

Python Programming

Módulo 06

Aplicaciones ejecutables

Distribuir aplicaciones como ejecutables con PyInstaller

Habíamos dicho que Python es un *lenguaje interpretado*, de modo que no se compila a un código que es ejecutado directamente por el procesador, sino que requiere, justamente, de un intérprete. Ahora bien, hemos desarrollado nuestra aplicación de escritorio y queremos distribuirla. ¿Debemos pretender que todos los usuarios que la utilicen tengan Python instalado? Eso sería una experiencia poco amigable.

Para solucionar este problema se hace uso de alguna herramienta que convierte nuestro programa de Python en un archivo ejecutable (.exe en Windows), de modo que el intérprete de Python no tenga que estar necesariamente instalado. Entre ellas se encuentran py2exe, PyInstaller y cx_Freeze. En este curso estaremos empleando la segunda, ya que es sencilla de utilizar.

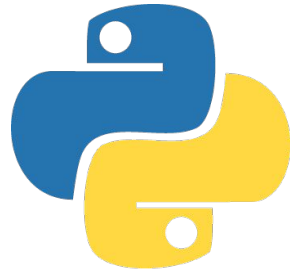
Podemos instalar PyInstaller vía pip:

```
python -m pip install pyinstaller
```

Su utilización es muy sencilla. Abrimos una terminal en la carpeta en donde se encuentre el *script* que queramos distribuir y ejecutamos:

```
pyinstaller miscript.py
```

Luego de unos cuantos mensajes en la consola, se habrán creado dos carpetas y un archivo: `build`, `dist` y `miscript.spec`, respectivamente. Dentro de `dist` encontrarás el archivo ejecutable junto con sus dependencias listo para ser distribuido y utilizado sin necesidad de tener un intérprete de Python.



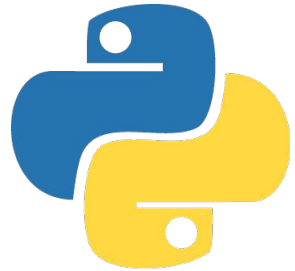
Si queremos aún más portabilidad haciendo que PyInstaller genere únicamente un archivo ejecutable sin dependencias, podemos usar la opción `--onefile`:

```
pyinstaller --onefile miscript.py
```

De forma similar, `--noconsole` indica que la consola no debe mostrarse (ideal para aplicaciones de escritorio):

```
pyinstaller --noconsole miscript.py
```

Encontrarás más ejemplos y la documentación completa en la [página oficial de PyInstaller \(pyinstaller.org\)](https://pyinstaller.org).



¡Sigamos trabajando!