

## 70. JFlex | Analizador léxico con Java

*Previo:*

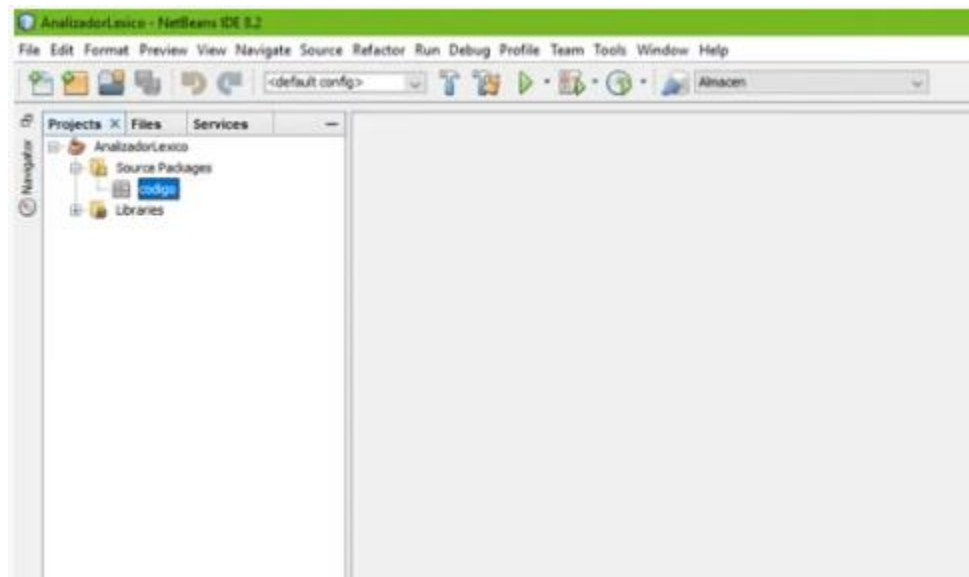
¿Qué es?

1. Patrón
2. Lexema
3. Token

*Explicación paso a paso:*

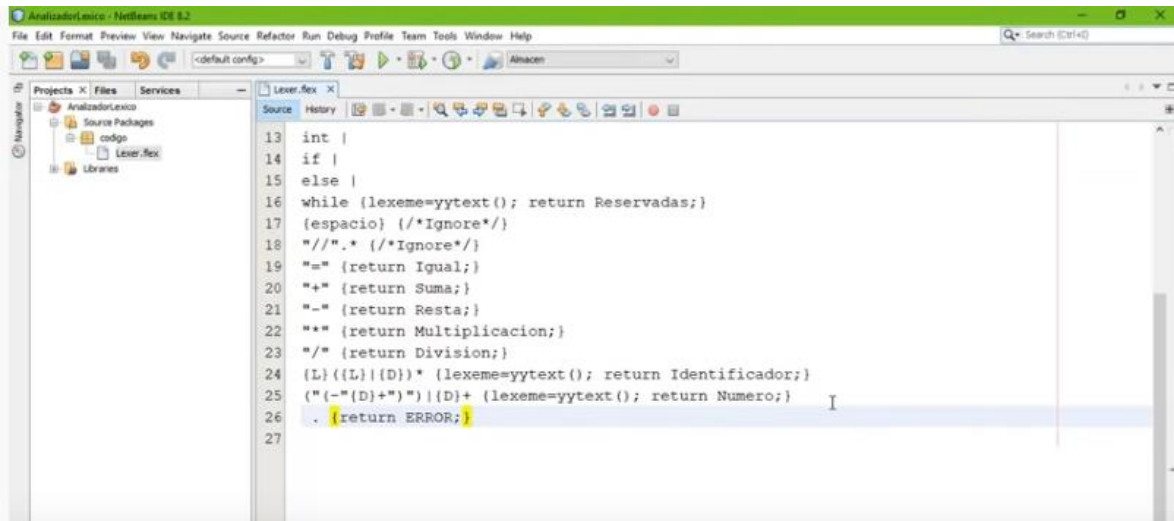
<https://youtu.be/5mIRrn2yEe8> 23mins 46segs

01:01 Creación básica del proyecto con la librería JFlex



1. Crear un proyecto llamado "AnalizadorLexico", sin MainClass.
2. Dentro del proyecto crear un nuevo Java Package llamado "codigo".
3. En las librerías del proyecto agregar un JAR/folder llamado JFlex.

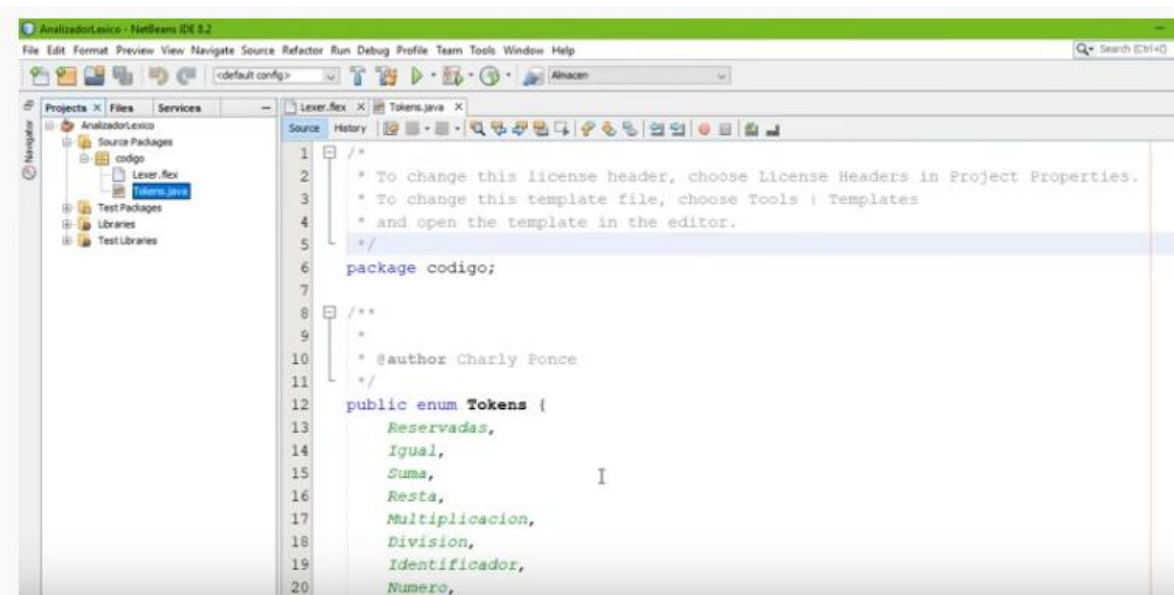
### 02:03 Reglas léxicas – Apoyo I.txt



```
13 int |
14 if |
15 else |
16 while {lexeme=yytext(); return Reservadas;}
17 {espacio} { /*Ignore*/ }
18 "/*.*" { /*Ignore*/ }
19 "=" {return Igual;}
20 "+" {return Suma;}
21 "-" {return Resta;}
22 "*" {return Multiplicacion;}
23 "/" {return Division;}
24 {L}({L}|{D}) * {lexeme=yytext(); return Identificador;}
25 ("{"|"}") {lexeme=yytext(); return Numero;}
26 . {return ERROR;}
27
```

1. Agregar en el paquete “código” un “Empty File”, se puede buscar en others, darle el nombre de Lexer.flex
2. Copiar y pegar las reglas léxicas del archivo “Apoyo I.txt”

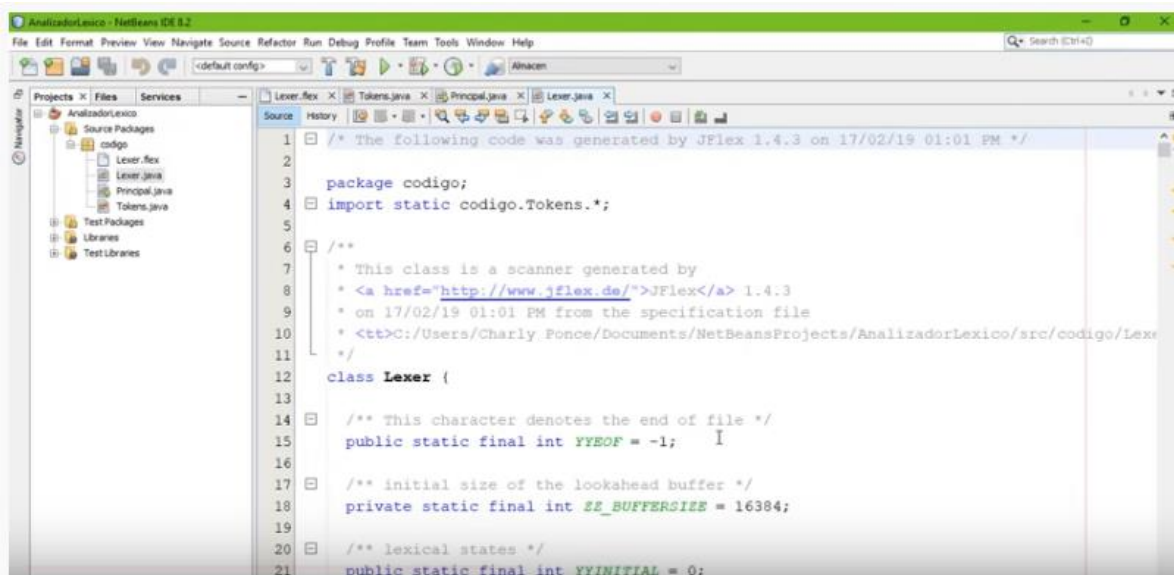
### 05:29 Enumeración Tokens – Apoyo II.txt



```
1 /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6 package codigo;
7
8 /**
9  *
10  * @author Charly Ponce
11  */
12 public enum Tokens {
13     Reservadas,
14     Igual,
15     Suma,
16     Resta,
17     Multiplicacion,
18     Division,
19     Identificador,
20     Numero,
21 }
```

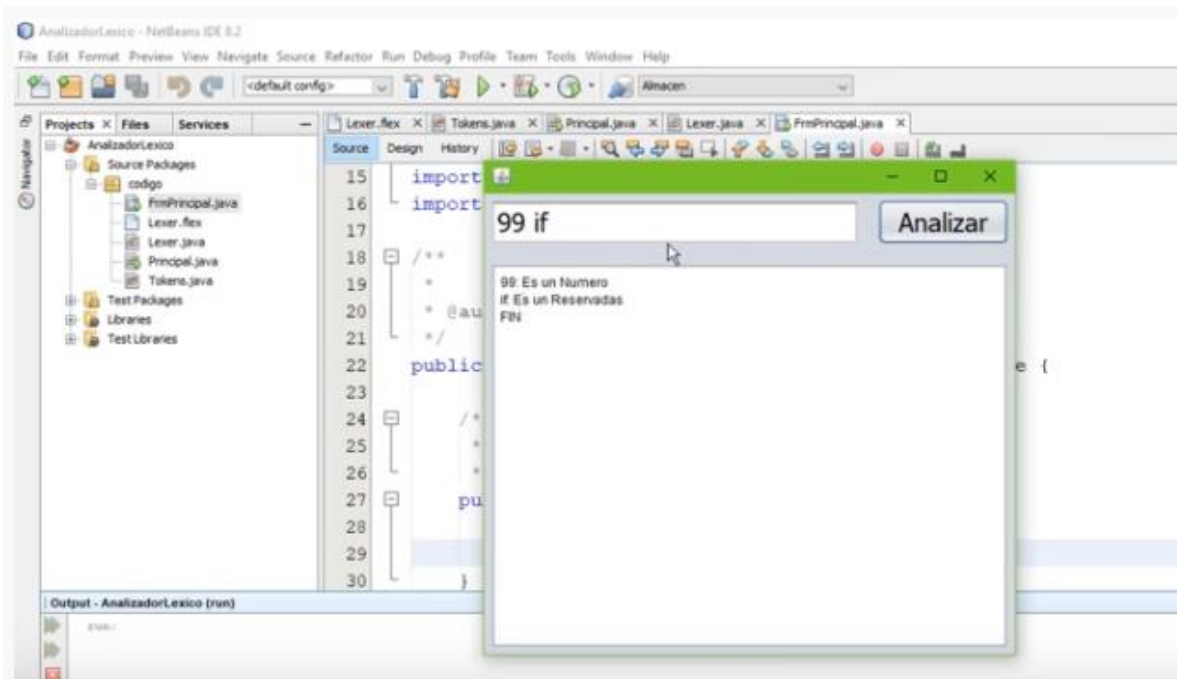
1. Agregar en el paquete “código” un “Java Enum”, se puede buscar en others, darle el nombre de Tokens
2. Se agregar los returns de la clase Lexer.flex, los puedes tomar del “Apoyo II.txt”

### 06:33 Analizador léxico JAVA – Apoyo III.txt



1. Agrega una "Java class" al paquete llamada "Principal".
2. Dentro de la nueva clase genera el main con "psvm" y tabulador.
3. Con clic derecho en el source package puedes ubicar la ruta para el ajuste del código en el "Apoyo III.txt"
4. Al ejecutar con clic derecho sobre la Principal, se va a crear la clase Lexer.java que permitirá realizar el análisis léxico.

### [10:11](#) Uso final (JFrame) – Apoyo IV.txt



1. Agregar un JFrame llamado FrmPrincipal.
2. Agrega un jTextField con letra de tamaño 24 y renombrada la variable como txtEntrada.
3. Agrega un botón con letra 24 que diga Analizar, renombrar la variable como btnAnalizar.
4. Agrega un Text Area renombrar como txtResultado.
5. En el botón de Analizar, codificar el análisis léxico, puedes tomar como base el archivo "Apoyo IV.txt". Ten cuidado de agregar las excepciones necesarias. Lo que realizar es:
  - a. Guarda los lexemas de entrada en un archivo de texto.
  - b. Crea un objeto tipo Lexer que usa como parámetro el archivo a ser analizado.
  - c. Va recorriendo cada lexema del archivo y muestra su clasificación de acuerdo con el token que le corresponde.

### Actividad

Entrega la impresión de pantalla donde cargues 8 lexemas y se muestren los tokens respectivos.