

Buenas prácticas de programación en Python

Lección 3: PROGRAMACIÓN DIRIGIDA POR LA DOCUMENTACIÓN

ACTIVIDAD LECCIÓN 3

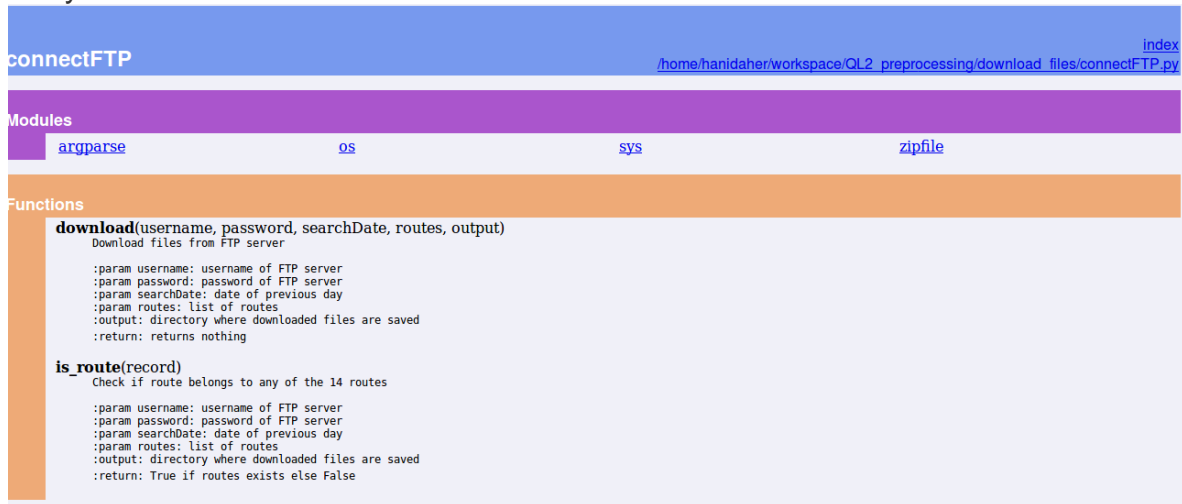
Como hemos estudiado en esta lección existen diversas herramientas que nos facilitan la tarea de documentar nuestro código. Para la actividad de esta lección se proponen dos ejercicios:

1. Busque una herramienta alternativa a Sphinx para generar la documentación de código en Python.
2. Realice la documentación de alguno de los códigos que ha desarrollado a lo largo de este curso. La documentación puede generarla con Sphinx o con la herramienta que ha buscado en el primer ejercicio.

Condiciones de entrega:

- Entregue un documento en formato PDF en el que describa la herramienta de documentación que ha encontrado. En dicho documento debe mostrarse un ejemplo básico del funcionamiento de esta herramienta.
- Entregue el código fuente.

1. PyDoc



El módulo `pydoc` genera automáticamente documentación de módulos de Python. La documentación se puede presentar como páginas de texto en la consola, servidos en un buscador web, o guardados en archivos HTML.

Para módulos, clases, funciones y métodos, la documentación mostrada es derivada del *docstring* (i.e. el atributo `__doc__`) 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 (véase `inspect.getcomments()`).

La función incorporada `help()` invoca el sistema de ayuda en línea en el interpretador interactivo, que usa `pydoc` para generar su documentación como texto en la consola. La misma documentación del texto se puede ver desde afuera del interpretador de python al ejecutar **pydoc** como un script en la consola del sistema operativo. Por ejemplo, ejecutando

```
pydoc sys
```

en la entrada de la consola mostrará la documentación sobre el módulo `sys`, en un estilo similar a las páginas del manual que se muestran por el comando **man** de Unix. El argumento de **pydoc** puede ser el nombre de una función, módulo, o paquete, o una referencia con puntos (*dotted reference*) de una clase, método, o función dentro de un módulo o módulo en un paquete. Si el argumento

de **pydoc** se parece a una ruta (*path*) (es decir, que contiene el separador de rutas para tu sistema operativo como una barra en Unix), y hace referencia a un archivo fuente de Python existente, entonces se produce la documentación para ese archivo.

Nota

Para encontrar los objetos y su documentación, **pydoc** importa el módulo o los módulos que serán documentados. Por lo tanto, cualquier código a nivel de módulo va a ser ejecutado en esa ocasión. Use `if __name__ == '__main__':` como protección para sólo ejecutar código cuando un archivo es invocado como un script y no sólo importado.

Cuando se imprime la salida a la consola, **pydoc** intenta paginar la salida para una lectura más fácil. Si la variable de entorno `PAGER` está puesta, **pydoc** usará su valor como el programa de paginación.

Especificar un bandera (*flag*) `-w` antes del argumento hará que la documentación HTML sea escrita en un archivo de la carpeta actual, en vez de mostrar el texto en la consola.

Especificar una bandera (*flag*) `-k` antes del argumento buscará las líneas de la sinopsis de todos los módulos disponibles por la palabra clave (*keyword*) dada como argumento, de nuevo en una manera similar al comando **man** de Unix. La sinopsis de un módulo es la primera línea de su cadena de documentación.

Puedes usar **pydoc** para empezar un servidor HTTP en la máquina local que va a servir la documentación para buscadores Web invitados. **pydoc -p 1234** empezará un servidor HTTP en el puerto 1234, permitiéndote buscar la documentación en `http://localhost:1234/` en tu buscador de preferencia. Especificar `0` como el número de puerto seleccionará un puerto arbitrario no usado.

pydoc -n <hostname> empezará el servidor escuchando al *hostname* (nombre del servidor) dado. Por defecto, el nombre del servidor es "localhost" pero si quieres que se llegue al servidor desde otras máquinas, quizás quieras cambiar el nombre por el cual el servidor responde. Durante el desarrollo esto es especialmente útil si quieres correr *pydoc* desde dentro de un contenedor.

pydoc -b empezará un servidor y adicionalmente abrirá un buscador web con una página del índice del módulo. Cada página servida tiene una barra de navegación arriba donde puedes obtener (*Get*) ayuda sobre una sección individual, buscar (*Search*) todos los módulos con una palabra clave y sus sinopsis, e ir a las páginas del índice del módulo (*Module index*), temas (*Topics*), y palabras clave (*Keywords*).

Cuando **pydoc** genera la documentación, se usa el entorno y ruta actual para localizar los módulos. Así, invocar **pydoc spam** precisamente documenta la versión del módulo que tu obtendrías si empezaras el interpretador de Python y escribieras `import spam`.

Se asume que la documentación de los módulos principales residen en `https://docs.python.org/X.Y/library/` donde `X` y `Y` son la versión mayor y menor de tu interpretador de Python. Esto puede ser sobre-escrito al poner a la variable de entorno `PYTHONDOS` una URL diferente o un directorio local conteniendo la páginas de referencia.

Podemos ver un ejemplo completo en <https://www.datacamp.com/community/tutorials/docstrings-python>

2. Código:

```
class Calculadora:
    """Calculadora

    Atributos
    -----
    num1:
        primer operando
    num2:
        segundo operando

    Metodos
    -----
    Sumar:
        Suma los operandos num1 y num2
    Restar:
        Resta los operandos num1 y num2
    Multiplicar:
```

```
    Multiplica los operandos num1 y num2
Dividir:
    Divide los operandos num1 y num2

Ejemplos
-----
>>> import Calculadora
>>> ot = Calculadora(num1,num2)
>>> resultado_suma = ot.Sumar()
>>> print(resultado_suma)
"""
def __init__(self,num1,num2):
    self.num1 = num1
    self.num2 = num2

def Sumar(self):
    print(self.num1 + self.num2)
    return self.num1 + self.num2

def Restar(self):
    print(self.num1 - self.num2)
    return self.num1 - self.num2

def Multiplicar(self):
    print(self.num1 * self.num2)
    return self.num1 * self.num2

def Dividir(self):
    if(self.num2 > 0):
        print(self.num1 / self.num2)
        return self.num1 / self.num2
    else:
        print("Error! no se puede dividir entre 0")
        return "Error"
```

RESULTADO:

Generación de documentación con sphinx

```
PS C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3  
\Actividad3> sphinx-quickstart
```

Bienvenido a la utilidad de inicio rápido de Sphinx 4.0.2.

Ingrese los valores para las siguientes configuraciones (solo pr
esione Entrar para
aceptar un valor predeterminado, si se da uno entre paréntesis).

Ruta raíz seleccionada: .

Tiene dos opciones para colocar el directorio de compilación par
a la salida de Sphinx.

O usas un directorio "_build" dentro de la ruta raíz, o separas
directorios "fuente" y "compilación" dentro de la ruta raíz.

> Separar directorios fuente y compilado (y/n) [n]: y

El nombre del proyecto aparecerá en varios lugares en la documen
tación construida.

> Nombre de proyecto: Docu project

> Autor(es): Marc

> Liberación del proyecto []: 1.0

Si los documentos deben escribirse en un idioma que no sea inglés,
s,

puede seleccionar un idioma aquí por su código de idioma. Sphinx
entonces

traducir el texto que genera a ese idioma.

Para obtener una lista de códigos compatibles, vea

[https://www.sphinx-doc.org/en/master/usage/configuration.html#co
nfval-language](https://www.sphinx-doc.org/en/master/usage/configuration.html#confval-language).

> Lenguaje del proyecto [en]:

```
Creando archivo C:\Users\Administrador\Desktop\MasterPython\BPPy  
thon\Leccion3\Actividad3\source\conf.py.  
Creando archivo C:\Users\Administrador\Desktop\MasterPython\BPPy  
thon\Leccion3\Actividad3\source\index.rst.  
Creando archivo C:\Users\Administrador\Desktop\MasterPython\BPPy  
thon\Leccion3\Actividad3\Makefile.  
Creando archivo C:\Users\Administrador\Desktop\MasterPython\BPPy  
thon\Leccion3\Actividad3\make.bat.
```

Terminado: se ha creado una estructura de directorio inicial.

Ahora debe completar su archivo maestro C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\source\index.rst y crear otros archivos fuente de documentación. Use el archivo Makefile para compilar los documentos, así ejecute el comando:

```
make builder
```

donde "builder" es uno de los constructores compatibles, por ejemplo, html, latex o linkcheck.

```

1  # Configuration file for the Sphinx documentation builder.
2  #
3  # This file only contains a selection of the most common options. For a full
4  # list see the documentation:
5  # https://www.sphinx-doc.org/en/master/usage/configuration.html
6
7  # -- Path setup -----
8
9  # If extensions (or modules to document with autodoc) are in another directory,
10 # add these directories to sys.path here. If the directory is relative to the
11 # documentation root, use os.path.abspath to make it absolute, like shown here.
12 #
13 import os
14 import sys
15 sys.path.insert(0, os.path.abspath('.'))
16
17
18 # -- Project information -----
19
20 project = 'Docu project'
21 copyright = '2021, Marc'
22 author = 'Marc'
23
24 # The full version, including alpha/beta/rc tags
25 release = '1.0'
26
27
28 # -- General configuration -----
29
30 # Add any Sphinx extension module names here, as strings. They can be
31 # extensions coming with Sphinx (named 'sphinx.ext.*') or your custom
32 # ones.
33 extensions = [
34 ]

```

```

PS C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc> sphinx-apidoc -f
-o source/ ../Ops/
Creando archivo source/Ops.rst.
Creando archivo source/Ops.Utils.rst.
Creando archivo source/modules.rst.

```


Modificamos el fichero index añadiendo modules

```
Doc > source > index.rst
1  .. Docu documentation master file, created by
2     sphinx-quickstart on Tue Jun 22 23:07:59 2021.
3     You can adapt this file completely to your liking, but it should at least
4     contain the root `toctree` directive.
5
6  Welcome to Docu's documentation!
7  =====
8
9  .. toctree::
10     :maxdepth: 2
11     :caption: Contents:
12
13     modules
14
15
16  Indices and tables
17  =====
18
19  * :ref:`genindex`
20  * :ref:`modindex`
21  * :ref:`search`
22
```

Desde win PowerShell no se reconoce el comando make. He tenido que ejecutarlo desde el cmd

```
C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc>make html
Ejecutando Sphinx v4.0.2
creando directorio de salida... hecho
compilando [mo]: los objetivos para 0 los archivos po que estan desactualizados
compilando [html]: los objetivos para 4 los archivos fuentes que estan desactualizados
actualizando ambiente[nueva configuración] 4añadido, 0 cambiado, 0 removido
leyendo fuentes...[100%] modules
C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc\source\Ops.rst:15: WA
RNING: Unknown directive type "automodule".

.. automodule:: Ops
   :members:
   :undoc-members:
   :show-inheritance:
C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc\source\Ops.Utils.rst:
10: WARNING: Unknown directive type "automodule".

.. automodule:: Ops.Utils.Calculadora
   :members:
   :undoc-members:
   :show-inheritance:
C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc\source\Ops.Utils.rst:
18: WARNING: Unknown directive type "automodule".

.. automodule:: Ops.Utils
   :members:
   :undoc-members:
   :show-inheritance:
C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc\source\index.rst:9: W
ARNING: toctree contains reference to nonexistent document 'modules'
buscando por archivos no actualizados...no encontrado
preparando ambiente... hecho
verificando consistencia... C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc\source\modules.rst: WARNING: documento no está incluido en ningún toctree
hecho
preparando documentos... hecho
escribiendo salida... [100%] modules
generando índices... genindex hecho
escribiendo páginas adicionales... search hecho
copying static files... hecho
copiando archivos extras... hecho
volcar el índice de búsqueda en English (code: en)... hecho
volcar inventario de objetos... hecho
compilación exitosa, 5 advertencias.

Las páginas HTML están en build\html.
C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc>
```

No puedo ver la documentación

**** EL ERROR QUE HE ENCONTRADO ES QUE DESPUES DE AÑADIR MODULES A index.rst**

Hay que lanzar el comando

C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc>sphinx-apidoc -f -o source/ ../Ops/

Entonces si hacemos make html

...

```
C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Ops\Utils\Calculadora.py:docstring of Ops.Utils.Calculadora.Calculadora.Sumar:7: WARNING: Unexpected section title.
```

Outputs

```
-----
buscando por archivos no actualizados...no encontrado
preparando ambiente... hecho
verificando consistencia... hecho
preparando documentos... hecho
escribiendo salida... [ 25%] Ops escribiendo salida...
escribiendo salida... [100%] modules
generando índices... genindex py-modindex hecho
escribiendo páginas adicionales... search hecho
copying static files... hecho
copiando archivos extras... hecho
volcar el índice de búsqueda en English (code: en)... hecho
volcar inventario de objetos... hecho
compilación exitoso, 15 advertencias.

Las páginas HTML están en build\html.
```

```
C:\Users\Administrador\Desktop\MasterPython\BPPython\Leccion3\Actividad3\Doc>
```

HTML

Docu

Navigation

Contents:

[Ops](#)

Quick search

Welcome to Docu's documentation!

Contents:

- [Ops](#)
 - [Ops package](#)

Indices and tables ¶

- [Index](#)
- [Module Index](#)
- [Search Page](#)

Ops.Utils package

Submodules

Ops.Utils.Calculadora module

```
class Ops.Utils.Calculadora.Calculadora(num1, num2) ¶
```

Bases: **object**

num1:

primer operando

num2:

segundo operando

Sumar:

Suma los operandos num1 y num2

Restar:

Resta los operandos num1 y num2

Multiplicar:

Multiplica los operandos num1 y num2