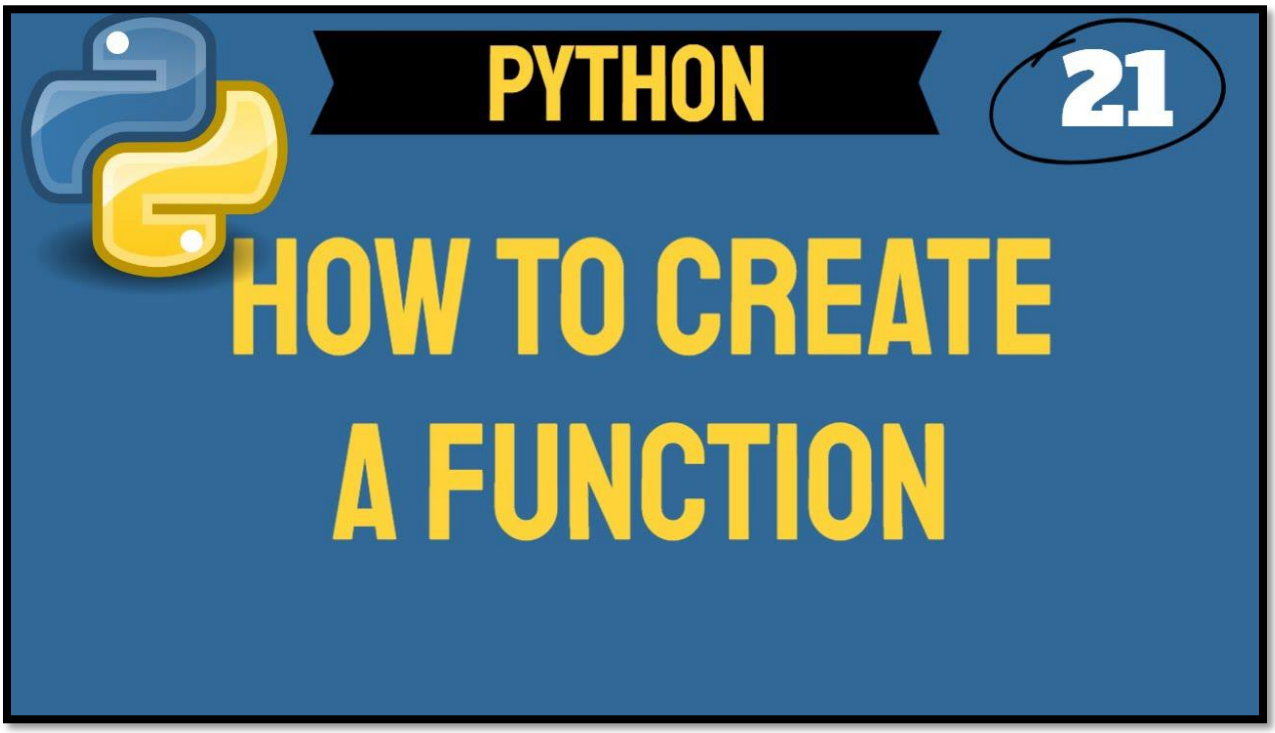
Rex Jones II

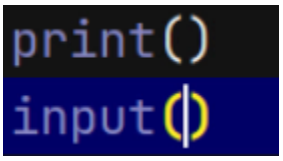
How To Create A Function



Python Video = https://youtu.be/0yNmNvHFOv4

In this session, I will talk about functions. A Python functions have a group of statements designed to perform a job. It's a good way to organize our code into reusable chunks. The reusable chunks of code benefit us if we need to update our code and if we need to reuse the code over and over.

We have already seen some functions such as print() and input(). These 2 functions are built into Python.



However we can also write our own functions. Whenever Python sees def, it knows we are defining a function. def is short for define and it is followed by a name. The name is our function. Do you recall the naming convention for a variable? Naming a function has the same convention as a variable. We should always have a descriptive name with lowercase letters. If there is more than 1 word then we separate each word with an underscore. This function name will be greet_the_user().Cannot forget the : colon which lets Python know we are getting ready to define a block of code. Therefore, any code lines that

comes after this 1st line will be part of our function. The 1st line is the function definition. Automatically, the next line is indented after clicking the Enter key.

The block of code will be print('\t Hello, how are you?') and print(''\t This Is A Python Function Tutorial'').

```
def greet_the_user():
    print('\t Hello, how are you?')
    print('\t This Is A Python Function Tutorial')
```

These 2 lines represent the body. We know it's the body because they are indented. The next line will be print('Start At The Top'). This line is not part of the function because it is not indented. Do you see the line under our function name greet_the_user? Python is showing this line because it wants us to add a Docstring comment. It's not required but if I hover, we see Missing docstring and insert docstring. Select Insert Docstring then write This Is A Python Function Example. A docstring explains the purpose of a function.

```
def greet_the_user():
    """
    This Is A Python Function Example
    """
    print('\t Hello, how are you?')
    print('\t This Is A Python Function Tutorial')

print('Start At The Top!!!')
```

Now, let's run. The only print statement we see in the console is Start At The Top.

```
Start At The Top!!!
```

That's because execution in Python always skip over a defined function. The defined function only describes our code.

If we want to execute a function. We must invoke the function also known as calling the function by writing the function name greet_the_user(). If we think about it, greet_the_user is also the same as we call print() and input(). It has a name and parenthesis. However, we have to be careful where we place

the function. It's very important. We are not allowed to place the function call before the name of the function. If we place it before the function, we get an error and the error says "Unresolved Reference". So the best way to place the call after the function name. Now, let's add one more print() statement "Stop At The Bottom".

```
print('Start At The Top!!!')
greet_the_user()
print('Stop At The Bottom!!!')
```

Let's run. We see Start At The Top!!!, Hello, how are you? This Is A Python Function Tutorial, Stop At The Bottom!!!

```
Start At The Top!!!
	Hello, how are you?
	This Is A Python Function Tutorial
Stop At The Bottom!!!
```

Here's the step-by-step process flow.

```
2   def greet_the_user():
3       """
4       This Is A Python Function Example
5       """
6       print('\t Hello, how are you?')
7       print('\t This Is A Python Function Tutorial')
8
9   print('Start At The Top!!!')
10  greet_the_user()
11  print('Stop At The Bottom!!!')
```

Like the last execution, code lines 6 and 7 were not executed within the function. Our program started at line 9 and stopped at line 11. Line 9 says Start At The Top then Python executes Line 10 which is the function call greet_the_user(). The program jumps up to the first line in the function then begin

executing the block of code for both print statements. When the program finish executing the function then execution return to Line 10 that called the function. Finally, the program moves to Line 11 Stop At The Bottom. The print statement within the function were only executed when we called the function at line 10.

Contact Info

- ✔ Email Rex.Jones@Test4Success.org
- ✔ YouTube https://www.youtube.com/c/RexJonesII/videos
- ✔ Facebook https://facebook.com/JonesRexII
- ✔ Twitter https://twitter.com/RexJonesII
- ✔ GitHub https://github.com/RexJonesII/Free-Videos
- ✔ LinkedIn https://www.linkedin.com/in/rexjones34/