

Analizador Léxico Sintáctico

Manual Técnico para Usuarios

Desarrollador



Juan Cardona



**UNIVERSIDAD
DE ANTIOQUIA**

1 8 0 3

Facultad de Ingeniería

Departamento de ingeniería de sistemas

Teoría de Lenguajes

2021-1

Aspectos Técnicos

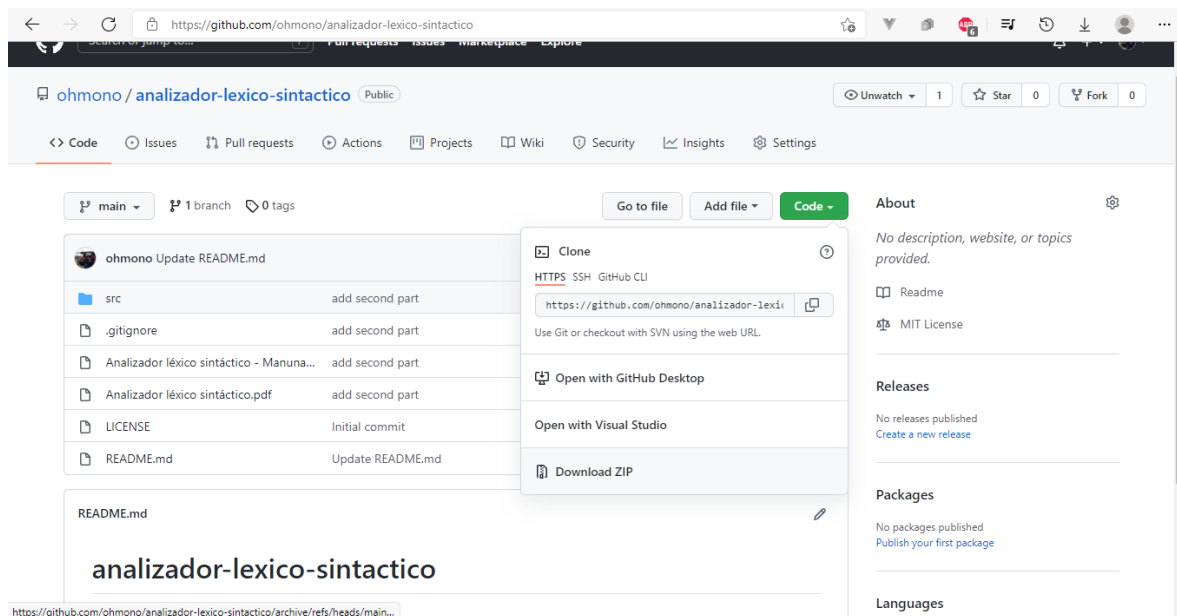
Este documento es el manual de técnico del **analizador léxico sintáctico** desarrollado por Juan Cardona.

El lenguaje de programación que utilice fue **Python 3.8.5** en el ambiente de desarrollo **Visual Studio Code** importando las librerías **PyQt5** (diseño de ambiente gráfico), **webbrowser** (dirección de url), y **re** (manejo para expresiones regulares).

No es necesario tener ningún programa instalado previo a la ejecución del **analizador léxico sintáctico**, pero si se quiere hacer alguna modificación es necesario tener instalado los programas y librerías anteriormente listados.

Obtener el Analizador Léxico Sintáctico

Para obtener el **Analizador Léxico Sintáctico** ingrese al repositorio de GitHub:
<https://github.com/ohmono/analizador-lexico>



En el botón Code, al dar clic, seleccione la opción Download ZIP, para obtener el archivo con el programa.

Descomprima el archivo e ingrese a la carpeta analizador-léxico

Nombre	Fecha de modificación	Tipo	Tamaño
src	9/05/2021 8:37 p. m.	Carpeta de archivos	
Analizador léxico - Manual técnico para usuarios.pdf	10/05/2021 6:34 p. m.	Microsoft Edge P...	363 KB
Analizador léxico.pdf	10/05/2021 5:22 p. m.	Microsoft Edge P...	378 KB
LICENSE	9/05/2021 8:39 p. m.	Archivo	2 KB
README.md	9/05/2021 8:39 p. m.	Archivo de origen ...	1 KB

Para acceder a el archivo ejecutable, ingrese en la carpeta **src**

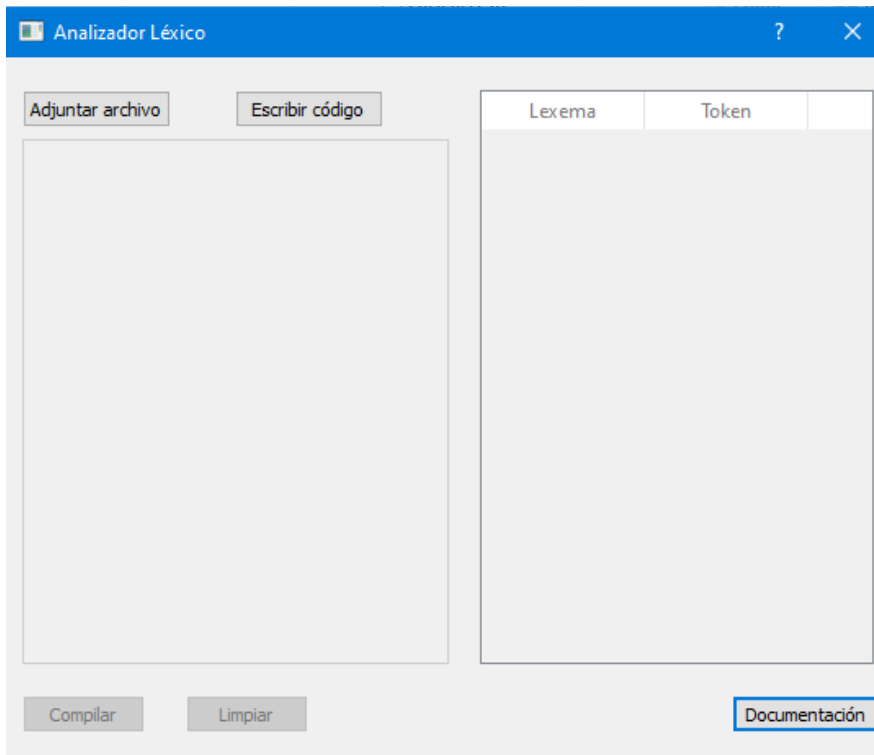
Nombre	Fecha de modificación	Tipo	Tamaño
dist	9/05/2021 8:23 p. m.	Carpeta de archivos	
app.py	9/05/2021 8:19 p. m.	Python File	1 KB
lexic.py	9/05/2021 8:22 p. m.	Python File	4 KB
prueba.txt	9/05/2021 7:18 p. m.	Documento de te...	1 KB
ui.py	9/05/2021 8:19 p. m.	Python File	5 KB

Allí están los archivos de la lógica del programa y la carpeta **dist** con el archivo ejecutable

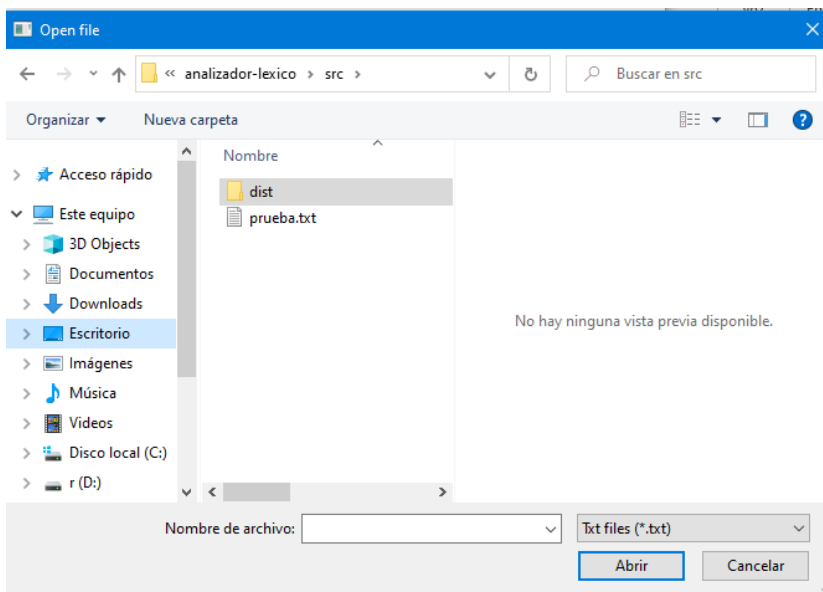
Nombre	Fecha de modificación	Tipo	Tamaño
app.exe	9/05/2021 8:24 p. m.	Aplicación	35.259 KB

El archivo **app.exe** corresponde al **analizador léxico sintáctico**.

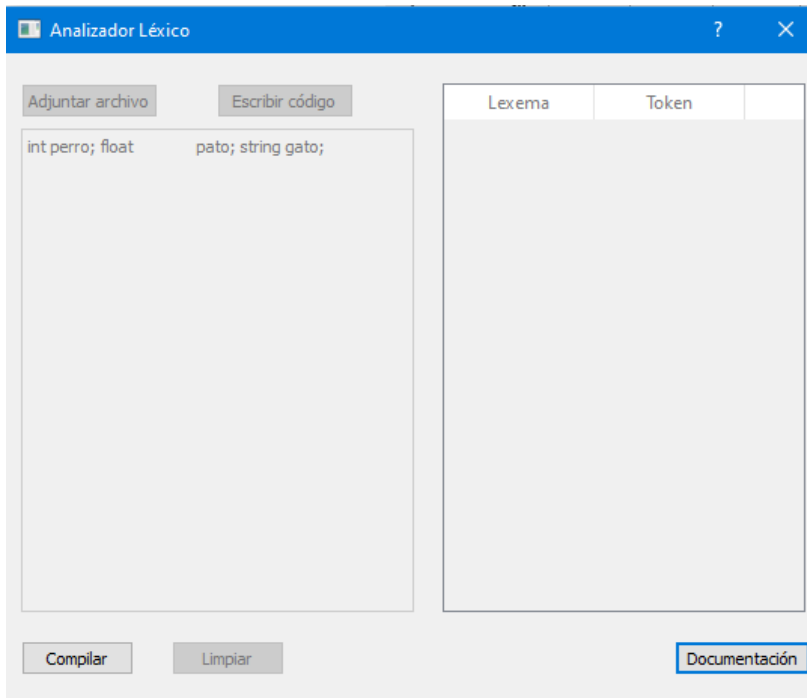
Usar el Analizador Léxico Sintáctico



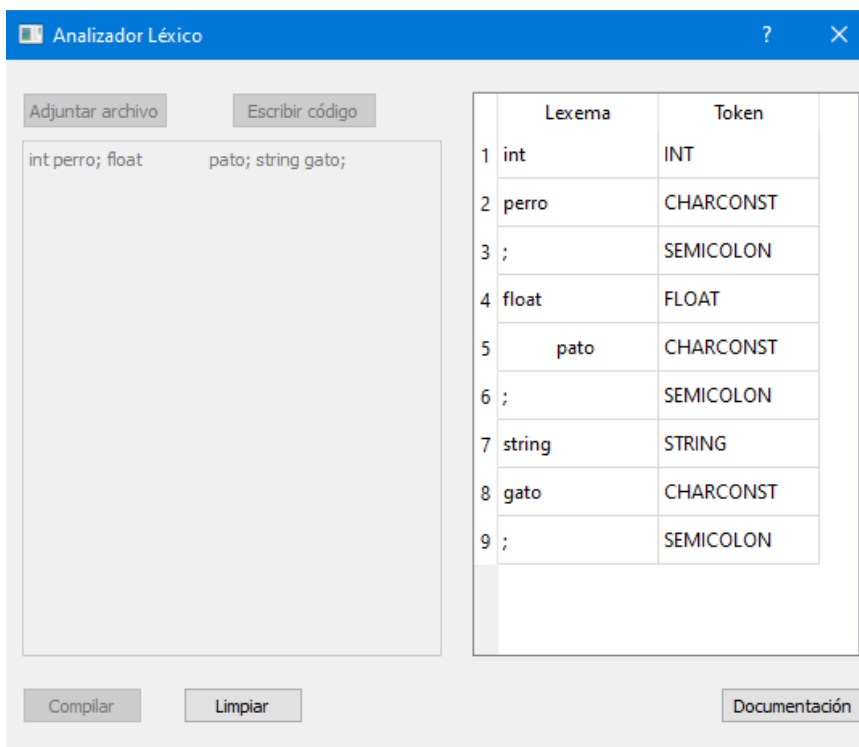
Con el botón **Adjuntar archivo** podemos adjuntar un archivo .txt para analizar



Luego de seleccionar un archivo .txt vemos que se despliega su contenido en el programa

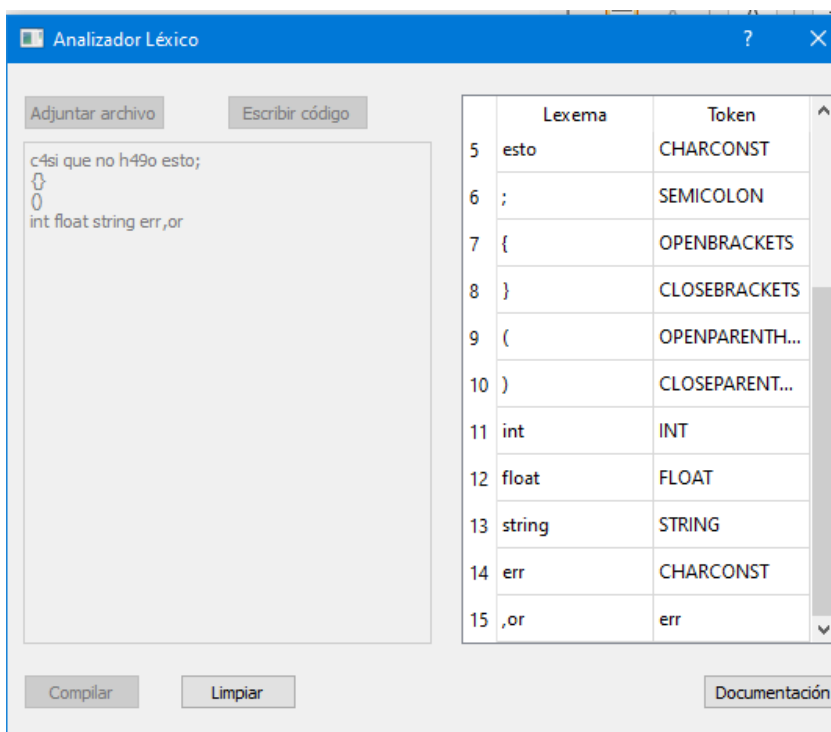
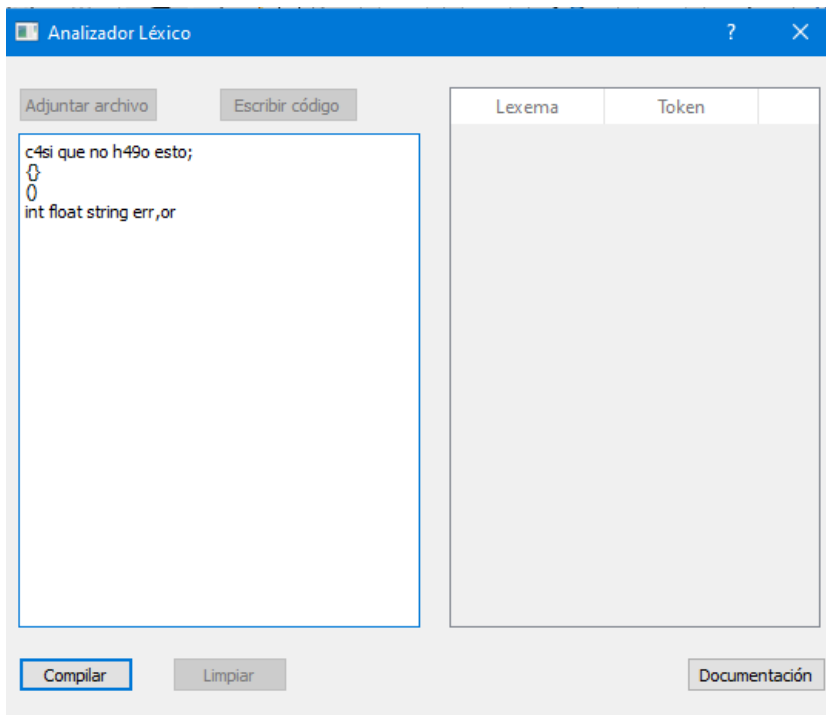


Y al darle en el botón **compilar** se nos mostrara una lista con los lexemas y sus tokens correspondientes



El botón **Limpiar** nos permite reiniciar la interfaz para ser usada de nuevo.

Con el botón **Escribir código** podemos ingresar manualmente el código a analizar



Por último, el botón **Documentación** nos direcciona al repositorio de GitHub donde se aloja el **Analizador Léxico Sintáctico**.

Lista de lexemas, tokens y terminales que reconoce y asigna el **Analizador Léxico Sintáctico**:

Lexema	Token	Terminales
esle	ELSE	ELSE
if	IF	IF
int	INT	T
float	FLOAT	
bool	BOOL	
string	STRING	
L+	CHARCONST	V
L+(L+D)*	IDENTIFY	
D+	INTCONST	D
D+,D+	FLOATCONST	
+	PLUS	O
-	MINUS	
*	MULTIPLICATION	
/	DIVISION	
=	ASIGNATION	=
==	BOOLCOPARISON	B
<	MINOR	
<=	MINOREQUAL	
<>	DIFFERENT	
>	HIGHER	
>=	HIGHEREQUAL	
(OPENPARENTHESIS	(
)	CLOSEPARENTHESIS)
{	OPENBRACKETS	{
}	CLOSEBRACKETS	}
;	SEMICOLON	;

- D (cualquier caracter numérico o dígito)
- L (cualquier carácter alfabético, en minúscula o mayúscula, o letra)

Gramática con la que se identifica si las secuencias de tokens identificadas son válidas:

1. $\langle S \rangle \rightarrow TV\langle A \rangle;\langle S \rangle$
2. $\langle S \rangle \rightarrow V=\langle X \rangle;\langle S \rangle$
3. $\langle S \rangle \rightarrow IF(\langle X \rangle B\langle X \rangle)\{\langle S \rangle\}ELSE\{\langle S \rangle\};\langle S \rangle$
4. $\langle S \rangle \rightarrow \lambda$
5. $\langle A \rangle \rightarrow =\langle B \rangle$
6. $\langle A \rangle \rightarrow \lambda$
7. $\langle B \rangle \rightarrow V$
8. $\langle B \rangle \rightarrow D$
9. $\langle X \rangle \rightarrow V$
10. $\langle X \rangle \rightarrow D$
11. $\langle X \rangle \rightarrow (\langle X \rangle \langle Z \rangle \langle X \rangle)$
12. $\langle Z \rangle \rightarrow O$
13. $\langle Z \rangle \rightarrow B$

Bibliografía

- Flórez Rueda, R. (2010). Introducción a los compiladores. Medellín, Colombia: Universidad de Antioquia
- Regular Expression Quick Reference. Retrieved May 10, 2021, from http://regexrenamer.sourceforge.net/help/regex_quickref.html
- PyQt5 - Qt for Python. Retrieved May 10, 2021, from <https://doc.qt.io/qtforpython/>