**Lenguaje: Cati**

**Desarrolladores:**

* Dulce Celeste Cruz Ramírez
* Oscar Eduardo Navarrete Castrejón

Cati es un lenguaje de programación construido en Java, está diseñado para ser fácil de usar e intuitivo. Su sintaxis es simple, se debe separar cada palabra/símbolo por un espacio, y cada línea debe terminar con < ; >. Es necesario respetar las letras mayúsculas y minúsculas, según sea el caso.

**Lista de palabras reservadas:**

|  |  |  |
| --- | --- | --- |
| **Token** | **Función** | **Lexema** |
| DEC | Declaración | DEC |
| Tipo | Categorizar variables | num, bool, cad, car |
| Variable | Nombre de variable |  |
| Asignación | Asignar valores a variables | = |
| Final | Terminar línea de código | ; |
| Decisión | Indicar una decisión | Si, Sino |
| Numero | Indicar que es un valor numérico | [0-9] |
| Lectura | Indica lectura | Lee |
| Imprime | Indica impresión | Imprime |
| Hacer | Indica inicio de ciclo | Hacer |
| Operador | Categoriza operadores | <, >, ==, !=, <=, >= |

SENTENCIA DEC

DEC es una palabra reservada de Cati para declarar nuevas variables en el programa, siguiendo la siguiente sintaxis.

Sintaxis:

DEC <tipo> <variable> ;

DEC <tipo> <variable1> & <variable2> & <variable3> …… ;

DEC <tipo> <variable> = <dato> ;

VARIABLES

Cati acepta variables que sigan los siguientes lineamientos:

* Debe empezar con una letra (ya sea mayúscula o minúscula)
* Acepta números y guion bajo, pero debe terminar con una letra o número.
* No acepta otros caracteres especiales
* Que no haya sido declarada esa variable previamente, si es así, no lo aceptará.

SENTENCIA Si, Sino

Son lexemas que indican condiciones, de los cuales es necesario que por cada Sino exista por lo menos un Si

Sintaxis:

Si ( <variable, tipo de dato > <operando> <variable, tipo de dato> ) [ ;

Sino ( <variable, tipo de dato > <operando> <variable, tipo de dato> ) [ ;

SENTENCIA Hacer

Hacer es una palabra reservada de Cati para indicar el inicio de un ciclo, en el que se índice con una variable previamente declarada e inicializada el inicio del ciclo, e indicar con otra variable o número, el termino de dicho ciclo

Sintaxis:

Hacer <variable> <operando> <numero> <+,-> ;

Hacer <variable> <operando> <variable> <+,-> ;

**Programa de prueba:**

Si ( a <= b ) [ ;

Si ( ) ;

Sino ( k <= 23 ) ;

Si ( h == ) ;

Si ) ;

Si ( c = D ) [ ;

Si ( c >= D ) [

Si ( "hola" != "hola" ) [ ;

Si ( "hola == 'H' ) [ ;

Si ( "hola" == 'Hola' ) [ ;

Si ( "hola" == h ) [ ;

DEC num a = 0 ;

DEC bool b = Verdadero ;

DEC num c ;

Hacer a < 5 ;

Hacer < 3 ;

Hacer a == 10 - ;

Hacer a == b ;

Hacer a <= c - ;

Hacer a ;

Hacer a > 2 \* ;

**Impresión en consola:**

run:

Linea 2 Error: No agrego datos a comparar

Linea 3 Error: Ausencia de simbolo [

Linea 4 Error: Falta segundo elemento a comparar

Linea 5 Error: Ausencia de '('

Linea 6 Error en el simbolo de comparacion

Linea 7 Error de termino de linea, ausencia de ;

Linea 9 Error: Primer dato de la comparacion no valido

Linea 10 Error: Segundo dato de la comparacion no valido

Linea 15 Error, falta operador de conteo

Linea 16 Error de sintaxis, se espera una variable

Linea 18 Error, diferentes tipos de dato

Linea 20 Error, se espera operador de decisión

Linea 21 Error, operador de conteo invalido, se espera '-' o '+'

Tabla de simbolos

Nombre Id Tipo Valor Lectura

a id0 num 0 N

b id1 bool Verdadero N

c id2 num 0 N