

Actividad relacionada con la 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.

Doxygen es un generador de documentación para C++, C, Java, Objective-C, Python, IDL (versiones Corba y Microsoft), VHDL y en cierta medida para PHP, C# y D. Dado que es fácilmente adaptable, funciona en la mayoría de sistemas Unix así como en Windows y Mac OS X. La mayor parte del código de Doxygen está escrita por Dimitri van Heesch.

Doxygen es un acrónimo de dox(document) gen(generator), generador de documentación para código fuente.

Varios proyectos como KDE usan Doxygen para generar la documentación de su API. KDevelop incluye soporte para Doxygen.

Ejemplo de código

```
"""@package pyexample_a
Documentation for this module.
More details.
"""

def func():
    """Documentation for a function.
    More details.
    """
    pass

class PyClass:
    """Documentation for a class.
    More details.
    """

    def __init__(self):
        """The constructor."""
        self._memVar = 0

    def PyMethod(self):
        """Documentation for a method."""
        pass
```

Compilando documentación

```
cd toplevel-source/
doxygen -g
doxygen Doxyfile
xdg-open ./html/index.html
```

Ejemplo de resultado

Class List

Here are the classes, structs, unions **and** interfaces **with** brief descriptions:

N pyexample_a

C PyClass

N pyexample_b Documentation **for** this module

C PyClass Documentation **for** a **class**

- 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.**

Código adjunto en la carpeta de este documento.