



UNIVERSIDAD POLITÉCNICA DE QUINTANA ROO

Formando Triunfadores

El estudiante baca pastor david sthepen perteneciente al séptimo cuatrimestre de la Ingeniería en software del grupo 27AV con la supervisión del catedrático Manuel Flores, realizaron el análisis y reporte de expediente de **CURSOS AUTOGESTIVOS**.

1. Introducción.

01

El objetivo principal de este documento es proporcionar una comprensión detallada del lenguaje de programación MikenCode, abarcando sus estructuras, funciones, variables y otros elementos fundamentales.

Este recurso está diseñado para ser de utilidad a cualquier persona que tenga interés en aprender y dominar este lenguaje de programación.

Objetivo:

El objetivo principal es crear un lenguaje de programación básico que sirva como una herramienta inspiradora para las nuevas generaciones de programadores, invitándolos a adentrarse en el emocionante mundo de la programación.

Valor del Producto:

Nuestro producto se distingue por ofrecer variables simples y estructuras de programación que son lo suficientemente grandes como para ser poderosas, pero al mismo tiempo, son fácilmente comprensibles incluso para aquellos sin experiencia previa en programación. Queremos que nuestra plataforma sea accesible y acogedora para todos, sin importar su nivel de conocimientos previos en este campo.

2. Lista de caracteres que Pueden usar.

02

Dentro de la documentación se van a estar definiendo variables como funciones que requieren una estructura de caracteres específicos en diferentes áreas. Este apartado es para definir las abreviaturas que se utilizaran en el desarrollo de este documento y ayudar a que el documento sea más fácil de comprender también de manera secundaria definiremos los caracteres que se pueden usar el en lenguaje de programación MikenCode.

| Abreviatura | Caracteres. | Nota. |
|-------------|---|---------------------------|
| L | Q,W,E,R,T,Y,U,I,O,P,A,S,D,F,G,H,J,K,L,Ñ,Z,X,C,V,B,N,M, q,w,e,r,t,y,u,i,o,p,a,s,d,f,g,h,j,k,l,ñ,z,x,c,v,b,n,m | Abecedario |
| N | 0,1,2,3,4,5,6,7,8,9 | Números |
| S | [,] , > , < , " , # , * , ; , - , _ , ¿ , ? , @ , (,) | símbolos |
| OM | + , - , / , * | Operadores Matemáticos |
| OL | = , => , <= , < , > , != | Operadores Lógicos |

3. Estructura

03

Encabezado:

Bibliotecas declaradas de la siguiente manera

```
@invoco_a: [Biblioteca a invocar]
```

Usando la palabra reservada **@invoco_a**:

Cuerpo:

El inicio de los procesos del programa se estructura de la siguiente manera

```
@Proceso: [nombre del proceso]
```

Usando la palabra reservada **@Proceso**:

! Solo se ejecutara por defecto el Proceso declarado Master:

```
@Proceso: Master
```

El cierre de los procesos del programa se estructura de la siguiente manera

```
@Terminado
```

! Ten en cuenta que no se puede declarar un Proceso dentro de un Proceso ejemplo de declaración inesperada

```
@Proceso: Master
```

```
  @Proceso: Segundo
```

```
  @Terminado
```

```
@Terminado
```

Ejecución de una parte del cuerpo

Dentro de un Proceso no se puede declarar un Proceso pero si se puede iniciar un proceso con la palabra reservada **@Iniciar**: [nombre]

```
@Proceso: Master
```

```
  @iniciar: Segundo
```

```
@Terminado
```

4. Sintaxis.

04

4.1 Asignación

Para la creación de variables en mikencode se utiliza la palabra reservada # [tipo de dato] en la siguiente tabla se denominan las estructuras para cada variable.

| Palabra reservada | Asignación de nombre | Aplicación de operador | Entradas de la lista de caracteres | Finalizar | Comentarios |
|-------------------|----------------------|------------------------|------------------------------------|-----------|---|
| #Numero | <L>,<N>,<S> | <=> | <N>.x10 | <;> | Solo acepta 10 números para declarar una cantidad. |
| #Texto | <L>,<N>,<S> | <=> | <L>.x55 | <;> | Solo acepta 55 caracteres un texto |
| #Gato | <L>,<N>,<S> | <=> | <0>,<1> | <;> | En referencia del gato de Schrödinger te pregunta si está vivo o muerto para un valor booleano. |
| #Decimal | <L>,<N>,<S> | <=> | <N>.x10 <.> <N>.x10 | <;> | Solo acepta 10 números enteros y a partir del punto Otros 10 números decimales |

4.2 Operación

Para la creación de operaciones en mikencode se utilizan las siguientes estructuras.

| Estructura de decisión | | | | | | | | | | |
|------------------------|---|------|---|---|----|---|----------------|---|----|----|
| Parte necesaria | | | | | | | Parte opcional | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| EsVerdad | <#Numero>, <#Texto>, <#Gato>, <#Decimal>, <N>, <L> | | <#Numero>, <#Texto>, <#Gato>, <#Decimal>, <N>, <L> | ¿ | // | ? | NoEsVerdad | ¿ | // | ? |

| Estructura finita <FOR> | | | | | | | | | | |
|-------------------------|-----------|------|-----|------------|-----------|------|-----|--------|----|-------------|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| EsVerdad | <#Numero> | | <N> | <Entonces> | <#Numero> | <OM> | <N> | <Para> | // | Fin Para |

| Estructura finita <FOR> | | | | | | | | |
|-------------------------|------------|----------|-------|---|------|---|-----|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Repetir | //procesos | Mientras | < (> | #Numero, #Texto, #Gato, #Decimal | | #Numero, #Texto, #Gato, #Decimal | <)> | ; |