

## Tarea: Construcción de un Analizador con ANTLR4 y JavaScript

Tema: 25914\_12

Se proporciona una gramática en notación EBNF que describe un lenguaje específico. Utilizando **ANTLR4** con **JavaScript**, implemente un analizador que procese un archivo de entrada (input.txt) con código fuente escrito en dicho lenguaje.

```
<DSL> ::= <Regla>+ <Accion>+
<Regla> ::= "regla" ' ' <Identificador> "{" "cuando" <Condicion> "entonces" <Comando> "}"
<Condicion> ::= <AccesoNoAutorizado> | <MultiplesIntentosFallidos> | <ActividadSospechosa>
<AccesoNoAutorizado> ::= "intentoAccesoFueraHorario"
<MultiplesIntentosFallidos> ::= "intentosFallidos" ">" <Digito>
<ActividadSospechosa> ::= "accesoAREcursosSensibles" "y" "noEsAdministrador"
<Comando> ::= "agregarUsuarioAListaNegra" | "agregarIPAListaNegra" | "activarMonitoreoDetallado"
<Identificador> ::= <Letra> (<Letra> | <Digito>)*
<Cadena> ::= <Caracter>+
<Letra> ::= "A" | "B" | "C" | ... | "Z" | "a" | "b" | "c" | ... | "z"
<Digito> ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
<Caracter> ::= <Letra> | <Digito> | " _ " | "." | "/"
```

El analizador deberá realizar las siguientes tareas:

1. **Análisis léxico y sintáctico:** realizar el análisis léxico y sintáctico sobre el código fuente e informar si la entrada es correcta o contiene errores. En caso de errores, indicar la línea en la que ocurren y la causa del problema.
2. **Tabla de lexemas-tokens:** Generar una tabla que contenga los lexemas y sus respectivos tokens reconocidos durante el análisis léxico.
3. **Árbol de análisis sintáctico:** Construir y mostrar el árbol de análisis sintáctico concreto de la entrada. Puede representarse en formato de texto.
4. **Interpretación:** Traducir el código fuente al lenguaje **JavaScript** y ejecutarlo como lo haría un intérprete básico.

El desarrollo debe entregarse cumpliendo las pautas para la entrega establecidas en el documento **Pautas de trabajo para analizador**.

### Ejemplo de Código

```
regla 'bloqueo_acceso_ilegal' { cuando intentoAccesoFueraHorario entonces agregarUsuarioAListaNegra }
accion 'agregarUsuarioAListaNegra' { }
```

### Traducción a JavaScript

```
let intentoAccesoFueraHorario = true;
const listaNegra = new Set();
function gestionarBloqueo() {
  if (intentoAccesoFueraHorario) {
    const usuario = "usuario_sospechoso";
    listaNegra.add(usuario);
    console.log('Bloqueando a ${usuario} por intento de acceso fuera de horario.');
```