
Sphinx Para Novatos Guía de Aprendizaje

Publicación 1.0

Igor F. Dávalos Rojas

16 de September de 2016

I	EL COMIENZO DEL CAOS	1
1.	PRIMEROS PASOS	3
1.1.	Buscando Ayuda sobre Sphinx	3
1.2.	Ayuda Encontrada	3
1.3.	Requerimientos Iniciales	4
II	ANEXO	5
2.	Enlaces Útiles	6
III	Indices and tables	7
	Bibliografía	9

Parte I

EL COMIENZO DEL CAOS

Es mi tabla de contenidos

- *EL COMIENZO DEL CAOS* (Página 2)
 - *PRIMEROS PASOS* (Página 3)
 - *Buscando Ayuda sobre Sphinx* (Página 3)
 - *Ayuda Encontrada* (Página 3)
 - *Requerimientos Iniciales* (Página 4)

PRIMEROS PASOS

La instalación de **Sphinx**¹ es sencilla. El uso del lenguaje **reStructuredText** es muy fácil de entender y sencillo de implementar cuando se lo utiliza a diario. Sabiendo esto, puedes elaborar documentos que van desde una estructuración sencilla a una muy compleja. Inicié esta referencia en español como una práctica más para entender como se puede sacar el máximo provecho a Sphinx y al lenguaje RST sin ser un desarrollador ni tener conocimientos sobre **Python**² o similares.

Buscando Ayuda sobre Sphinx

La idea del creador de Sphinx (desarrollado en Python) fue tener a disposición de los desarrolladores una aplicación que les permita elaborar sus guías, manuales, y demás documentos de forma sencilla (para ellos, claro), dándoles las herramientas necesarias con las cuales puedan enlazar imágenes, tablas, código de sus programas y lo que vieran conveniente.

En este contexto, es que me topé con que la mayor parte de la documentación está más destinada a ser entendida por desarrolladores y no así por usuarios novatos o también llamados usuarios finales o comunes. Pues para un usuario común que necesita una aplicación para crear documentos de uso diario existen otras alternativas que son *fáciles* de aprender, como por ejemplo: ApacheOpenOffice writer, LibreOffice writer, Abiword, MS Word, Wordperfect, Etc.

Ayuda Encontrada

Existe bastante ayuda para aprender sobre reStructuredText, principalmente en su [página oficial](http://docutils.sourceforge.net/rst.html) (<http://docutils.sourceforge.net/rst.html>) y en otras páginas encontrarás muchas referencias como en estas páginas que indico en [Enlaces Útiles](#) (Página 6).

Como indiqué en [PRIMEROS PASOS](#) (Página 3), la explicación sobre Sphinx es bastante técnica, osea se asume que el lector tiene conocimientos sobre Python o de programación, ya que las explicaciones están destinadas más que todo a la estructura del documento y no así a su maquetación final. Dejando así un gran vacío al usuario común sobre *cómo* darle cuerpo al documento final. Si tienes conocimientos sobre Tex o Latex o similares, no tendrás ningún problema en adaptar la presentación final a tus gustos y necesidades; caso contrario quedas invitado a seguir leyendo esta *Guía de Aprendizaje* y a realizar en tu editor de textos favorito, los ejemplos que que se van desarrollando.

¹ Página web de Sphinx (<http://sphinx-doc.org/>)

² Página web de Python (<http://www.python.org/>)

Requerimientos Iniciales

Lo principal que necesitas es *Saber leer y escribir*, eso es, saber leer exactamente lo que está escrito y escribirlo de igual modo. Esto referido a la sintaxis del formato, no al contenido que estarás creando. Por ejemplo, no obtienes lo mismo con `'esto' = 'esto'`; que con `'esto' = esto`.

necesitarás usar un `editor de textos` como el que viene instalado por defecto en tu sistema operativo.

- En **Linux** puedes disponer de los siguientes:
 - `medit`, `geany`, `gedit`, `nano` y muchos más.
- Para **Windows** tienes:
 - `block de notas`, el `notepad++` este último muy recomendado pues posee complementos para el resaltado de código y permite cambiar la codificación de caracteres.

Parte II

ANEXO

Enlaces Utiles

Listado de enlaces de utilidad para aprender sobre reStructuredText y Sphinx.

aaa

Parte III

Indices and tables

- genindex
- modindex
- search

Bibliografía

[CIT2016] Ejemplo de una cita bibliográfica (Que a veces es bibliográfica otras solo de referencia.)