

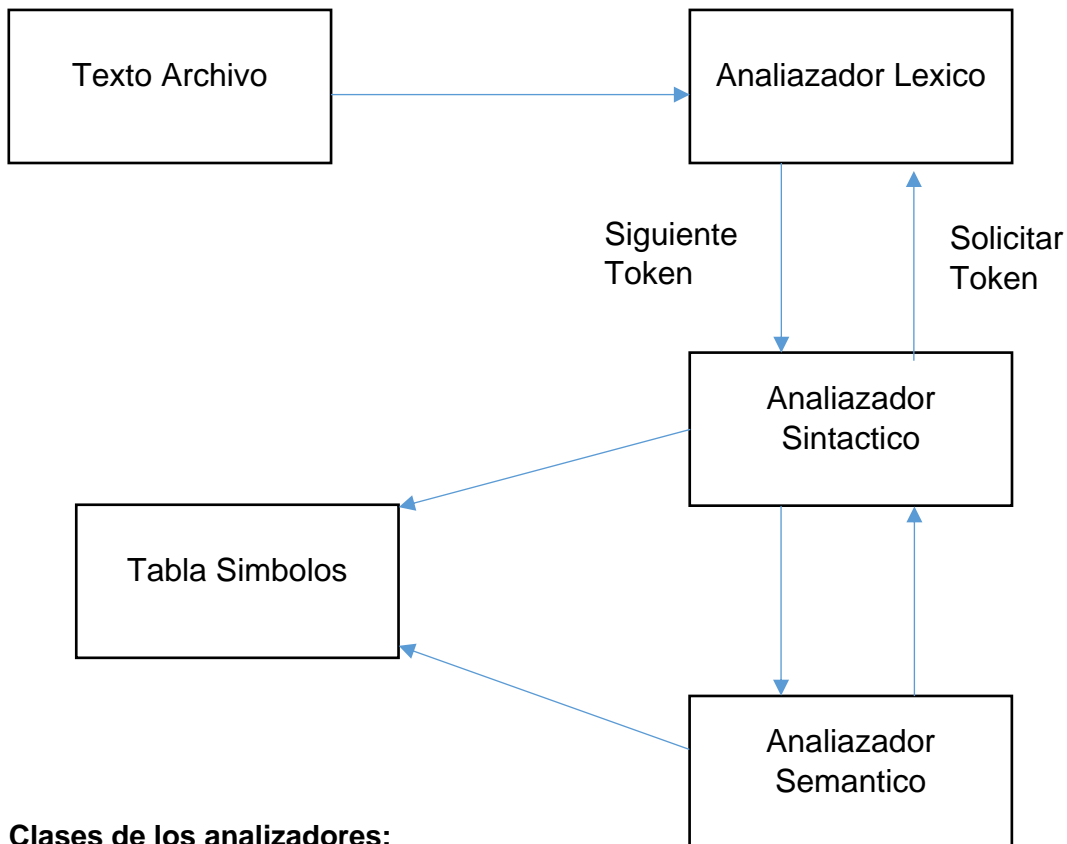
# **MANUAL TECNICO**

## **(IDE Animaciones)**

# DESCRIPCION DEL PROGRAMA

El programa cuenta con un analizador léxico y sintáctico. Estos se usan para determinar los objetos que se crearan para integracion del componente del programa. Cuenta con cuatro analizadores de estos tipos, una para el lienzo, uno para imágenes, uno para tiempos y el ultimo para el pintar.

Los archivos de lectura para los analizadores son de extensión .json la gramática del archivo tiene una estructura tipo json. De esta manera se implementa el sistema de lectura de los analizadores. Ya que el programa está hecho en el Lenguaje Java, se utilizan las herramientas de jflex y cup para la creación de los analizadores.



## Clases de los analizadores:

AnalizadorLexicoLnz -----.lnz

AnalizadorSintacticoLnz -----.lnz

AnalizadorLexicoClrs -----.clrs

AnalizadorSintacticoClrs ----- .clrs

AnalizadorLexicoTmp ----- .tmp

AnalizadorSintacticoTmp ----- .tmp

AnalizadorLexicoPnt ----- .pnt

AnalizadorSintacticoPnt ----- .pnt





# GRAMATICAS

Gramática archivo .lnz

## Expresiones Regulares

Caracter = [ \_ ]

Letra = ([a-zA-Z] | ñ | Ñ)

LetraHEX = ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9])

Texto = ({Letra} | {Numero} | {Caracter})\*

Digito = [0-9]

Numero = {Digito} {Digito}\*

Espacio = [ " '\t\r\n ]+

## Palabras Reservadas

LIENZOS, nombre , tipo, Fondo, Red, Blue, Green, HEX, tamaño, cuadro, dimension\_x, dimension\_y, png, gif, :, {, }.

**M = {Q, A, q<sub>0</sub>, F}**

**Q** = INICIO, ESTRUCTURA\_LIENZOS, ESTRUCTURA\_PARAMETROS, SO, PARAMETROS, ATRIBUTOS, TIPO\_IMAGEN, ESTRUCTURA\_FONDO, ATRIB\_RGB, ATRIB\_HEX, ESTRUCTURA\_TAMANIO, ATRIB\_TAMANIO, IDENTIFICADORES.

**A** = DOS\_PUNTOS, COMA, COMILLAS, LLAVES\_A, LLAVES\_C, LIENZOS, TAMANIO, IDENTIFICADOR, NUMERO, FONDO, CODIGO\_HEX, NOMBRE, TIPO, PNG, GIF, HEX, RED, BLUE, GREEN, CUADRO, DIMENSION\_X, DIMENSION\_Y.

**q<sub>0</sub>** = INICIO

**F** = INICIO, ESTRUCTURA\_LIENZOS, ESTRUCTURA\_PARAMETROS, SO, PARAMETROS, ATRIBUTOS, TIPO\_IMAGEN, ESTRUCTURA\_FONDO, ATRIB\_RGB, ATRIB\_HEX, ESTRUCTURA\_TAMANIO, ATRIB\_TAMANIO, IDENTIFICADORES.

## Producciones

INICIO ::= LLAVES\_A ESTRUCTURA\_LIENZOS LLAVES\_C;

ESTRUCTURA\_LIENZOS ::= LIENZOS DOS\_PUNTOS LLAVES\_A ESTRUCTURA\_PARAMETROS LLAVES\_C;

ESTRUCTURA\_PARAMETROS ::= IDENTIFICADOR DOS\_PUNTOS LLAVES\_A PARAMETROS LLAVES\_C SO;

SO ::= COMA ESTRUCTURA\_PARAMETROS  
| ;

PARAMETROS ::= ATRIBUTOS COMA ATRIBUTOS COMA ATRIBUTOS COMA ATRIBUTOS;

ATRIBUTOS ::= NOMBRE DOS\_PUNTOS COMILLAS IDENTIFICADOR COMILLAS  
| TIPO DOS\_PUNTOS COMILLAS TIPO\_IMAGEN COMILLAS  
| FONDO DOS\_PUNTOS LLAVES\_A ESTRUCTURA\_FONDO LLAVES\_C  
| TAMANIO DOS\_PUNTOS LLAVES\_A ESTRUCTURA\_TAMANIO LLAVES\_C;

TIPO\_IMAGEN ::= PNG  
| GIF;

ESTRUCTURA\_FONDO ::= ATRIB\_RGB COMA ATRIB\_RGB COMA ATRIB\_RGB;  
| ATRIB\_HEX;

ATRIB\_RGB ::= RED DOS\_PUNTOS NUMERO  
| BLUE DOS\_PUNTOS NUMERO  
| GREEN DOS\_PUNTOS NUMERO;

ATRIB\_HEX ::= HEX DOS\_PUNTOS CODIGO\_HEX;

ESTRUCTURA\_TAMANIO ::= ATRIB\_TAMANIO COMA ATRIB\_TAMANIO COMA ATRIB\_TAMANIO;

ATRIB\_TAMANIO ::= CUADRO DOS\_PUNTOS ENTERO  
| DIMENSION\_X DOS\_PUNTOS NUMERO  
| DIMENSION\_Y DOS\_PUNTOS NUMERO

## Gramática archivo .clrs

### Expresiones Regulares

Caracter = [ \_ ]

Letra = ([a-zA-Z] | ñ | Ñ)

LetraHEX = ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9]) ([a-fA-F] | [0-9])

Texto = ({Letra} | {Numero} | {Caracter})<sup>+</sup>

Digito = [0-9]

Numero = {Digito} {Digito}<sup>\*</sup>

Espacio = [ \t\r\n ]<sup>+</sup>

### Palabras Reservadas

COLORES, nombre , tipo, Fondo, Red, Blue, Green, HEX, :, {, }, ", .

**M = {Q, A, q<sub>0</sub>, F}**

**Q** = INICIO , ESTRUCTURA\_COLORES, ESTRUCTURA\_PARAMETROS, S0, S1, PARAMETROS, ESTRUCTURA\_FONDO, ATRIB\_RGB, ATRIB\_HEX.

**A** = CORCHE\_A, CORCHE\_C, DOS\_PUNTOS, COMA, COMILLAS, LLAVES\_A, LLAVES\_C, COLORES, TAMANIO, IDENTIFICADOR, NUMERO, FONDO, CODIGO\_HEX, TIPO, PNG, GIF, HEX, RED, BLUE, GREEN.

**q<sub>0</sub>** = INICIO

**F** = INICIO , ESTRUCTURA\_COLORES, ESTRUCTURA\_PARAMETROS, S0, S1, PARAMETROS, ESTRUCTURA\_FONDO, ATRIB\_RGB, ATRIB\_HEX.

### Producciones

INICIO ::= LLAVES\_A ESTRUCTURA\_COLORES LLAVES\_C;

ESTRUCTURA\_COLORES ::= COLORES DOS\_PUNTOS LLAVES\_A ESTRUCTURA\_PARAMETROS LLAVES\_C;

ESTRUCTURA\_PARAMETROS ::= IDENTIFICADOR DOS\_PUNTOS LLAVES\_A PARAMETROS LLAVES\_C S0;

S0 ::= COMA ESTRUCTURA\_PARAMETROS

| ;

PARAMETROS ::= IDENTIFICADOR DOS\_PUNTOS LLAVES\_A ESTRUCTURA\_FONDO LLAVES\_C S1;

S1 ::= COMA PARAMETROS

| ;

ESTRUCTURA\_FONDO ::= ATRIB\_RGB COMA ATRIB\_RGB COMA ATRIB\_RGB;

| ATRIB\_HEX;

ATRIB\_RGB ::= RED DOS\_PUNTOS NUMERO

| BLUE DOS\_PUNTOS NUMERO

| GREEN DOS\_PUNTOS NUMERO;

ATRIB\_HEX ::= HEX DOS\_PUNTOS CODIGO\_HEX;



## Gramática archivo .tmp

### Expresiones Regulares

Caracter = [ \_ ]

Letra = ([a-zA-Z] | ñ | Ñ)

Texto = ({Letra}|{Numero}|{Caracter})+

Digito = [0-9]

Numero = {Digito} {Digito}\*

Espacio = [ \t\r\n ]+

### Palabras Reservadas

TIEMPOSS, nombre , tipo, inicio, fin, imagenes, id, duración,[, ],", :, {, }.

**M = {Q, A, q<sub>0</sub>, F}**

**Q** = INICIO, ESTRUCTURA\_TIEMPOS, ESTRUCTURA\_PARAMETROS, S0, PARAMETROS, ATRIBUTOS, ESTRUCTURA\_IMAGENES, S1, ATRIB\_IMAGENES, IDENTIFICADORES.

**A** = CORCHETE\_A, CORCHETE\_C, DOS\_PUNTOS, COMA, COMILLAS, LLAVES\_A, LLAVES\_C, DURACION, ID, IMAGENES, FIN, INICIO\_RSRVD.

**q<sub>0</sub>** = INICIO

**F** = INICIO, ESTRUCTURA\_TIEMPOS, ESTRUCTURA\_PARAMETROS, S0, PARAMETROS, ATRIBUTOS, ESTRUCTURA\_IMAGENES, S1, ATRIB\_IMAGENES, IDENTIFICADORES.

### Producciones

INICIO ::= LLAVES\_A ESTRUCTURA\_ TIEMPOS LLAVES\_C;

ESTRUCTURA\_TIEMPOS ::= TIEMPOS DOS\_PUNTOS LLAVES\_A ESTRUCTURA\_PARAMETROS LLAVES\_C;

ESTRUCTURA\_PARAMETROS ::= IDENTIFICADOR DOS\_PUNTOS LLAVES\_A PARAMETROS LLAVES\_C S0;

S0 ::= COMA ESTRUCTURA\_PARAMETROS  
| ;

PARAMETROS ::= ATRIBUTOS COMA ATRIBUTOS COMA ATRIBUTOS;

ATRIBUTOS ::= INICIO\_RSRVD DOS\_PUNTOS COMILLAS IDENTIFICADOR COMILLAS  
| FIN DOS\_PUNTOS COMILLAS IDENTIFICADOR COMILLAS  
| IMAGENES DOS\_PUNTOS CORCHETE\_A ESTRUCTURA\_IMAGENES CORCHETE \_C;

ESTRUCTURA\_IMAGENES ::= LLAVES\_A ATRIB\_IMAGENES LLAVES\_C S1;

S1 ::= COMA ESTRUCTURA\_IMAGENES  
| ;

ATRIB\_IMAGENES ::= ID DOS\_PUNTOS COMILLAS IDENTIFICADOR COMILLAS COMA DURACION DOS\_PUNTOS NUMERO  
| DURACION DOS\_PUNTOS NUMERO COMA ID DOS\_PUNTOS COMILLAS IDENTIFICADOR COMILLAS

## Gramática archivo .pnt

### Expresiones Regulares

Caracter = [ \_ ]

Letra = ([a-zA-Z] | ñ | Ñ)

Texto = ({Letra}|{Numero}|{Caracter})+

Digito = [0-9]

Numero = {Digito} {Digito}\*

Espacio = [ \t\r ]+

### Palabras Reservadas

VARS, int, String, boolean, id, else, while, AND, OR, ISNTRUCCIONES, PINTAR, pintar, true, false, [, ], ", :, {, }, ,, : , = , + , - , \* , / , < , > , ( , ) , , , < = , > = , < > .

**M = {Q, A, q<sub>0</sub>, F}**

**Q** = INICIO, INICIO\_VARS, INICIO\_INSTRUCCIONES, SO, COMENTARIO, ESTRUCTURA\_VARS, DECLARACION\_VARIABLES, ASIG\_VARS\_INT, OPERACIÓN, CADENA\_ASIG\_INT, ASIG\_VARS\_STRING, CONCATENACION, TIPO\_TEXTO, CADENA\_ASIG\_STRING, ASIG\_VARS\_BOOLEAN, TIPO\_BOOL, CADENA\_ASIG\_BOOLEAN, ESTRUCTURA\_INSTRUCCIONES, ASIGNACION\_VALORES, ASIG, OPERADOR\_LOGICO, OPERACION\_LOGICA, OPERADOR\_RACIONAL, ESTRUC\_PINTAR, PRMT, PRMT1, ESTRUC\_WHILE, CONDICION\_L, CODICIONES, CONTENIDO, ESTRUC\_IF\_ELSE, ESTRUC\_ELSE, TIPO\_DATO, OPERADOR\_ARITMETICO, TIPOS\_INT, OPERA\_LOGICA, OPERACION2, TIPOS\_INT2, PRMT2, DATO, E, LO\_QUE\_SEA, LO\_QUE\_SEA\_SALTO, FINALIZAR, LO, LOA, OPERACION\_BOOLEANA, DATOS\_BOOLEANOS, TIPO\_PINTAR.

**A** = CORCHETE\_A, CORCHETE\_C, DOS\_PUNTOS, COMA, COMILLAS, LLAVES\_A, LLAVES\_C, IDENTIFICADOR, PUNTO\_COMA, COMENTAR\_LINEA, VARS, IGUAL, MAS, MENOS, ASTERISCO, DIAGONAL, MENOR\_QUE, MAYOR\_QUE, PARENTESIS\_A, PARENTESIS\_C, PUNTO, PINTAR\_MIN, INT, STRING, BOOLEAN, IF, ELSE, WHILE, AND, OR, INSTRUCCIONES, PINTAR, NUMERO, COMENTAR\_A, COMENTAR\_C, TRUE, FALSE, MAYOR\_IGUAL, MENOR\_IGUAL, IGUAL\_IGUAL, DIFERENTE, PUNTO\_PUNTO, SALTO.

**q<sub>0</sub>** = INICIO

**F** = INICIO, INICIO\_VARS, INICIO\_INSTRUCCIONES, SO, COMENTARIO, ESTRUCTURA\_VARS, DECLARACION\_VARIABLES, ASIG\_VARS\_INT, OPERACIÓN, CADENA\_ASIG\_INT, ASIG\_VARS\_STRING, CONCATENACION, TIPO\_TEXTO, CADENA\_ASIG\_STRING, ASIG\_VARS\_BOOLEAN, TIPO\_BOOL, CADENA\_ASIG\_BOOLEAN, ESTRUCTURA\_INSTRUCCIONES, ASIGNACION\_VALORES, ASIG, OPERADOR\_LOGICO, OPERACION\_LOGICA, OPERADOR\_RACIONAL, ESTRUC\_PINTAR, PRMT, PRMT1, ESTRUC\_WHILE, CONDICION\_L, CODICIONES, CONTENIDO, ESTRUC\_IF\_ELSE, ESTRUC\_ELSE, TIPO\_DATO, OPERADOR\_ARITMETICO, TIPOS\_INT, OPERA\_LOGICA, OPERACION2, TIPOS\_INT2, PRMT2, DATO, E, LO\_QUE\_SEA, LO\_QUE\_SEA\_SALTO, FINALIZAR, LO, LOA, OPERACION\_BOOLEANA, DATOS\_BOOLEANOS, TIPO\_PINTAR.

### Producciones

INICIO ::= INICIO\_VARS INICIO\_INSTRUCCIONES ;

INICIO\_VARS ::= VARS CORCHETE\_A ESTRUCTURA\_VARS CORCHETE\_C;

INICIO\_INSTRUCCIONES ::= INSTRUCCIONES PARENTESIS\_A IDENTIFICADOR PARENTESIS\_C CORCHETE\_A ESTRUCTURA\_INSTRUCCIONES CORCHETE\_C S0;

S0 ::= INICIO\_INSTRUCCIONES  
| ;

COMENTARIO ::= DIAGONAL ASTERISCO LO\_QUE\_SEA ASTERISCO DIAGONAL  
| DIAGONAL DIAGONAL LO\_QUE\_SEA SALTO\_LINEA;

ESTRUCTURA\_VARS ::= DECLARACION\_VARIABLES ESTRUCTURA\_VARS  
| COMENTARIO ESTRUCTURA\_VARS  
| ;

DECLARACION\_VARIABLES ::= INT IDENTIFICADOR ASIG\_VARS\_INT PUNTO\_COMA  
| STRING IDENTIFICADOR ASIG\_VARS\_STRING PUNTO\_COMA  
| BOOLEAN IDENTIFICADOR ASIG\_VARS\_BOOLEAN PUNTO\_COMA;

ASIG\_VARS\_INT ::= IGUAL OPERACION CADENA\_ASIG\_INT  
| ;

OPERACION ::= OPERACION SUMA OPERACION  
| OPERACION RESTA OPERACION  
| NUMERO  
| IDENTIFICADOR;

CADENA\_ASIG\_INT ::= COMA IDENTIFICADOR IGUAL OPERACION CADENA\_ASIG\_INT  
| COMA IDENTIFICADOR CADENA\_ASIG\_INT  
| ;

ASIG\_VARS\_STRING ::= IGUAL CONCATENACION CADENA\_ASIG\_STRING;  
| ;

CONCATENACION ::= CONCATENACION SUMA CONCATENACION  
| COMILLAS TIPO\_TEXTO COMILLAS  
| IDENTIFICADOR

TIPO\_TEXTO ::= IDENTIFICADOR | NUMERO |;

CADENA\_ASIG\_STRING ::= COMA IDENTIFICADOR IGUAL CONCATENACION CADENA\_ASIG\_STRING  
| COMA IDENTIFICADOR CADENA\_ASIG\_STRING  
|;

ASIG\_VARS\_BOOLEAN ::= IGUAL TIPO\_BOOL CADENA\_ASIG\_BOOLEAN  
| ;

TIPO\_BOOL ::= TRUE | FALSE;

CADENA\_ASIG\_BOOLEAN ::= COMA IDENTIFICADOR IGUAL TIPO\_BOOL CADENA\_ASIG\_BOOLEAN  
| COMA IDENTIFICADOR CADENA\_ASIG\_BOOLEAN  
|;

ESTRUCTURA\_INSTRUCCIONES ::= ASIGNACION\_VALORES  
| ESTRUCT\_PINTAR  
| ESTRUCT\_WHILE  
| ESTRUCT\_IF\_ELSE  
|;

ASIGNACION\_VALORES ::= IDENTIFICADOR IGUAL ASIG PUNTO\_COMA

ASIG ::= ASIG MAS ASIG  
| ASIG MENOS ASIG

|ASIG OPERADOR\_RACIONAL ASIG OPERADOR\_LOGICO

|COMILLAS TIPO\_TEXTO COMILLAS

|NUMERO

|IDENTIFICADOR

|TIPO\_BOOL;

OPERADOR\_LOGICO ::= AND OPERACION\_LOGICA OPERADOR\_RACIONAL OPERACION\_LOGICA OPERADOR\_LOGICO

|OR OPERACION\_LOGICA OPERADOR\_RACIONAL OPERACION\_LOGICA OPERADOR\_LOGICO;

|;

OPERACION\_LOGICA ::= OPERACION

|IDENTIFICADOR

|TIPO\_BOOL;

OPERADOR\_RACIONAL ::= IGUAL\_IGUAL

|MENOR\_QUE

|MAYOR\_QUE

|MENOR\_IGUAL

|MAYOR\_IGUAL

|DIFERENTE;

ESTRUC\_PINTAR ::= PINTAR PARENTESIS\_A PRMT COMA PRMT COMA PRMT1 COMA PRMT1 PARENTESIS\_C PUNTO\_COMA;

PRMT ::= PRMT SUMA PRMT

|PRMT RESTA PRMT

|COMILLAS TIPO\_TEXTO COMILLAS

|NUMERO

|IDENTIFICADOR;

PRMT1 ::= PRMT1 SUMA PRMT1

|PRMT1 RESTA PRMT1

| NUMERO PUNTO PUNTO NUMERO

| NUMERO

| IDENTIFICADOR;

ESTRUC\_WHILE ::= WHILE PARENTESIS\_A CONDICION\_L PARENTESIS\_C LLAVES\_A CONTENIDO LLAVES\_C;

CONDICION\_L ::= CODICIONES OPERADOR\_RACIONAL CODICIONES OPERADOR\_LOGICO

| IDENTIFICADOR

| TIPO\_BOOL;

CODICIONES ::= OPERACIÓN

| IDENTIFICADOR

| TIPO\_BOOL

| COMILLAS TIPO\_TEXTO COMILLAS

CONTENIDO ::= ESTRUCTURA\_INSTRUCCIONES;

ESTRUC\_IF\_ELSE ::= IF PARENTESIS\_A CONDICION\_L PARENTESIS\_C LLAVES\_A CONTENIDO LLAVES\_C ESTRUC\_ELSE;

ESTRUC\_ELSE ::= ELSE LLAVES\_A CONTENIDO LLAVES\_C

|;