

Proyecto de aplicación para realizar conversiones de sistemas numéricos

1. Descripción del proyecto

La aplicación Conversión Numérica es una herramienta educativa desarrollada en Python que permite convertir números entre distintos sistemas de numeración: binario, octal, decimal y hexadecimal.

La interfaz gráfica fue creada con **Tkinter** y estilos modernos (`ttk.Style`) para ofrecer una experiencia intuitiva, clara y funcional. La aplicación valida automáticamente las entradas del usuario y permite trabajar con números negativos, mostrando resultados precisos en todos los sistemas soportados.

2. Objetivos del proyecto

Objetivo general

Desarrollar una aplicación interactiva que permita la conversión de números entre sistemas: binario, octal, decimal y hexadecimal, con interfaz gráfica amigable y validación de entradas.

Objetivos principales

- Facilitar la conversión rápida entre distintos sistemas de numeración.
- Garantizar que las entradas del usuario sean válidas y coherentes con el sistema seleccionado.
- Implementar una interfaz moderna y clara, con botones funcionales y resultados visibles.
- Permitir copiar los resultados al portapapeles para uso externo.
- Generar un ejecutable .exe que funcione en Windows sin necesidad de instalar Python.

3. Tecnologías y herramientas utilizadas

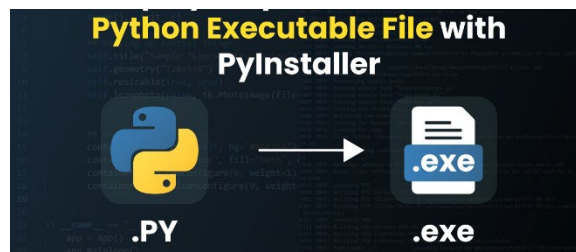
- **Python 3.13.5:** Lenguaje de programación principal, elegido por su simplicidad y amplia comunidad.



- **Tkinter:** Biblioteca para creación de interfaces gráficas.



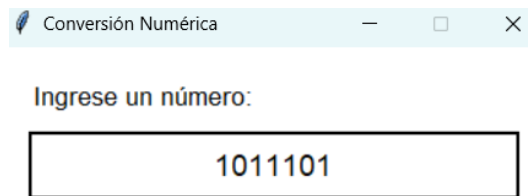
- **ttk (Tkinter themed widgets):** Para aplicar estilos modernos y consistentes a botones, etiquetas y cajas de texto.
- **PyInstaller:** Para generar un ejecutable .exe independiente del entorno Python.



- **Pillow (opcional):** Para convertir imágenes a íconos .ico y personalizar la apariencia del .exe.

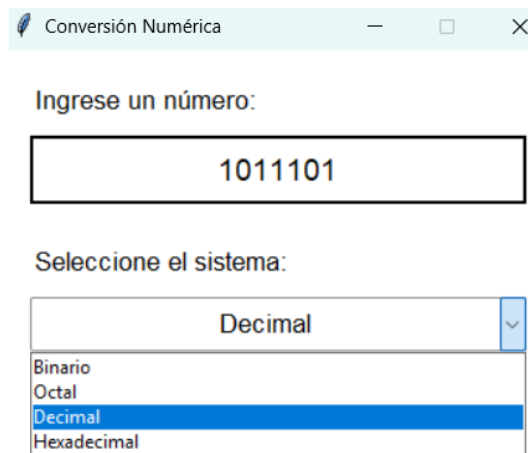
4. Funcionalidades principales

- **Entrada de número:** Permite ingresar un número en el sistema seleccionado.



The screenshot shows a window titled 'Conversión Numérica' with standard window controls. Below the title bar, the text 'Ingrese un número:' is displayed. A text input field contains the binary number '1011101'.

- **Selección del sistema:** Cuatro opciones: Binario, Octal, Decimal y Hexadecimal.



This screenshot shows the same application window with the input field containing '1011101'. Below it, the text 'Seleccione el sistema:' is followed by a dropdown menu. The dropdown is open, showing four options: 'Binario', 'Octal', 'Decimal' (which is highlighted in blue), and 'Hexadecimal'.

- **Botones principales:**
 - **Calcular:** Realiza la conversión a los otros sistemas numéricos.
 - **Limpiar:** Borra la entrada y los resultados.
 - **Copiar:** Copia los resultados al portapapeles.
 - **Salir:** Cierra la aplicación.



The image displays four rectangular buttons stacked vertically. From top to bottom, they are: a green button labeled 'Calcular', a red button labeled 'Limpiar', a blue button labeled 'Copiar', and a purple button labeled 'Salir'.

- **Validación automática:** El programa verifica que los números ingresados sean válidos para el sistema seleccionado.

The screenshot shows a window titled 'Conversión Numérica'. It contains the following elements:

- A label 'Ingrese un número:' followed by a text input field containing '01011101'.
- A label 'Seleccione el sistema:' followed by a dropdown menu currently set to 'Hexadecimal'.
- Four buttons stacked vertically: 'Calcular' (green), 'Limpiar' (red), 'Copiar' (blue), and 'Salir' (purple).
- A label 'Resultados:' followed by a text area displaying the converted values:

Binario:	0b100000001000100010000001
Octal:	0o100210401
Decimal:	16847105
Hexadecimal:	0X1011101

- **Resultados en tiempo real:** Muestra los valores convertidos en los cuatro sistemas numéricos.
- **Interfaz centrada y moderna:** La ventana se abre centrada en pantalla y utiliza estilos visuales consistentes.

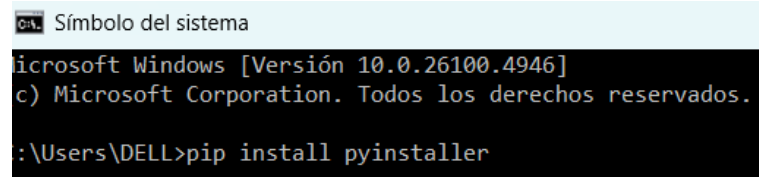
5. Instrucciones de instalación y uso

5.1 Requisitos

- Python 3.x (si se ejecuta el script .py).
- Librerías: tkinter (incluida con Python).
- Para generar un ejecutable .exe:

```
pip install pyinstaller
```

Ejemplo:



La imagen muestra una captura de pantalla de una terminal de Windows. En la parte superior, hay una barra azul con el texto "Símbolo del sistema". Debajo, el texto de la terminal indica: "Microsoft Windows [Versión 10.0.26100.4946]", "Copyright (c) Microsoft Corporation. Todos los derechos reservados.", y la línea de comandos "C:\Users\DELL>pip install pyinstaller".

5.2 Uso del script .py

1. Descargar el archivo Conversion_Sistemas_Numericos.py.
2. Abrirlo en Python IDLE o desde la terminal:

```
python Conversion_Sistemas_Numericos.py
```

3. Interactuar con la aplicación:
 - Ingresar el número a convertir.
 - Seleccionar el sistema de origen.
 - Presionar Calcular para obtener resultados.
 - Usar Copiar para enviar resultados al portapapeles.
 - Usar Limpiar para borrar datos.
 - Presionar Salir para cerrar la ventana.

5.3 Uso del ejecutable .exe

1. Ejecutar Conversion_Sistemas_Numericos.exe (no requiere Python).

```
C:\WINDOWS\system32\cmd.exe
1186 INFO: Building EXE because EXE-00.toc is non existent
1186 INFO: Building EXE from EXE-00.toc
1186 INFO: Copying bootloader EXE to D:\Familia\Diego\ITECSUR\Tareas Itecsur\Deberes Electronica\Examen_Convesion_Sistemas_Numericos\dist\Conversion_Sistemas_Numericos.exe
1284 INFO: Copying icon to EXE
1341 INFO: Copying 0 resources to EXE
1341 INFO: Embedding manifest in EXE
1350 INFO: Appending PKG archive to EXE
1441 INFO: Fixing EXE headers
2203 INFO: Building EXE from EXE-00.toc completed successfully.
2224 INFO: Build complete! The results are available in: D:\Familia\Diego\ITECSUR\Tareas Itecsur\Deberes Electronica\Examen_Convesion_Sistemas_Numericos\dist

:\Familia\Diego\ITECSUR\Tareas Itecsur\Deberes Electronica\Examen_Convesion_Sistemas_Numericos>REM 5'0Ãã Mensaje de finalizaciÃ³n

:\Familia\Diego\ITECSUR\Tareas Itecsur\Deberes Electronica\Examen_Convesion_Sistemas_Numericos>echo =====
=====

:\Familia\Diego\ITECSUR\Tareas Itecsur\Deberes Electronica\Examen_Convesion_Sistemas_Numericos>echo .exe generado correctamente en la carpeta dist\
.exe generado correctamente en la carpeta dist\

:\Familia\Diego\ITECSUR\Tareas Itecsur\Deberes Electronica\Examen_Convesion_Sistemas_Numericos>echo =====
=====

:\Familia\Diego\ITECSUR\Tareas Itecsur\Deberes Electronica\Examen_Convesion_Sistemas_Numericos>pause
Presione una tecla para continuar . . .
```

2. La ventana se abrirá centrada en pantalla con el mismo diseño que el script .py.
3. Seguir los pasos de uso como en el script .py.
4. Para personalizar el ícono, incluir mi_ícono.ico en la misma carpeta antes de generar el .exe.



5.4 Generación del ejecutable con PyInstaller

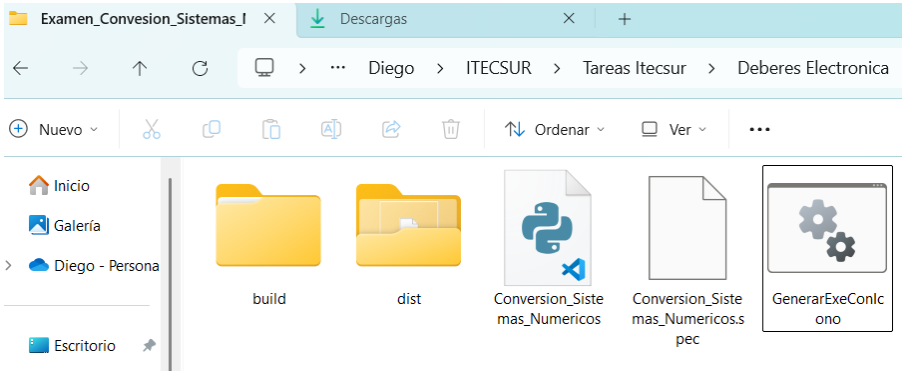
- 1. Abrir CMD en la carpeta del proyecto.
- 2. Renombrar el archivo (opcional) para evitar espacios:

```
rename "Conversion_Sistemas Numericos.py" Conversion_Sistemas_Numericos.py
```

- 3. Ejecutar PyInstaller:

```
pyinstaller --onefile --windowed --icon=mi_icono.ico Conversion_Sistemas_Numericos.py
```

- 4. El .exe se encontrará en la carpeta dist\.



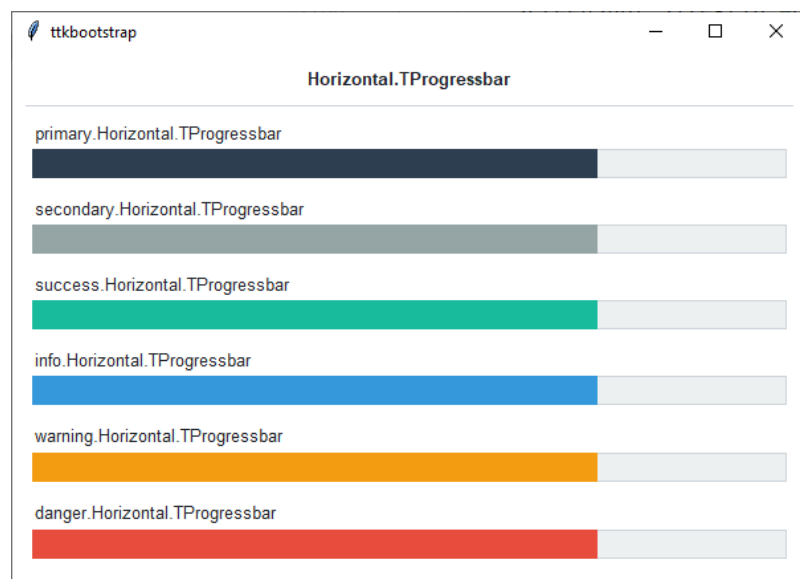
6. Ejemplos de uso

Entrada	Sistema de origen	Resultados
1010	Binario	Decimal: 10, Octal: 12, Hexadecimal: A
15	Decimal	Binario: 1111, Octal: 17, Hexadecimal: F
1F	Hexadecimal	Binario: 11111, Decimal: 31, Octal: 37

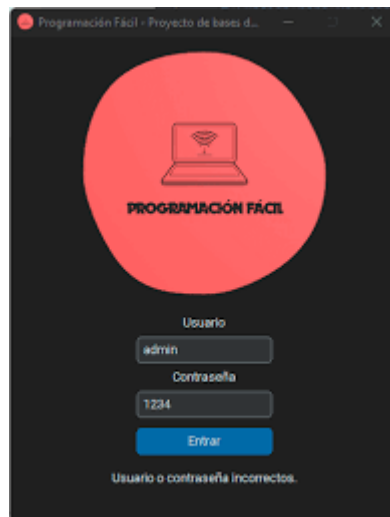
7. Posibles mejoras futuras

- Añadir más sistemas de numeración (ej: base 3, base 5).
- Exportar resultados a un archivo CSV o TXT.
- Soporte multilenguaje en la interfaz.
- Mejorar la estética con librerías como **ttkbootstrap** o **customtkinter**.

Ejemplo de ventana con la librería **ttkbootstrap**:



Ejemplo de ventana con la librería **customtkinter**:



8. Glosario de conceptos

- Binario: Sistema de base 2, utiliza solo los dígitos 0 y 1.
- Octal: Sistema de base 8, utiliza los dígitos del 0 al 7.
- Decimal: Sistema de base 10, utilizado comúnmente en la vida diaria.
- Hexadecimal: Sistema de base 16, utiliza dígitos del 0 al 9 y letras A-F.