



Python Programming

Lesson 5

Function and Returning

Presented by Advaspire Team



Function / Myblock

Function, in a simple term, is a collection of code that have a specific purpose or action. Kind of similar to how dictionary works, it can store lines of code inside of it and when you need to use it, just use the key.

For those that learned scratch, this may be similar to myblock.



Using a function can save you a lot of time because you don't need to write down the same code again if you need to use a particular code. Once you have defined a function, it can be used multiple time in your program.



Create a function

To create a function, just type (def) then a descriptive name for it. Then put the bracket and the colon at then. When you press enter to insert a code, it should start at the name of the function. This is called indent. Consider it as inside the function.

Now that you are inside the function, write down the code that you wish to execute.

```
def sayhey():  
    print_("Hey")
```



Execute the Function

To execute a function, we call it as Call. But you actually just type down the function to execute it. Just make sure to put the bracket at end.

```
def sayhey():  
    print("Hey")  
  
sayhey()
```

Maybe you are wondering, can we insert information in the bracket?

Yes, we can. But first you need to put a placeholder in the def line. For example, I use name as the PH, then when I want to call it, I insert a name in the bracket.

```
def sayhey(name):  
    print("Hey " + name)  
  
sayhey('Haziq')
```



Multiple Info in Parameter

Like other storage type, you can use multiple info in it. What you need to do is just use the comma.

```
def sayhey(name, age):  
    print("Hey " + name + ' you are ' + age + ' years old')  
  
sayhey('Haziq', '22')
```

C:\Users\haziq\PycharmProjects\
Hey Haziq you are 22 years old

Make sure to insert all the necessary value in the calling before you run it. Or it will result in error.

```
sayhey('Haziq')  
TypeError: sayhey() missing 1 required positional argument: 'age'
```



Return Statement

Sometime when using a python function, as in the one you just learned, you may want it to return an answer or statement. Depending on the code, sometime it wont give you any value in the terminal section.

```
C:\Users\haziq\PycharmProjects\pytho  
Process finished with exit code 0
```



For example here, we want to create a function that will return the value of a cube or square. Normally by using a calculator you just typed it like so: 2^2 or 2^3 . Or in python you can type like this $2^{**}3$. But to make it even simple, we are creating a function for it.

```
def cube(num):  
    num*num*num
```



```
cube(2)
```

```
def power(num):  
    num*num*num
```

```
print(power(3))
```



As you can see, when you call the function, nothing happens. So this is where the return statement will come in handy. In a way, when defining, python can't seem to process code that involves math operators like + or -.

To use the return statement, just type return before the code like in the example below. So the return will give a signal to python, telling them that the user wants a feedback or answer for the code. Then just print the call with the parameter value inside. It will then give you the answer in terminal

```
def power(num):  
    return num*num*num  
  
print(power(3))
```




Store / Input Using Return

You can store or use input function with the define and return. The variable is easy to do but if you want to use input and concatenation, it will require some thinking first. Especially if you are using string and integer in a line.

```
def cube(num1):  
    return num1*num1*num1  
result = cube(3)  
print(result)
```

```
def power(num):  
    return num*num*num  
  
num = int(input("Please enter a number to cube : "))  
  
result = str(power(num))  
print("The cube for " + str(num) + " is " + result)
```



You can direct message your teacher and ask your question through [Slack Robotene Community](#) or arrange a [One-to-One Consultation](#) with your teacher.



Any Questions?



Thank you :)