



---

# ANÁLISIS SINTÁCTICO III

MTRO. MIGUEL ÁNGEL ROMO MARTÍNEZ

# ANÁLISIS SINTÁCTICO I – LEX/CUP

- Se generan los patrones (expresiones regulares) para el análisis léxico y sintáctico en los archivos:
  - Lexer.flex
  - LexerCup.flex
- Se ajustaron los token para ser utilizados como terminales (símbolos), esto en el archivo:
  - Token.java



# ANÁLISIS SINTÁCTICO II - LEX/CUP

- Se estableció para el análisis el manejo de errores (con set), además del método get(). Se declararon las terminales, el inicializador para el análisis y las producciones, esto en el archivo:
  - Syntax.cup
- Se establecieron las rutas para la generación de los 2 autómatas finitos, el árbol sintáctico y la tabla de símbolos:
  - Lexer.java
  - LexerCUP.java
  - sym.java
  - Syntax.java.



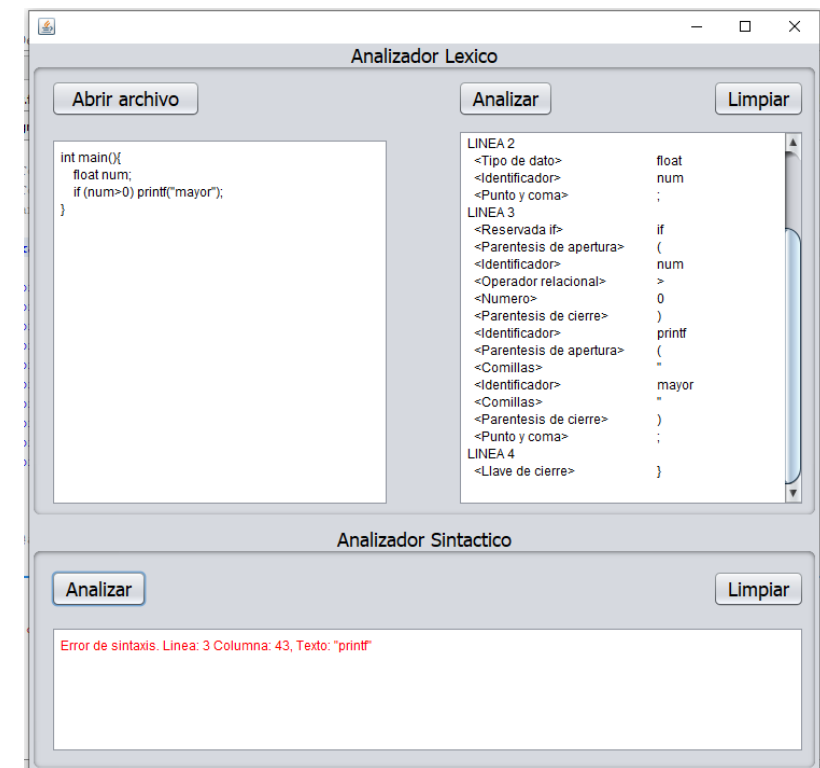
# ANÁLISIS SINTÁCTICO III - LEX/CUP

- Uso del Frame - Design

- Manejo de Paneles

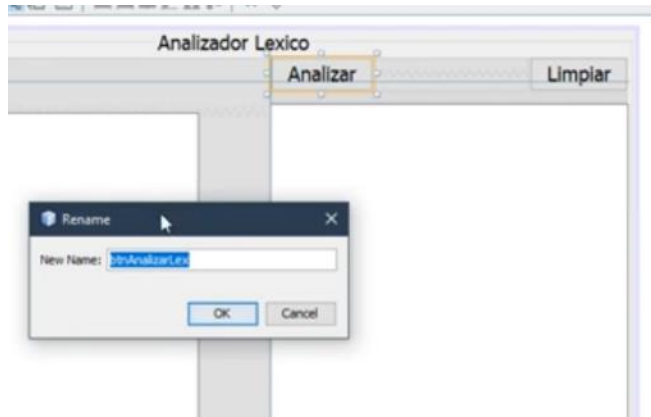


- Título: propiedades-borde-título de borde-título

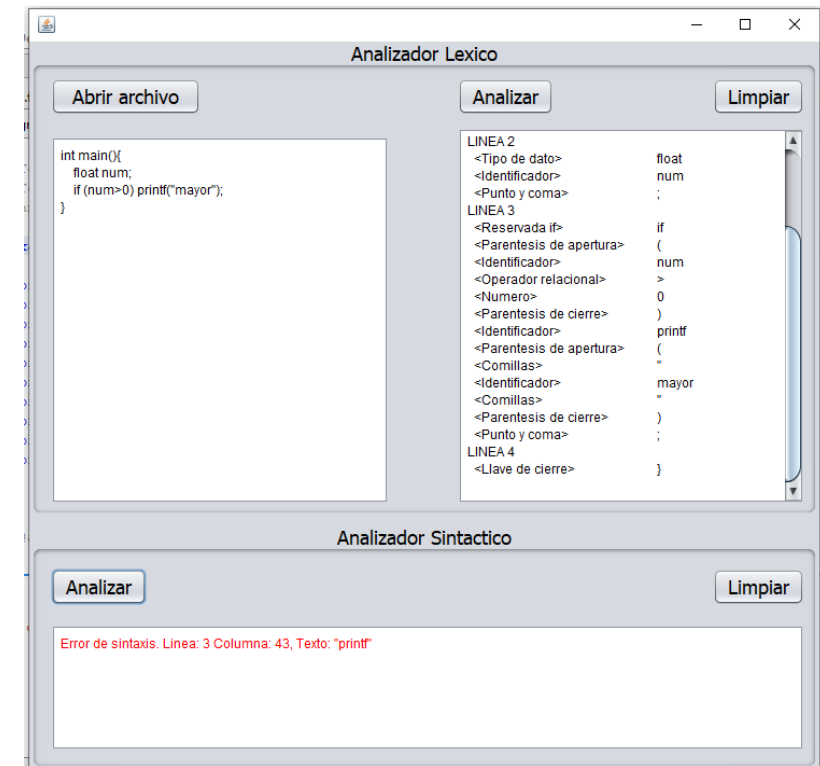


# ANÁLISIS SINTÁCTICO III - LEX/CUP

- Uso del Frame - Design
  - Se utilizan botones. Cambiar nombres.

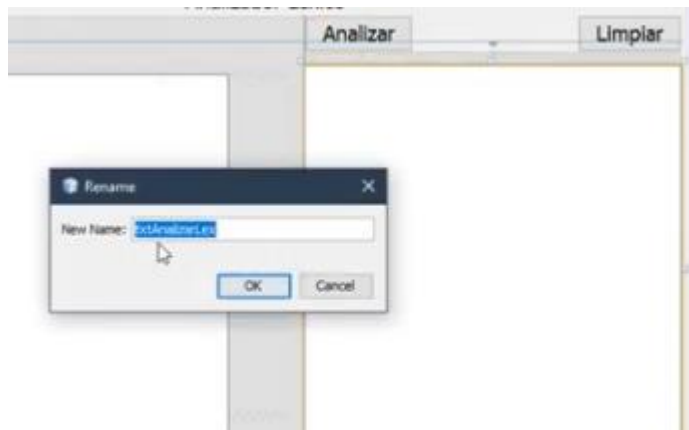


- btnArchivo – cargar archivos de texto
- btnAnalizarLex
- btnLimpiarLex

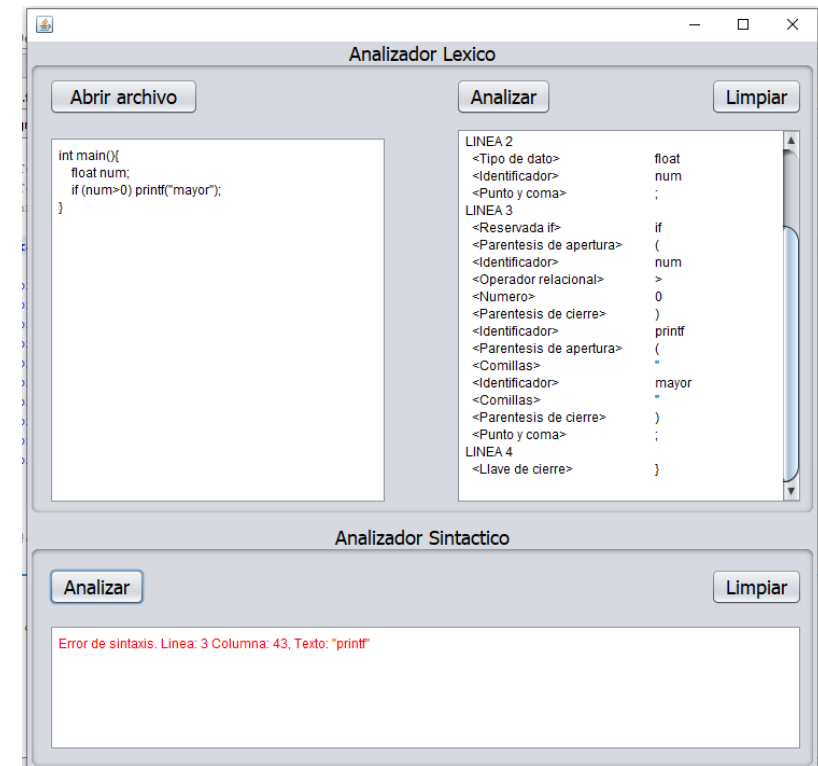


# ANÁLISIS SINTÁCTICO III - LEX/CUP

- Uso del Frame - Design
  - Se utilizan text area.



- txtArchivo
- txtAnalizarLex
- txtAnalizarSin



# ANÁLISIS SINTÁCTICO III - LEX/CUP

- Uso del Frame - Source

- Después del constructor agrega el método “analizarLexico”. Además realiza la llamada del mismo desde btnAnalizarLex. Permite que se cargue la excepción.

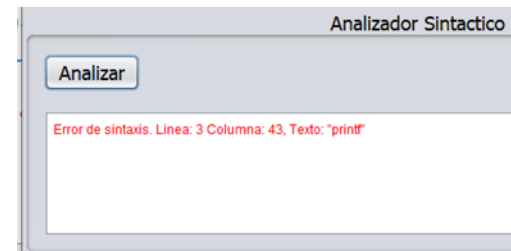
```
*  
* @author Dell  
*/  
public class FrmPrincipal extends javax.swing.JFrame {  
  
    /**  
     * Creates new form FrmPrincipal  
     */  
    public FrmPrincipal() {  
        initComponents();  
    }  
}
```

```
private void btnAnalizarLexActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btnAnalizarLexActionPerformed  
    try {  
        analizarLexico();  
    } catch (IOException ex) {  
        Logger.getLogger(FrmPrincipal.class.getName()).log(Level.SEVERE, null, ex);  
    }  
} //GEN-LAST:event_btnAnalizarLexActionPerformed
```

# ANÁLISIS SINTÁCTICO III - LEX/CUP

## ■ Uso del Frame - Source

- Carga el Código del analizador sintáctico en el btnAnalizadorSin.
- Agrega la línea de código para que se limpien los espacios de analisis.



```
private void btnLimpiarLexActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btnLimpiarLexActionPerformed
    // TODO add your handling code here:
    txtAnalizarLex.setText(null);
}//GEN-LAST:event_btnLimpiarLexActionPerformed

private void btnLimpiarSinActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btnLimpiarSinActionPerformed
    // TODO add your handling code here:
    txtAnalizarSin.setText(null);
}//GEN-LAST:event_btnLimpiarSinActionPerformed
```





# GRACIAS

**MAROMOM@UP.EDU.MX**

EVIDENCIA:

CON UNA IMPRESIÓN DE  
PANTALLA MUESTRA QUE  
FUNCIONAN TUS  
ANALIZADORES.