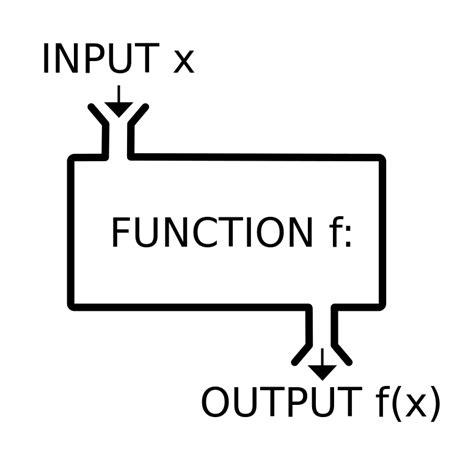
Python Functions Examples: Call, Indentation, Arguments & Return Values

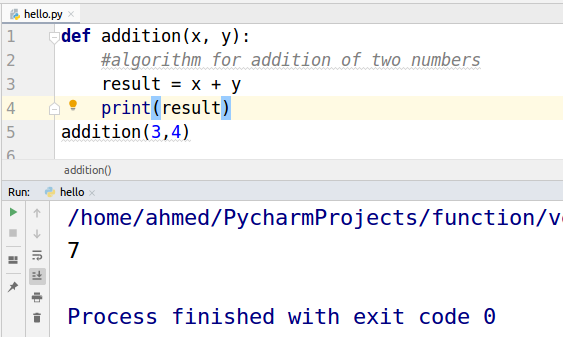
## What is a function?

Consider a function a box just as shown below with two sides open one for input and other for output. This box takes input then does some operations on the given input and provides you with output. If this box is a function of addition then when you give it two numbers 3 and 4 as input it will output 7. If this box is a function of multiplication then with the same input 3 and 4 it will output 12.



Let’s make an ‘addition’ function.

## Addition Function Example

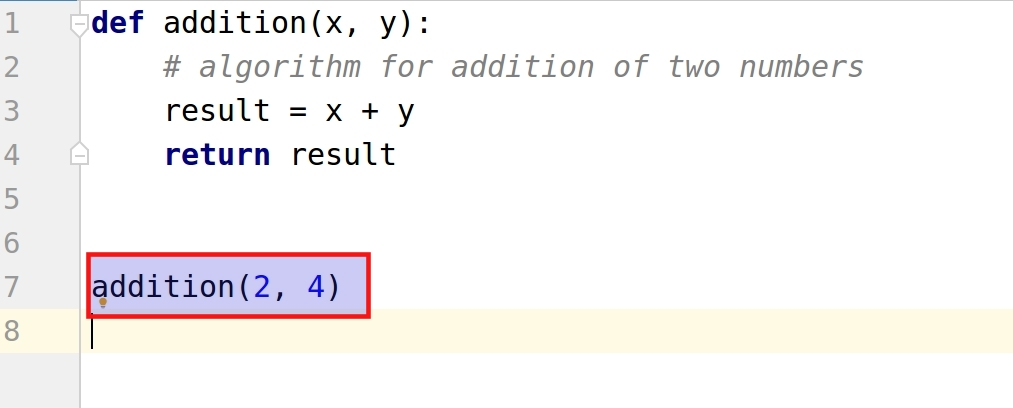


**def** is a keyword that is used to define a function. “Addition” is the name of the function.

x and y are the input values. The **result** is a variable. When we passed 3 and 4 as input the function performed the addition operation on the inputs and stored the answer in the result variable. Which is then printed by the print function.

## Calling Function

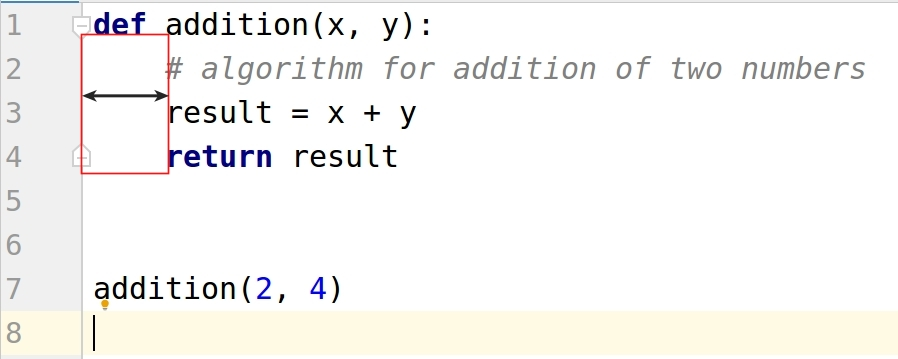
The red rectangle in the screenshot below is an example of function calls. While calling any function the name of the function is used and we give parameters in the round brackets as shown in the figure below.



## Indentation

In other languages like c++ semicolon is used to define that this line of code has ended. But Python has its own way of defining syntax. Traditionally, Indentation of one tab or 4 spaces is used by the Python community. You can use any number of spaces but you have to use the same number of spaces in your entire code.

Indentation in Python makes it a highly readable language. As shown below that indentation block is necessary to define a function. This indentation says that “I am a part of this function”. Below the call “addition(2, 4)” is not indented as it is not part of the function. The function ends at return.

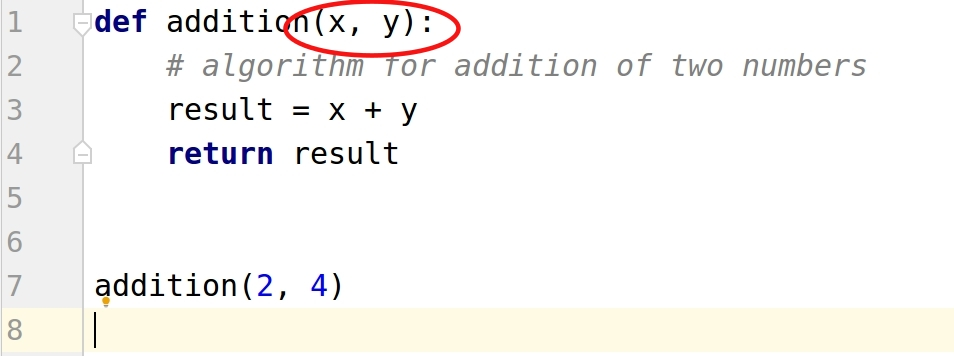


If we don’t indent the code then we will face an “IndentationError”.

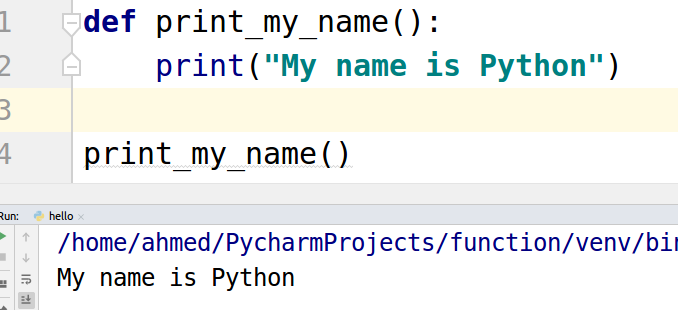
## 

## Function Arguments

In the above example, the x and y are the arguments of the function. Arguments are just what will be the input of the function.



A function can be without arguments and can also have as many arguments as required. Let’s take a look at a function without arguments.

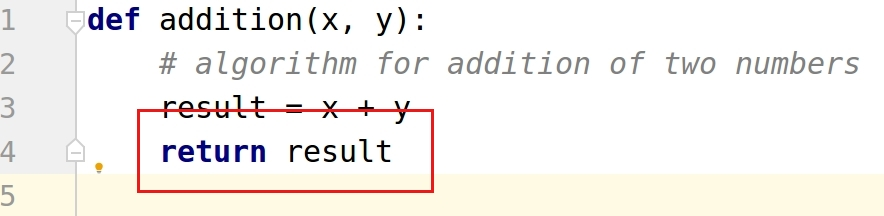


As it can be seen that the above function does not contain “return”. Such functions return “None”.

## 

## Return Value

A “return” keyword is used to return a value from the function or return the output. As shown in the example below the “result” variable will be returned whenever an “addition” function is called.



Let’s combine all the concepts in one example.

