

# Guide 5 functions

April 21, 2019

Guía 4: Ciclos

Ingeniería en Estadística, Universidad de Valparaíso

Profesor: Eduardo Jorquera - eduardo.jorquera@postgrado.uv.cl

## 1 Nociones básicas

Si tienes que realizar una tarea de código mayor en tamaño y complejidad, es posible que dividiendo esta gran tarea en otras más pequeñas y más simples el código sea más sencillo de escribir. Las unidades de los programas, que pueden ser llamados como base, toman un “input” que luego es procesado, y luego puede generar un “output” el cual puede ser llamado como **función**.

Esto es útil al tratarse por ejemplo de programación modular, la cual consiste en dividir el programa en pequeñas partes. Esto evita que la persona que crea el programa reescriba reiteradas veces el mismo código una y otra vez, lo cual puede reducir el largo del programa.

El nombre de una función puede ser cualquier cadena de caracteres alfanumérica, así mismo como cuando se trata de ponerle nombre a una variable u objeto.

Las funciones tienen argumentos, los cuales denotan o que se ingresa a la función; es decir, el “input” de la función. En la literatura también se conoce como parámetros de la función, pero esto último puede enredar el lenguaje estadístico al que seguramente estarás ya acostumbrado, entonces por comodidad mejor llamarle argumento.

El valor de retorno de una función puede o no existir de manera explícita. Si el valor es explícito, éste será el valor que será el resultado en la función. De lo contrario, un objeto vacío será el “output” de la función.

La sintaxis básica para crear una función es la siguiente:

```
In [ ]: def <nombre de la función>(<lista de parámetros>):  
        <cuerpo de la función>  
        return <el "output" de la función>
```

## 2 Funcionamiento

Considere una función que multiplica dos números pasados como argumentos:

```
In [1]: def producto(num1, num2):  
        prod=num1*num2  
        print("el producto de", num1, "*", num2, "es", prod)  
        producto(2,4)
```

el producto de 2 \* 4 es 8

Así también, se puede usar return para explicitar el resultado de la función

```
In [2]: def producto(num1,num2):  
        prod=num1*num2  
        return str("el producto de ")+str(num1)+" * "+str(num2)+" es "+str(prod)  
        producto(2,4)
```

```
Out[2]: 'el producto de 2 * 4 es 8'
```

O aún mucho más simple, una función puede no tener un objeto de retorno:

```
In [3]: def hola():  
        print("hola")  
        hola()
```

hola

Y así puedes jugar con varias cosas...

```
In [4]: def cowsay(cdc): #cdc: cadena de caracteres  
        print("","_" * (len(cdc)+2))  
        print("< "+cdc+" >")  
        print("","-" * (len(cdc)+2))  
        print("      \  ^__^")  
        print("      \  (oo)\_____")  
        print("      (__) \       )\/\ ")  
        print("              ||----w |")  
        print("              ||     ||")  
        cowsay("moo")
```

```
-----  
< moo >  
-----  
      \  ^__^  
      \  (oo)\_____  
      (__) \       )\/\ "  
              ||----w |  
              ||     ||
```

### 3 Actividad

Dada una lista, es posible crear una función que busque un elemento en una lista. Por ejemplo, si ingreso la lista [1,2,3,4,5,6] y el valor 4, es deseable que muestre la posición 3 (recordando que las posiciones son contadas desde el 0).

Arregle el siguiente código para que muestre la posición de lo encontrado contando a partir del 0.

```
In [5]: def buscar(L,item):
        bandera=0
        for i in L:
            if i==item:
                bandera=1
                print("posición", i)
        if bandera ==0:
            print("no encontrado")
L=[1,2,5,9,10]
buscar(L,5)
```

posición 5

## 4 Tarea

1. Escriba una función que calcule la media de  $n$  números ingresados por el usuario.
2. Escriba una función que calcule la moda de  $n$  números ingresados por el usuario.
3. Escriba una función que calcule la mediana de  $n$  números ingresados por el usuario.
4. Escriba una función que calcule la desviación estándar de  $n$  números ingresados por el usuario.
5. Escriba una función que encuentre el máximo de una lista de números ingresada por el usuario.
6. Escriba una función que encuentre el mínimo de una lista de números ingresada por el usuario.