

Tema 20: Distribución de Aplicaciones Python

De Código Fuente a Ejecutable (.exe) con PyInstaller

Curso de Python con Qt Designer

17 de diciembre de 2025

1. Introducción

¡Felicitaciones! Has llegado al final del desarrollo. Hasta ahora, tu aplicación (Gestor de Gastos) vive dentro de archivos .py que solo funcionan en tu computadora porque tienes Python instalado.

El objetivo de este tema es convertir esos archivos en un **Ejecutable independiente (.exe)** que puedas enviar a cualquier persona (cliente, amigo, profesor) y que funcione en su computadora sin necesidad de instalar Python ni librerías.

Para ello utilizaremos una herramienta estándar en la industria llamada **PyInstaller**.

2. Paso 1: Preparación del Entorno

Antes de comenzar, el orden es fundamental. Asegúrate de tener una carpeta limpia exclusivamente para tu proyecto final. Esta carpeta debe contener los siguientes archivos:

- `main.py` (El script principal con la lógica).
- `gastos_ui.py` (La interfaz del gestor compilada).
- `calculadora_ui.py` (La interfaz de la calculadora compilada).
- `datos_gastos.json` (Opcional: Si ya tienes datos guardados, si no, el programa lo creará).

Nota Importante

Asegúrate de que tu programa funciona correctamente ejecutando `main.py` antes de intentar compilarlo. Si hay errores en el código, el ejecutable también fallará.

3. Paso 2: Instalación de PyInstaller

PyInstaller es una librería externa. Debemos instalarla usando pip. Abre tu terminal (consola o símbolo del sistema) en la carpeta de tu proyecto y ejecuta:

```
pip install pyinstaller
```

4. Paso 3: Creación del Ejecutable

Vamos a utilizar un comando específico para generar un archivo único, limpio y sin la ventana negra de la consola de fondo.

En tu terminal, asegúrate de estar en la carpeta del proyecto y ejecuta:

```
pyinstaller --name="GestorGastos" --onefile --noconsole main.py
```

4.1. Explicación de los parámetros:

- **-name="GestorGastos"**: Define el nombre que tendrá tu archivo .exe.
- **-onefile**: Empaquetá todo (librerías, imágenes, código) en un solo archivo. Sin esto, se generaría una carpeta con miles de archivos sueltos.
- **-noconsole (o -w)**: Indica que es una aplicación gráfica (GUI). Esto oculta la ventana negra de comandos (CMD) que normalmente aparece al ejecutar scripts de Python.
- **main.py**: Es tu archivo principal, el punto de entrada de la aplicación.

5. Paso 4: Localizar tu Aplicación

Una vez que la terminal muestre el mensaje `Completed successfully`, verás que han aparecido nuevas carpetas en tu proyecto:

1. **build/** (Archivos temporales, puedes borrarlos).
2. **main.spec** (Configuración de compilación, puedes borrarlo).
3. **dist/** (Distribution). **¡Aquí está tu programa!**

Abre la carpeta **dist**. Dentro encontrarás el archivo **GestorGastos.exe**. Puedes mover este archivo al Escritorio, a una memoria USB o enviarlo por correo. ¡Es 100 % independiente!

6. Bonus: Añadir un Icono Personalizado

Para darle un acabado profesional, puedes añadir un ícono (.ico).

1. Consigue una imagen en formato .ico (ej: **dinero.ico**) y ponla junto a **main.py**.
2. Ejecuta el siguiente comando:

```
pyinstaller --name="GestorPro" --onefile --noconsole --icon="dinero.ico" main.py
```