

Introducción a Python creando Scripts con \$ Click_ (v2.0)

Por: @Perla_GCastillo, @Jose_Carlos_Mz y @rctorr

unix | utf-8 | python

**Invitan #PythonMexico,
#Datos.club**

**Gracias a NFO Systems y
RedHat**

Presentación

- Nombre
- A qué te dedicas

```
Editar Ver Buscar Terminar  
#!/usr/bin/env python  
# -*- coding: utf-8 -*-
```

```
import sys
```

```
def main(argv):  
    """ Función principal del script """
```

```
    print("Hola", argv[1], "!")
```

```
if __name__ == "__main__":  
    main(sys.argv)
```

```
unix | utf-8 | python
```

genpass.py, Ver 0.1

genpass.py:

Es un generador de claves usando como base el alfabeto inglés, tanto en minúsculas como en mayúsculas.

Opciones:

--len N Indica la longitud de la clave y por omisión es 8, si indicas un valor menor a 8 recibirás un horrible mensaje.

--n N Indica el número de clave a generar.

--símbolos Indica si se incluyen o no algunos símbolos como parte del alfabeto.

Por omisión no se incluyen.

l.py PATH, Ver 0.1

Toma los datos de cada archivo en PATH y los imprime en la salida estándar en formato de columnas de ancho fijo, si PATH no se proporciona entonces se usa el directorio actual.

Se incluyen las siguientes columnas:

- Nombre
- Extensión
- Tamaño
- Fecha

l.py PATH, Ver 0.1 (revisión)

```
rctorr@gatev ~/repos/PythonMexico/Crea-tus-propios-comandos-en-Python
.py
hola 684 08-10-2018 18:36
img 4096 08-10-2018 18:36
hola-0.1.py .py 602 08-10-2018 18:36
hola-0.2.py .py 714 08-10-2018 18:36
holaclick-0.1.py .py 363 08-10-2018 18:40
hola-0.5.py .py 684 08-10-2018 18:36
hola-0.3.py .py 724 08-10-2018 18:36
holaclick-0.2.py .py 560 08-10-2018 19:31
hola-0.4.py .py 1148 08-10-2018 18:36
l.py .py 685 08-10-2018 21:34
```

l.py --tohtml PATH, Ver 0.2

Opciones:

--tohtml Imprime en la salida estándar la lista de archivos en formato HTML

```
python l.py --tohtml Descargas/ > descargas.html
```

l.py --tohtml PATH, Ver 0.2.1

Opciones:

--tohtml Imprime en la salida estándar la lista de archivos en formato HTML

```
python l.py --tohtml Descargas/ > descargas.html
```

```
<html>
```

```
<head>
```

```
  <styles src="main.css"></styles>
```

```
</head>
```

```
<body>
```

```
  <!-- tabla →
```

```
</body>
```


l.py --tocsv PATH, Ver 0.3

Opciones:

--tocsv Imprime en la salida estándar la lista de archivos en formato CSV

```
rctorr@gatev ~/repos/PythonMexico/Crea-tus-propios-comandos-en-Python $ python l
-0.2.py --tocsv
img,,4096,08-10-2018 18:36
hola-0.1.py,.py,602,08-10-2018 18:36
hola-0.2.py,.py,714,08-10-2018 18:36
holaclick-0.1.py,.py,363,08-10-2018 18:40
hola-0.5.py,.py,684,08-10-2018 18:36
l-0.1.py,.py,685,08-10-2018 21:44
hola-0.3.py,.py,724,08-10-2018 18:36
holaclick-0.2.py,.py,560,08-10-2018 19:31
l-0.2.py,.py,1202,08-10-2018 21:59
archivos.csv,.csv,414,08-10-2018 22:00
hola-0.4.py,.py,1148,08-10-2018 18:36
```

l.py --toxls PATH, Ver 0.4

Opciones:

--toxls Guarda la lista de archivo en formato XLS en el archivo salida.xls

catxml.py [Opciones] PATHFILE, Ver 0.1

Imprime en la salida estándar el contenido de PATHFILE en forma de árbol.

Opciones:

--raw Imprime el contenido del archivo sin cambio alguno.

catxml.py [Opciones] PATHFILE, Ver 0.2

Imprime en la salida estándar el contenido de PATHFILE en forma de árbol.

Opciones:

--raw Imprime el contenido del archivo sin cambio alguno.

--node NAME Imprime todos los nodos cuyo nombre sea NAME

xmltocsv.py [Opciones] PATHFILE, Ver 0.1

Lee de PATHFILE el valor de los nodos indicados por --node nom1, nom2, ..., nomN e imprime en la salida estándar en formato CSV una línea conteniendo el valor de los nodos solicitados

Salida:

```
nom1, nom2, ..., nomN  
val1, val2, ..., valN
```

Opciones:

--node NAME Indica el nombre del nodo a leer

xmltocsv.py [Opciones] PATHFILE, Ver 0.2

Lee de PATHFILE el valor de los nodos indicados por --node nom1, nom2, ..., nomN e imprime en la salida estándar en formato CSV una línea conteniendo el valor de los nodos solicitados.

Si PATHFILE no se indica, se lee desde la entrada estándar la lista de archivos con extensión xml desde los cuales se obtendrán los valores. Si algún archivo no tiene extensión xml, se descarta, si algún archivo no tiene algún nodo se omite ese nodo.

Opciones:

--node NAME1, NAME2, ..., NAMEN Indica el nombre de los nodos a leer

xmltocsv.py [Opciones] PATHFILE, Ver 0.2

Salida:

```
nom1, nom2, ..., nomN  
val1, val2, ..., valN  
val1, val2, ..., valN  
...  
val1, val2, ..., valN
```

Para más info:

Telegram:

<https://t.me/pythonCDMX>

Repo:

<https://github.com/rctorr/PythonMexico>

iGracias!

@rctorr

@Perla_GCastillo

@Jose_Carlos_Mz
