2021

201807117

MADELINE ARIANA PÉREZ QUIÑÓNEZ

8-5-2021

Gramática



# INICIO

## ES LA ESTRUCTURA QUE TENDRÁ TODO EL DOCUMENTO

INICIO 🡪

OPCION\_INICIO

# Ambito global

## Se podrá declarar y asignar variables, declarar metodos y funciones. y tambien lleva la funcion exec

OPCION\_INICIO 🡪

OPCION\_INICIO CUERPO

| CUERPO

CUERPO 🡪

DECLARACION\_VARIABLES

| METODOS\_FUNCIONES

| CAMBIARVALOR\_VAR

| exec identificador parenA parenC ptcoma

| exec identificador parenA LISTA\_VALORES parenC ptcoma

| error ptcoma

# Metodos y funciones

## Se declarar metodos y funciones con y sin parametros, asimismo los metodos y funciones pueden o no pueden llevar cuerpo

METODOS\_FUNCIONES 🡪

TIPO identificador parenA parenC llaveA OPCIONES\_CUERPO llaveC

|TIPO identificador parenA LISTA\_PARAMETROS parenC llaveA OPCIONES\_CUERPO llaveC

|TIPO identificador parenA parenC llaveA llaveC

|TIPO identificador parenA LISTA\_PARAMETROS parenC llaveA llaveC

|void identificador parenA parenC llaveA llaveC

|void identificador parenA LISTAPARAMETROS parenC llaveA llaveC

|void identificador parenA parenC llaveA OPCIONES\_CUERPO llaveC

|void identificador parenA LISTA\_PARAMETROS parenC llaveA OPCIONES\_CUERPO llaveC

# parametros

## Se podrá declarar uno a más parametros

LISTA\_PARAMETROS 🡪

LISTA\_PARAMETROS coma PARAMETROS

| PARAMETROS

PARAMETROS🡪

TIPO identificador

# LLamada

## es la funcion llamada, manda a ejecutar metodos y funciones, con y sin parametros.

LLAMADA 🡪

identificador parenA LISTA\_VALORES parenC

|identificador parenA parenC

# lista de valores

## es una lista que puede llevar 1 o mas expresiones. se utiliza para las llamadas y para declaracion de un tipo de vector

LISTA\_VALORES 🡪

LISTA\_VALORES coma VALORES

| VALORES

VALORES 🡪

EXPRESION

# tipo

## es en donde se almacena los tipos de variables y funciones que se pueden utilizar en este lenguaje

TIPO🡪

Double

| String

| Boolean

| int

| Char

# cuerpo de metodos, funciones y ciclos

## es una lista de instrucciones que se ejecutará mas adelante

OPCIONES\_CUERPO🡪

OPCIONES\_CUERPO CUERPO\_METODOS\_FUNCIONES

| CUERPO\_METODOS\_FUNCIONES

| error ptcoma

CUERPO\_METODOS\_FUNCIONES 🡪

DECLARACION\_VARIABLES

| CICLOS

| CAMBIARVALOR\_VARIABLES

| LLAMADA ptcoma

| INCREMENTO\_DECREMENTO ptcoma

| SENTENCIA\_TRANS

| IMPRIMIR

# variables

## es en donde se declaran los diferentes tipos de variables

DECLARACION\_VARIABLES 🡪

TIPO identificador ptcoma

| TIPO identificador signoigual EXPRESION ptcoma

| DECLARACION\_VECTOR

| DECLARACION\_LISTA

## es en donde se le asigna distintos valores a las variables

CAMBIARVALOR\_VAR 🡪

identificador signoigual EXPRESION ptcoma

|CAMBIARVALOR\_VECTOR

|CAMBIARVALOR\_LISTA

|AGREGAR\_LISTA

## es en donde se declaran los vectores de diferentes tipos

DECLARACION\_VECTOR 🡪

TIPO corcheteA corcheteC identificador signoigual new TIPO corcheteA EXPRESION corcheteC ptcoma

| TIPO corcheteA corcheteC identificador signoigual llaveA LISTA\_VALORES llaveC ptcoma

## es en donde se asigna un valor a una posicion del vector

CAMBIARVALOR\_VECTOR 🡪

identificador corcheteA EXPRESION corcheteC signoigual EXPRESION ptcoma

## es en donde se declaran las listas de diferentes tipos

DECLARACION\_LISTA🡪

list menor TIPO mayor identificador signoigual new list menor TIPO mayor ptcoma

| list menor TIPO mayor identificador signoigual tochararray parenA EXPRESION parenC ptcoma

## es en donde se asigna un valor a una posicion de una lista

CAMBIARVALOR\_LISTA 🡪

identificador corcheteA corcheteA EXPRESION corcheteC corcheteC signoigual EXPRESION ptcoma

## es en donde se agrega un valor al final de una lista

AGREGAR\_LISTA 🡪

identificador punto add parenA EXPRESION parenC ptcoma

# casteos

## estructura y tipos de variables a los que se les puede hacer casteos

TIPO\_CASTEO 🡪

Double

| int

| Char

CASTEOS 🡪

parenA TIPO\_CASTEO parenC EXPRESION

| tostring parenA EXPRESION parenC

# incremento y decremento

## es la produccion en donde se le incrementa o decrementa de uno en uno a la varible correspondiente al identificador

INCREMENTO\_DECREMENTO 🡪

identificador masmas

| identificador menosmenos

# funciones

## es la produccion en donde esta la estructura de los casteos y las funciones nativas y especiales

FUNCIONES 🡪

CASTEOS

| tochararray parenA EXPRESION parenC

| length parenA SACAR\_LONG parenC

| toLower parenA EXPRESION parenC

| toUpper parenA EXPRESION parenC

| truncate parenA EXPRESION parenC

| round parenA EXPRESION parenC

| typeof parenA EXPRESION parenC

# sacar longitud

## es la produccion en donde estan declarados los tipos de valores que se les puede sacar la longitud (cadenas e identificadores de variables string, vectores y listas)

SACAR\_LONG 🡪

cadenatexto

| identificador

# IMPRIMIR

## es la produccion en donde esta la estructura de la funcion print que puede llevar dentro una expresion o puede ir vacio

IMPRIMIR 🡪

print parenA EXPRESION parenC ptcoma

|print parenA parenC ptcoma

# Sentencias ciclicas y de control

## es la produccion en donde esta la lista de ciclos y sentencias de control, incluyendo el operador ternario

CICLOS 🡪

WHILE

|IF

|ELSEIF

|FOR

|SWITCH

|DO\_WHILE

|OPE\_TERNARIO ptcoma

## es en donde se declara la estructura del ciclo for que puede venir con cuerpo o vacio

FOR 🡪

for parenA FOR\_DEC EXPRESION ptcoma FOR\_ASIG parenC llaveA OPCIONES\_CUERPO llaveC

|for parenA FOR\_DEC EXPRESION ptcoma FOR\_ASIG parenC llaveA llaveC

### Estructura de la declaracion o asignacion del contador

FOR\_DEC 🡪

DECLARACION\_VARIABLES

|CAMBIARVALOR\_VAR

### Estructura de asignacion del contador (incremento o decremento)

FOR\_ASIG 🡪

identificador signoigual EXPRESION

|identificador masmas

| identificador menosmenos

## es en donde se declara la estructura del ciclo while que puede venir con cuerpo o vacio

WHILE 🡪

while parenA EXPRESION parenC llaveA OPCIONES\_CUERPO llaveC

|while parenA EXPRESION parenC llaveA llaveC

## es en donde se declara la estructura del ciclo do while que puede venir con cuerpo o vacio

DO\_WHILE 🡪

do llaveA OPCIONES\_CUERPO llaveC while parenA EXPRESION parenC ptcoma

|do llaveA llaveC while parenA EXPRESION parenC ptcoma

## es en donde se declara la estructura de la sentencia de control switch que puede venir con cuerpo o vacio. tambien puede venir con uno a mas casos

SWITCH 🡪

switch parenA EXPRESION parenC llaveA LISTA\_CASOS llaveC

### Estructura de la lista que puede llevar uno o mas casos que pueden ir con cuerpo o vacios

LISTA\_CASOS 🡪

LISTA\_CASOS CASOS

| CASOS

CASOS 🡪

case EXPRESION dospuntos OPCIONES\_CUERPO

| default dospuntos OPCIONES\_CUERPO

| case EXPRESION dospuntos

| default dospuntos

## es en donde se declara la estructura de la sentencia de control else if y el if else que puede venir con cuerpo o vacio. esto lleva una lista de else y else if.

ELSEIF 🡪

if parenA EXPRESION parenC llaveA OPCIONES\_CUERPO llaveC LISTA\_ELSEIF

| if parenA EXPRESION parenC llaveA llaveC LISTA\_ELSEIF

### Estructura de la lista que puede llevar uno o mas else if o else que pueden ir con cuerpo o vacios

LISTA\_ELSEIF 🡪

LISTA\_ELSEIF UN\_ELSEIF

| UN\_ELSEIF

UN\_ELSEIF 🡪

else if parenA EXPRESION parenC llaveA OPCIONES\_CUERPO llaveC

| else llaveA OPCIONES\_CUERPO llaveC

| else if parenA EXPRESION parenC llaveA llaveC

| else llaveA llaveC

## es en donde se declara la estructura de la sentencia de control if que puede venir con cuerpo o vacio.

IF 🡪

if parenA EXPRESION parenC llaveA OPCIONES\_CUERPO llaveC

| if parenA EXPRESION parenC llaveA llaveC

## es en donde se declara la estructura de la opereracion ternaria, que tiene como funciones las expresiones

OPE\_TERNARIO 🡪

EXPRESION interrogacion EXPRESION dospuntos EXPRESION

# Sentencias transferncia

## es la produccion en donde esta la lista de sentencias de transferencias con su estructura (break, continue, return)

SENTENCIA\_TRANS 🡪

break ptcoma

|continue ptcoma

|return ptcoma

|return EXPRESION ptcoma

# expresion

## es la produccion en donde esta la lista de tipos de valores, diversas funciones, operaciones aritmeticas, logicas, realcionales, entre otras.

EXPRESION 🡪

EXPRESION suma EXPRESION

| EXPRESION resta EXPRESION

| EXPRESION multiplicacion EXPRESION

| EXPRESION division EXPRESION

| EXPRESION exponente EXPRESION

| EXPRESION modulo EXPRESION

| resta EXPRESION

| parenA EXPRESION parenC

| EXPRESION igualacion EXPRESION

| EXPRESION diferenciacion EXPRESION

| EXPRESION menor EXPRESION

| EXPRESION menorigual EXPRESION

| EXPRESION mayor EXPRESION

| EXPRESION mayorigual EXPRESION

| EXPRESION or EXPRESION

| EXPRESION and EXPRESION

| not EXPRESION

| OPE\_TERNARIO

| INCREMENTO\_DECREMENTO

| identificador corcheteA EXPRESION corcheteC

| identificador corcheteA corcheteA EXPRESION corcheteC corcheteC

| FUNCIONES

| LLAMADA

| entero

| true

| false

| cadenatexto

| identificador

| decimal

| caracter

# no terminales

|  |  |
| --- | --- |
| "new" nnew | "int" int |
| "Double" Double | "Boolean" Boolean |
| "Char" Char | "String" String |
| "true" true | "false" false |
| "list" list | "add" add |
| "if" if | "else" else |
| "switch" switch | "case" case |
| "default" default | "break" break |
| "while" while | "for" for |
| "do" do | "continue" continue |
| "return" return | "void" void |
| "print" print | "toLower" toLower |
| "toUpper" toUpper | "length" length |
| "truncate" truncate | "round" round |
| "typeof" typeof | "toString" tostring |
| "toCharArray" tochararray | "exec" exec |
| "==" igualacion | "=" signoigual |
| "!=" diferenciacion | "<=" menorigual |
| "<" menor | ">=" mayorigual |
| ">" mayor | "?" interrogacion |
| ":" dospuntos | "||" or |
| "&&" and | "!" not |
| "(" parenA | ")" parenC |
| "++" masmas | "--" menosmenos |
| "^" exponente | "-" resta |
| "+" suma | "\*" multi |
| "/" division | "%" modulo |
| ";" ptcoma | "." punto |
| "{" llaveA | "}" llaveC |
| "[" corcheteA | "]" corcheteC |
| "," coma | [0-9]+("."[0-9]+)\b decimal; |
| [0-9]+\b entero; | ([a-zA-Z])([a-zA-Z0-9\_])\* identificador |
| ["\”"]([^"\”"])?["\”"] cadenatexto | ["\'"]([^"\'"])?["\'"] caracter |