

Manual Técnico

Proyecto 2



Lenguajes Formales y de Programación
Francisco Daniel Peruch de León
Carné: 202100639

En este documento se presenta los detalles técnicos de la aplicación, el cual fue desarrollado en el lenguaje Python.

Se crearon diferentes ficheros de Python para poder crear y manejar las diferentes funcionalidades de la aplicación, los cuales se detallan a continuación:

Ventana:

En este fichero se desarrollaron las diferentes funciones para poder crear la interfaz gráfica de la aplicación, la cual fue desarrollada utilizando la librería Tkinter.

Usando las distintas funciones que provee la librería Tkinter se crearon distintos botones y área de texto que fueron de utilidad para indicar al usuario de una manera agradable las funciones de la aplicación. Las funciones creadas en este fichero son las siguientes:

- *limpiar_texto()*: Esta función fue utilizada para poder limpiar el área de texto si tiene alguna cadena texto escrita en el mismo y fue utilizado en el botón llamado “Nuevo” como el parámetro command.
- *seleccionar_archivo()*: Esta función fue utilizada para poder encontrar un archivo previamente guardado en el dispositivo, haciendo que al momento de seleccionar el archivo este se muestre en un área de texto y así pueda ser editado. Esta función fue utilizada en el botón llamado “Abrir” como el parámetro command.
- *guardar()*: Esta función fue utilizada para tomar la cadena de texto que está en el área de texto y guardarla en un archivo de nombre “Entrada.txt” y fue utilizado en el botón llamado “Guardar” como el parámetro command.
- *guardar_archivo()*: Esta función fue utilizada para poder tomar la cadena de texto que está en el área de texto y guardarlo en una ruta específica con un nombre específico. Esta función fue utilizada en el botón llamado “GuardarComo” como el parámetro command.
- *finalizar_aplicacion()*: Esta función fue utilizada para finalizar la aplicación y así cerrar la ventana en ejecución. Esta función fue utilizada en el botón llamado “Salir” como el parámetro command.

Errores:

En este fichero se encuentra una clase llamada “Errores”, la cual tiene los siguientes atributos que serán útiles para crear objetos que guarden la información de un error encontrado durante el análisis correspondiente:

- fila: En este atributo se guarda la fila donde ocurrió el error.
- columna: En este atributo se guarda la columna donde ocurrió el error:
- lexema: En este atributo se guarda el carácter que fue un error en el análisis.
- id: En este se guarda el número del identificador correspondiente al error encontrado.

analizador:

En este fichero se realiza el análisis léxico y sintáctico al texto que contiene el archivo .txt que se está analizando. Para realizar el análisis se necesitó la implementación de una autónoa (Figura 1), la cual fue utilizada para poder analizar cada carácter dentro del texto y revisar si correspondía a un patrón en cierto estado. Dentro de la clase “Automata”, se encuentran los siguientes atributos y la siguiente función:

- letras: En este atributo se guardan los caracteres aceptados para un token del tipo letras.
- cadena: En este atributo se guarda la parte actual del texto del archivo .txt que está siendo analizado.
- fila: En este atributo se guarda la fila donde se encuentra el carácter analizado.
- columna: En este atributo se guarda la columna donde se encuentra el carácter analizado.
- estado actual: En este atributo se guarda el estado donde se encuentra la autónoa en el transcurso de la ejecución del programa.
- estado anterior: En este atributo se encuentra el estado que es anterior al estado donde se encuentra actualmente la autónoa.
- estado aceptacion: En este atributo se guarda el estado que se considera de aceptación.
- tabla tokens: Este atributo es una lista que guarda los objetos de la clase “Token” que guarda la información del lexema encontrado.
- tabla errores: Este atributo es una lista que guarda los objetos de la clase “Errores” que guarda la información de los errores encontrados en la ejecución.

- analizar(): Esta función realiza la implementación de la automática utilizada para el análisis del texto contenido el archivo.txt.
- guardar_tokens(): En esta función se crean los diferentes objetos de la clase "Token" y se guardan en la lista "tabla_tokens".

Se utilizó el método del árbol para la generar el autómata necesario para construir la aplicación que hiciera el análisis léxico y sintáctico de la cadena de texto a analizar y así crear sentencias básicas para ejecutar código de MongoDB. A continuación, se detalla el proceso:

Expresiones Regulares Utilizadas:

- Comentario de una línea
-.--(letras)*
- Comentario de Varias líneas
/.*(letras)*.*/
- CrearDB
(letras)*.(letras)*.=. (letras)*. (letras)*.(.);
- Eliminar DB
(letras)*.(letras)*.=. (letras)*. (letras)*.(.);
- CrearColeccion
(letras)*.(letras)*.=. (letras)*. (letras)*.(.(letras)*.)).;
- EliminarColeccion
(letras)*.(letras)*.=. (letras)*.(letras)*.(.(letras)*.)).;
- Insertar Único
(letras)*.(letras)*.=. (letras)*. (letras)*.(.(letras)*.))., ".{".(letras)*." : ".letras.".,".(letras)*." : ".(letras)*.}.".);
- ActualizarUnico
(letras)*.(letras)*.=. (letras)*. (letras)*.(.(letras)*.))., ".{".(letras)*." : ".letras.".,".{\$. (letras)*.{".(letras)*." : ".(letras)*.}.".);
- EliminarUnico
(letras)*.(letras)*.=. (letras)*. (letras)*.(.(letras)*.))., ".{".(letras)*." : ".letras.".,".{.".);
- BuscarTodo
(letras)*.(letras)*.=. (letras)*. (letras)*.(.(letras)*.)).;
- BuscarUnico
(letras)*.(letras)*.=. (letras)*. (letras)*.(.(letras)*.)).;

Expresión Regular Completa:

```
((-.-.(letras)*)?.( /.*(letras)*.*/)?.( (letras)*(letras)*.=. (letras)*. (letras)*.(.)?;).(
(letras)*(letras)*.=. (letras)*(letras)*.(“(letras)*.”);)?.( (letras)*(letras)*.=.
(letras)*. (letras)*.(“(letras)*.”.,“{“(letras)*.”. : “.letras.”.,“(letras)*.”.
:“(letras)*.”.”);)?.( (letras)*(letras)*.=. (letras)*.
(letras)*.(“(letras)*.”.,“{“(letras)*.”. : “.letras.”.},,{$. (letras)*.{“(letras)*.”. :
“(letras)*.”.”);)?.( (letras)*(letras)*.=. (letras)*.
(letras)*.(“(letras)*.”.,“{“(letras)*.”. : “.letras.”.}.”);)?.( (letras)*(letras)*.=.
(letras)*. (letras)*.(“(letras)*.”);)?*
```

Tabla de Tokens:

letras={ a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z }

Token	Lexema
Palabra Reservada	CrearDB
Palabra Reservada	EliminarDB
Palabra Reservada	CrearColeccion
Palabra Reservada	EliminarColeccion
Palabra Reservada	InsertarUnico
Palabra Reservada	ActualizarUnico
Palabra Reservada	EliminarUnico
Palabra Reservada	BuscarTodo
Palabra Reservada	BuscarUnico
Palabra Reservada	set
Palabra Reservada	autor
Palabra Reservada	nombre
Asignacion	=
slash	/
DosPuntos	:
PuntoComa	;
Llaves	{ }
Parentesis	()
Comillas	“ ”
Coma	,
simbolo	\$
identificador	Combinaciones de caracteres pertenecientes al conjunto “letras”

Árbol:

El diagrama del árbol se puede visualizar de mejor manera accediendo al siguiente link:

https://drive.google.com/file/d/138Bjsr_TiFbFB_H2zYOWEAivU8rgHJY/view?usp=sharing

Tabla de Siguietes:

A continuación, se muestra la tabla de siguietes realizada durante el método del árbol.

Nodo	Siguietes
1-	2
2-	3
3-	1,4,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
4letras	1,4,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
5/	6
6*	7,8
7letras	7,8
8*	9
9/	1,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
10letras	10,12
11letras	11,12
12=	13,14,15
13letras	13,14,15
14letras	14,15
15(16
16)	17
17;	1,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
18letras	18,19,20
19letras	19,20
20=	21,22,23
21letras	21,22,23
22letras	22,23
23(24
24"	25,26
25letras	25,26
26"	27
27)	28
28;	1,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
29letras	29,30,31
30letras	30,31
31=	32,33,34
32letras	32,33,34
33letras	33,34

34(35
35“	36
36letras	36,37
37”	38
38)	39
39;	1,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
40letras	40,41,42
41letras	41,42
42=	43,44,45
43letras	43,44,45
44letras	44,45
45(46
46“	47,48
47letras	47,48
48”	49
49,	50
50“	51
51{	52
52"	53,54
53letras	53,54
54"	55
55:	56
56"	57,58
57letras	57,58
58"	59
59,	60
60"	61,62
61letras	61,62
62"	63
63:	64
64"	65,66
65letras	65,66
66"	67
67}	68
68”	69
69)	70
70;	1,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
71letras	71,72,73
72letras	72,73
73=	74,75,76
74letras	74,75,76
75letras	75,76
76(77
77“	78
78letras	78,79
79”	80
80,	81

81“	82
82{	83
83"	84,85
84letras	84,85
85"	86
86:	87
87"	88,89
88letras	88,89
89"	90
90}	91
91,	92
92{	93
93\$	94,95
94letras	94,95
95:00:00	96
96{	97
97"	98,99
98letras	98,99
99"	100
100:	101
101"	102,103
102letras	102,103
103"	104
104}	105
105}	106
106”	107
107)	108
108;	1,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
109letras	109,110,111
110letras	110,111
111=	112,113,114
112letras	112,113,114
113letras	113,114
114(115
115“	116,117
116letras	116,117
117”	118
118,	119
119“	120
120{	121
121"	122,123
122letras	122,123
123"	124
124:	125
125"	126,127
126letras	126,127
127"	128

128}	129
129"	130
130)	131
131;	1,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
132letras	132,133,134
133letras	133,134
134=	135,136,137
135letras	135,136,137
136letras	136,137
137(138
138"	139,140
139letras	139,140
140"	141
141)	142
142;	1,5,10,11,12,18,19,20,29,30,31,40,41,42,71,72,73,109,110,111,132,133,134,143,\$
143#	

Tabla de Transiciones:

En el siguiente link se puede visualizar la tabla de transiciones realizada durante el método del árbol:

<https://docs.google.com/spreadsheets/d/1N-ZYZTEbCTUod4bpyNRR2D13sQgqTXU3/edit?usp=sharing&oid=103223961737514966801&rtpof=true&sd=true>


Gráfico del autómatas finito determinista:

En el siguiente link se puede visualizar el gráfico del AFD utilizado durante el análisis:

<https://drive.google.com/file/d/1BPjwhthJe4NFcEhtP9tXM5dQVL7RSNqS/view?usp=sharing>

Gramática Utilizada:

 **Alfabeto**=Terminales{
letras={a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z}}

 **NoTerminales**= $\{q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}, q_{12}, q_{13}, q_{14}, q_{15}, q_{16}, q_{17}, q_{18}, q_{19}, q_{20}, q_{21}, q_{22}, q_{23}, q_{24}, q_{25}, q_{26}, q_{27}, q_{28}, q_{29}, q_{30}, q_{31}, q_{32}, q_{33}, q_{34}, q_{35}, q_{36}, q_{37}, q_{38}, q_{39}, q_{40}, q_{41}, q_{42}, q_{43}, q_{44}, q_{45}, q_{46}, q_{47}, q_{48}, q_{49}, q_{50}, q_{51}, q_{52}, q_{53}, q_{54}, q_{55}, q_{56}, q_{57}, q_{58}\}$

 **Estado Inicial:** q_1

 **Estado de Aceptación:** q_9, q_{14}

 **Transiciones:**

- $q_1 \rightarrow -q_2$
- $q_1 \rightarrow /q_3$
- $q_1 \rightarrow \text{letras}q_4$
- $q_1 \rightarrow =q_5$
- $q_2 \rightarrow -q_6$
- $q_3 \rightarrow *q_7$
- $q_4 \rightarrow \text{letras}q_4$
- $q_4 \rightarrow =q_5$
- $q_5 \rightarrow \text{letras}q_5$
- $q_5 \rightarrow (q_8$
- $q_6 \rightarrow -q_9$
- $q_7 \rightarrow *q_{10}$
- $q_7 \rightarrow \text{letras}q_{11}$
- $q_8 \rightarrow)q_{12}$
- $q_8 \rightarrow "q_{13}$
- $q_9 \rightarrow -q_2$
- $q_9 \rightarrow /q_3$
- $q_9 \rightarrow \text{letras}q_9$
- $q_9 \rightarrow =q_5$
- $q_{10} \rightarrow /q_{14}$
- $q_{11} \rightarrow *q_{10}$
- $q_{11} \rightarrow \text{letras}q_{11}$
- $q_{12} \rightarrow ;q_{14}$
- $q_{13} \rightarrow \text{letras}q_{16}$
- $q_{13} \rightarrow "q_{15}$
- $q_{14} \rightarrow -q_2$
- $q_{14} \rightarrow /q_{10}$
- $q_{14} \rightarrow \text{letras}q_4$
- $q_{14} \rightarrow =q_5$

- q15 ->)q17
- q15 -> ,q18
- q16 -> letrasq19
- q17 -> ;q14
- q18 -> "q21
- q19 -> letrasq16
- q19 -> "q15
- q20 ->)q22
- q20 -> ,q23
- q21 -> {q24
- q22 -> ;q14
- q23 -> "q25
- q24 -> "q26
- q25 -> {q27
- q26 -> letras q26
- q26 -> "q28
- q27 -> "q29
- q28 -> :q30
- q29 -> letras q29
- q29 -> "q31
- q30 -> "q32
- q31 -> :q33
- q32 -> letras q32
- q32 -> "q34
- q33 -> "q35
- q34 -> }q37
- q34 -> ,q36
- q35 -> letras q35
- q35 -> "q38
- q36 -> letras q39
- q37 -> "q40
- q38 -> }q41
- q38 -> ,q36
- q39 -> letrasq39
- q39 -> "q42
- q40 ->)q43
- q41 -> "q40
- q41 -> ,q44
- q42 ->)q45
- q43 -> ;q14
- q44 -> {q46

- q45 -> ;q14
- q46 -> \$q47
- q47 -> letrasq47
- q47 -> :q48
- q48 -> {q49
- q49 -> "q50
- q50 -> letrasq50
- q50 -> "q51
- q51 -> :q52
- q52 -> "q53
- q53 -> letrasq53
- q53 -> "q54
- q54 -> }q55
- q55 -> }q56
- q56 -> "q57
- q57 ->)q58
- q58 -> ;q14

