What are the variable number of arguments in Python?

*argv   **kwargs   communitycreator

An argument is a variable, value, or object passed to a function or method as input. In Python, there are two ways to define a function that can take a variable number of arguments. Different forms of this type are:

1. Positional arguments
2. Keyword arguments

Positional arguments

Positional arguments are arguments that need to be included in the proper position or order. The first positional argument always needs to be listed first when the function is called. The second positional argument needs to be listed second, the third positional argument listed third, etc. **General syntax** for positional argument is: `function(*iterable)`

```
1  def positional_args(*argv):
2      for arg in argv:
3          print (arg)
4
5  positional_args ('Hello', 'Welcome', 'to', 'GITAM')
6
```

The arguments passed by the calling function are received by `*argv`. The passed arguments will be unpacked one by one within the function.

```
1  def positional_args (arg1, *argv):
2      print ("First argument :", arg1)
3      for arg in argv:
4          print("Next argument through *argv :", arg)
5
6  positional_args ('Hello', 'Welcome', 'to', 'GITAM')
7
```

Keyword arguments

A keyword argument is an argument passed to a function or method that is preceded by a keyword and an equal sign. The general form is:

`function(keyword = value)`

Where function is the function name, the keyword is the keyword argument, and value is the value or object passed as that keyword.

In the code below, `**kwargs` is used to access the keyword arguments.

```
1  def key_args(**kwargs):
2      for key, value in kwargs.items():
3          print ("%s == %s" %(key, value))
4
5  key_args (first ='GITAM', mid ='for', last='GITAM')
6
```

```
1  def key_args (arg1, **kwargs):
2      for key, value in kwargs.items():
3          print ("%s == %s" %(key, value))
4
5  key_args ("Hi", first ='GITAM', mid ='for', last='GITAM')
6
```

*args vs. **kwargs

```
1  def var_arg(*args,**kwargs):
2      print("args: ", args)
3      print("kwargs: ", kwargs)
4
5  var_arg ('GITAM','for','GITAM',first="GITAM",mid="for",last="GITAM")
6
```