

# INDICE

---

Introducción .....	2
Objetivos .....	2
Conocimientos Previos .....	2
Requerimientos.....	2
Descripción de variables estáticas .....	3
Descripción de métodos .....	3
Descripción de funciones .....	4
Descripción de Tokens Permitidos .....	4
Expresiones regulares utilizadas.....	5
Autómata Finito Determinista .....	5
Gramática libre de contexto .....	6

# Introducción

---

El presente documento describe los aspectos técnicos informáticos del analizador léxico y sintáctico para la generación de páginas web, de manera que cualquier técnico informático que posea los conocimientos necesarios, pueda entender y comprender la lógica dentro del programa, para así poder darle mantenimiento y de ser necesario, actualizar el mismo.

## Objetivos

---

Instruir al lector para el uso adecuado del código del programa, así como dar a conocer la lógica detrás del programa, de manera que se el lector la pueda comprender para facilitar la manipulación del código del programa.

## Conocimientos Previos

---

Los conocimientos que deberán tener las personas que manejen el programa son:

- Conocimiento en Python
- Manejo de POO
- Manejo tkinter
- Expresiones regulares
- Autómatas Finitos Determinísticos

## Requerimientos

---

- Python 3.8 o posterior
- 512Mb RAM
- Intel Pentium III 800MHz o posterior
- Windows xp o posterior, GNU Linux o Mac Os

## Descripción de variables estáticas

---

Variable	Clase	Descripción
menu	main	Variable que representa todos los métodos de la clase Menús.
controles	instancias	Variable que representa los controles que se crean del html.
propiedades	instancias	Variable que representa las propiedades para generar el css.
colocaciones	instancias	Variable que representa la colocación de los controles para el html.
token	instancias	Variable que representa todos los métodos de la clase Token.
error	instancias	Variable que representa todos los métodos de la clase Error.
root	menus	Variable que representa la ventana principal del programa.
analizadorLexico	menus	Variable que representa todos los métodos de la clase analizador.
analizadorSintactico	menus	Variable que representa todos los métodos de la clase sintáctico.
generador	sintactico	Variable que representa todos los métodos de la clase generador.
contadorTokens	sintactico	Variable que contabiliza el token verificado.

## Descripción de métodos

---

Método	Clase	Descripción
AbrirArchivo	Menus	Abre un archivo en formato "gpw" y opera la carga de datos en el sistema para almacenarlos en un cuadro de texto.
AbrirManual	Menus	Abre un manual desde el frontend, en formato pdf.
AnalizarTexto	Menus	Realiza un análisis léxico de los datos que se encuentran ingresados en el cuadro de texto.
Nuevo	Menus	Guarda o no lo que se encuentra en el cuadro de texto, borra la información que se encuentra en el mismo.
Guardar	Menus	Guarda un documento previamente cargado en el cuadro de texto.
GuardarComo	Menus	Guarda el texto que se encuentra en el cuadro de texto con un nombre elegido por el usuario, en formato "lfp"
InsertarErrores	Menus	Inserta los errores del último análisis léxico a una tabla al frontend.
InsertarTokens	Menus	Inserta los tokens del último análisis léxico a una tabla al frontend.
Informacion	Menus	Muestra la información del desarrollador del programa.
Analizar	Analizador	Realiza un análisis léxico del texto plano que recibe.
NuevoToken	Token	Agrega un nuevo token a la lista de tokens.
NuevoError	Error	Agrega un nuevo error a la lista de errores.
Sintactico	Sintactico	Realiza un análisis sintáctico del listado de tokens que se generaron del análisis léxico.
ReporteHTML	generadorHtml	Genera un html con los datos que fueron recibidos del análisis léxico y sintáctico.
ReporteCSS	generadorHtml	Genera el css con los datos que fueron recibidos del análisis léxico y sintáctico.

## Descripción de funciones

---

Función	Clase	Descripción
VentanaErrores	Menus	Retorna una ventana con botones y una tabla para la gestión de los errores que se encuentran en el listado de errores.
VentanaTokens	Menus	Retorna una ventana con botones y una tabla para la gestión de los tokens que se encuentran en el listado de tokens.
EditorVentana	Menus	Retorna la posición que debe de tener una ventana con determinadas características, para que esta quede centrada en pantalla.
MostrarEliminarVentana	Menus	Muestra una pantalla oculta y elimina una ventana.
OcultarVentana	Menus	Ocultar una ventana.
EliminarVentana	Menus	Elimina una ventana.

## Descripción de Tokens Permitidos

---

Tipo	Patrón	Lexema
Etiqueta	Palabra reservada "Etiqueta"	Etiqueta
Boton	Palabra reservada "Boton"	Boton
Check	Palabra reservada "Check"	Check
RadioBoton	Palabra reservada "RadioBoton"	RadioBoton
Texto	Palabra reservada "Texto"	Texto
AreaTexto	Palabra reservada "AreaTexto"	AreaTexto
Clave	Palabra reservada "Clave"	Clave
Contenedor	Palabra reservada "Contenedor"	Contenedor
Apertura	Caracter <	<
Cierre	Caracter >	>
Admiracion	Caracter !	!
Guion	Caracter -	-
Diagonal	Caracter /	/
Asterisco	Caracter *	*
Punto y coma	Caracter ;	;
Punto	Caracter .	.
Paréntesis Abierto	Caracter (	(
Paréntesis Cerrado	Caracter )	)
Entero	Secuencia de 1 o más dígitos	36
Decimal	Secuencia de 1 o más dígitos, seguidos por punto, seguido por 1 o más dígitos	187.32
Cadena de texto	Secuencia de 1 o más letras del abecedario, incluyendo tildes, dígitos, espacios, caracteres especiales como tabulaciones y símbolos.	Tengo \$4, compraré un m&m.

Tipo	Patrón	Lexema
Comentario linea	Secuencia de 2 diagonales, seguidos por cualquier carácter, símbolo o signo.	//Comentario #1
Comentario lineas	Secuencia de una diagona, seguido por un asterisco, seguido por cualquier carácter, símbolo, signo o salto de línea, seguido por un asterisco y por último una diagonal.	/* Comentario con #n cantidad de líneas, símbolos como \$, ) */

## Expresiones regulares utilizadas

Caracteres = [A-Za-z]+

Entero = [0-9]+

Decimal = [0-9]+[\.][0-9]+

## Autómata Finito Determinista

Pasos para obtener el autómata:

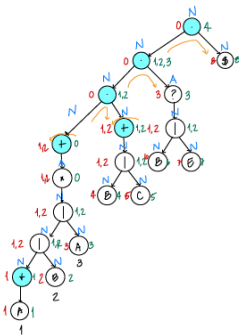


Tabla de Siguietes:

Est	A	B	C	D	E
S0 =	{<}	{0,1}	{car...}	-	-
S1 =	-	{<}	{car}	-	-
S2 =	-	{<}	{car}	-	-
S3 =	{<}	-	{car...}	-	-
S4 =	-	{car}	-	-	-
S5 =	-	-	{car...}	{,}	-
S6 =	-	-	-	{,}	-

Est	A	B	C	D	E
S0 =	S0	S1	S2	-	-
S1 =	-	S0	S1	-	-
S2 =	-	S0	-	S0	-
S3 =	S0	-	S2	-	-
S4 =	-	S0	-	-	-
S5 =	-	-	-	S0	-
S6 =	-	-	-	-	S0

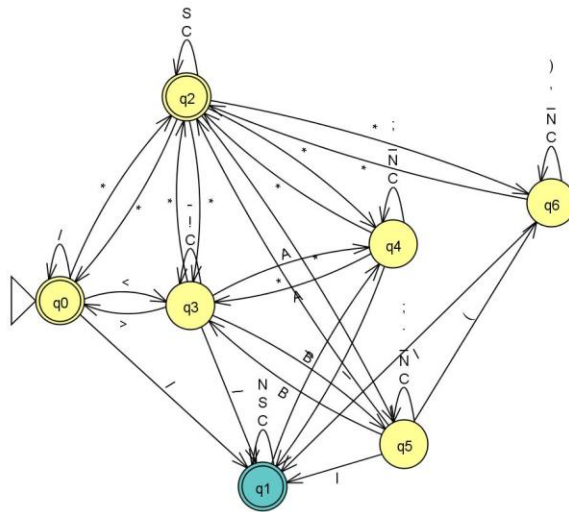
C = Caracter

E = Entero

D = Decimal

A = Control

B = Propiedad



## Gramática libre de contexto

La siguiente gramática se utiliza para realizar el análisis sintáctico:

```
SINTACTICO = LISTA_INSTRUCCIONES  
            | COMENTARIO
```

```
COMENTARIO = // Comentario  
            | /* Comentario */
```

```
LISTA_INSTRUCCIONES = INSTRUCCION  
                    | EOF
```

```
INSTRUCCION = <!-- CONTROLES  
              | <!-- PROPIEDADES  
              | <!-- COLOCACION
```

```
CONTROLES = Controles LISTA_CONTROLES -->
```

```
LISTA_CONTROLES = CONTROL ID ; LISTA_CONTROLES  
    | COMENTARIO LISTA_CONTROLES  
    | Controles
```

```
PROPIEDADES = propiedades LISTA_PROPIEDADES -->
```

```
LISTA_PROPIEDADES = CONTROL . PROPIEDAD ( VALORES ; LISTA_PROPIEDADES  
    | COMENTARIO LISTA_PROPIEDADES  
    | propiedades
```

```
VALORES = VALOR VALORES  
    | VALOR )
```

```
COLOCACION = Colocacion LISTA_COLOCACIONES -->
```

```
LISTA_COLOCACIONES = ID . PROPIEDAD ( VALORES ; LISTA_COLOCACIONES  
    | COMENTARIO LISTA_PROPIEDADES  
    | Colocacion
```

```
TIPOS = ID  
    | this
```

```
TIPOS2 = setPosicion  
    | add
```