



Centro de Electricidad y  
**Automatización Industrial**  
Regional Valle del Cauca



Python



## ✓ Distribuible

Ya tenemos nuestros módulos y paquetes ¿**Cómo podemos distribuirlos para que otros lo utilicen en su entorno de ejecución?**

## ✓ ¿Cómo crear un paquete distribuible?

La creación de un distribuible requiere que creamos un archivo adicional al mismo nivel del paquete principal llamado **setup.py** el cual debe contener una información básica sobre el paquete distribuido.

ejemplo:

```
from setuptools import setup
setup(
    name="mi_paquete_distribuible",
    version="0.1",
    description="Este es un paquete que quiero distribuir",
    author="Olga P Moreno G",
    author_email="omorenog@sena.edu.co",
    url="http://www.mipagina.com",
    packages=['mi_paquete_distribuible', 'mi_paquete_distribuible.subpaquete']
)
```

El anterior archivo setup.py debe encontrarse en la misma ruta en la que se encuentra el paquete principal "**mi\_paquete\_distribuible**"

Ubicados en esa ruta ejecutaremos la siguiente instrucción:

```
python setup.py sdist
```

Lo anterior creará una carpeta llamada **dist** con un archivo **.zip** o **tar.gz** con nuestro distribuible, el cual ya podemos compartir con quienes queramos.

## ✓ ¿Cómo instalar un paquete distribuido?

Una vez contamos con un archivo distribuido, nos ubicamos en la ruta donde se encuentra el archivo **.zip** o **tar.gz** y allí ejecutamos la siguiente instrucción:

```
pip install archivo_distribuido.zip
o
pip install archivo_distribuido.tar.gz
```

Después de instalado nuestro paquete distribuido ya podemos importarlo en cualquiera de nuestros programas.

## ✓ Apropiación

1. Cree un paquete llamado **mi\_paquete** el cual debe contener un módulo llamado **modulo\_base** con una función llamada **sumatodo** que reciba cualquier cantidad de números y retorne su sumatoria. Recuerde incluir en el paquete un archivo `__init__.py` para que python lo identifique como un paquete.
2. Cree un archivo distribuible del paquete **mi\_paquete** creado en el punto anterior.
3. Instale el archivo distribuible en su equipo
4. Verifique la instalación con los comandos **list** y **show** de PIP
5. Cree un pequeño programa en python que haga uso de la función **sumatodo** que se encuentra en el **modulo\_base** del paquete **mi\_paquete**

## ✓ Ejecutable

Ahora nuestra atención se centra en crear un archivo ejecutable de un programa desarrollado en Python.

### ¿Cómo crear un archivo ejecutable?

Lo primero que debemos hacer es instalar el módulo **pyinstaller** con el comando

```
pip install pyinstaller
```

Una vez instalado pyinstaller simplemente debemos ubicarnos en la ruta donde se encuentra el archivo de python .py del cual queremos generar el ejecutable y ejecutar el siguiente comando:

```
pyinstaller --onefile programa_python.py
```

A partir de allí se crearán varias carpetas y en una de ellas (la carpeta dist)encontraremos el archivo ejecutable (.exe) de nuestro programa.

### ¿Cómo ejecutar el archivo .exe

Simplemente nos dirigimos a la ubicación en la que se encuentra el archivo y lo abrimos

## ✓ Apropiación

1. Realice el juego de **adivine el número** en el que el usuario tiene 3 oportunidades para adivinar un número entre 0 y 10 generado aleatoriamente. El juego termina cuando el usuario adivina el número usando sus tres oportunidades (el usuario gana) o cuando se usan las tres oportunidades sin adivinar el número (el computador gana).
2. Cree un ejecutable .exe del programa
3. Ejecute el programa desde cualquier equipo (incluso si no tiene python instalado)

Comienza a programar o [generar](#) con IA.

No fue posible conectarse al servicio de reCAPTCHA. Comprueba tu conexión a Internet y vuelve a cargar la página para obtener un desafío de reCAPTCHA.