

# Manual de Usuario: **Analizador Léxico**

Carlos Enrique de León de la Cruz – 1626423

Ana Massielle Coti Rodas – 2477523

## 1. ¿Qué es el Analizador Léxico?

El **Analizador Léxico** es una herramienta que examina un archivo de texto en busca de:

- **Errores léxicos:** Caracteres no permitidos o mal formados.
- **Tokens:** Elementos válidos como palabras reservadas, operadores, números, y más.

La aplicación te muestra el contenido del archivo, los errores detectados, y la clasificación de los tokens encontrados.

## 2. Requisitos para usar el programa

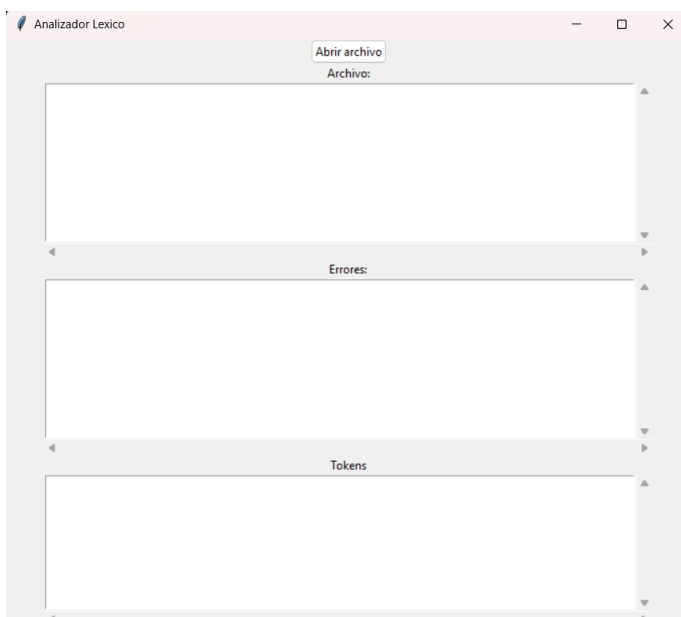
Antes de usar el programa, asegúrate de contar con:

- **Python 3** instalado en tu equipo.
- Las librerías y archivos necesarios para el programa (Tkinter).

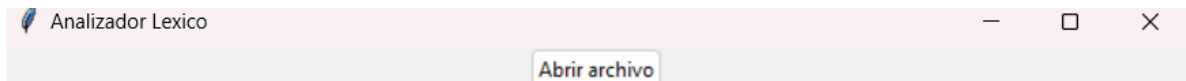
## 3. Pasos para usar el Analizador Léxico

### 3.1. Abrir el programa

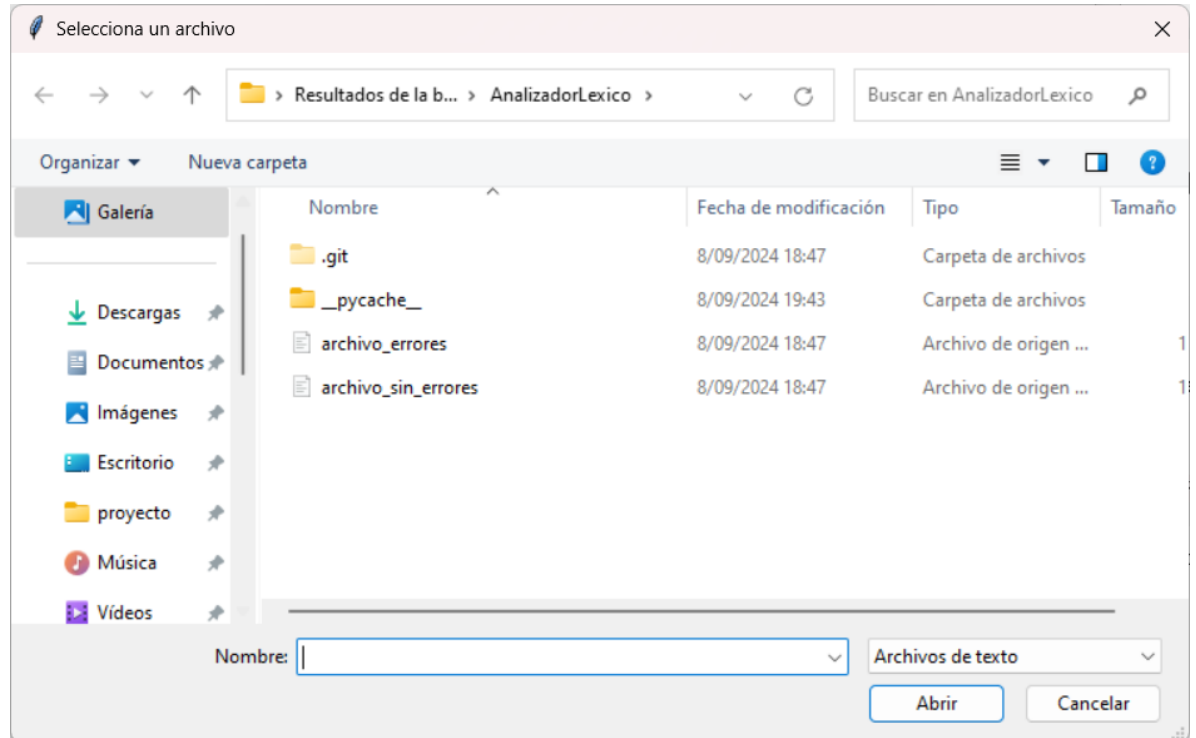
1. **Ejecuta el programa.** Al abrirlo, verás una ventana con varias secciones y el botón **Abrir archivo** en la parte superior.



2. **Presiona el botón "Abrir archivo".** Se abrirá una ventana para que selecciones el archivo de texto que deseas analizar. Navega hasta el archivo y selecciónalo.

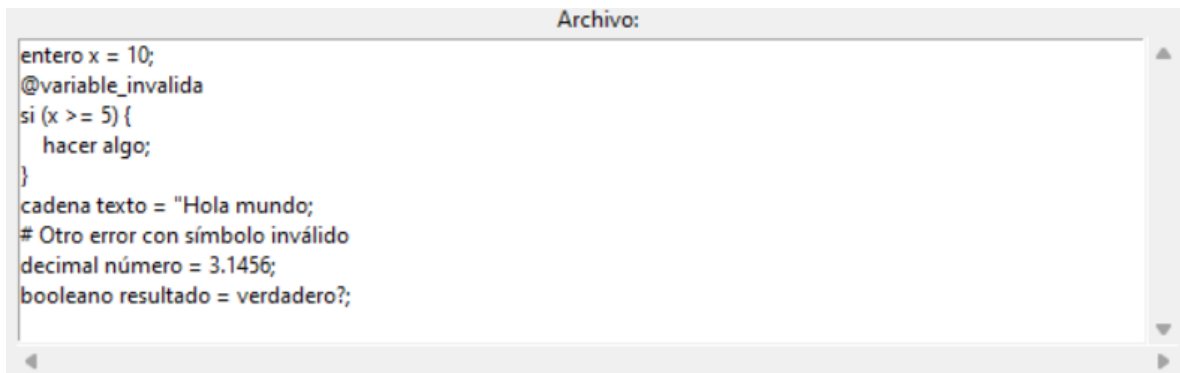


3. El contenido del archivo que seleccionaste aparecerá en la sección **Archivo**.



### 3.2. Analizar el archivo

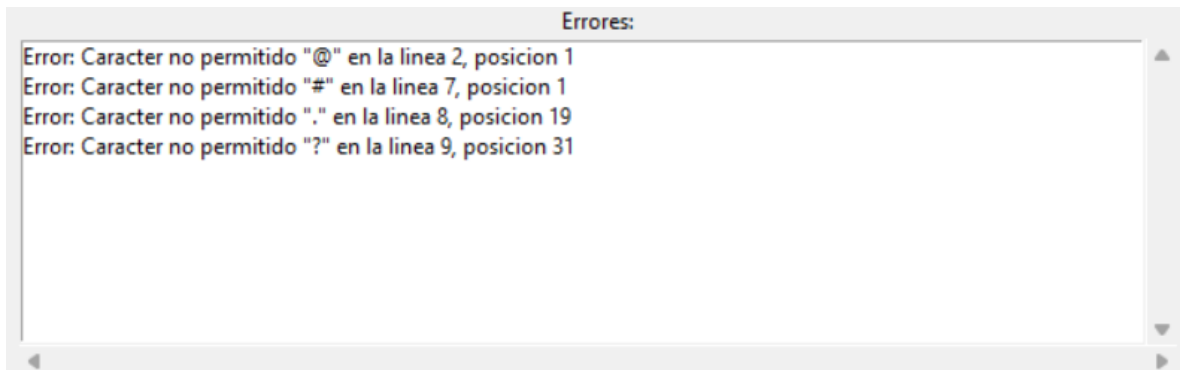
- El programa realiza automáticamente tres tareas importantes:
  1. **Mostrar el contenido del archivo** en la primera caja de texto (bajo **Archivo**).
  2. **Buscar errores léxicos** en el archivo. Si se encuentra un error, se mostrará en la segunda caja de texto (bajo **Errores**). Aquí verás detalles como el carácter no permitido, la línea donde se encuentra y la posición exacta dentro de esa línea.
  3. **Clasificar los tokens** válidos, como palabras reservadas, operadores, y números. Esta información se muestra en la tercera caja de texto (bajo **Tokens**).



```
Archivo:
entero x = 10;
@variable_invalida
si (x >= 5) {
    hacer algo;
}
cadena texto = "Hola mundo;
# Otro error con símbolo inválido
decimal número = 3.1456;
booleano resultado = verdadero?;
```

### 3.3. Interpretar los resultados

- **Errores:** Si el archivo contiene caracteres no válidos o errores, los verás en la sección de **Errores**. Por ejemplo, si hay un símbolo no permitido.



```
Errores:
Error: Caracter no permitido "@" en la línea 2, posicion 1
Error: Caracter no permitido "#" en la línea 7, posicion 1
Error: Caracter no permitido "." en la línea 8, posicion 19
Error: Caracter no permitido "?" en la línea 9, posicion 31
```

- **Tokens:** Los tokens son los elementos válidos que el programa detecta. Estos incluyen palabras como entero, si, o símbolos como = y >=. La sección de **Tokens** te mostrará cuántos y qué tipos de tokens ha identificado el programa



### 3.4. Cerrar el programa

Cuando termines de analizar el archivo, simplemente cierra la ventana para salir del programa.