

Jeferson Manuel Aguilar Panjoj

202130718

MANUAL TÉCNICO

Este manual proporciona una descripción del sistema de análisis léxico implementado en Java. El presente programa permite a los usuarios ingresar código fuente a través de una interfaz gráfica, analizar el código para identificar y categorizar tokens, y visualizar los resultados en tiempo real. La aplicación utiliza un analizador léxico para descomponer el código en tokens y un visualizador para mostrar los resultados en una interfaz gráfica.

El sistema se compone de tres componentes principales:

Clase Principal

La clase Principal es la ventana principal de la aplicación. Gestiona la interacción del usuario y muestra los resultados del análisis léxico.

Componentes:

- **Área de Texto (jTextArea1):** Permite al usuario ingresar el código fuente.
- **Panel de Tokens (jPanel1):** Muestra los tokens identificados como cuadros de colores.
- **Botón de Historial (jButton1):** Muestra el historial de tokens en una ventana separada.

Métodos Clave:

- **Principal():** Constructor que inicializa los componentes y configura el escuchador de eventos para el área de texto.
- **LectorTextArea():** Configura un `DocumentListener` en el área de texto para detectar cambios y llamar al método `analizarCodigo()` en respuesta a modificaciones.
- **analizarCodigo():** Lee el texto del área de texto, utiliza el analizador léxico para obtener tokens, filtra los espacios en blanco y actualiza el panel con los tokens.
- **actualizarPanel(List<Token> tokens):** Actualiza el panel `jPanel1` para mostrar los tokens como cuadros de colores. Ajusta el tamaño y disposición de los cuadros en función del número de tokens y el tamaño del panel.
- **obtenerColorPorHex(String colorHex):** Convierte un valor hexadecimal de color a un objeto `Color`.
- **lectorTextoDelTextArea():** Obtiene el texto actual del área de texto.
- **jButton1ActionPerformed(java.awt.event.ActionEvent evt):** Muestra una ventana con el historial de tokens en formato de tabla.

Clase Analizador

La clase Analizador es responsable de descomponer el código fuente en tokens. Implementa la lógica de análisis léxico que identifica diferentes tipos de tokens.

Métodos Clave:

- **analizar(String codigo):** Analiza el código fuente y devuelve una lista de tokens. Identifica palabras reservadas, identificadores, números enteros y decimales, operadores, comentarios, cadenas, paréntesis, llaves, corchetes, comas, puntos y tokens especiales.

Clase Token

La clase Token representa un token en el código fuente e incluye información sobre su tipo, lexema, ubicación en el código y color asociado.

Atributos:

- **tipo:** El tipo de token (ej. "IDENTIFICADOR", "PALABRA RESERVADA", etc).
- **lexema:** El texto del token.
- **linea:** La línea del código donde aparece el token.
- **columna:** La columna del código donde aparece el token.
- **color:** El color asociado al token en formato hexadecimal.

Métodos Clave:

- **getTipo():** Retorna el tipo de token.
- **getLexema():** Retorna el lexema del token.
- **getLinea():** Retorna la línea del token.
- **getColumna():** Retorna la columna del token.
- **getColor():** Retorna el color del token.
- **toString():** Retorna una representación en cadena del token.

Código Fuente Detallado

Clase Principal

La clase Principal maneja la interfaz gráfica de usuario y la interacción con el usuario. Los métodos principales son:

- **analizarCodigo():** Procesa el texto del área de texto, utilizando el analizador léxico para identificar tokens, elimina los espacios en blanco y actualiza el panel de tokens.
- **actualizarPanel(List<Token> tokens):** Configura el diseño del panel para mostrar los tokens. Calcula el número de filas y columnas basado en el número de tokens y ajusta el tamaño de los cuadros de colores.
- **obtenerColorPorHex(String colorHex):** Convierte el color en formato hexadecimal a un objeto Color.

Clase Analizador

La clase Analizador es responsable de la lógica de análisis léxico. Los métodos principales incluyen:

- **analizar(String codigo):** Descompone el código en tokens, identificando palabras reservadas, identificadores, números, operadores, comentarios y cadenas de texto. Utiliza el valor de columna y línea para registrar la posición de cada token en el código fuente.

Clase Token

La clase Token encapsula la información de un token identificado. Sus atributos permiten la representación y el análisis de cada token en términos de tipo, lexema, ubicación y color.