

# Passing\_arguments

December 6, 2018

Written by Nadège Guiglielmoni.

Here are examples of how you can pass arguments to a python script. You can go through these script to understand how they work.

## 1 With SYS

```
In [2]: python passing_args.py a b c
```

```
The number of parameters is : 4
```

```
The first parameter is : a
```

```
The third parameter is : c
```

The script calls for a third argument, so if you do not pass at least 3 arguments, here is what happens :

```
In [3]: python passing_args.py banana
```

```
The number of parameters is : 2
```

```
The first parameter is : banana
```

```
Traceback (most recent call last):
```

```
  File "passing_args.py", line 7, in <module>
```

```
    print("The third parameter is : {0}".format(sys.argv[3]))
```

```
IndexError: list index out of range
```

But you can pass as many arguments as you want :

```
In [5]: python passing_args.py banana dog "bla bla bla" 1 beer toronto
```

```
The number of parameters is : 7
```

```
The first parameter is : banana
```

```
The third parameter is : bla bla bla
```

You may notice that the number of parameters displayed is 7, while there are only 6 arguments. This is because the name of the script counts as argument number 0.

## 2 With ARGPARSE

```
In [6]: python passing_args_2.py -h
```

```
usage: passing_args_2.py [-h] -e ESR -c COMPLIMENT -g GIFT
```

```
optional arguments:
```

```
-h, --help            show this help message and exit
-e ESR, --esr ESR      Name of one of your fellow ESR.
-c COMPLIMENT, --compliment COMPLIMENT
                        Compliment that you want to make to this ESR
-g GIFT, --gift GIFT   Compliment that you want to make to this ESR
```

Well now this looks fancy, there is help ! When you use argparse, you can pass -h to the script and it will display the parameters you can pass to the function.

```
In [7]: python passing_args_2.py -e Ramon -c wonderful -g beer
```

```
Ramon is wonderful. When I visit this ESR, I will bring a beer.
```

```
In [8]: python passing_args_2.py --esr Ramon --compliment "so smart" --gift "beer and waffles"
```

```
Ramon is so smart. When I visit this ESR, I will bring a beer and waffles.
```

The parameters do not need to be passed in a specific order :

```
In [9]: python passing_args_2.py --esr Ksenia --gift "box of chocolates" --compliment adorable
```

```
Ksenia is adorable. When I visit this ESR, I will bring a box of chocolates.
```