

Python

Function

Class IX

Lab 15

**Lab Objectives:**

* Creating function, return values
* Lambda
* Uses of lambda function

Function

A function is a block of code which only runs when it is called.

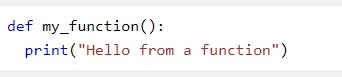
You can pass data, known as parameters, into a function.

A function can return data as a result.

Creating a Function

In Python a function is defined using the def keyword:

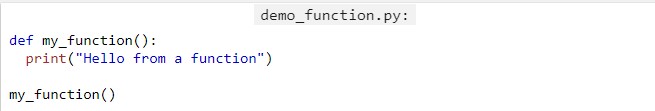
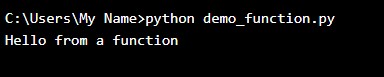
Example



Calling a Function

To call a function, use the function name followed by parenthesis:

Example



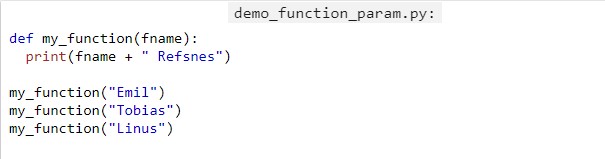
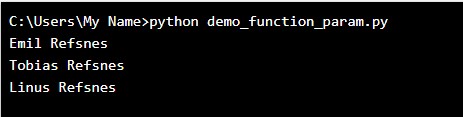
Parameters

Information can be passed to functions as parameter.

Parameters are specified after the function name, inside the parentheses. You can add as many parameters as you want, just separate them with a comma.

The following example has a function with one parameter (fname). When the function is called, we pass along a first name, which is used inside the function to print the full name:

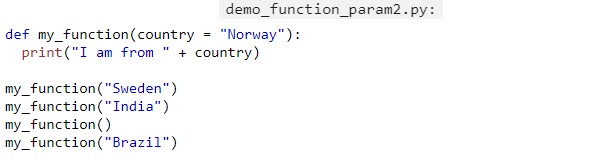
Example:

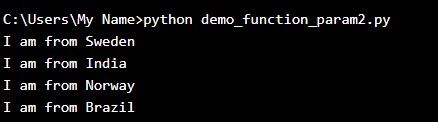


Default Parameter Value

The following example shows how to use a default parameter value.

If we call the function without parameter, it uses the default value:

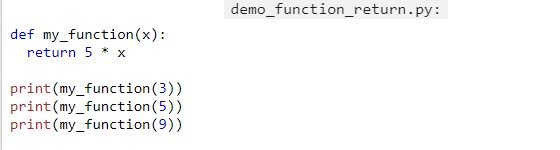
Example

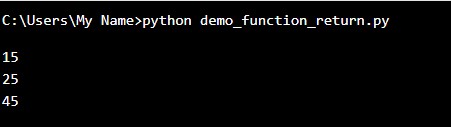


Return Values

To let a function, return a value, use the return statement:

Example





Python Lambda

A lambda function is a small anonymous function.

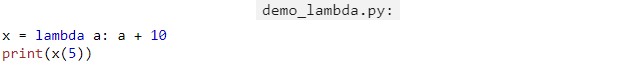
A lambda function can take any number of arguments, but can only have one expression.

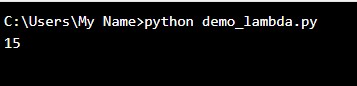
Syntax

lambda arguments: expression

### **Example**

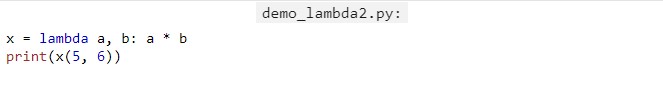
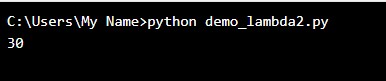
A lambda function that adds 10 to the number passed in as an argument, and print the result:

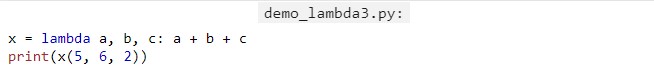


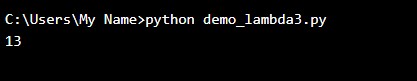


Lambda functions can take any number of arguments:

Example

A lambda function that multiplies argument a with argument b and print the result:

A lambda function that sums argument a, b, and c and print the result:



Why Use Lambda Functions?

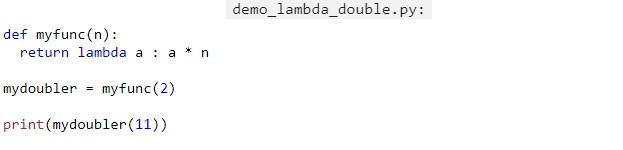
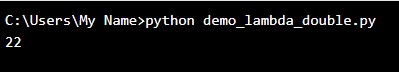
The power of lambda is better shown when you use them as an anonymous function inside another function.

Say you have a function definition that takes one argument, and that argument will be multiplied with an unknown number:



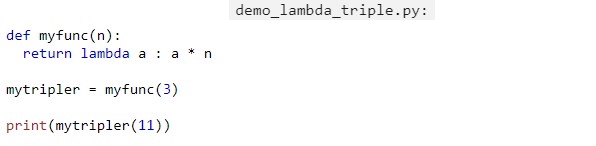
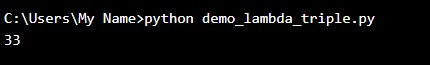
Use that function definition to make a function that always doubles the number you send in:

Example



Or, use the same function definition to make a function that always triples the number you send in:

Example



Or, use the same function definition to make both functions, in the same program:

Example

