**SMILEBASIC**

Esta basado en un lenguaje BASIC. Como se sabe basic es de los mejores lenguajes de programación para los que no saben programar y asi aprendemos a programar desde cero ¡Tan pronto como inicie SmileBASIC, puede comenzar a programar!

**INSTRUCCIONES**

**PRINT:** Instrucción en la cual se muestra impreso cualquier cadena de carácter, números, simbolos, etc entre comillas. Si se escribe 2 print y el priemro se le escribe al final ‘;’ salen impreso los 2 como uno solo pero si se le escribe ‘,’ al primero salen los dos print separados, siempre en la misma línea pero con un espacio.

EJEMPLO:

1. PRINT “hello”;
2. PRINT “world”

SALIDA:

helloworld

1. PRINT “hello”,
2. PRINT “world”

SALIDA:

hello world

**INPUT:** Instrucción en la cual se le ingresa valor a la variable durante la salida (con el teclado) en espera de un valor especifico a la variable ya sea numero o carácter y luego presiona ENTER. Si el número de elementos de entrada es insuficiente, se mostrará "Rehacer desde el inicio" para volver a ingresar

EJEMPLO:

1. INPUT “Valor de la base: ”;B
2. A = 4
3. C = B \* A
4. PRINT “El área de un triangulo es”; C

SALIDA:

Valor de la base: ? 6

El área de un triangulo es 24

**LINPUT:** Instrucción igual a INPUT con la excepción de que sólo se le ingrese una cadena de caracteres y siempre en espera hasta que se presiona ENTER

EJEMPLO:

1. LINPUT “Nombre Completo: ”; Name$
2. PRINT “Su nombre es”; Name$

SALIDA:

Nombre Completo: ? Edwin Nuñez

Su nombre es Edwin Nuñez

**GOTO:** Instrucción que hace un salto de línea de la primera hasta la ultima línea y siempre debe empezar con @ antes de los caracteres que se le escriba a lado. hará que este programa siempre regrese a la primera línea, por lo que para detenerlo, presione ENTER.

EJEMPLO:

1. @PRIMERO
2. INPUT “Valor de la base: ”;B
3. A = 4
4. C = B \* A
5. PRINT “El área de un triangulo es”; C

6 GOTO @PRIMERO

SALIDA:

Valor de la base: ? 6

El área de un triangulo es 24

Valor de la base: ? 2

El área de un triangulo es 8

Valor de la base: ?

**GOSUB:** Instrucción que llama una subrutina de acuerdo con el valor de la variable principal.

EJEMPLO:

1 ON ID GOSUB @SUB\_A, @SUB \_B   
2 PRINT "SALIR"   
3 @SUB \_A   
4 PRINT "ID = 0": RETURN  
5 @SUB \_B   
6 PRINT "ID = 1" : RETURN

**IF:** Instrucción para verificar el valor de una variable si cumple o no con la condición y ejecutar las instrucciones dentro del IF

**THEN:** instrucción que se coloca al final del IF es decir justo a lado de la condición y lo que esta a punto de ejecutarse cuando se cumple la condición

**ELSE:** Instrucción que se usa en caso que no se cumpla la condición IF para que cumpla entonces la alternativa.

**ELSEIF:** Instrucción que se usa en caso que no se cumpla la condición IF y al mismo tiempo se verifica si cumple o no otra condición.

**ENDIF:** Instrucción para marcar por terminado toda la condición IF

EJEMPLO:

1. @PRIMERO
2. INPUT “Valor de la base: ”;B
3. IF B == 0 THEN
4. PRINT “continuar”;
5. ELSEIF B <= 2 THEN
6. B = B + 1
7. ELSE
8. A = 4
9. C = B \* A

10 PRINT “El área de un triangulo es”; C

11 GOTO @PRIMERO

12 ENDIF

**ON:** Instrucción que se extiende de acuerdo con el valor de la variable de control. El numero inicial seria 0. También llama subrutinas.

EJEMPLO:

1 ON ID GOTO @SALTO\_A, @SALTO \_B   
2 PRINT "FINALIZADO"   
3 @SALTO \_A   
4 PRINT "ID = 0"  
5 @SALTO \_B   
6 PRINT "ID = 1"