MASTERY 07

CALLING FUNCTIONS

		Built-in Functions		
abs()	delattr()	hash()	memoryview()	set()
all()	dict()	help()	min()	setattr()
any()	dir()	hex()	next()	slice()
ascii()	divmod()	id()	object()	sorted()
bin()	enumerate()	input()	oct()	staticmethod()
bool()	eval()	int()	open()	str()
breakpoint()	exec()	isinstance()	ord()	sum()
bytearray()	filter()	issubclass()	pow()	super()
bytes()	float()	iter()	print()	tuple()
callable()	format()	len()	property()	type()
chr()	frozenset()	list()	range()	vars()
classmethod()	getattr()	locals()	repr()	zip()
compile()	globals()	map()	reversed()	__import__()
complex()	hasattr()	max()	round()	

- In this mastery we will learn how to use already existing functions in Python, and how to import others from Python modules.
- As we may know, there are some built-in-functions in Python, and to use them, we just need to type them and introduce text, numbers, or any other element inside the parenthesis

mastery 05 (finished).py | Mastery 06.py | mastery 07.py | Mastery 07.1.py

```
1  import math
2  print(math.pi)
3  print(math.sqrt(2))
4
5  from math import pi
6  print(pi)
7  from math import sqrt
8  print (sqrt(2))
9
10
11
```

Shell

```
>>>
>>>
>>>
>>>
>>> %Run 'Mastery 07.1.py'

  3.141592653589793
  1.4142135623730951
  3.141592653589793
  1.4142135623730951
```

- In the other hand, the are some functions that we need to import from python modules in order to use them.
- As an example, we can import mathematical expressions or operations.
- There are two ways to do this, the first one is just by typing “import math” and then, calling the expression that we want to use (see example)
- The other way, is just importing an specific expression by typing “from math import pi”, and then just print the value.