# Lista de Tokens en el lenguaje PCL

## Comentarios

|  |  |  |  |
| --- | --- | --- | --- |
| Descripción  del comentario | Tipo | [[1]](#endnote-1)Identificador | ¿Tomado en cuenta? |
| Comentario que inicia con // | Comentario de línea | COMMENT |  |
| Comentario (\* *contenido* \*) | Comentario de bloque | COMMENT |  |
| Comentario { *contenido* } | COMMENT |  |

## Identificadores

Tipos: Constantes, variables, tipos de datos, procedimientos, funciones

|  |  |
| --- | --- |
| Detalle | ¿Tomado en cuenta? |
| Secuencia de 1 a 127 caracteres |  |
| Inicia con una letra |  |
| No tiene espacios ni símbolos:  &, ¡, \* |  |
| No es una palabra reservada |  |
| No existe diferencia entre mayúscula y minúscula |  |

## Operadores

|  |  |  |  |
| --- | --- | --- | --- |
| OPERADORES | [[2]](#endnote-2)Identificador | Tipo | ¿Tomado en cuenta? |
| , | COMA | Coma |  |
| ; | PUNTO\_COMA | Punto y coma |  |
| ++ | INC | Incremento |  |
| -- | DEC | Decremento |  |
| >= | MAYOR\_IGUAL | Mayor o igual que |  |
| > | MAYOR | Mayor |  |
| <= | MENOR\_IGUAL | Menor o igual que |  |
| < | MENOR | Menor |  |
| <> |  | Diferencia |  |
| = | ASIGN | Asignación |  |
| + | SUMA | Suma |  |
| - | RESTA | Resta |  |
| \* | MUL | Multiplicación |  |
| / |  |  |  |
| ( | PAR\_APER | Paréntesis de apertura |  |
| ) | PAR\_CERR | Paréntesis de cerrado |  |
| [ | BRA\_APER | Bracket de apertura |  |
| ] | BRA\_CERR | Bracket de cerrado |  |
| := |  |  |  |
| . | PUNTO | Punto |  |
| : | DOS\_PUNTO | Dos puntos |  |
| += | SUMA\_ASIGN | Suma y asignación |  |
| -= | RESTA\_ASIGN | Resta y asignación |  |
| \*= | MUL\_ASIGN | Multiplicación y asignación |  |
| /= |  |  |  |
| >> | SHIFT\_DER | Shift derecho |  |
| << | SHIFT\_IZQ | Shift izquierdo |  |
| <<= | SHIFT\_IZQ\_ASIGN | Shift izquierdo y asignación |  |
| >>= | SHIFT\_DER\_ASIGN | Shift derecho y asignación |  |
| OPERADORES QUE SON PALABRAS RESERVADAS | | | |
| NOT | NOT | Operador lógico NOT |  |
| OR | OR | Operador lógico OR |  |
| AND | AND | Operador lógico AND |  |
| XOR | XOR | Operador lógico XOR |  |
| DIV | DIV | División |  |
| MOD | MOD | Módulo |  |

## Palabra reservada

|  |  |
| --- | --- |
| PALABRA RESERVADA | ¿Tomado en cuenta? |
| AND |  |
| ARRAY |  |
| BEGIN |  |
| BOOLEAN |  |
| BYTE |  |
| CASE |  |
| CHAR |  |
| CONST |  |
| DIV |  |
| DO |  |
| DOWNTO |  |
| ELSE |  |
| END |  |
| FALSE |  |
| FILE |  |
| FOR |  |
| FORWARD |  |
| FUNCTION |  |
| GOTO |  |
| IF |  |
| IN |  |
| INLINE |  |
| INT |  |
| LABEL |  |
| LONGIT |  |
| MOD |  |
| NIL |  |
| NOT |  |
| OF |  |
| OR |  |
| PACKED |  |
| PROCEDURE |  |
| PROGRAM |  |
| READ |  |
| REAL |  |
| RECORD |  |
| REPEAT |  |
| SET |  |
| SHORINT |  |
| STRING |  |
| THEN |  |
| TO |  |
| TRUE |  |
| TYPE |  |
| UNTIL |  |
| VAR |  |
| WHILE |  |
| WRITE |  |
| XOR |  |

## Literales

|  |  |
| --- | --- |
| Detalle | ¿Tomado en cuenta? |
| Posee al menos un dígito de cada lado del punto decimal, sea cero.  5 -> 5.0  0.5 -> .5 |  |
| Uso denotación científica |  |
| Los strings y caracteres está entre “” |  |
| Los string pueden ser varias líneas |  |
| Caracteres representados con el signo # seguido por un número |  |

1. El identificador es utilizado en la clase SYM para manejar el código único para cada token reconocido [↑](#endnote-ref-1)
2. El identificador es utilizado en la clase SYM para manejar el código único para cada token reconocido [↑](#endnote-ref-2)