

Documentación de UnoGenerator

Version: 0.9.0

1 Introducción

UnoGenerator uses Libreoffice UNO API python bindings to generate documents. So in order to use, you need to launch a `--headless libreoffice` instance. You can easily launch `unogenerator_start` script with bash.

UnoGenerator has standard templates to help you with edition, although you can use your own templates. Puedes editar éste o crear el tuyo. This document has been created with 'standard.odt' files that you can find inside this python module.

1.1 Instalación

Puedes usar pip para instalar este paquete python:

```
pip install unogenerator
```

1.2 Ejemplo 'Hola Mundo' de ODT

Este un ejemplo 'Hello World'. Has generado el ejemplo en los formatos odt, docx y pdf

```
from unogenerator import ODT_Standard
doc=ODT_Standard()
doc.addParagraph("Hello World", "Heading 1")
doc.addParagraph("Easy, isn't it", "Standard")
doc.save("hello_world.odt")
doc.export_docx("hello_world.docx")
doc.export_pdf("hello_world.pdf")
doc.close()
```

2 ODT

Los ficheros ODT son generados rápidamente con UnoGenerator. Hay una plantilla predefinida en el código llamada 'standard.odt' para ayudar con la edición.

2.1 Llamando al constructor ODT

Puedes llamar al constructor ODT de estas maneras:

- ODT with standard template (Recomended). There is a predefined template in code called 'standard.odt', inside this python module, to help you with edition, although you can use your own ones. Con este modo puedes crear nuevos documentos

```
from unogenerator import ODT_Standard
doc=ODT_Standard()
```

- ODT with template or file (Recomended). With this mode you can read your files to overwrite them or use your file as a new template to create new documents

```
from unogenerator import ODT
doc=ODT('yourdocument.odt')
```

- ODT sin plantilla With this mode you can write your files with Libreoffice default styles. If you want to create new ones, you should write them using Libreoffice API code

```
from unogenerator import ODT
doc=ODT()
```

2.2 Estilos

To call default Libreoffice paragraph styles you must use their english name. Puedes ver sus nombres con este método:

```
doc.print_styles()
```

2.3 Tablas

We can create tables with diferent font sizes and formats:[15, 70, 15]

Concepto	Valor	Commnet
Texto	Esto es un texto	Bueno
Fecha y hora	2021-10-28 22:10:39.638207	Bueno
Fecha	2021-10-28	Bueno
Float	12.121	Bueno
Divisa	-12.12 €	Bueno
Porcentaje	33.33 %	Bueno

Concepto	Valor	Commnet
Texto	Esto es un texto	Bueno
Fecha y hora	2021-10-28 22:10:39.638207	Bueno
Fecha	2021-10-28	Bueno
Float	12.121	Bueno

Divisa	-12.12 €	Bueno
Porcentaje	33.33 %	Bueno

2.4 Listas y listas numeradas

- Lista simle
 - Lista simle
 - Lista simle
- Lista simle
 - Lista simle
- Lista simle

2.5 Imágenes

Este es un ejemplo de imagen as char:  . Ahora sigo escribiendo sin problemas.

Como puedes ver, puedo reutilizar la imagen cien veces. El tamaño del fichero no aumentará porque uso referencias. 

El siguiente párrafo es generado con el método illustration



2.6 Buscar y reemplazar

Below this paragraph is a paragraph with a % REPLACEME % (Without white spaces) text and it's going to be replaced after all document is been generated

This paragraph was set at the end of the code after a find and replace command.

This paragraph was set after replacement.

This paragraph was set after a page break.

This paragraph was set after a page break with Landscape style.

3 ODS

3.1 ODS 'Hello World' example

This is a Hello World example. You'll get the example in ods, xlsx and pdf formats:

```
from unogenerator import ODS_Standard
doc=ODS_Standard()
doc.addCellMergedWithStyle("A1:E1", "Hello world", style="BoldCenter")
doc.save("hello_world.ods")
doc.export_xlsx("hello_world.xlsx")
doc.export_pdf("hello_world.pdf")
doc.close()
```