

## Lenguaje

letra = [a-zA-Z]

numero = [0-9]+ id

={letra}{{letra}|"\_"|[0-9]}\* simbolo

Simbolo =[!-}]

**Terminales:** entero, decimal, id, cadena, igual, puntocomma, var, let, constvar, truevar, falsevar, ifpr, pariz, parder, llaveiz, llaveder, menorq, elsepr, imprimir, diferente, forpr, mas, menos, whilepr, dopr, switchpr, casepr, breakpr, defaultpr, dpuntos, mayorq, andpr, orpr, notpr, por, pot, div, porcentaje, menorigual, mayorigual, digual, coma, requirepr, classpr;

**No Terminales:** INICIO, INSTRUCCION, EXPRESION, VARIABLES, PRVARIABLES, VALOR, IF, CONDICION, CUERPOIF, RELACIONAL, ELSE, CUERPOELSE, PRINTCONSOLE, ELSEIF, CUERPOELSEIF, FOR, SETTINGFOR, CUERPOFOR, INICIOFOR, INCFOR, ASIGNACION, WHILE, DOWHILE, CUERPOWHILE, SWITCH, CUERPOSWITCH, FINALSWITCH, CUERPOCASE, LOGICO, INSTRUCCIONL, INC, DEC, CALLM, CUERPOMETODO, CUERPOMETODO2, IMPORT, CLASS, CUERPOCLASE, METODO, METODOBODY

INICIO ::= INSTRUCCION

| INSTRUCCION ;

INSTRUCCION ::= VARIABLES

| IF

| FOR

| ASIGNACION

| WHILE

| DOWHILE

| SWITCH

| LLAMADAM

| INSTRUCCION

VARIABLES ::= PRVARIABLE id igual VALORVARIABLE

PRVARIABLE ::= var

| let

| const

VALORVARIABLE ::= entero

| decimal

| cadena

| true

| false

| IMPORT

IF ::= if ( CONDICION ) { CUERPOIF }

| if ( CONDICION ) { CUERPOIF } ELSE

| if ( CONDICION ) { CUERPOIF } ELSEIF

| if ( CONDICION ) { CUERPOIF } ELSEIF ELSE

CONDICION ::= RELACIONAL

RELACIONAL ::= EXPRESION == EXPRESION

| EXPRESION != EXPRESION

| EXPRESION < EXPRESION

| EXPRESION > EXPRESION

| EXPRESION <= EXPRESION

| EXPRESION >= EXPRESION

EXPRESION ::= id

| entero

| decimal

| cadena

| true

| false

| EXPRESION + EXPRESION

| EXPRESION - EXPRESION

| EXPRESION \* EXPRESION

| EXPRESION / EXPRESION

| EXPRESION \*\* EXPRESION

| EXPRESION % EXPRESION

| - EXPRESION

| ( EXPRESION )

| RELACIONAL

| LOGICO

CUERPOIF ::= INSTRUCCION

| break ;

ELSE ::= else { CUERPOELSE }

CUERPOELSE ::= INSTRUCCION

| break ;

ELSEIF ::= CUERPOELSEIF

| ELSEIF

CUERPOELSEIF ::= else if (CONDICION) { CUERPOIF }

FOR ::= for ( SETTINGSFOR ) { CUERPOFOR }

SETTINGSFOR ::= INICIOFOR ; RELACIONAL ; INCFOR

INICIOFOR ::= VARIABLES

| ASIGNACION

ASIGNACION ::= id = EXPRESION

INCFOR ::= id mas mas

| id menos menos

CUERPOFOR ::= INSTRUCCION

| break ;

WHILE ::= while ( RELACIONAL ) { CUERPOWHILE }

CUERPOWHILE ::= INSTRUCCION

| break ;

DOWHILE ::= do { CUERPOWHILE } while ( RELACIONAL ) SWITCH ::= switch ( id ) {  
CUERPOSWITCH FINALSCHITCH }

CUERPOSWITCH ::= case EXPRESION : CUERPOCASE break ;  
| CUERPOSWITCH

CUERPOCASE ::= INSTRUCCION

FINALSWITCH ::= default : CUERPOCASE

LOGICO ::= EXPRESION && EXPRESION  
| EXPRESION || EXPRESION  
| ! EXPRESION

LLAMADAM ::= id ( )  
| id ( CUERPOMETODO )

CUERPOMETODO ::= EXPRESION  
| EXPRESION CUERPOMETODO2

CUERPOMETODO2 ::= , EXPRESION  
| CUERPOMETODO2

IMPORT ::= require ( cadena )

CLASS ::= class id ( ) { CUERPOCLASE }

CUERPOCLASE ::= INSTRUCCION

METODO ::= id () { CUERPOMETODO }  
| id (PARAMETRO) { CUERPOMETODO }

PARAMETRO ::= EXPRESION  
| EXPRESION PARAMETRO2

PARAMETRO2 ::= , EXPRESION  
| PARAMETRO**2**