

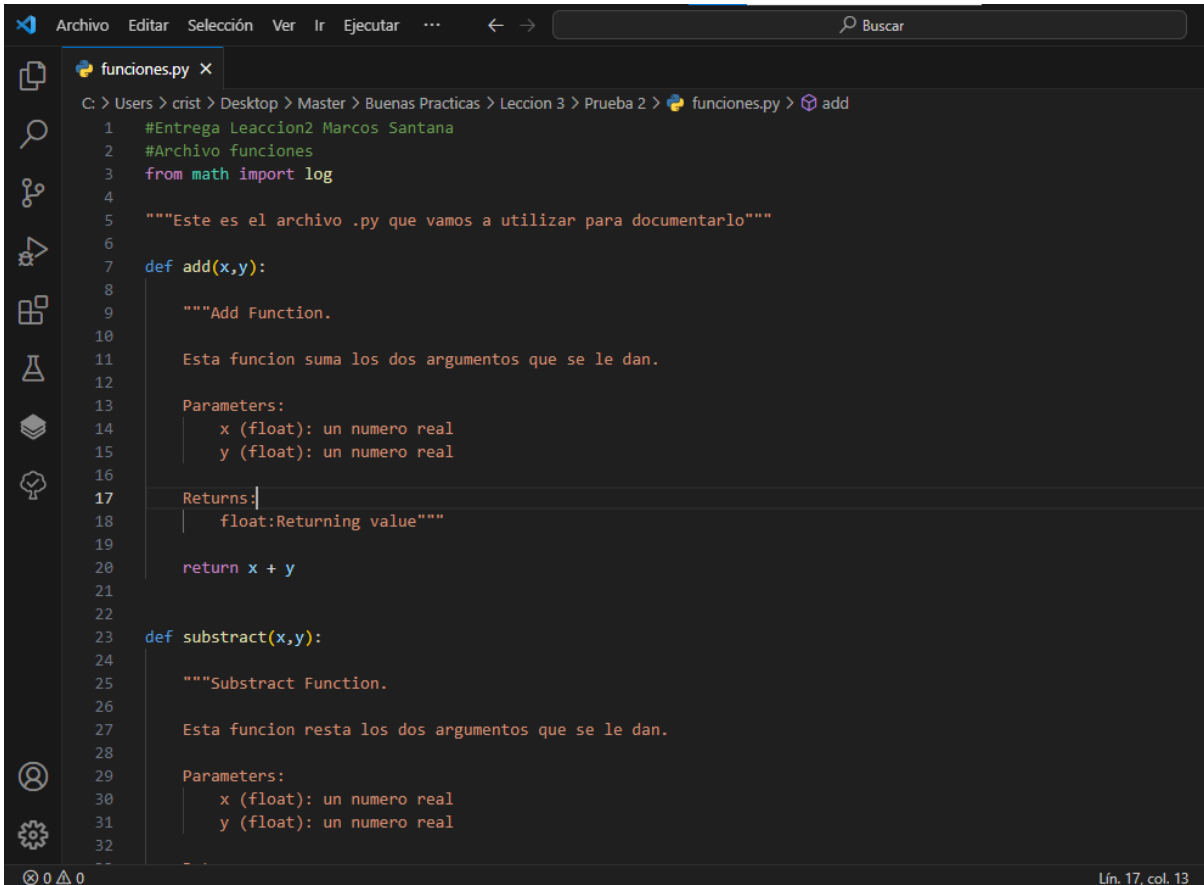
# USO DE OTRA HERRAMIENTA PARA DOCUMENTAR DIFERENTE DE Sphinx

Realizado por Marcos Santana Pastor para la asignatura de Buenas Prácticas de programación en Python. Entrega 3

El módulo `pydoc` genera automáticamente documentación a partir de módulos de Python. La documentación puede presentarse como páginas de texto en la consola, enviarse a un navegador web o guardarse en archivos HTML.

Para módulos, clases, funciones y métodos, la documentación mostrada es derivada del *docstring* del objeto, y recursivamente de sus miembros que se puedan documentar. Si no existe el *docstring*, `pydoc` trata de obtener una descripción del bloque de comentarios arriba de la definición de la clase, función o método en el archivo fuente, o encima del módulo.

Veamos un ejemplo de cómo usarlo. Vamos a utilizar el mismo Script de Python que vamos a documentar con Sphinx pero en este caso con `pydoc`. Está documentado siguiendo las normas de *docstring*.



```
funciones.py X
C:\Users\crist\Desktop\Master\Buenas Practicas\Leccion 3\Prueba 2\funciones.py
1 #Entrega Leccion2 Marcos Santana
2 #Archivo funciones
3 from math import log
4
5 """Este es el archivo .py que vamos a utilizar para documentarlo"""
6
7 def add(x,y):
8
9     """Add Function.
10
11     Esta funcion suma los dos argumentos que se le dan.
12
13     Parameters:
14         x (float): un numero real
15         y (float): un numero real
16
17     Returns:
18         float:Returning value"""
19
20     return x + y
21
22
23 def subtract(x,y):
24
25     """Substract Function.
26
27     Esta funcion resta los dos argumentos que se le dan.
28
29     Parameters:
30         x (float): un numero real
31         y (float): un numero real
32
33     """
```

Ahora, abriendo el terminal, escribimos `python -m pydoc` y nos mostrará las diferentes opciones de documentación que tenemos y una breve descripción.

```
C:\Users\cris\Desktop\Master\Buenas Practicas\Leccion 3\Prueba 2>python -m pydoc
pydoc - the Python documentation tool

pydoc <name> ...
    Show text documentation on something. <name> may be the name of a
    Python keyword, topic, function, module, or package, or a dotted
    reference to a class or function within a module or module in a
    package. If <name> contains a '\', it is used as the path to a
    Python source file to document. If name is 'keywords', 'topics',
    or 'modules', a listing of these things is displayed.

pydoc -k <keyword>
    Search for a keyword in the synopsis lines of all available modules.

pydoc -n <hostname>
    Start an HTTP server with the given hostname (default: localhost).

pydoc -p <port>
    Start an HTTP server on the given port on the local machine. Port
    number 0 can be used to get an arbitrary unused port.

pydoc -b
    Start an HTTP server on an arbitrary unused port and open a Web browser
    to interactively browse documentation. This option can be used in
    combination with -n and/or -p.

pydoc -w <name> ...
    Write out the HTML documentation for a module to a file in the current
    directory. If <name> contains a '\', it is treated as a filename; if
    it names a directory, documentation is written for all the contents.
```

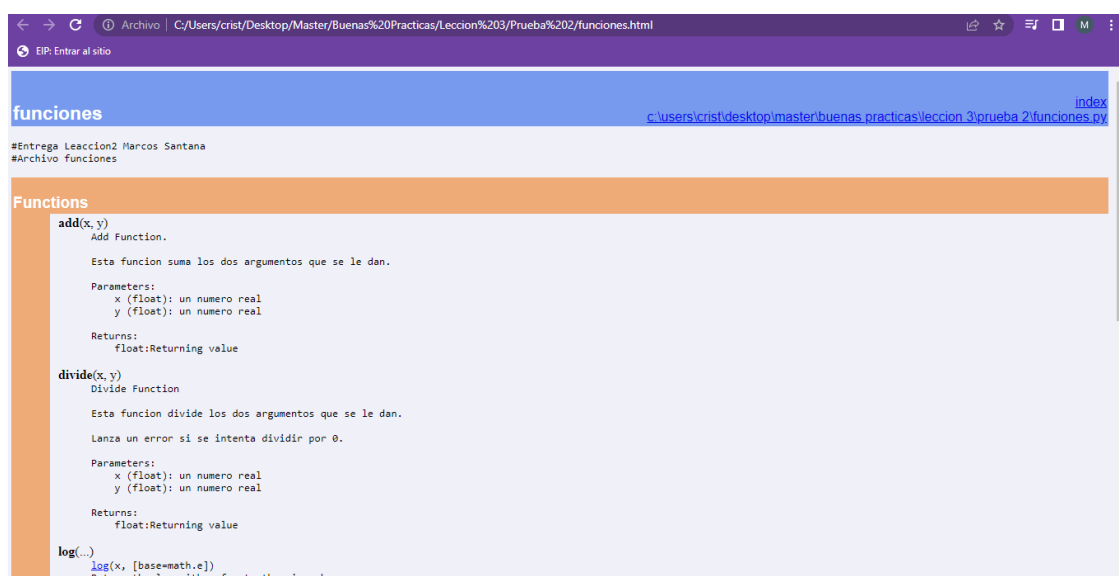
En este caso intentaremos obtener algo lo más similar posible a lo que tenemos con Sphinx, por lo tanto generamos un html con toda nuestra documentación. Esto lo haremos con la siguiente instrucción en la terminal.

```
C:\Users\cris\Desktop\Master\Buenas Practicas\Leccion 3\Prueba 2>python -m pydoc -w funciones
wrote funciones.html
```

Es decir,

`python -m pydoc -w <nombre del archivo python que queremos documentar>`

Obtenemos así el HTML con toda la documentación



```
← → C Archivo | C:/Users/crist/Desktop/Master/Buenas%20Practicas/Leccion%203/Prueba%202/funciones.html
EIP: Entrar al sitio

log(...)
log(x, [base=math.e])
Return the logarithm of x to the given base.
If the base not specified, returns the natural logarithm (base e) of x.

logarithm(x)
Logarithm Function
Esta funcion calcula el logaritmo neperiano del argumento que le dan.
Lanza un error en caso de que el valor de entrada no sea mayor que 0.
Parameters:
  x (float): un numero real
  y (float): un numero real
Returns:
  float:Returning value

multiply(x,y)
Substract Function
Esta funcion multiplica los dos argumentos que se le dan.
Parameters:
  x (float): un numero real
  y (float): un numero real
Returns:
  float:Returning value

substract(x,y)
Substract Function.
Esta funcion resta los dos argumentos que se le dan.
Parameters:
  x (float): un numero real
```

```
← → C Archivo | C:/Users/crist/Desktop/Master/Buenas%20Practicas/Leccion%203/Prueba%202/funciones.html
EIP: Entrar al sitio

If the base not specified, Returns the natural logarithm (base e) of x.

logarithm(x)
Logarithm Function
Esta funcion calcula el logaritmo neperiano del argumento que le dan.
Lanza un error en caso de que el valor de entrada no sea mayor que 0.
Parameters:
  x (float): un numero real
  y (float): un numero real
Returns:
  float:Returning value

multiply(x,y)
Substract Function
Esta funcion multiplica los dos argumentos que se le dan.
Parameters:
  x (float): un numero real
  y (float): un numero real
Returns:
  float:Returning value

substract(x,y)
Substract Function.
Esta funcion resta los dos argumentos que se le dan.
Parameters:
  x (float): un numero real
  y (float): un numero real
Returns:
  float:Returning value
```