Companies

Reliance Owned Companies

## Exams

JEE Mains

JEE Advanced

GATE CS

NEET

UGC NET

## Write & Earn

Write an Article

Improve an Article

Pick Topics to Write

Share your Experiences

Internships

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

Got It !