



Universidad Autónoma de Chiapas

Licenciatura en Ingeniería de Desarrollo y Tecnologías
de Software

Sub 1. Análisis léxico

Actividad 1.6

Docente: D.S.C Luis Gutiérrez Alfaro

Alumno: Gilberto Gutiérrez Vázquez

Matrícula: A221701

6° Semestre

Grupo M

Lugar: Tuxtla Gutiérrez, Chiapas

Fecha: 18/08/2024

Tabla de tokens

Token	Patrón	Lexema
NUMERO_ENTERO	\d+	Número entero
SUMA	+	+
RESTA	-	-
MULTIPLICACION	*	*
DIVISION	/	/
PARENTESIS_IZQ	((
PARENTESIS_DER))
ESPACIO	\s+	Espacio en blanco
ASIGNACION	\=	=
PRINT	print	print
PUNTO	.	.
COMA	,	,
PUNTO_Y_COMA	;	;
DOS_PUNTOS	:	:
COMILLAS_DOBLAS	"	"
COMILLAS_SIMPLES	'	'
CORCHETE_IZQ	[[
CORCHETE_DER]]
LLAVE_IZQ	{	{
LLAVE_DER	}	}
MENOR	<	<
MAYOR	>	>
MENOR_IGUAL	<=	<=
MAYOR_IGUAL	>=	>=
DIFERENTE	!=	!=
INCREMENTO	++	++
DECREMENTO	--	--
AND	&&	&&
OR	\	\
NOT	!	!
MODULO	%	%
GETTER	\bgetter\b	getter
SETTER	\bsetter\b	setter
PRIVATE	\bprivate\b	private
PROTECTED	\bprotected\b	protected

PUBLIC	\bpublic\b	public
IF	\bif\b	if
ELSE	\belse\b	else
FOR	\bfor\b	for
WHILE	\bwhile\b	while
DO	\bdo\b	do
SWITCH	\bswitch\b	switch
CASE	\bcase\b	case
BREAK	\bbreak\b	break
CONTINUE	\bcontinue\b	continue
RETURN	\breturn\b	return
FUNCTION	\bfunction\b	function
CLASS	\bclass\b	class
CONSTRUCTOR	\bconstructor\b	constructor
NUEVO_OBJETO	\bnew\b	new
THIS	\bthis\b	this
STATIC	\bstatic\b	static
VOID	\bvoid\b	void
NULL	\bnull\b	null
BOOLEANO	\btrue\b	false
IDENTIFICADOR	[a-zA-Z_][a-zA-Z_0-9]*	Hola_Mundo

Pruebas de código

```
92         return tokens
93
94
95     code = "x = 2 + 4 * (2 - 8) = 8 , if (
96     tokens = tokenize(code)
97
98     for i, token in enumerate(tokens, start=0):
99         print(f'{token}')
100
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PC

```
PS C:\Users\Gilberto\Desktop\Ejercicios compila
s/Python/Python311/python.exe "c:/Users/Gilbert
ico2.py"
```

```
( 'IDENTIFICADOR', 'x' )
( 'ASIGNACION', '=' )
( 'NUMERO_ENTERO', '2' )
( 'SUMA', '+' )
( 'NUMERO_ENTERO', '4' )
( 'MULTIPLICACION', '*' )
( 'PARENTESIS_IZQ', '(' )
( 'NUMERO_ENTERO', '2' )
( 'RESTA', '-' )
( 'NUMERO_ENTERO', '8' )
( 'PARENTESIS_DER', ')' )
( 'ASIGNACION', '=' )
( 'NUMERO_ENTERO', '8' )
( 'COMA', ',' )
( 'IF', 'if' )
( 'PARENTESIS_IZQ', '(' )
( 'IDENTIFICADOR', 'z' )
( 'MAYOR', '>' )
( 'NUMERO_ENTERO', '0' )
( 'PARENTESIS_DER', ')' )
( 'FUNCTION', 'function' )
( 'IDENTIFICADOR', 'suma' )
( 'PARENTESIS_IZQ', '(' )
```

Ln 100, Col 1

```

10 def tokenize(code):
11     tokens = []
12     for token in code.split():
13         tokens.append(token)
14     return tokens
15
16 code = "x = 2 + 4 * (2 - 8) = 8 , y = 3"
17 tokens = tokenize(code)
18
19 for i, token in enumerate(tokens, 1):
20     print(f'{i}. {token}')
21
22 input()

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\Gilberto\Desktop\Ejercicios comp
Programs/Python/Python311/python.exe "c:/Use
Analizador lexico2.py"
```

1. ('IDENTIFICADOR', 'x')
2. ('ASIGNACION', '=')
3. ('NUMERO_ENTERO', '2')
4. ('SUMA', '+')
5. ('NUMERO_ENTERO', '4')
6. ('MULTIPLICACION', '*')
7. ('PARENTESIS_IZQ', '(')
8. ('NUMERO_ENTERO', '2')
9. ('RESTA', '-')
10. ('NUMERO_ENTERO', '8')
11. ('PARENTESIS_DER', ')')