

MANUAL DE USUARIO

PLATAFORMA DE SALUD PROYECTO DEL CURSO IPC2: APLICACIÓN WEB CON JAVA Y ANGULAR

¿QUÉ ES ESTE PROYECTO?

¡Bienvenido al Manual de Usuario del proyecto del Analizador Léxico Simulado de Python! Este proyecto ofrece una implementación sencilla de un analizador léxico que emula el comportamiento del lenguaje de programación Python. El proyecto se ha dividido en dos partes, el backend y el frontend, para facilitar su comprensión y uso. A continuación, se detallan los pasos para poner en marcha este analizador léxico:

1. Requisitos Previos:

Antes de comenzar, asegúrese de tener instalados los siguientes componentes:

Node.js:

Puede descargarse e instalarlo desde nodejs.org.

2. Iniciar el Backend:

El backend del proyecto se ha desarrollado en Java utilizando librerías como JakartaEE para crear una API que atenderá las solicitudes de los clientes.

3. Iniciar el Frontend:

La interfaz de usuario del proyecto se ha creado utilizando el framework JavaScript Vue.js. Para ejecutarla, seguir estos pasos:

- Descargar el proyecto y descomprimirlo en su ubicación preferida.
- Abrir una terminal y navegar hasta la carpeta del proyecto frontend.
- Ejecute el comando `npm install` para instalar las dependencias necesarias.
- Una vez completada la instalación, use `npm run dev` para iniciar el servidor de desarrollo.
- Abrir el navegador web y navegar a la dirección proporcionada por el servidor de desarrollo.

4. Interacción con el Analizador Léxico:

Una vez que el frontend esté en funcionamiento, tendrá acceso a la interfaz del analizador léxico simulado. Ingrese el código en el área designada y este será automáticamente reconocido. El analizador procesa el código y mostrará los resultados en la interfaz.

INTERFAZ Y SUS PARTES

Analizador léxico parser-py - Lenguajes Formales SS 2023

Escribe el código aquí:

Limpiar

El cursor está en: L1:C1

```
def if "cadena " : ( 4 + 5 )
mi_id_
#comentario
while
for
>
<
'cadena en comillas simples'

#un error se puede presentar como la siguiente línea

" mi cadena que no cierra
```

LEER ARCHIVO

ANALIZAR

Tokens validos encontrados:

Lexema	Tipo	Patrón	Línea	Columna
def	KEYWORD	def	1	4
if	KEYWORD	if	1	7
"cadena "	STRING	"*"	1	17
:	PUNCTUATION_COLON	^:	1	19
(PUNCTUATION_PARENTHESIS_OPEN	(1	21

Items per page: 5 1-5 of 28 |< < > >|


Errores / Tokens invalidos:

→ EDITOR DE TÉXTO

La aplicación proporciona un pequeño y simple editor de texto con resaltado de sintaxis a medida de que se va escribiendo. En la parte superior muestra la información del cursor y su posición. También se proporciona un pequeño botón para limpiar el editor y comenzar de nuevo.

Escribe el código aquí:

 Limpiar

El cursor está en: L2:C6 

```
def if "cadena " : ( 4 + 5 )
mi_id_|
#comentario
while
for
≥
<
'cadena en comillas simples'












#un error se puede presentar como la siguiente línea

" mi cadena que no cierra
```

→ TABLAS DE INFORMACIÓN DE TOKENS

Este componente muestra información en formato de tabla acerca de los tokens que sean encontrados en el código así como los errores que se detecten.

También a través de un botón, se permite la generación de un gráfico del lexema detectado

 Tokens validos encontrados: 					
Lexema	Tipo	Patrón	Linea	Columna	
def	KEYWORD	def	1	4	
	SPACE		1	4	
if	KEYWORD	if	1	7	
	SPACE		1	7	
"cadena "	STRING	"*"	1	17	
Items per page: <div>5 ▾</div> 1-5 of 36    					

🔍 Tokens validos encontrados:



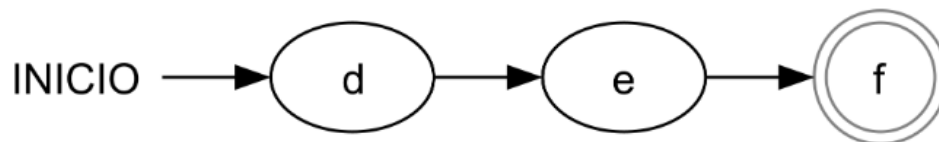
⚠ Errores / Tokens invalidos:



Lexema	Tipo	Linea	Columna	
" mi cadena que no cierra	INVALID_UNIDENTIFIED	12	1	

Items per page: 5 1-1 of 1 < >

Gráfico para: def



→ ACCIONES DE CÓDIGO

Se proporciona dos acciones que pueden ser realizadas sobre el editor de texto, que son, la carga de archivos para ingresar código por medio de un archivo y su contenido, y el análisis manual del mismo en caso haya un problema



LEER ARCHIVO



ANALIZAR

→ CARGA DE ARCHIVOS

Al seleccionar la opción de *Leer Archivo*, se mostrará un diálogo pidiendo que se seleccione desde la máquina local un archivo en formato de texto plano o con extensión .py para poder importar su contenido al editor de texto y analizarlo



Leer desde archivo



Aquí puedes seleccionar un archivo de texto plano para leerlo y analizarlo



Archivo de texto

d1.png

 **Tipo de archivo detectado:**

 image/png

 **Ocurrió un error al leer el archivo:**

El tipo de archivo parece no ser valido, solo se permiten archivos de texto plano.

USAR ARCHIVO

→ CONFIGURACIÓN

En la esquina superior derecha se proporciona un pequeño menú con dos opciones, que son, si se desea mostrar u ocultar los tokens de tipo *espacio* o de tipo *nueva línea*.

