**Tecnológico de Costa Rica**

**1° Proyecto**

**Analizador Léxico y semántico**

**Curso: Compiladores e interpretes**

**Profesor**

**Allan Rodríguez Davila**

**Estudiantes**

**Cristopher Zúñiga Jiménez**

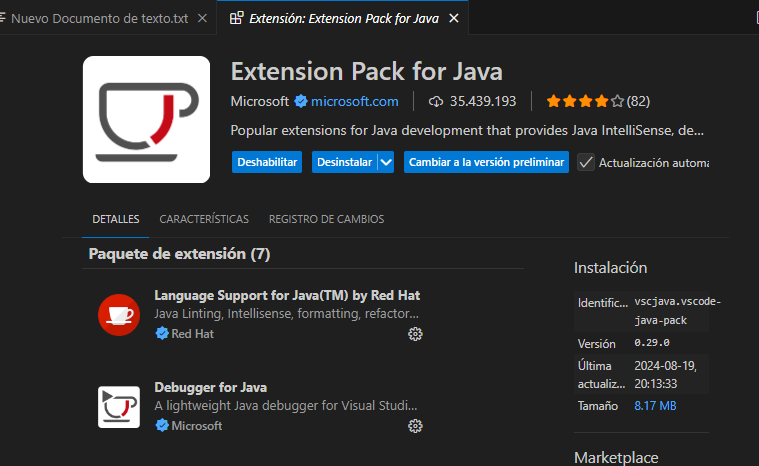
**2025**

# Descripción del Problema

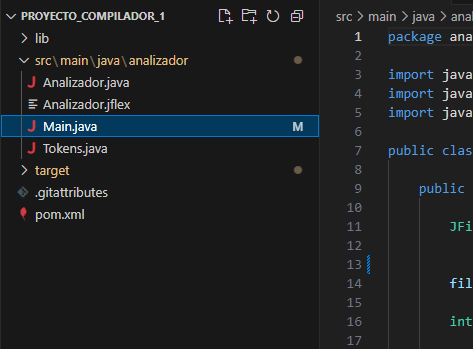
Un grupo de desarrolladores desea crear un nuevo lenguaje imperativo, ligero, que le permita realizar operaciones básicas para la configuración de chips, ya que esta es una industria que sigue creciendo constantemente, y cada vez estos chips necesitan ser configurados por lenguajes más ligeros y potentes. Es por esto por lo que este grupo de desarrolladores requiere desarrollar su propio lenguaje para el desarrollo de sistemas empotrados.

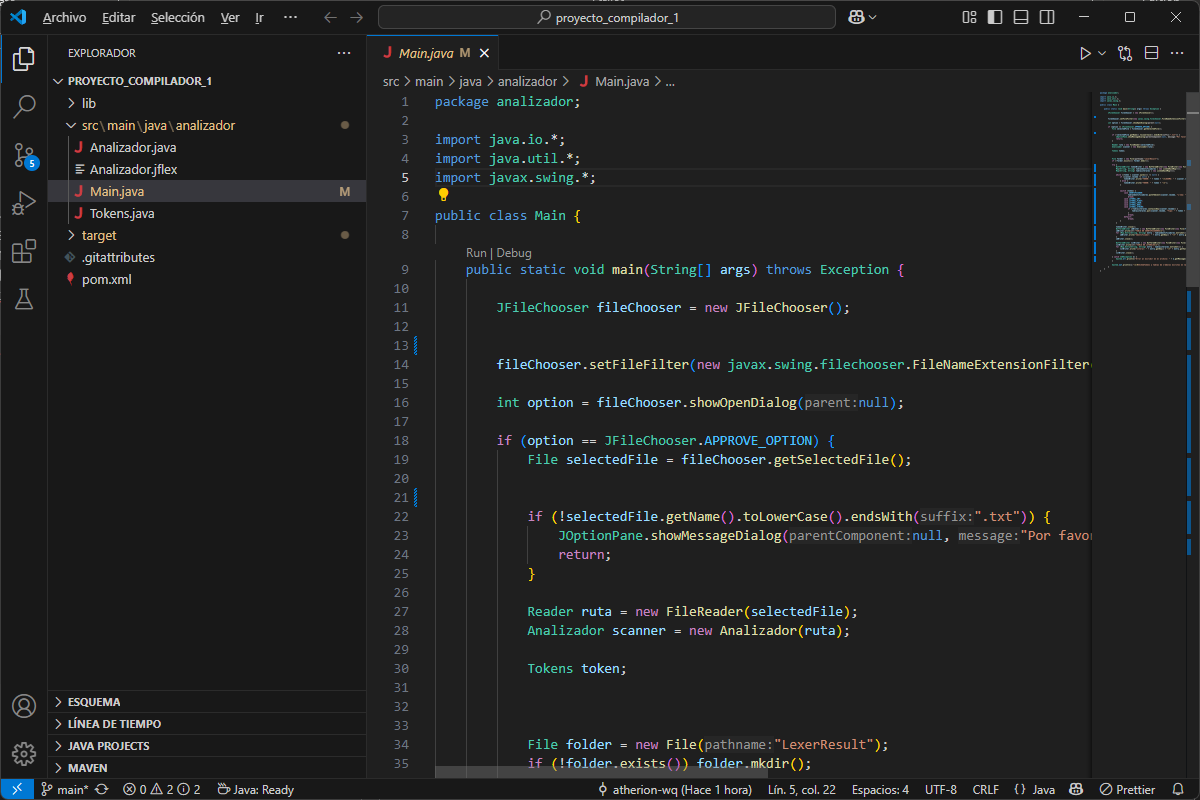
# Manual de Usuario y Pruebas de funcionalidad.

Para ejecutar el proyecto, es necesario tener instalado **Visual Studio Code** junto con la extensión **Extensión Pack for Java**. Con estas herramientas, ya es posible compilar y ejecutar el código sin configuraciones adicionales.

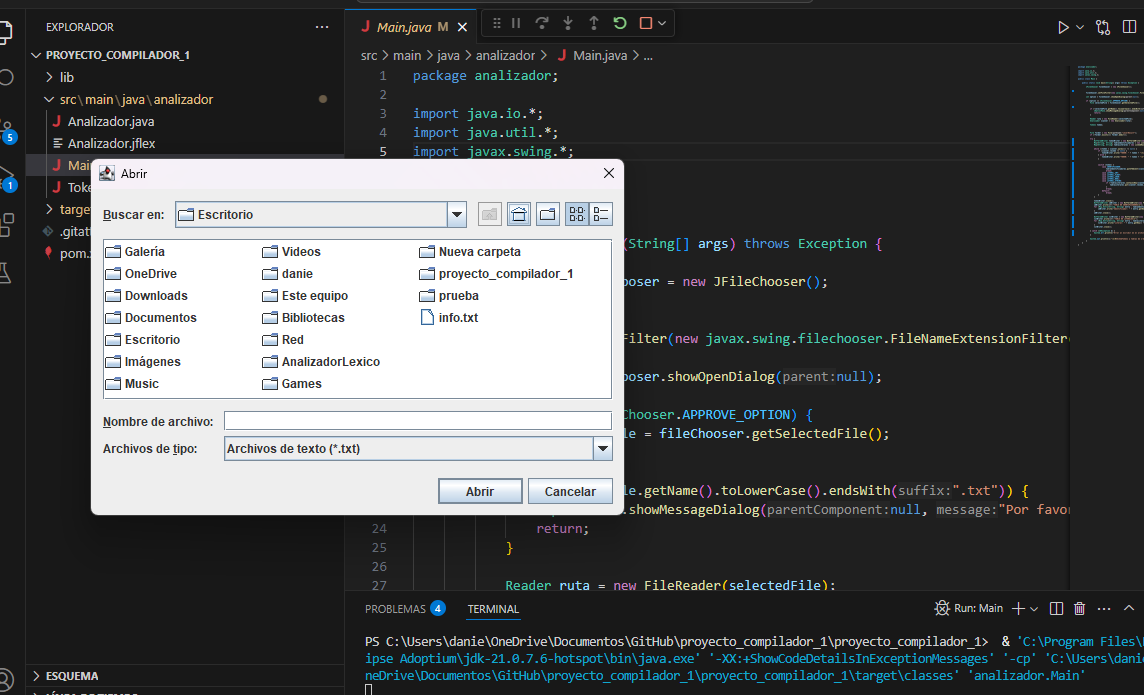


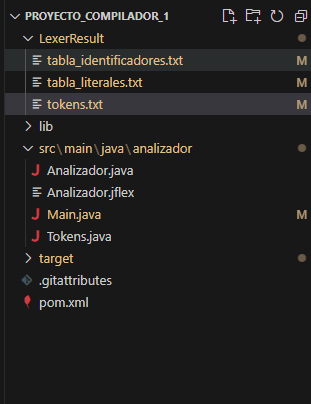
Una vez abierto el proyecto, diríjase al archivo Main.java y presione el botón de **ejecución (play)** ubicado en la parte superior del editor.



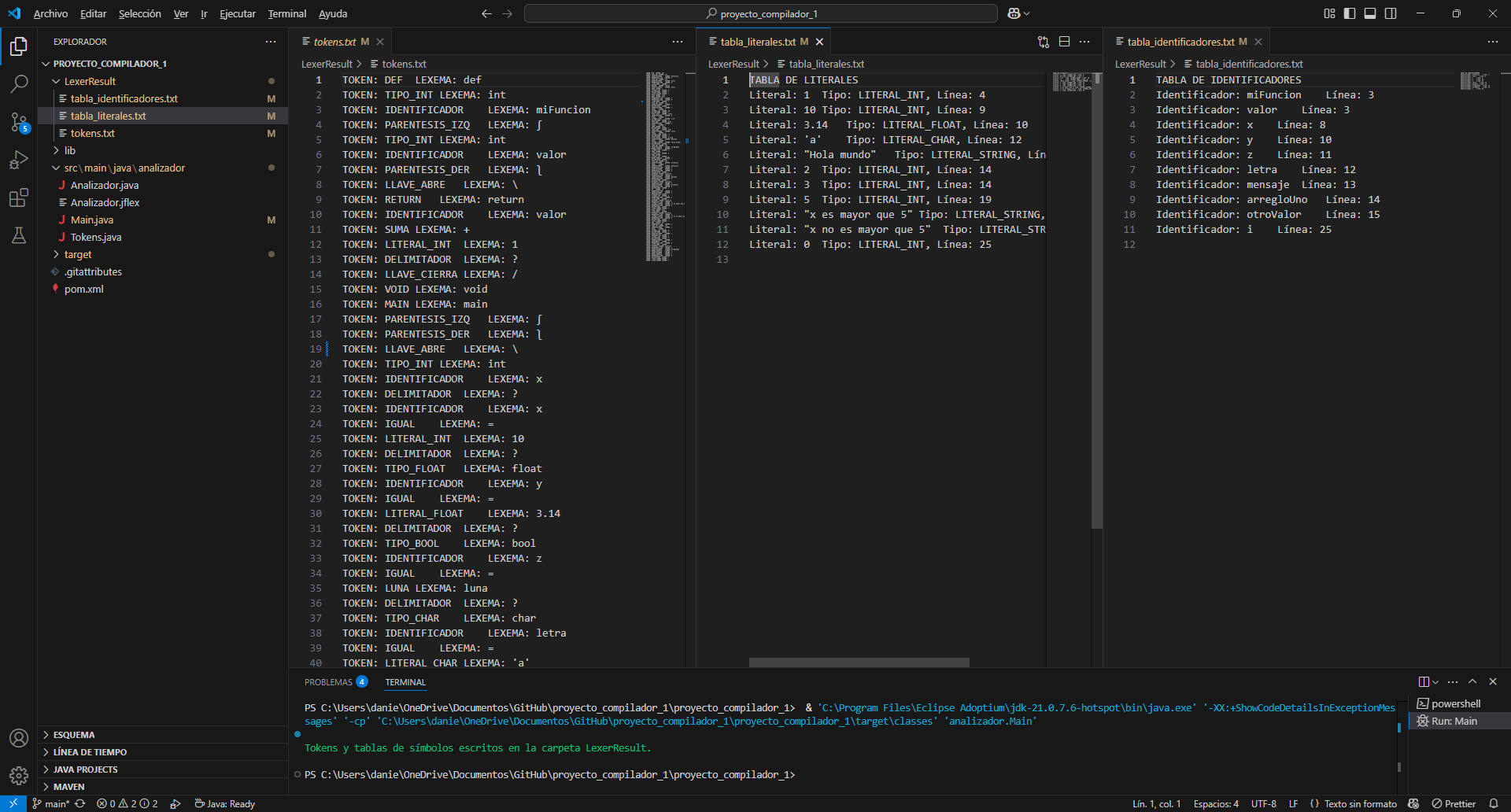


El programa abrirá una consola en la que se deberá seleccionar un archivo con formato .txt. A partir de este archivo, el programa generará una lista de tokens y dos tablas que se almacenarán en una carpeta llamada **LexerResult**, ubicada en la raíz del repositorio.





En los resultados generados, se puede observar la lista de **tokens** junto con sus respectivos **lexemas**, y una **tabla de símbolos** que contiene información detallada sobre cada token: su tipo y la línea en la que aparece en el archivo de entrada.



# Diseño del Programa

* Se crea una carpeta con las tablas para comodidad.
* Desarrollo de interfaz para la selección del archivo más cómoda para el usuario.
* Se creó en base a la gramática desarrollada en la tarea y se realizaron los cambios propuestos por el profesor. Además, se hizo replanteamiento por problemas por la precedencia en algunos casos.

# Librerías Usadas

* JFlex

# Análisis de Resultados

Se llega a la conclusión de que realiza el análisis léxico correspondiente y demarca aquellos que no corresponden a la gramática propuesta. Pero, se presentaron problemas con la precedencia de algunos lexemas y la ambigüedad del código, eso llevando a una gran cantidad de warnings a la hora de tratar de ejecutar el cup, múltiple de los errores vienen por parte de las expresiones matemáticas debido a que su producción es similar una con otra, se intentó la corrección de estos warnings mas no se lograron solucionar todos por lo tanto el analizar sintáctico no se encuentra funcional. El analizador léxico y tabla de símbolos funciona como corresponde, pero todo lo demás no será incluida porque provoca que el código no sea ejecutable.