

# GRAMATICA

## EXPRESIONES REGULARES

- `//".*` comentarios simples
- `"/*(\\*+[^/*])|([\\^*]))*\\**"/` comentarios con múltiples líneas
- `[a-zA-Z][a-zA-Z0-9_]*` 'ID' Identificadores tanto para variables, funciones etc
- `[0-9]+(\\. "[0-9]+)\\b` 'DECIMAL'; números con decimales
- `[0-9]+\\b` 'ENTERO' números enteros
- `\\((\\\\\\\\)|(\\\\[n\\r])*)'` 'CARACTERES' caracteres especiales
- `[^"\\]+ "\\\" "" "\\n""\\t" "\\\" "" "\\\" ["]` CADENA cadena de caracteres

## SIMBOLOS:

// simbolos reservados

```
"," 'PTCOMA';
 "(" 'PARIZQ';
 ")" 'PARDER';
 "." 'PUNTO';
 ":" 'DOSPUNTOS';
 "," 'COMA';
 "[" 'CORIZR';
 "]" 'CORDER';
 "{" 'LLAVEIZQ';
 "}" 'LLAVEDER';
```

// palabras reservadas

```
"print" 'RPRIN';
"true" 'TRUE';
>false" 'FALSE';
"for" 'RFOR';
"if" 'RIF';
"else" 'RELSE';
"switch" 'RSWITCH'
```

```
"case"    'RCASE'
"default" 'RDEFAULT'
"break"   'RBREAK'
"do"      'RDO';
"while"   'RWHILE';
"void"    'RVOID';
"toLowerCase" 'RTOLOWER';
"toUpperCase" 'RTOUPPER';
"length"   'RLENGTH';
"truncate" 'RTRUNCATE';
"round"    'RROUND';
"typeof"   'RTYPEOF';
"toString" 'RTOSTRING';
"main"     'RMAIN';
"new"      'RNEW';
```

//Relacionales

```
"=="      'IGUALQ';
"!="      'DIFERENTEQ';
"<="      'MENORIGUALQUE';
">="      'MAYORIGUALQUE';
"<"       'MENORQUE';
">"       'MAYORQUE';
```

//LOGICOS

```
"||"      'OR';
"&&"      'AND';
"!"       'NOT';
```

//TERNARIO

```
"?"       'TERNARIO';
```

//ASIGANACION

"=" 'IGUAL';

// aritmeticos

"++" 'INCRE';

"--" 'DECRE';

"+" 'MAS';

"-" 'MENOS';

"\*" 'POR';

"/" 'DIVISION';

"^" 'POTENCIA';

"%" 'MODULO';

// tipos de variables

"int" 'RENTERO';

"string" 'RSTRING';

"char" 'RCHAR';

"boolean" 'RBOOLEAN';

"double" 'RDOUBLE';

## GRAMATICAS

// -----> declaracion de terminales

'PTCOMA'; 'PARIZQ'; 'PARDER'; 'PUNTO'; 'DOSPUNTOS'; 'COMA'; 'CORIZR'; 'CORDER'; 'LLAVEIZQ'; 'LLAVEDER';	'RPRIN'; 'TRUE'; 'FALSE'; 'RFOR'; 'RIF'; 'RELSE'; 'RSWITCH' 'RCASE' 'RDEFAULT' 'RBREAK' 'RDO';	'RVOID'; 'RTOLOWER'; "t" 'RTOUPPER'; 'RLENGTH'; 'RTRUNCATE'; 'ROUND'; 'RTYPEOF'; 'RTOSTRING'; 'RMAIN'; 'RNEW'; 'TERNARIO';	'RWHILE'; 'IGUALQ'; 'DIFERENTEQ'; 'MENORIGUALQUE'; 'MAYORIGUALQUE'; 'MENORQUE'; 'MAYORQUE'; 'OR'; 'AND'; 'NOT'; 'INCRE'; 'DECRE';	'MAS'; 'MENOS'; 'POR'; 'DIVISION'; 'POTENCIA'; 'MODULO'; 'RENTERO'; 'RSTRING'; 'RCHAR'; 'RBOOLEAN'; " 'RDOUBLE';
--	--	--	--	--

// -----> declaracion de no terminales

INICIO	WHILE	LENGTH
INSTRUCCIONES	MAIN	TRUNCATE
INSTRUCCION	ARRAY_DECLARACION	ROUND
DEFPRINT	ARGUMENTOS	TYPEOF
DECLARAR	STATEMENT	TOSTRING
INCREDECRE	PARAMETROS	ACCEDERVAR
GUARDARFUNCION	PARAMETRO	ARITMETICA
LLAMADAFUNCION	TIPO	RELACIONALES
FOR	EXPRESION	LOGICOS
CONTROLIF	EXPRESION	OPERACIONESUNARIOS
DOWHILE	PRIMITIVO	TOLOWER
CONTROLELSE	CONTROLIF	TOUPPER

// -----> Precedencias

*Precedence left* 'TERNARIO' 'DOSPUNTOS'

*Precedence left* 'OR'

*Precedence left* 'AND'

*Precedence right* 'NOT'

*Precedence left* 'IGUALQ' 'DIFERENTEQ' 'MENORQUE' 'MENORIGUALQUE' 'MAYORQUE' 'MAYORIGUALQUE'

*Precedence left* 'MAS' 'MENOS'

*Precedence left* 'POR' 'DIVISION' 'MODULO'

*Precedence nonassoc* 'POTENCIA'

*Precedence right* 'UMENOS '

*Precedence right* 'PARIZQ' 'PARDER'

*Precedence right* 'INCRE' 'DECRE'

```
// -----> Producciones
```

```
INICIO
```

```
-----: INSTRUCCIONES
```

```
;
```

```
INSTRUCCIONES
```

```
: INSTRUCCIONES INSTRUCCION
```

```
| INSTRUCCION
```

```
;
```

```
INSTRUCCION
```

```
: DEFPRINT
```

```
| DECLARAR PTCOMA
```

```
| INCREDECRE
```

```
| GUARDARFUNCION
```

```
| LLAMADAFUNCION PTCOMA
```

```
| FOR
```

```
| CONTROLIF
```

```
| DOWHILE
```

```
| WHILE
```

```
| MAIN PTCOMA
```

```
| ARRAY_DECLARACION PTCOMA
```

```
;
```

```
// GRAMATICA ARREGLOS
```

```
ARRAY_DECLARACION
```

```
: TIPO CORIZR CORDER ID IGUAL LLAVEIZQ ARGUMENTOS LLAVEDER
```

```
| TIPO CORIZR CORDER ID IGUAL RNEW TIPO CORIZR EXPRESION CORDER
```

```
;
```

```
//GRAMATICA MAIN
```

```
MAIN
```

```
: RMAIN ID PARIZQ PARDER
```

```
| RMAIN ID PARIZQ ARGUMENTOS PARDER
```

```
;
```

```
// GRAMATICA IMPRIMIR
```

```
DEFPRINT
```

```
    : RPRIN PARIZQ EXPRESION PARDER PTCOMA
```

```
;
```

```
//GRAMATICA DECLARAR Y ASIGNAR
```

```
DECLARAR
```

```
    : TIPO ID
```

```
    | TIPO ID IGUAL EXPRESION
```

```
;
```

```
// GRAMATICA PARA DECLARAR FUNCIONES
```

```
GUARDARFUNCION
```

```
    : TIPO ID PARIZQ PARDER STATEMENT
```

```
    | TIPO ID PARIZQ PARAMETROS PARDER STATEMENT
```

```
    | RVOID ID PARIZQ PARDER STATEMENT
```

```
    | RVOID ID PARIZQ PARAMETROS PARDER STATEMENT
```

```
;
```

```
//GRAMATICA CICLO DOWHILE
```

```
DOWHILE
```

```
    :RDO STATEMENT RWHILE PARIZQ EXPRESION PARDER PTCOMA
```

```
;
```

```
//GRAMATICA PARA CICLO WHILE
```

```
WHILE
```

```
    : RWHILE PARIZQ EXPRESION PARDER STATEMENT
```

```
;
```

```
// STATEMENT
```

```
STATEMENT
```

```
    : LLAVEIZQ INSTRUCCIONES LLAVEDER
```

```
;
```

```
// PARAMETROS
```

```
PARAMETROS
```

```
: PARAMETROS COMA PARAMETRO
```

```
| PARAMETRO
```

```
;
```

```
PARAMETRO
```

```
: TIPO ID
```

```
;
```

```
//-----
```

```
// GRAMATICA PARA CICLO FOR
```

```
FOR
```

```
: RFOR PARIZQ DECLARAR PTCOMA EXPRESION PTCOMA OPERACIONESUNARIOS PARDER STATEMENT
```

```
;
```

```
//GRAMATICA PARA SENTENCIA IF
```

```
CONTROLIF
```

```
: RIF PARIZQ EXPRESION PARDER STATEMENT CONTROLELSE
```

```
;
```

```
CONTROLELSE
```

```
: RELSE STATEMENT
```

```
| RELSE CONTROLIF
```

```
| EPSILON
```

```
;
```

```
// INCREMENTO Y DECREMENTO COMO INSTRUCCION
```

```
INCREDECRE
```

```
: ID INCRE PTCOMA
```

```
| ID DECRE PTCOMA
```

```
;
```

```
//RETORNO DE EXPRESIONES
```

```
EXPRESION
```

: PARIZQ EXPRESION PARDER

| PRIMITIVO

| ACCEDERVAR

| ARITMETICA

| RELACIONALES

| LOGICOS

| OPERACIONESUNARIOS

| TOLOWER

| TOUPPER

| LENGTH

| TRUNCATE

| ROUND

| TYPEOF

| TOSTRING

;

// llamada a funciones

LLAMADAFUNCION

: ID PARIZQ PARDER

| ID PARIZQ ARGUMENTOS PARDER

;

// argumentos

ARGUMENTOS

: ARGUMENTOS COMA EXPRESION

;

//GRAMATICA PARA TOLOWER

TOLOWER

: RTOLOWER PARIZQ EXPRESION PARDER

;

//GRAMATICA PARA TOUPPER

TOUPPER

: RTOUPPER PARIZQ EXPRESION PARDER



;

//GRAMATICA PARA LENGTH

LENGTH

: RLENGTH PARIZQ EXPRESION PARDER

;

//GRAMATICA PARA TRUNCATE

TRUNCATE

: RTRUNCATE PARIZQ EXPRESION PARDER

;

//GRAMATICA PARA ROUND

ROUND

: RROUND PARIZQ EXPRESION PARDER

;

//GRAMATICA PARA TYPEOF

TYPEOF

: RTYPEOF PARIZQ EXPRESION PARDER

;

//GRAMATICA PARA TOSTRING

TOSTRING

: RTOSTRING PARIZQ EXPRESION PARDER

;

// OPERACION ARITMETICA

ARITMETICA

: EXPRESION MAS EXPRESION

| EXPRESION MENOS EXPRESION

| EXPRESION POR EXPRESION

| EXPRESION DIVISION EXPRESION

| EXPRESION POTENCIA EXPRESION

| EXPRESION MODULO EXPRESION

| MENOS EXPRESION U MENOS

;

// GRAMATICA RELACIONALES

RELACIONALES

: EXPRESION IGUALQ EXPRESION

| EXPRESION MENORQUE EXPRESION

| EXPRESION MAYORQUE EXPRESION

| EXPRESION DIFERENTEQ EXPRESION

| EXPRESION MAYORIGUALQUE EXPRESION

| EXPRESION MENORIGUALQUE EXPRESION

;

//GRAMATICA LOGICOS

LOGICOS

: EXPRESION OR EXPRESION

| EXPRESION AND EXPRESION

| NOT EXPRESION

;

//GRAMATICA INCREMENTO DECREMENTO

OPERACIONES UNARIOS

: ID INCRE

| ID DECRE

;

// ACCEDER A UNA VARIABLE

ACCEDERVAR

: ID

;

PRIMITIVO

: ENTERO

| DECIMAL

```
| CADENA  
| CHARACTER  
| TRUE  
| FALSE  
;
```

```
// GRAMATICA TIPO
```

```
TIPO
```

```
: RENTERO  
| RDOUBLE  
| RSTRING  
| RCHAR  
| RBOOLEAN  
;
```