UNIVERSIDAD AUTÓNOMA DE CHIAPAS.

FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN, CAMPUS I.

LICENCIATURA EN INGENIERÍA EN DESARROLLO Y TECNOLOGÍAS DE SOFTWARE.

SEXTO SEMESTRE, GRUPO: “M”

MATERIA: COMPILADORES.

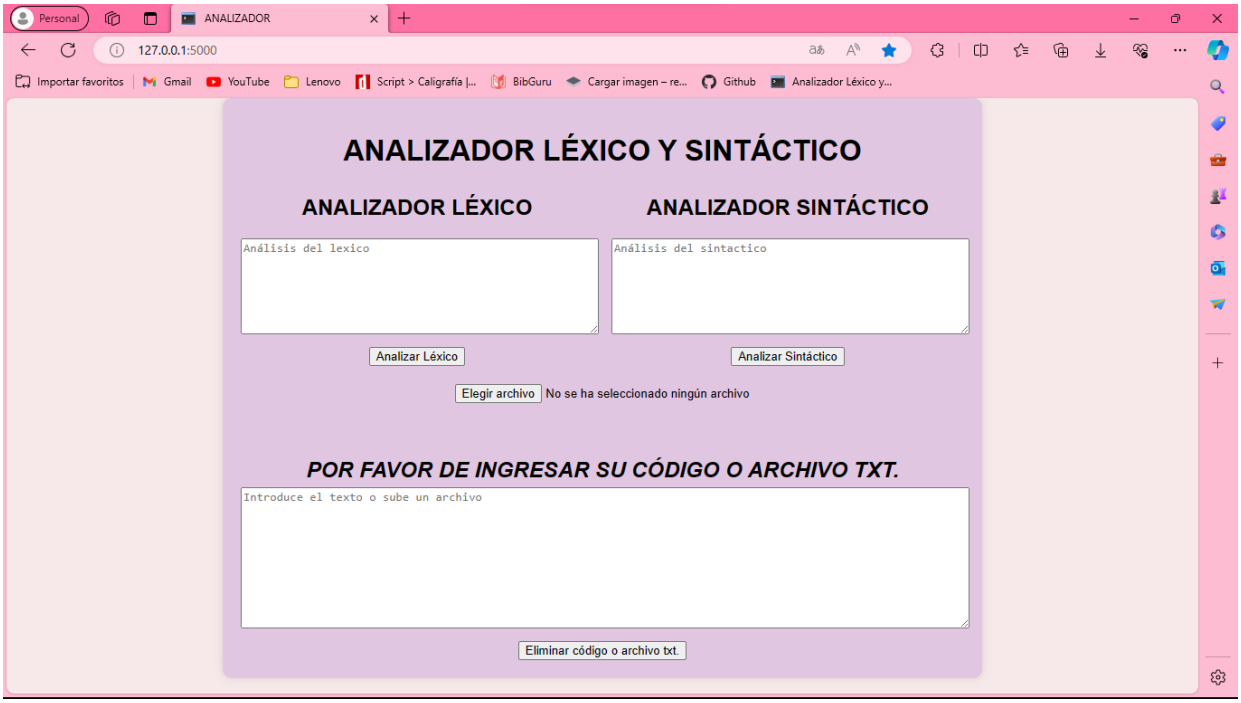
DOCENTE: DR. LUIS GUTIÉRREZ ALFARO.

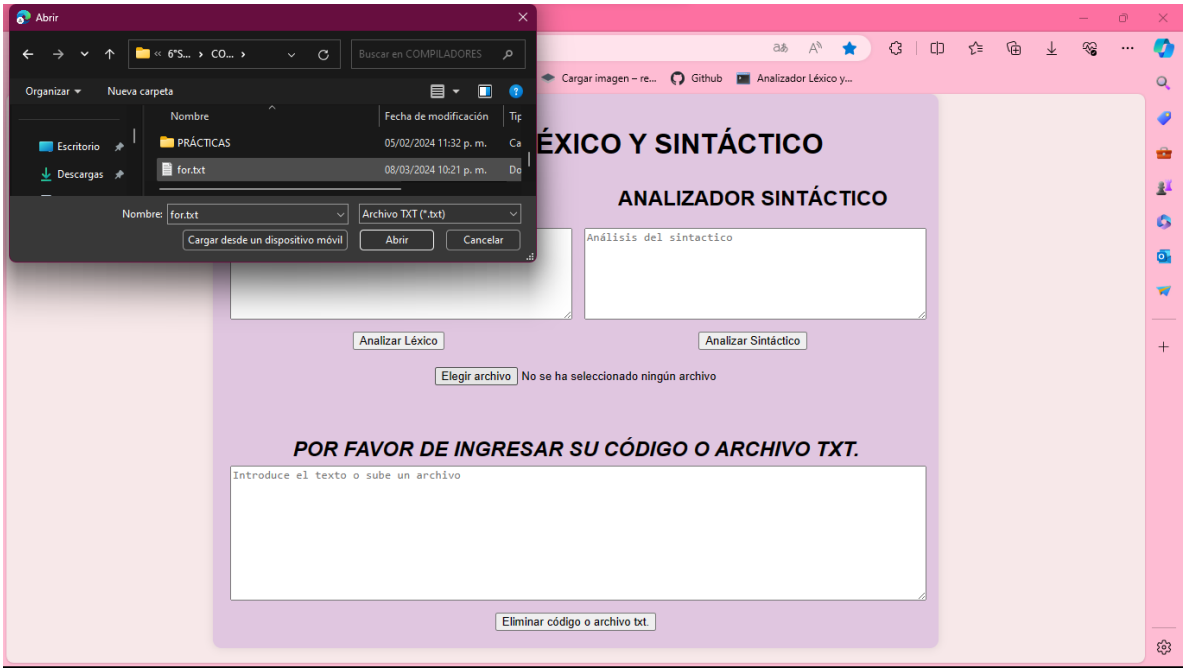
ALUMNA: CIGARROA HERNÁNDEZ LUISA FERNANDA (A210118).

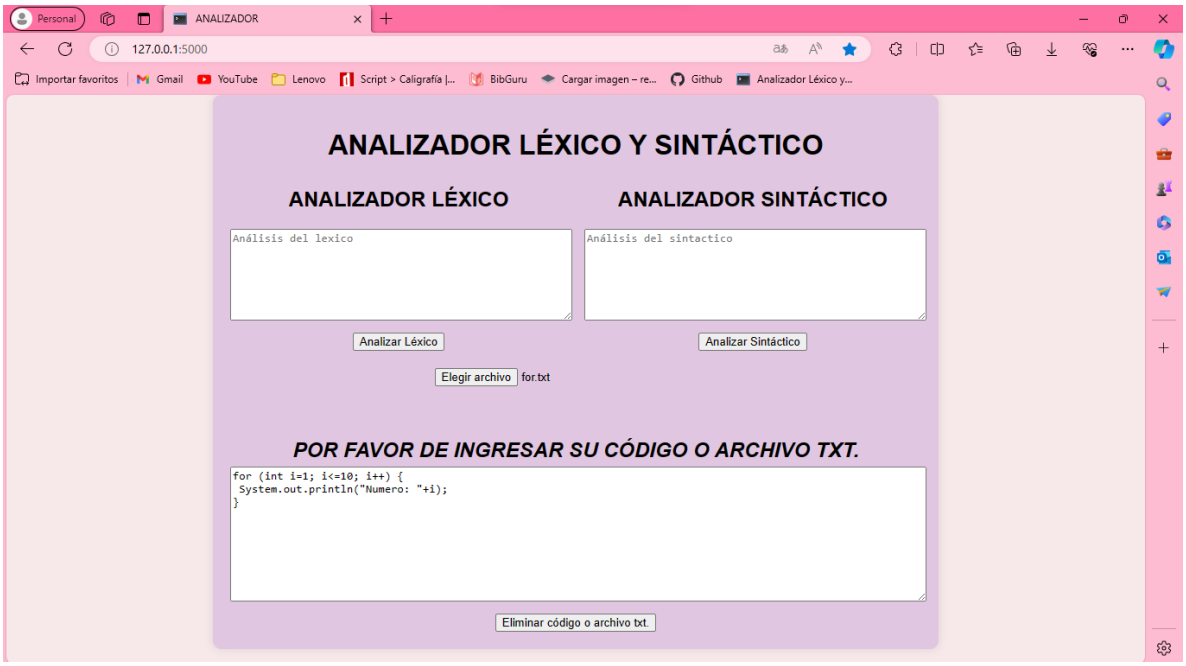
“ANALIZADOR LÉXICO Y SINTÁCTICO”

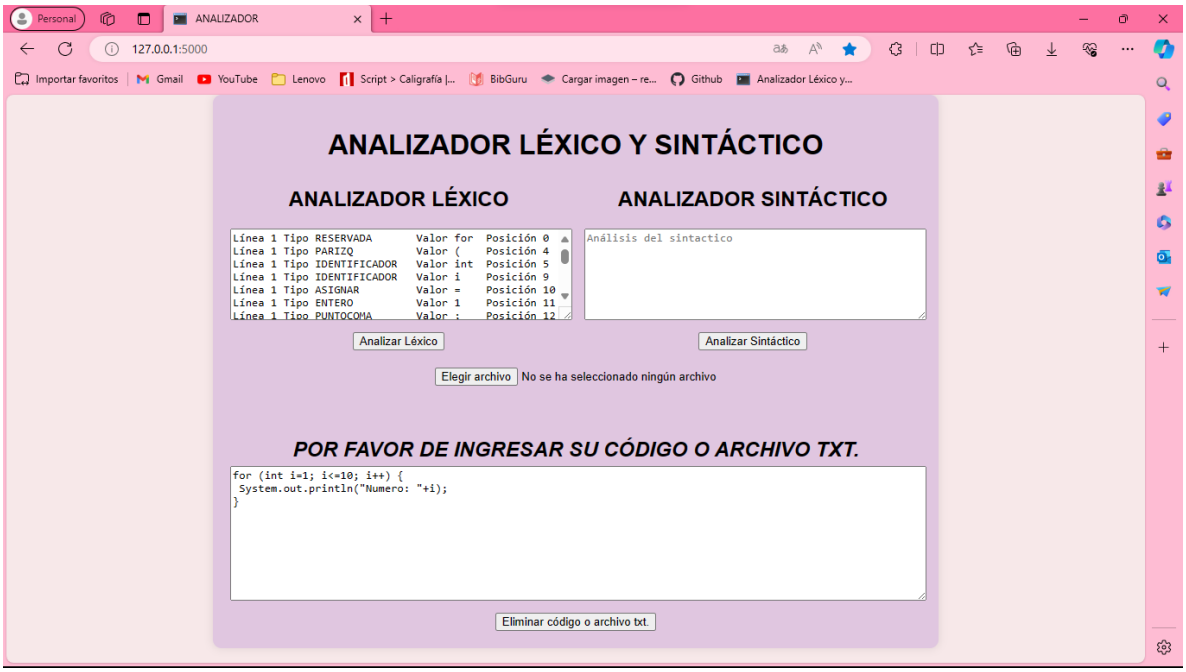
FECHA DE ENTREGA: SÁBADO 09 DE MARZO DE 2024.

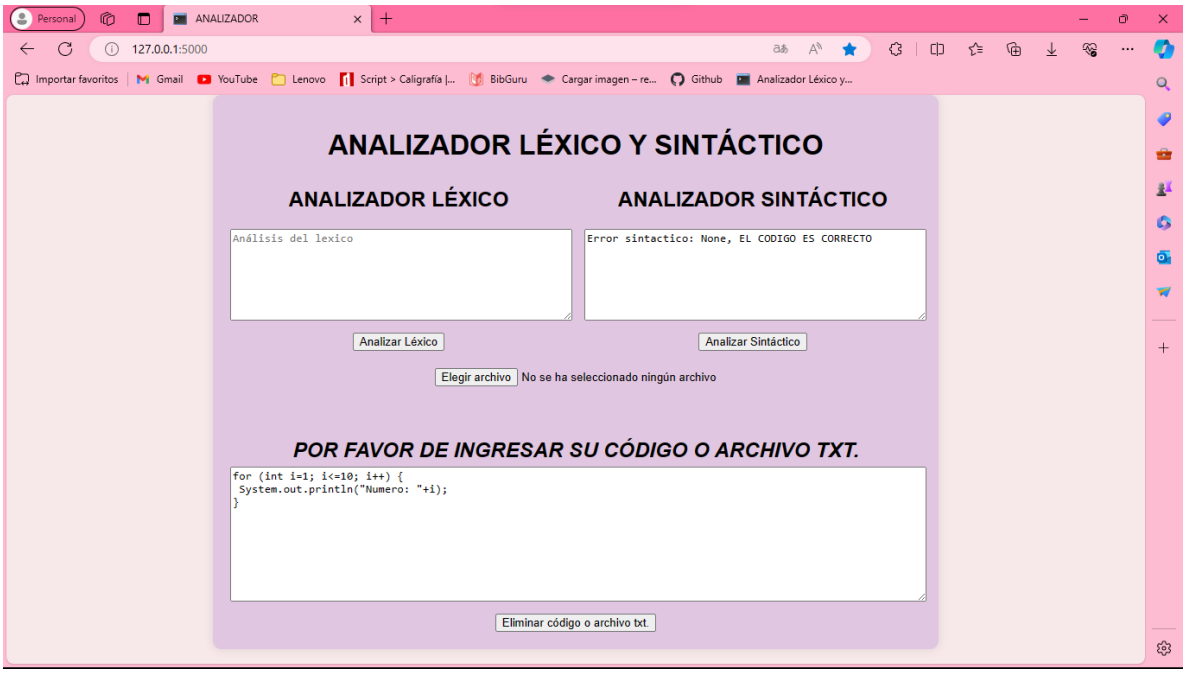
ANALIZADOR LÉXICO Y SINTÁCTICO

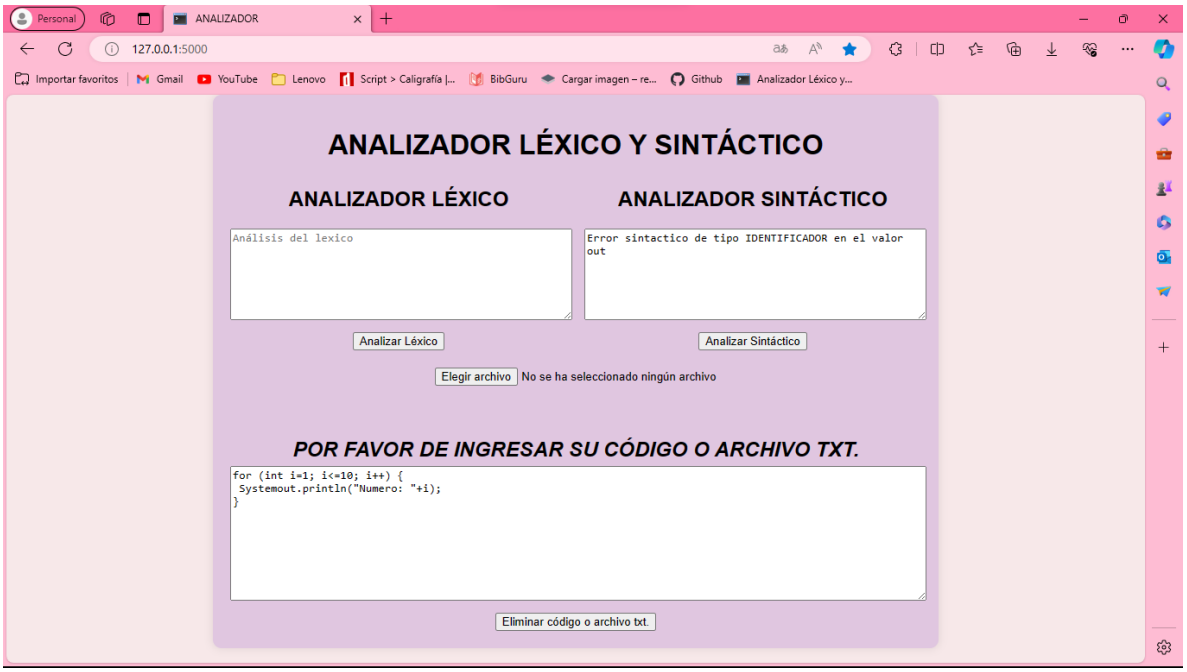


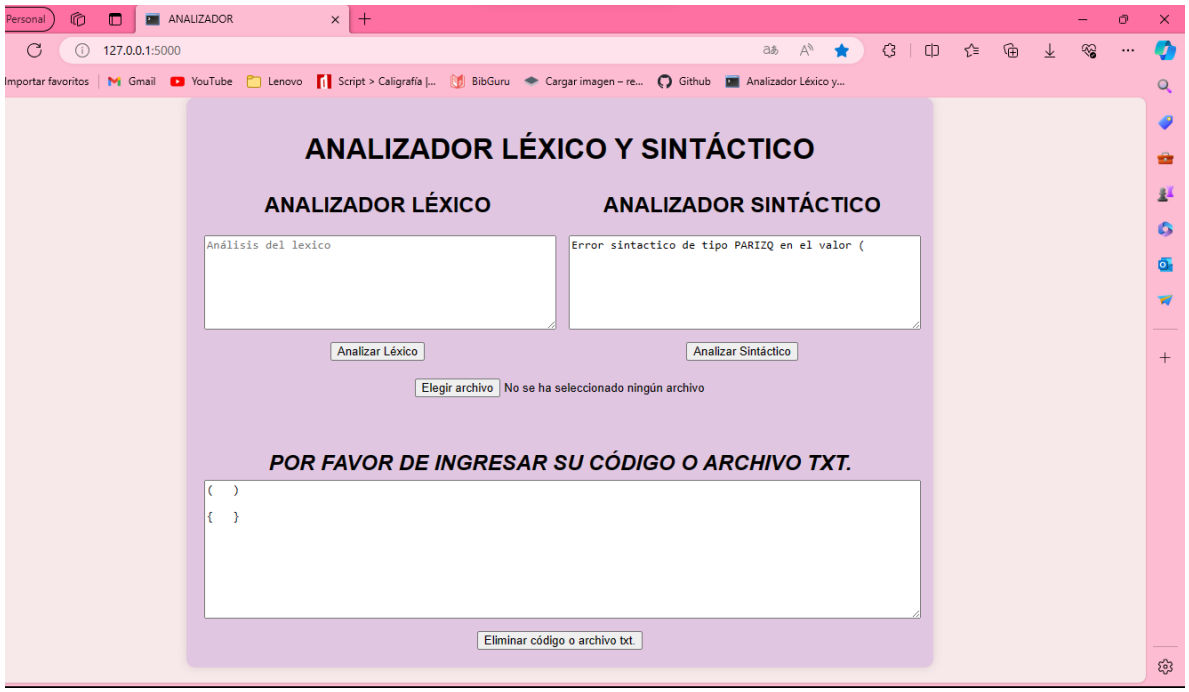


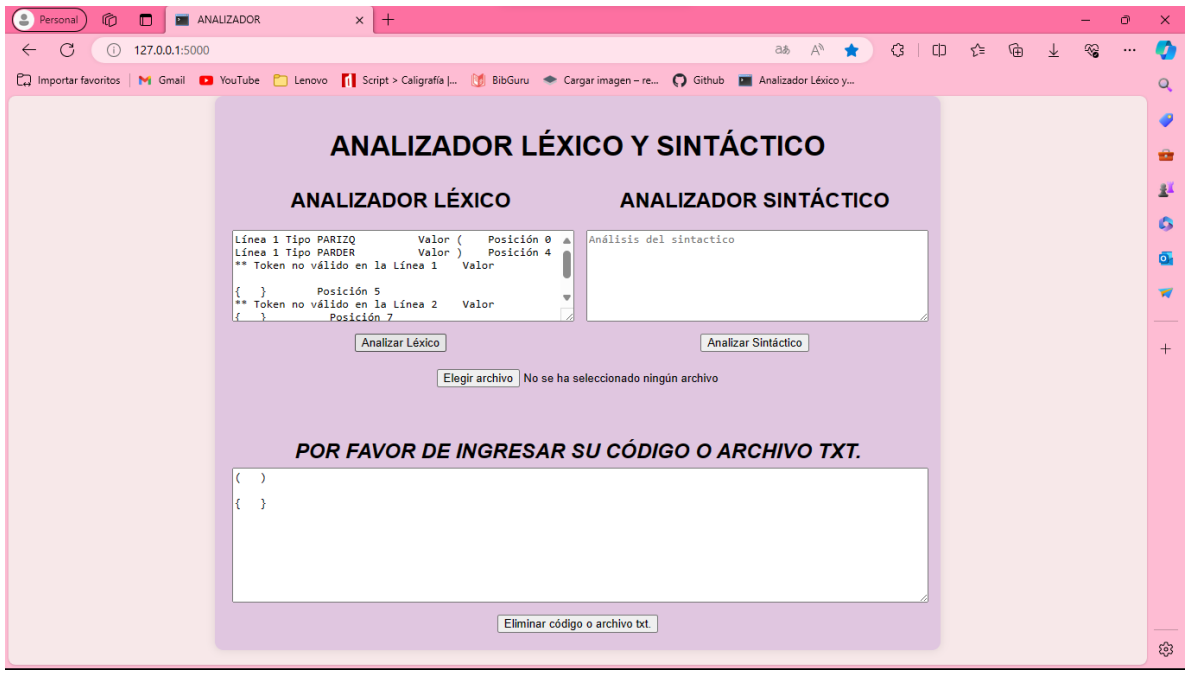

ANALIZADOR
127.0.0.1:5000
Importar favoritos
Gmail
YouTube
Lenovo
Script > Caligrafía |...
BibGuru
Cargar imagen – re...
Github
Analizador Léxico y...
ANALIZADOR LÉXICO Y SINTÁCTICO
ANALIZADOR LÉXICO
ANALIZADOR SINTÁCTICO
Análisis del lexico
Análisis del sintactico
Analizar Léxico
Analizar Sintáctico
Elegir archivo
for.txt
POR FAVOR DE INGRESAR SU CÓDIGO O ARCHIVO TXT.
for (int i=1; i<=10; i++) {
System.out.println("Numero: "+i);
}
Eliminar código o archivo txt.


ANALIZADOR
127.0.0.1:5000
Importar favoritos
Gmail
YouTube
Lenovo
Script > Caligrafía |...
BibGuru
Cargar imagen – re...
Github
Analizador Léxico y...
ANALIZADOR LÉXICO Y SINTÁCTICO
ANALIZADOR LÉXICO
ANALIZADOR SINTÁCTICO
Línea 1 Tipo RESERVADA Valor for Posición 0
Línea 1 Tipo PARIZQ Valor (Posición 4
Línea 1 Tipo IDENTIFICADOR Valor int Posición 5
Línea 1 Tipo IDENTIFICADOR Valor i Posición 9
Línea 1 Tipo ASIGNAR Valor = Posición 10
Línea 1 Tipo ENTERO Valor 1 Posición 11
Línea 1 Tipo PUNTOCOMA Valor ; Posición 12
Análisis del sintactico
Analizar Léxico
Analizar Sintáctico
Elegir archivo
No se ha seleccionado ningún archivo
POR FAVOR DE INGRESAR SU CÓDIGO O ARCHIVO TXT.
for (int i=1; i<=10; i++) {
System.out.println("Numero: "+i);
}
Eliminar código o archivo txt.

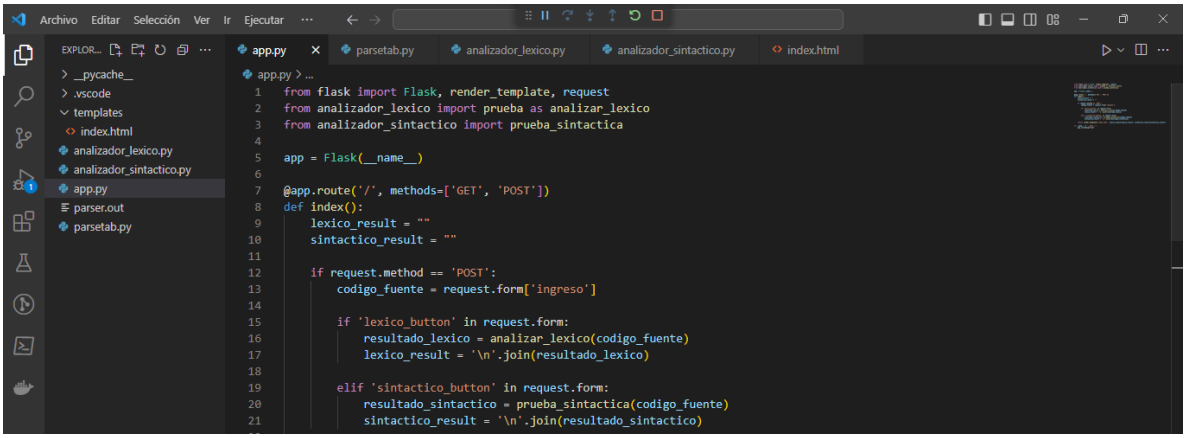

ANALIZADOR
127.0.0.1:5000
Importar favoritos
Gmail
YouTube
Lenovo
Script > Caligrafía |...
BibGuru
Cargar imagen – re...
Github
Analizador Léxico y...
ANALIZADOR LÉXICO Y SINTÁCTICO
ANALIZADOR LÉXICO
ANALIZADOR SINTÁCTICO
Análisis del lexico
Error sintactico: None, EL CODIGO ES CORRECTO
Analizar Léxico
Analizar Sintáctico
Elegir archivo
No se ha seleccionado ningún archivo
POR FAVOR DE INGRESAR SU CÓDIGO O ARCHIVO TXT.
for (int i=1; i<=10; i++) {
System.out.println("Numero: "+i);
}
Eliminar código o archivo txt.


ANALIZADOR
127.0.0.1:5000
Importar favoritos
Gmail
YouTube
Lenovo
Script > Caligrafía |...
BibGuru
Cargar imagen – re...
Github
Analizador Léxico y...
ANALIZADOR LÉXICO Y SINTÁCTICO
ANALIZADOR LÉXICO
ANALIZADOR SINTÁCTICO
Análisis del lexico
Error sintactico de tipo IDENTIFICADOR en el valor out
Analizar Léxico
Analizar Sintáctico
Elegir archivo
No se ha seleccionado ningún archivo
POR FAVOR DE INGRESAR SU CÓDIGO O ARCHIVO TXT.
for (int i=1; i<=10; i++) {
Systemout.println("Numero: "+i);
}
Eliminar código o archivo txt.

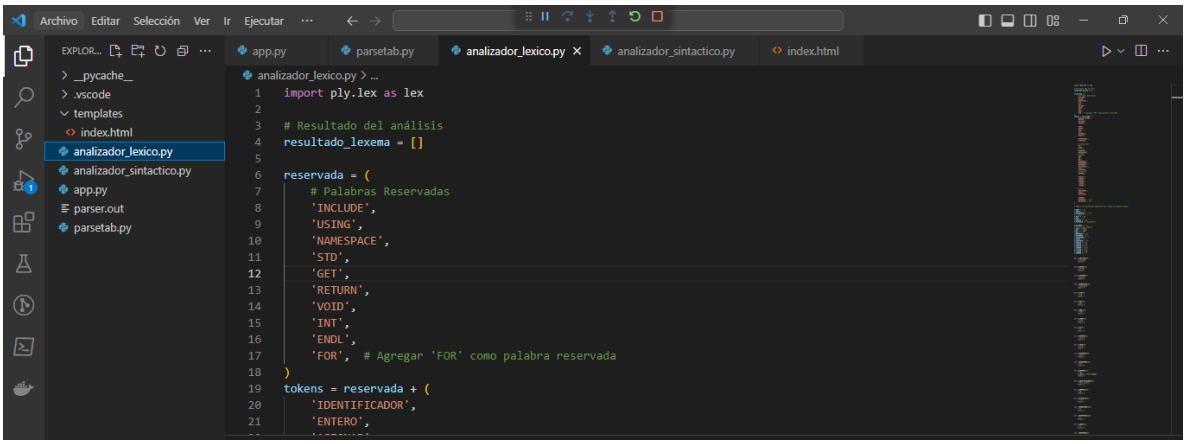

ANALIZADOR
127.0.0.1:5000
Importar favoritos
Gmail
YouTube
Lenovo
Script > Caligrafía |...
BibGuru
Cargar imagen – re...
Github
Analizador Léxico y...
ANALIZADOR LÉXICO Y SINTÁCTICO
ANALIZADOR LÉXICO
ANALIZADOR SINTÁCTICO
Análisis del lexico
Error sintactico de tipo PARIZQ en el valor (
Analizar Léxico
Analizar Sintáctico
Elegir archivo
No se ha seleccionado ningún archivo
POR FAVOR DE INGRESAR SU CÓDIGO O ARCHIVO TXT.
( )
{ }
Eliminar código o archivo txt.


ANALIZADOR
127.0.0.1:5000
Importar favoritos
Gmail
YouTube
Lenovo
Script > Caligrafía |...
BibGuru
Cargar imagen – re...
Github
Analizador Léxico y...
ANALIZADOR LÉXICO Y SINTÁCTICO
ANALIZADOR LÉXICO
ANALIZADOR SINTÁCTICO
Línea 1 Tipo PARIZQ Valor (Posición 0
Línea 1 Tipo PARDER Valor) Posición 4
** Token no válido en la Línea 1 Valor
{ } Posición 5
** Token no válido en la Línea 2 Valor
{ } Posición 7
Análisis del sintactico
Analizar Léxico
Analizar Sintáctico
Elegir archivo
No se ha seleccionado ningún archivo
POR FAVOR DE INGRESAR SU CÓDIGO O ARCHIVO TXT.
( )
{ }
Eliminar código o archivo txt.

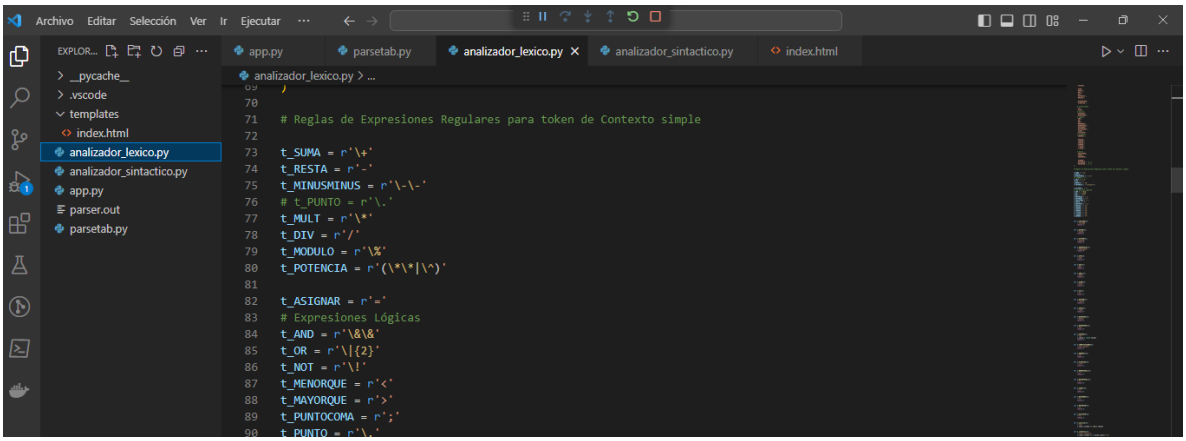
ANEXO ALGUNAS CAPTURAS DEL CÓDIGO



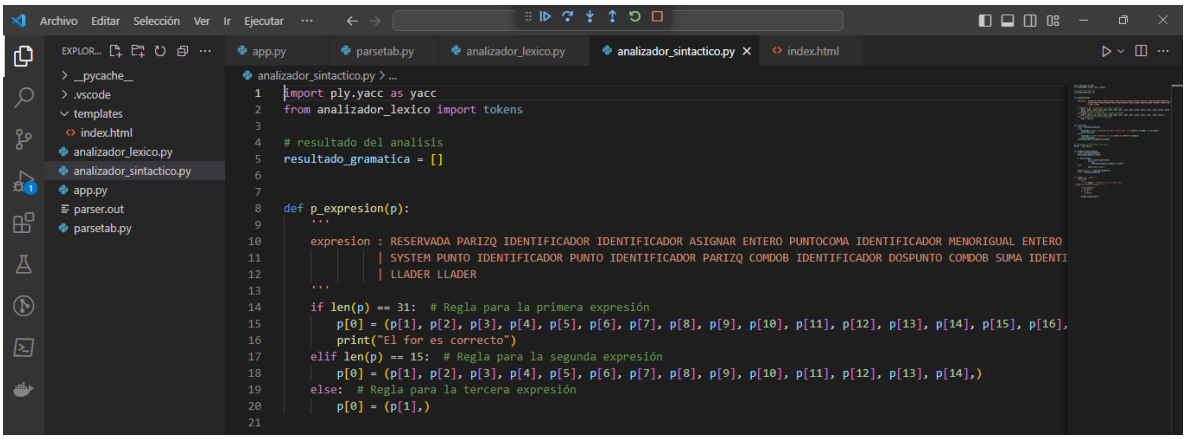
```python
from flask import Flask, render_template, request
from analizador_lexico import prueba as analizar_lexico
from analizador_sintactico import prueba_sintactica

app = Flask(__name__)

@app.route('/', methods=['GET', 'POST'])
def index():
    lexico_result = ""
    sintactico_result = ""

    if request.method == 'POST':
        codigo_fuente = request.form['ingreso']

        if 'lexico_button' in request.form:
            resultado_lexico = analizar_lexico(codigo_fuente)
            lexico_result = '\n'.join(resultado_lexico)

        elif 'sintactico_button' in request.form:
            resultado_sintactico = prueba_sintactica(codigo_fuente)
            sintactico_result = '\n'.join(resultado_sintactico)
```



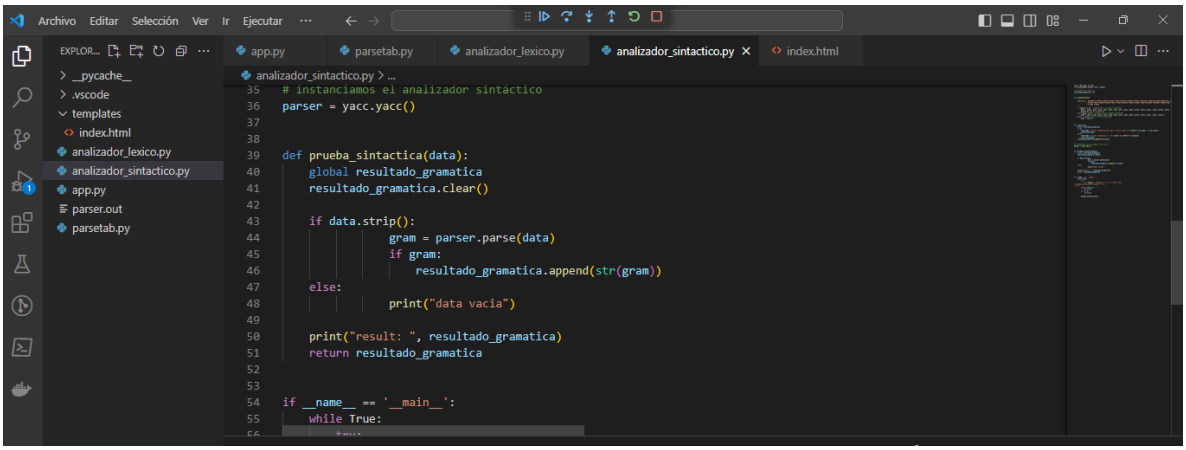
```python
import ply.lex as lex

# Resultado del análisis
resultado_lexema = []

reservada = (
    # Palabras Reservadas
    'INCLUDE',
    'USING',
    'NAMESPACE',
    'STD',
    'GET',
    'RETURN',
    'VOID',
    'INT',
    'ENDL',
    'FOR',  # Agregar 'FOR' como palabra reservada
)
tokens = reservada + (
    'IDENTIFICADOR',
    'ENTERO',
```



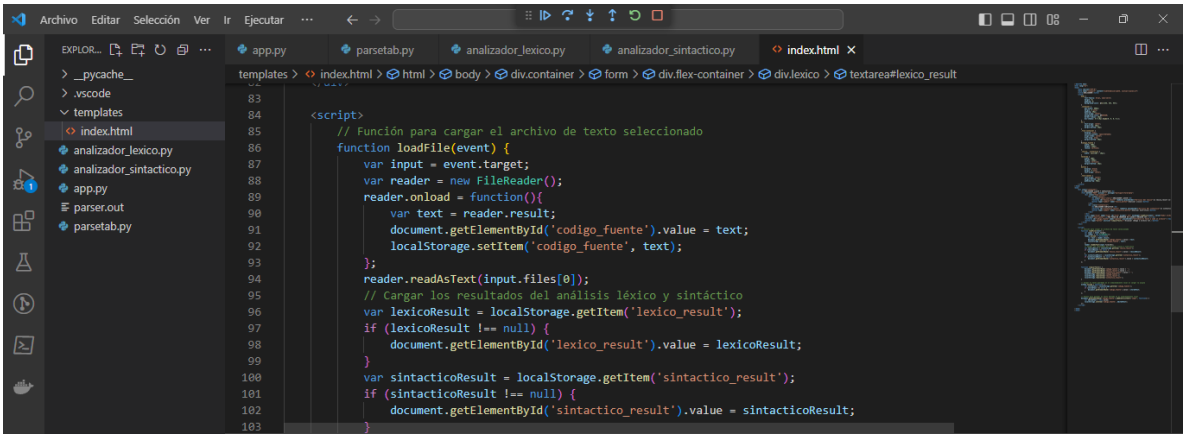
```python
)

# Reglas de Expresiones Regulares para token de Contexto simple

t_SUMA = r'\+'
t_RESTA = r'-'
t_MINUSMINUS = r'\-\-'
# t_PUNTO = r'\.'
t_MULT = r'\*'
t_DIV = r'/'
t_MODULO = r'\%'
t_POTENCIA = r'(\*\*|\^)'

t_ASIGNAR = r'='
# Expresiones Lógicas
t_AND = r'\&\&'
t_OR = r'\|{2}'
t_NOT = r'\!'
t_MENORQUE = r'<'
t_MAYORQUE = r'>'
t_PUNTOCOMA = r';'
t_PUNTO = r'\.'
```



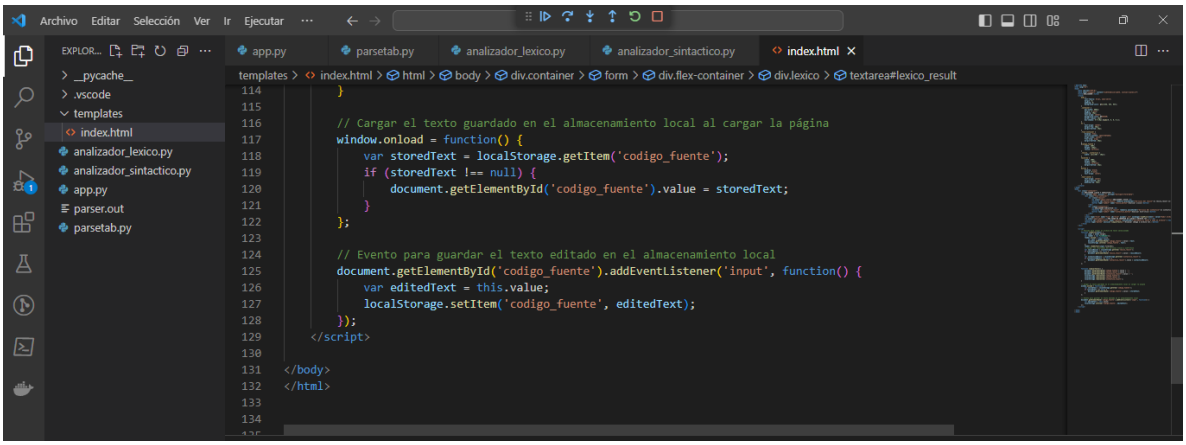
```python
import ply.yacc as yacc
from analizador_lexico import tokens

# resultado del analisis
resultado_gramatica = []


def p_expresion(p):
    '''
    expresion : RESERVADA PARIZQ IDENTIFICADOR IDENTIFICADOR ASIGNAR ENTERO PUNTOCOMA IDENTIFICADOR MENORIGUAL ENTERO
              | SYSTEM PUNTO IDENTIFICADOR PUNTO IDENTIFICADOR PARIZQ COMDOB IDENTIFICADOR DOSPUNTO COMDOB SUMA IDENTI
              | LLADER LLADER
    '''
    if len(p) == 31:  # Regla para la primera expresión
        p[0] = (p[1], p[2], p[3], p[4], p[5], p[6], p[7], p[8], p[9], p[10], p[11], p[12], p[13], p[14], p[15], p[16],
        print("El for es correcto")
    elif len(p) == 15:  # Regla para la segunda expresión
        p[0] = (p[1], p[2], p[3], p[4], p[5], p[6], p[7], p[8], p[9], p[10], p[11], p[12], p[13], p[14],)
    else:  # Regla para la tercera expresión
        p[0] = (p[1],)
```



```python
# instanciamos el analizador sintáctico
parser = yacc.yacc()


def prueba_sintactica(data):
    global resultado_gramatica
    resultado_gramatica.clear()

    if data.strip():
                gram = parser.parse(data)
                if gram:
                    resultado_gramatica.append(str(gram))
    else:
                print("data vacia")

    print("result: ", resultado_gramatica)
    return resultado_gramatica


if __name__ == '__main__':
    while True:
```



```
templates > index.html > html > body > div.container > form > div.flex-container > div.lexico > textarea#lexico_result

    <script>
        // Función para cargar el archivo de texto seleccionado
        function loadFile(event) {
            var input = event.target;
            var reader = new FileReader();
            reader.onload = function(){
                var text = reader.result;
                document.getElementById('codigo_fuente').value = text;
                localStorage.setItem('codigo_fuente', text);
            };
            reader.readAsText(input.files[0]);
            // Cargar los resultados del análisis léxico y sintáctico
            var lexicoResult = localStorage.getItem('lexico_result');
            if (lexicoResult !== null) {
                document.getElementById('lexico_result').value = lexicoResult;
            }
            var sintacticoResult = localStorage.getItem('sintactico_result');
            if (sintacticoResult !== null) {
                document.getElementById('sintactico_result').value = sintacticoResult;
            }
```



LINK DEL GIT

https://github.com/luisafer200/ANALIZADOR-L-XICO-SINT-CTICO