

Let's learn more about Calling Functions.

Built-in Functions							
abs()	isinstance()	property()	unichr()	getattr()	memoryview()	hex()	
divmod()	pow()	tuple()	callable()	map()	round()	object()	
input()	sum()	bool()	format()	repr()	complex()	slice()	
open()	<pre>basestring()</pre>	filter()	locals()	xrange()	hash()	dir()	
all()	execfile()	len()	reduce()	cmp()	min()	id()	
enumerate()	issubclass()	range()	unicode()	globals()	set()	oct()	
int()	print()	type()	chr()	max()	delattr()	sorted()	
ord()	super()	bytearray()	frozenset()	reversed()	help()		
str()	bin()	float()	long()	zip()	next()		
any()	file()	list()	reload()	compile()	setattr()		
eval()	iter()	raw_input()	vars()	hasattr()	dict()		

Python has many, many built-in functions.

You could never learn them all in this course.

You will learn a few, but it's far more important that you learn how to learn more about functions.



Good programmers create documentation for their functions.

Documentation is extra information about a function that makes it clear how it works. There are many ways to write documentation, but they usually have a few things in common.

Function Documentation						
Name: len Description: The function "len" returns the number of characters in a string (its length). Syntax:						
Parameters: a string (String): The string to calculate the length of						
Returns: Integer Example:						
len("Hello World")						

You should look for five major parts.

The function's name, its parameters, its return value, its description, and examples how to use the function.



The Arguments that are passed into a function are also known as Parameters.

Parameters are the formal names that describe those arguments.

In the example here, the arguments are 3 and 10.

The documentation tells us that the corresponding parameters are length and width.



A common task is to nest function calls inside of other function calls.

This means that return value of the inner function becomes the argument of the outer function.



When you need to call methods on the same string variable or value, it's a little different. You will repeat the period, name, and parentheses each time.

In this example, we invoke the title() method on the string "hello world", and then the count() method.

By placing them side-by-side, you can "chain" the results from left to right.



Since calls are just another expression, you can freely combine them with each other. Observe the code show, where we combine string addition, a string method call, and a function.

The result of the entire expression is stored inside this new "string_length" variable. Notice the order of execution, this can be tricky to follow!



You already know two functions: Print is a function that takes in arguments and writes to the console.

Input is a function reads from the console and returns it as a string value.

These two functions are strange, because they have side effects.

The Input function puts a textbox on the console, and the print function writes text to the console.