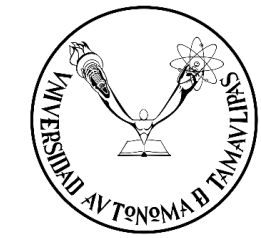
Universidad Autónoma de Tamaulipas
Facultad de Ingeniería Tampico



ASIGNATURA

Programación de Sistemas Base 1

8vo. Semestre – Grupo “G”

2025 -1

Unidad 3 Actividad 3

Proyecto Itegrador

Docente: Muñoz Quintero Dante Adolfo

Alumnos:

Duran Lugo Jesus Antonio

Hernández Lara Ana Patricia

Antonio Guzmán Lucero Iris

Saldaña Sánchez Carlos Alfonso

Introducción:

En este proyecto se aborda la creación de un analizador léxico y sintáctico diseñado específicamente para un lenguaje simplificado de definición de tablas. Esta herramienta fundamental en el proceso de compilación, se encarga de desglosar el código fuente en unidades significativas denominadas tokens y, posteriormente, verificar que la estructura de estos tokens se ajuste a las reglas gramaticales preestablecidas para el lenguaje en cuestión. A través de una interfaz gráfica intuitiva, el usuario podrá ingresar definiciones de tablas y observar en detalle el resultado del análisis, incluyendo la identificación de los componentes léxicos y la validación de su correcta sintaxis.

Objetivo (5 puntos importantes):

- Implementar un analizador léxico eficiente: Capaz de identificar y clasificar correctamente los tokens (palabras clave, identificadores, delimitadores) del lenguaje de definición de tablas.
- Desarrollar un analizador sintáctico robusto: Que verifique la conformidad de la secuencia de tokens con la gramática definida para el lenguaje, asegurando la correcta estructura de las definiciones de tablas y sus columnas.
- Incorporar un análisis semántico básico: Para detectar errores como la declaración de variables con nombres duplicados dentro de una misma tabla.
- Crear una interfaz gráfica de usuario: Que permita la fácil entrada del código fuente y la visualización clara de los tokens identificados y los resultados del análisis sintáctico y semántico, incluyendo mensajes de error detallados.
- Proporcionar una herramienta comprensible: Que sirva como ejemplo práctico de los conceptos fundamentales del análisis léxico y sintáctico en el contexto del procesamiento de lenguajes.

Este proyecto es un analizador léxico y sintáctico para un lenguaje simplificado de definición de tablas. A continuación, se detalla su funcionamiento, lo que hace y los resultados que produce, basándose en los archivos de código proporcionados.

Descripción del Proyecto: Analizador Léxico y Sintáctico para Definición de Tablas

Este proyecto se enfoca en el desarrollo de un analizador léxico y sintáctico para un lenguaje simplificado diseñado para la definición de tablas, cumpliendo con los objetivos establecidos en la materia de Programación de Sistemas de Base 1. El objetivo principal es aplicar las técnicas de análisis léxico y sintáctico para procesar un lenguaje estructurado, demostrando la comprensión de las etapas fundamentales de un compilador.

Análisis Léxico:

Se ha implementado un analizador léxico (lexer) en Java, específicamente en el archivo Lexico.java. Este lexer realiza la tokenización del código fuente, dividiéndolo en unidades léxicas significativas llamadas tokens. El proyecto incluye el manejo de errores léxicos, utilizando la clase ErrorAnalizador.java para reportar caracteres no reconocidos, lo cual satisface el requerimiento de la especificación.

Análisis Sintáctico:

Se ha desarrollado un analizador sintáctico (parser) en el archivo Sintactico.java. Este parser implementa un análisis sintáctico descendente recursivo. El parser verifica que la secuencia de tokens se ajuste a la gramática definida para el lenguaje, y además, construye una estructura intermedia que representa la información analizada, incluyendo el nombre de la tabla, las variables y sus tipos. Esto cumple con el requisito de generar un AST o estructura intermedia.

Funcionalidad Adicional:

El proyecto incorpora una interfaz gráfica de usuario (GUI) desarrollada con Swing en Main.java, permitiendo a los usuarios ingresar código y visualizar los resultados del análisis. Se ha implementado el manejo de errores sintácticos, utilizando ErrorAnalizador.java para reportar errores en la sintaxis del código, como la ausencia de la palabra clave "Table" o el uso de tipos de variables no válidos. Estas características cumplen con los componentes opcionales de la especificación.

Fuentes y contenido relacionado

Componentes Principales y su Función

El proyecto está estructurado en varias clases Java, cada una con una responsabilidad específica:

***Main*.java:**

Función: Es la clase principal que contiene el método main y crea la interfaz gráfica de usuario (GUI) para la aplicación.

Funcionamiento:

- Inicializa la ventana principal titulada "Analizador Léxico y Sintáctico Simplificado".
- Configura dos áreas de texto: una para que el usuario ingrese el código y otra para mostrar los resultados del análisis.
- Incluye un botón "Analizar Código" que, al ser presionado, activa el proceso de análisis.
- Inicializa las clases ErrorAnalizador, Lexico y Sintactico.
- Cuando se analiza el código, primero llama al analizador léxico y luego al sintáctico.

- Muestra los tokens encontrados y el resultado del análisis sintáctico en el área de resultados.

Lexico.java (Analizador Léxico):

Función: Se encarga de tomar el código fuente como entrada y dividirlo en una secuencia de componentes léxicos llamados *tokens*.

Funcionamiento:

- Recorre el código fuente línea por línea y carácter por carácter.
- Identifica palabras clave ("Create", "Table", "Text", "Number", "Bool"), identificadores (nombres de tablas y variables), y delimitadores ('{', '}', ';').
- Crea objetos Token para cada componente léxico encontrado, almacenando el lexema (el string del token), su fila y las columnas de inicio y fin.
- Si encuentra un carácter inesperado que no pertenece a la gramática léxica definida, reporta un error léxico a través de ErrorAnalizador.

Resultados: Una lista de objetos Token.

Sintactico.java (Analizador Sintáctico):

Función: Toma la lista de tokens generada por el analizador léxico y verifica si la secuencia de tokens sigue la gramática definida para el lenguaje. También realiza un análisis semántico básico, como verificar la duplicidad de nombres de variables.

Funcionamiento:

- Implementa un analizador sintáctico descendente recursivo.
- Tiene reglas para la estructura esperada del código, que parece ser una sentencia "Create Table" seguida de un nombre de tabla, llaves, y dentro de las llaves, declaraciones de variables (tipo y nombre, terminadas en punto y coma).
- La gramática principal que parece seguir es: CreateTable -> "Create" "Table" IDENTIFICADOR "{" ListaDeclaraciones "}" ListaDeclaraciones -> DeclaracionVariable ListaDeclaraciones | ε DeclaracionVariable -> Tipo IDENTIFICADOR ";" Tipo -> "Text" | "Number" | "Bool"
- Utiliza un método esperar(String tipoEsperado) para consumir el siguiente token si coincide con lo esperado, o reportar un error sintáctico si no.

- Valida que se defina al menos una variable en la tabla.
- Comprueba si un nombre de variable ya ha sido definido (error semántico).

Resultados:

- Un mensaje de éxito si el análisis sintáctico es correcto.
- El nombre de la tabla identificada.
- Una lista de los nombres de las variables y sus tipos correspondientes.
- Mensajes de error (a través de ErrorAnalizador) si se encuentran errores sintácticos o semánticos.

Token.java:

Función: Es una clase simple para representar un token.

Estructura: Contiene campos para el token (el lexema o valor del token), la fila donde se encontró, y las posiciones x (columna de inicio) e y (columna de fin).

ErrorAnalizador.java:

Función: Proporciona una forma estandarizada de mostrar mensajes de error al usuario.

Funcionamiento: Crea una ventana emergente (un JFrame) que muestra el mensaje de error y un botón "Aceptar" para cerrar la ventana. Los mensajes pueden tener saltos de línea.

Tokens Encontrados:

Una lista de todos los tokens identificados por el analizador léxico, cada uno con su tipo (palabra clave, identificador, delimitador), el lexema y su posición (fila, columna). Por ejemplo:

--- Tokens Encontrados ---

Token{token='Create', fila=1, x=0, y=5}

Token{token='Table', fila=1, x=7, y=11}

Token{token='MiTabla', fila=1, x=13, y=19}

Token{token='{', fila=1, x=21, y=21}

Token{token='Text', fila=2, x=2, y=5}

Token{token='Nombre', fila=2, x=7, y=12}

Token{token=';', fila=2, x=13, y=13}

Análisis Sintáctico:

Si el código es sintácticamente correcto según la gramática definida:

- Un mensaje "Análisis sintáctico completado exitosamente."
- El nombre de la tabla: Tabla: [NombreDeLaTabla]
- Las variables y sus tipos: Variables: [[Variable1, Variable2, ...]] y Tipos: [[Tipo1, Tipo2, ...]]

Si hay errores sintácticos o semánticos (como un token inesperado, una variable duplicada o ninguna variable definida):

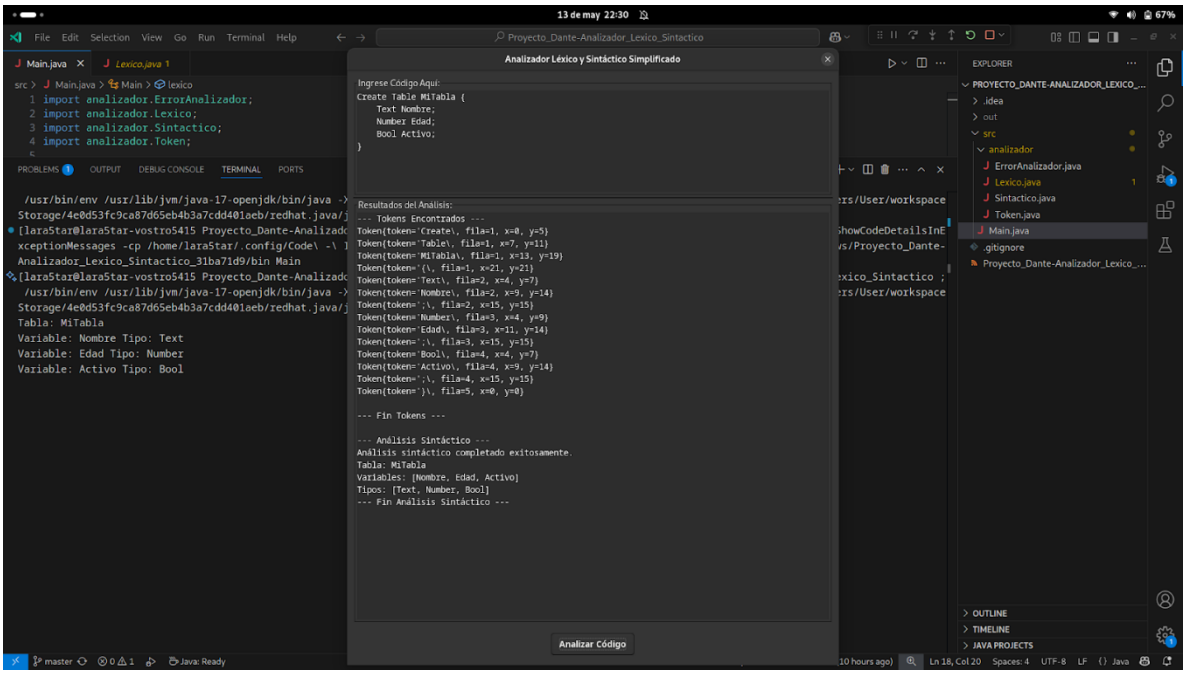
- Un mensaje como "El análisis sintáctico falló. Revise la ventana de errores."
- Adicionalmente, la clase ErrorAnalizador mostrará una ventana emergente con un mensaje de error detallado, indicando la naturaleza del error y, a menudo, la posición (fila y columna) donde ocurrió.

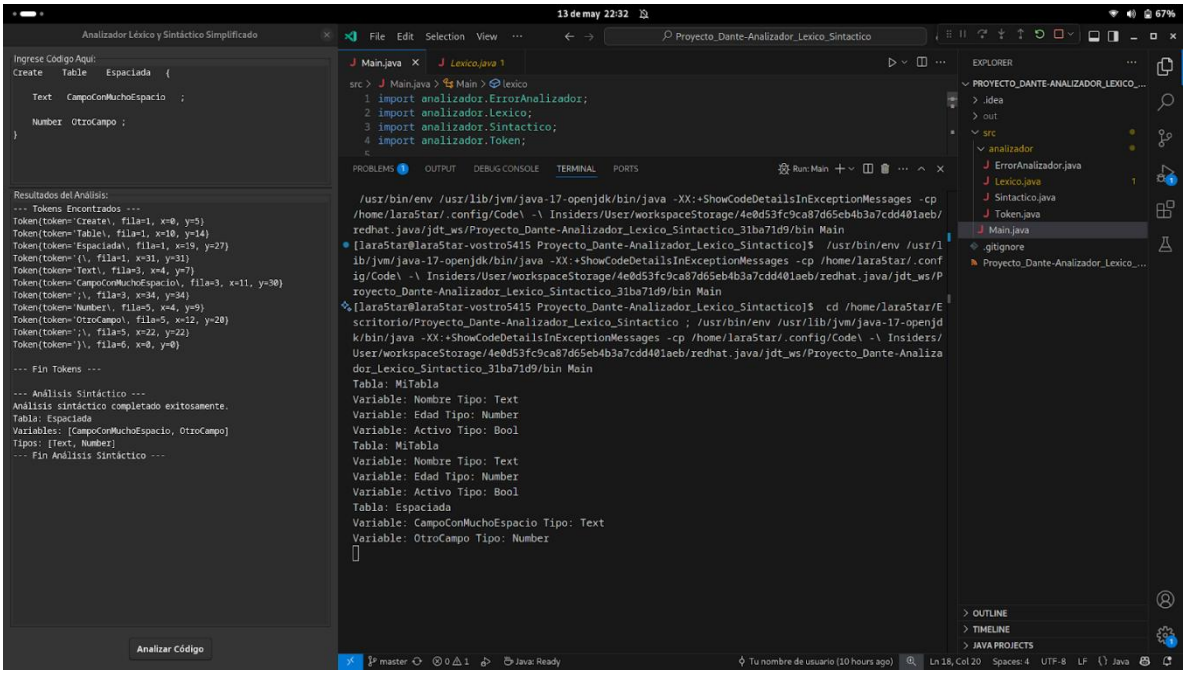
Flujo General de Trabajo

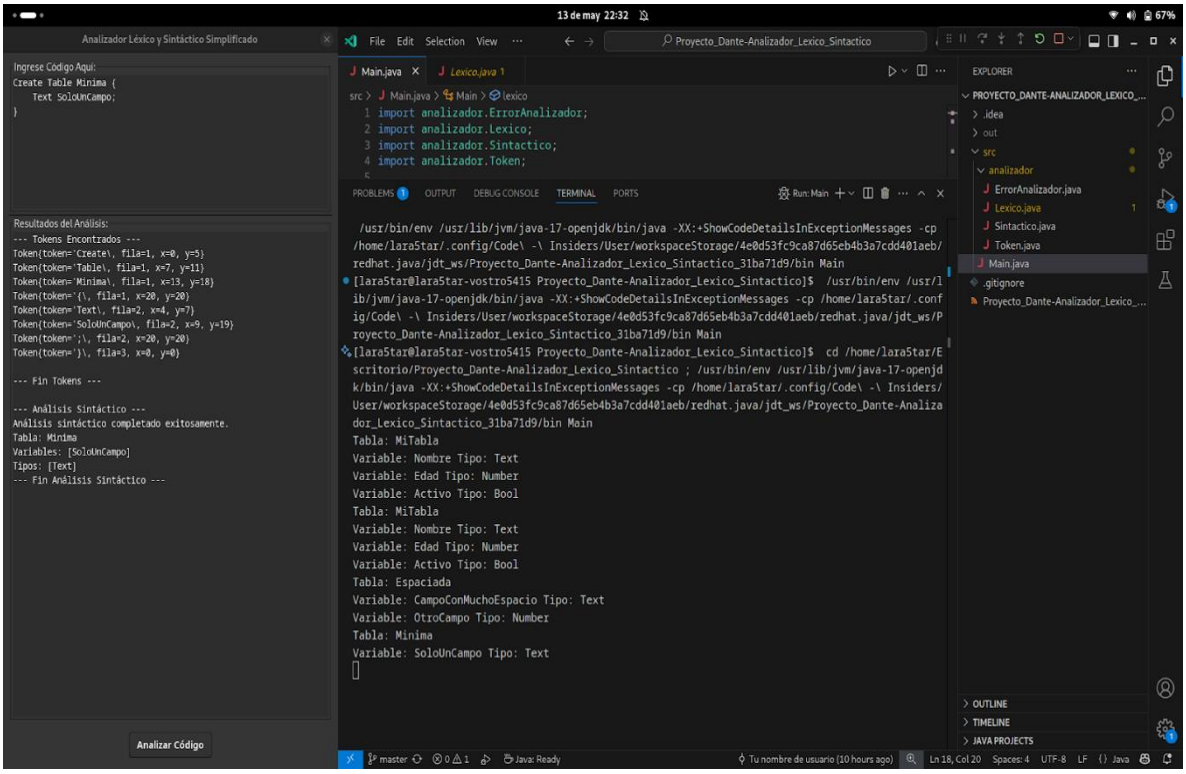
1. El usuario introduce el código en el JTextArea areaCodigo.
2. El usuario hace clic en el botón botonAnalizar.
3. El método analizarCodigo() en Main.java se ejecuta.
4. Se limpia el areaResultados.
5. Se llama al método analizar() de la instancia de Lexico con el código fuente.
    - Lexico devuelve una lista de Token o una lista vacía si hay errores léxicos graves (mostrados por ErrorAnalizador).
    - Los tokens se muestran en areaResultados.

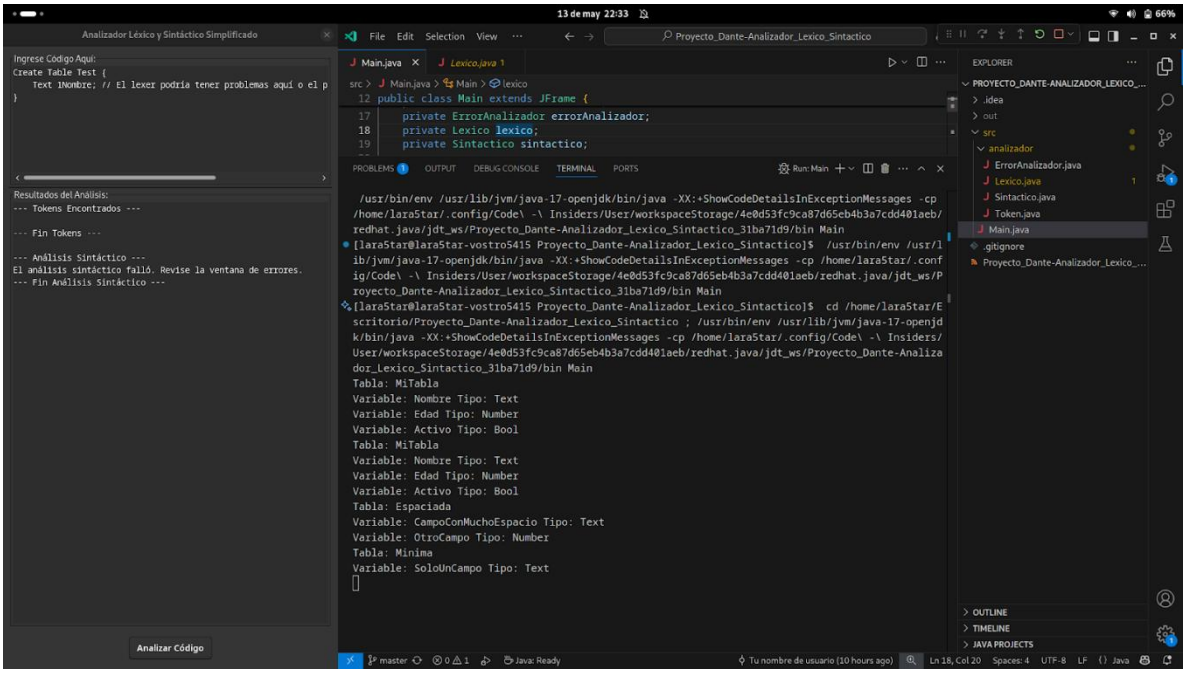
6. Se llama al método analizar() de la instancia de Sintactico con la lista de tokens.
    - Sintactico verifica la gramática y la semántica básica.
    - Si tiene éxito, se muestran los detalles de la tabla (nombre, variables, tipos).
    - Si falla, se muestra un mensaje de fallo y ErrorAnalizador muestra el error específico.

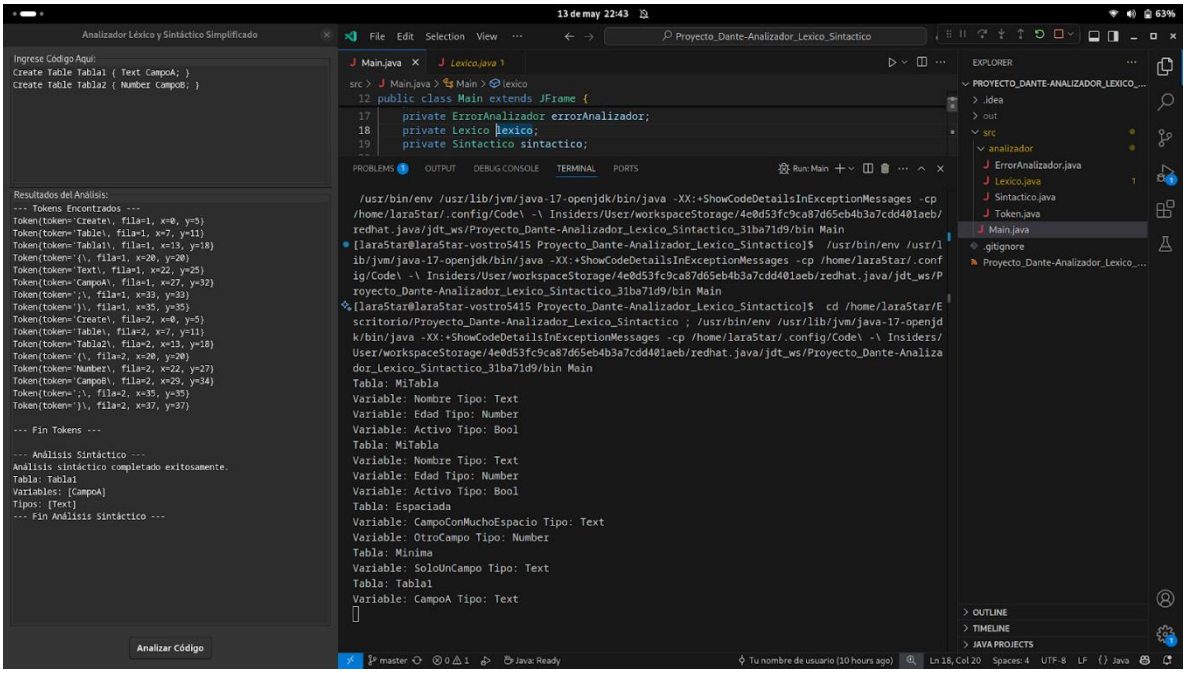
Pruevas Realizadas y casos de error analizados

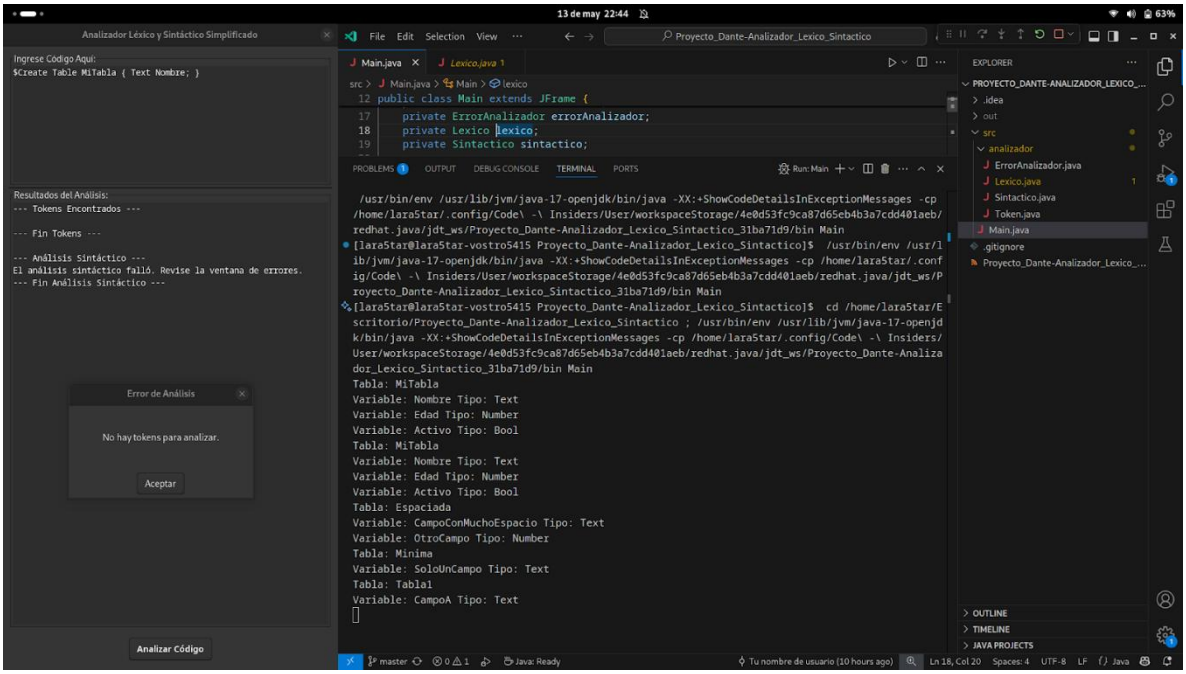


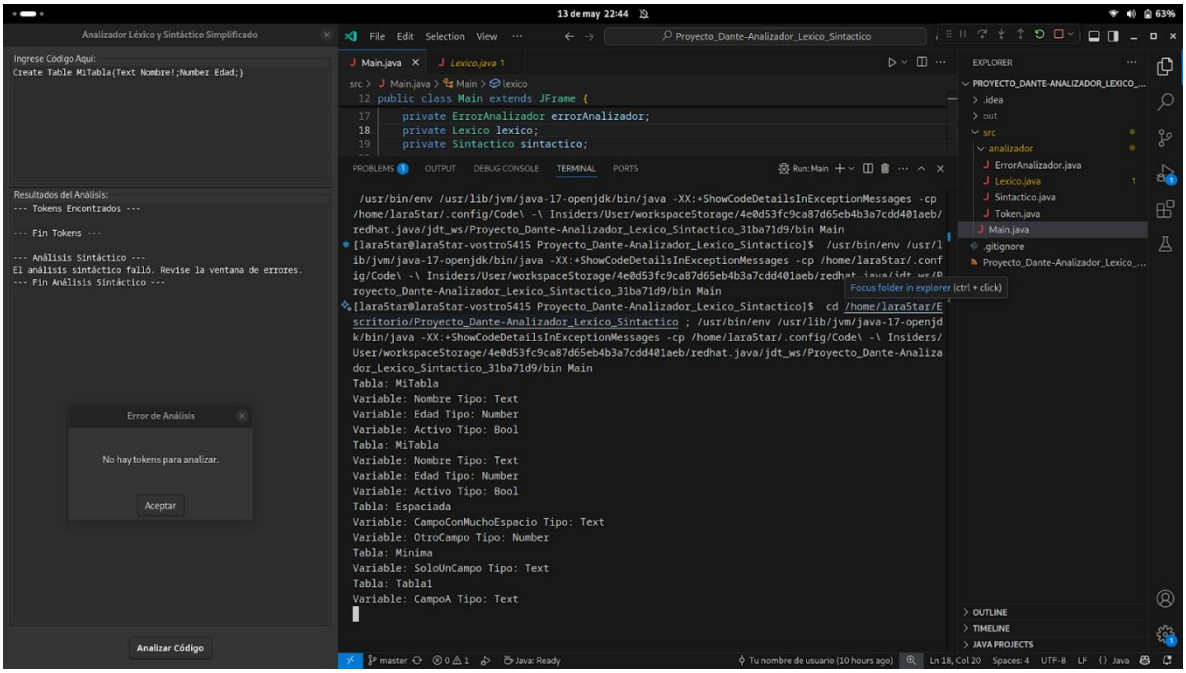


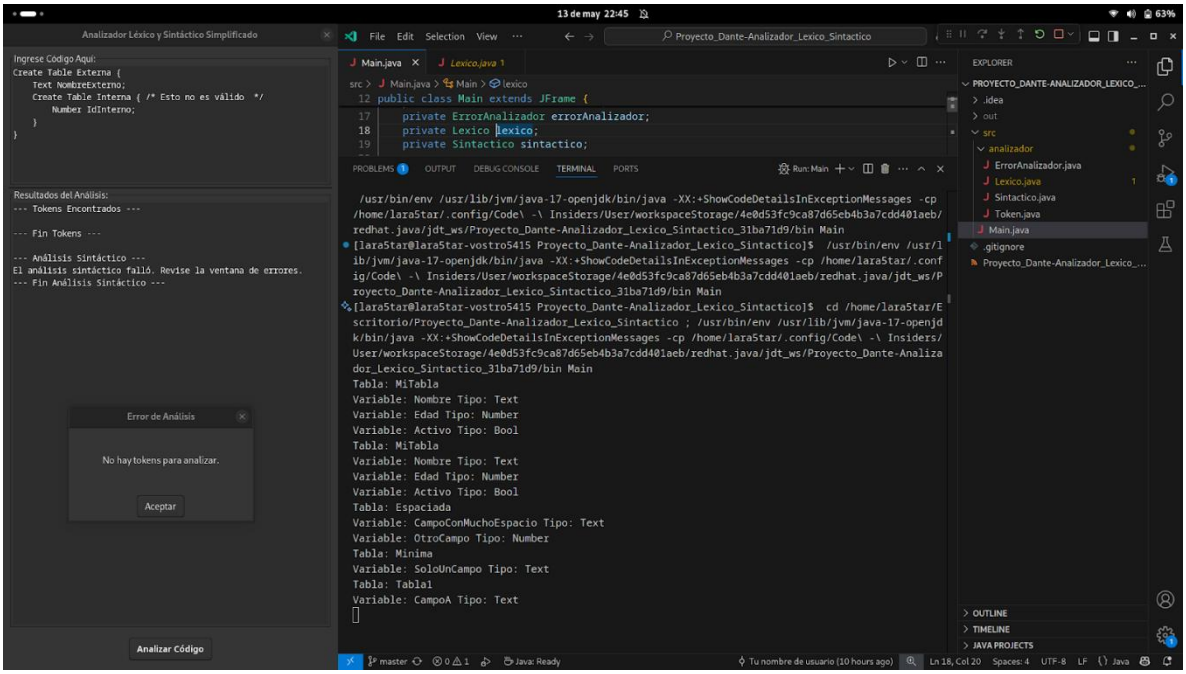

13 de may 22:32
Analizador Léxico y Sintáctico Simplificado
Ingrese Código Aquí:
Create Table Minima {
Text SoloUnCampo;
}
Resultados del Análisis:
--- Tokens Encontrados ---
Token{token='Create\, fila=1, x=0, y=5}
Token{token='Table\, fila=1, x=7, y=11}
Token{token='Minima\, fila=1, x=13, y=18}
Token{token='{\, fila=1, x=20, y=20}
Token{token='Text\, fila=2, x=4, y=7}
Token{token='SoloUnCampo\, fila=2, x=9, y=19}
Token{token=';\, fila=2, x=20, y=20}
Token{token='}\, fila=3, x=0, y=0}
--- Fin Tokens ---
--- Análisis Sintáctico ---
Análisis sintáctico completado exitosamente.
Tabla: Minima
Variables: [SoloUnCampo]
Tipos: [Text]
--- Fin Análisis Sintáctico ---
Analizar Código
Proyecto_Dante-Analizador_Lexico_Sintactico
Main.java
Lexico.java
import analizador.ErrorAnalizador;
import analizador.Lexico;
import analizador.Sintactico;
import analizador.Token;
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Run: Main
/usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
[lara5tar@lara5tar-vostro5415 Proyecto_Dante-Analizador_Lexico_Sintactico]$ /usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
[lara5tar@lara5tar-vostro5415 Proyecto_Dante-Analizador_Lexico_Sintactico]$ cd /home/lara5tar/Escritorio/Proyecto_Dante-Analizador_Lexico_Sintactico ; /usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: Espaciada
Variable: CampoConMuchoEspacio Tipo: Text
Variable: OtroCampo Tipo: Number
Tabla: Minima
Variable: SoloUnCampo Tipo: Text
EXPLORER
PROYECTO_DANTE-ANALIZADOR_LEXICO_...
.idea
out
src
analizador
ErrorAnalizador.java
Lexico.java
Sintactico.java
Token.java
Main.java
.gitignore
Proyecto_Dante-Analizador_Lexico_...
OUTLINE
TIMELINE
JAVA PROJECTS
master
Java: Ready
Tu nombre de usuario (10 hours ago)
Ln 18, Col 20
Spaces: 4
UTF-8
LF
Java


13 de may 22:33
Analizador Léxico y Sintáctico Simplificado
Ingrese Código Aquí:
Create Table Test {
Text 1Nombre; // El lexer podría tener problemas aquí o el p
}
Resultados del Análisis:
--- Tokens Encontrados ---
--- Fin Tokens ---
--- Análisis Sintáctico ---
El análisis sintáctico falló. Revise la ventana de errores.
--- Fin Análisis Sintáctico ---
Analizar Código
Proyecto_Dante-Analizador_Lexico_Sintactico
Main.java
Lexico.java
public class Main extends JFrame {
private ErrorAnalizador errorAnalizador;
private Lexico lexico;
private Sintactico sintactico;
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Run: Main
/usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
[lara5tar@lara5tar-vostro5415 Proyecto_Dante-Analizador_Lexico_Sintactico]$ /usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
[lara5tar@lara5tar-vostro5415 Proyecto_Dante-Analizador_Lexico_Sintactico]$ cd /home/lara5tar/Escritorio/Proyecto_Dante-Analizador_Lexico_Sintactico ; /usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: Espaciada
Variable: CampoConMuchoEspacio Tipo: Text
Variable: OtroCampo Tipo: Number
Tabla: Minima
Variable: SoloUnCampo Tipo: Text
EXPLORER
PROYECTO_DANTE-ANALIZADOR_LEXICO_...
.idea
out
src
analizador
ErrorAnalizador.java
Lexico.java
Sintactico.java
Token.java
Main.java
.gitignore
Proyecto_Dante-Analizador_Lexico_...
OUTLINE
TIMELINE
JAVA PROJECTS
master
Java: Ready
Tu nombre de usuario (10 hours ago)
Ln 18, Col 20
Spaces: 4
UTF-8
LF
Java


Analizador Léxico y Sintáctico Simplificado
Ingrese Código Aquí:
Create Table Tabla1 { Text CampoA; }
Create Table Tabla2 { Number CampoB; }
Resultados del Análisis:
--- Tokens Encontrados ---
Token{token='Create\', fila=1, x=0, y=5}
Token{token='Table\', fila=1, x=7, y=11}
Token{token='Tabla1\', fila=1, x=13, y=18}
Token{token='{\, fila=1, x=20, y=20}
Token{token='Text\, fila=1, x=22, y=25}
Token{token='CampoA\, fila=1, x=27, y=32}
Token{token=';\, fila=1, x=33, y=33}
Token{token='}\, fila=1, x=35, y=35}
Token{token='Create\, fila=2, x=0, y=5}
Token{token='Table\, fila=2, x=7, y=11}
Token{token='Tabla2\, fila=2, x=13, y=18}
Token{token='{\, fila=2, x=20, y=20}
Token{token='Number\, fila=2, x=22, y=27}
Token{token='CampoB\, fila=2, x=29, y=34}
Token{token=';\, fila=2, x=35, y=35}
Token{token='}\, fila=2, x=37, y=37}
--- Fin Tokens ---
--- Análisis Sintáctico ---
Análisis sintáctico completado exitosamente.
Tabla: Tabla1
Variables: [CampoA]
Tipos: [Text]
--- Fin Análisis Sintáctico ---
Analizar Código
13 de may 22:43
File Edit Selection View
Proyecto_Dante-Analizador_Lexico_Sintactico
Main.java
Lexico.java
public class Main extends JFrame {
private ErrorAnalizador errorAnalizador;
private Lexico lexico;
private Sintactico sintactico;
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Run: Main
/usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
[lara5tar@lara5tar-vostro5415 Proyecto_Dante-Analizador_Lexico_Sintactico]$ /usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
[lara5tar@lara5tar-vostro5415 Proyecto_Dante-Analizador_Lexico_Sintactico]$ cd /home/lara5tar/Escritorio/Proyecto_Dante-Analizador_Lexico_Sintactico ; /usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: Espaciada
Variable: CampoConMuchoEspacio Tipo: Text
Variable: OtroCampo Tipo: Number
Tabla: Miníma
Variable: SoloUnCampo Tipo: Text
Tabla: Tabla1
Variable: CampoA Tipo: Text
EXPLORER
PROYECTO_DANTE-ANALIZADOR_LEXICO_...
.idea
out
src
analizador
ErrorAnalizador.java
Lexico.java
Sintactico.java
Token.java
Main.java
.gitignore
Proyecto_Dante-Analizador_Lexico_...
OUTLINE
TIMELINE
JAVA PROJECTS
master
Java: Ready
Tu nombre de usuario (10 hours ago)
Ln 18, Col 20 Spaces: 4 UTF-8 LF Java


Analizador Léxico y Sintáctico Simplificado
Ingrese Código Aquí:
$Create Table MiTabla { Text Nombre; }
Resultados del Análisis:
--- Tokens Encontrados ---
--- Fin Tokens ---
--- Análisis Sintáctico ---
El análisis sintáctico falló. Revise la ventana de errores.
--- Fin Análisis Sintáctico ---
Error de Análisis
No hay tokens para analizar.
Aceptar
Analizar Código
13 de may 22:44
File Edit Selection View
Proyecto_Dante-Analizador_Lexico_Sintactico
Main.java
Lexico.java
public class Main extends JFrame {
private ErrorAnalizador errorAnalizador;
private Lexico lexico;
private Sintactico sintactico;
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Run: Main
/usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
[lara5tar@lara5tar-vostro5415 Proyecto_Dante-Analizador_Lexico_Sintactico]$ /usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
[lara5tar@lara5tar-vostro5415 Proyecto_Dante-Analizador_Lexico_Sintactico]$ cd /home/lara5tar/Escritorio/Proyecto_Dante-Analizador_Lexico_Sintactico ; /usr/bin/env /usr/lib/jvm/java-17-openjdk/bin/java -XX:+ShowCodeDetailsInExceptionMessages -cp /home/lara5tar/.config/Code\ -\ Insiders/User/workspaceStorage/4e0d53fc9ca87d65eb4b3a7cdd401aeb/redhat.java/jdt_ws/Proyecto_Dante-Analizador_Lexico_Sintactico_31ba71d9/bin Main
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: Espaciada
Variable: CampoConMuchoEspacio Tipo: Text
Variable: OtroCampo Tipo: Number
Tabla: Miníma
Variable: SoloUnCampo Tipo: Text
Tabla: Tabla1
Variable: CampoA Tipo: Text
EXPLORER
PROYECTO_DANTE-ANALIZADOR_LEXICO_...
.idea
out
src
analizador
ErrorAnalizador.java
Lexico.java
Sintactico.java
Token.java
Main.java
.gitignore
Proyecto_Dante-Analizador_Lexico_...
OUTLINE
TIMELINE
JAVA PROJECTS
master
Java: Ready
Tu nombre de usuario (10 hours ago)
Ln 18, Col 20 Spaces: 4 UTF-8 LF Java


Analizador Léxico y Sintáctico Simplificado
Ingrese Código Aquí:
Create Table MiTabla{Text Nombre!;Number Edad;}
Resultados del Análisis:
--- Tokens Encontrados ---
--- Fin Tokens ---
--- Análisis Sintáctico ---
El análisis sintáctico falló. Revise la ventana de errores.
--- Fin Análisis Sintáctico ---
Error de Análisis
No hay tokens para analizar.
Aceptar
Analizar Código
Main.java
Lexico.java
public class Main extends JFrame {
private ErrorAnalizador errorAnalizador;
private Lexico lexico;
private Sintactico sintactico;
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: Espaciada
Variable: CampoConMuchoEspacio Tipo: Text
Variable: OtroCampo Tipo: Number
Tabla: Minima
Variable: SoloUnCampo Tipo: Text
Tabla: Tabla1
Variable: CampoA Tipo: Text


Analizador Léxico y Sintáctico Simplificado
Ingrese Código Aquí:
Create Table Externa {
Text NombreExterno;
Create Table Interna { /* Esto no es válido */
Number IdInterno;
}
}
Resultados del Análisis:
--- Tokens Encontrados ---
--- Fin Tokens ---
--- Análisis Sintáctico ---
El análisis sintáctico falló. Revise la ventana de errores.
--- Fin Análisis Sintáctico ---
Error de Análisis
No hay tokens para analizar.
Aceptar
Analizar Código
Main.java
Lexico.java
public class Main extends JFrame {
private ErrorAnalizador errorAnalizador;
private Lexico lexico;
private Sintactico sintactico;
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: MiTabla
Variable: Nombre Tipo: Text
Variable: Edad Tipo: Number
Variable: Activo Tipo: Bool
Tabla: Espaciada
Variable: CampoConMuchoEspacio Tipo: Text
Variable: OtroCampo Tipo: Number
Tabla: Minima
Variable: SoloUnCampo Tipo: Text
Tabla: Tabla1
Variable: CampoA Tipo: Text

Enlace al repositorio del proyecto:
https://docs.google.com/document/d/1B-pRRODyvEnkCwkuKFDXL90yNJPGsVUc6YHK7GOZDh8/edit?tab=t.0