

Librería “string” [LUA]

La librería que veremos a continuación es referente a los **strings** y con las funciones que hay en ellas notarán que los **strings** son más que simples objetos que carecen de valor.

Retorna el **Valor Numérico** del caracter **s** según su equivalente decimal en la [Tabla ASCII](#)

Veamos un ejemplo de la Tabla:

64	@	96	`
65	A	97	a
66	B	98	b
67	C	99	c
68	D	100	d

Vemos que el valor decimal de “@” es 64, así como el de “b” es 98.

Ejemplo 1:

string.byte(“A”) = 65

Ejemplo 2:

Str = “C”

string.byte(s) **string.byte(Str) = 67**

Los valores decimales de la **Tabla ASCII** van desde el 0 hasta el 255.

El modo abreviado de usar esta función es así:

Ejemplo 3:

s = “B”

s:byte() = 66

Ejemplo 4:

s = “d”

s:byte() = 100

Esta función tiene más modo de uso, pero al menos por el momento no serán de gran utilidad y de ahí que las haya omitido.

Retorna el caracter asignado por la **Tabla ASCII** de un número entre 0 y 255. Ejemplo:

string.char(...) **string.char(65) = “A”**

string.char(66) = “B”

string.char(65, 66, 100) = “ABd”

Retorna las posiciones del inicio y final del string **s** dentro de un string mayor **ptr**.

Ejemplo 1:

L = "Demo lua string"

s = "lua"

string.find(s, L) = 6, 8

string.find(s, ptr) 6 es el inicio y 8 es el final:

D	e	m	o		l	u	a		s
1	2	3	4	5	6	7	8		

La manera abreviada es:

s:find(L) = 6, 8

Si el string **s** no pertenece al string **ptr**, entonces retorna **nil**

Inserta valores al string **s**. El modo general de asignar los valores el por medio del string "%s".

Ejemplo 1:

string.format("\\be%s", l.shadow)

string.format

Es decir que inserta el valor de **l.shadow** al string "\\be%s".

(s, ...)

Hay dos formas abreviadas de usar esta función en el Kara Effector:

· **format(s, ...)**

· **F(s, ...)**

Ejemplo 2:

F("\\pos(%s,%s)", syl.center, 20*pi)

Por cada vez que aparezca "%s" en el string, se debe asignar el valor que se insertará en esa posición.

Este modo es el más usado en el Kara Effector y es la función string que más se usa en él. Los demás modos no son tan relevantes para hacer Efectos Karaokes.

Reemplaza al string **s** que pertenece al string **ptr**, por un valor u otro string llamado **replace**, **n** cantidad de veces.

El modo abreviado de uso es:

ptr:gsub(s, replace, n)

n es un entero positivo, es opcional en la función.

Ejemplo 1:

ptr = "el sol en la mañana"

ptr:gsub("a", "X")

string.gsub = "el sol en lX mXñXnX"

(s, ptr, replace, n) La función reemplazó al carácter "a" por "X" todas las veces.

Ejemplo 2:

ptr = "el sol en la mañana"

ptr:gsub("a", "X", 2)

= "el sol en lX mXñana"

El reemplazo se hizo solo dos veces, dado que **n** = 2

Esta función también es mucho más extensa de lo que aparenta, pero de eso nos ocuparemos más adelante.

Retorna la cantidad de caracteres que tiene el string **s**.

El modo abreviado es:

s:len()

Ejemplo 1:

s = "&HFFFFFF&"

string.len(s) **s:len()** = 9

Ejemplo 2:

s = "El sol"

s:len() = 6

No olvidemos que el espacio entre palabra y palabra también es un caracter.

Retorna al string **s** con todas las letras mayúsculas en él convertidas en minúsculas.

El modo abreviado es:

s:lower()

Ejemplo 1:

string.lower(s)

s = "La Noche"

s:lower() = "la noche"

Ejemplo 2:

s = "EL SOL"

s:lower() = "el sol"

Repite al string **s** una **n** cantidad de veces.

El modo abreviado es:

s:rep(n)

Ejemplo 1:

string.rep(s, n) s = "Demo"

s:rep(3) = "DemoDemoDemo"

Ejemplo 2:

s = "lua "

s:rep(5) = "lua lua lua lua lua"

Invierte al string **s**.

El modo abreviado es:

s:reverse()

Ejemplo 1:

string.reverse(s) s = "Demo"

s:reverse() = "omeD"

Ejemplo 2:

s = "La Luna y el Sol"

s:reverse() = "loS le y anuL aL"

Recorta al string **s** desde la posición **i** hasta la posición **j**.

Tanto **i** como **j** son números enteros y pueden ser positivos o negativos. Al ser positivos la posición se empieza a contar de izquierda a derecha, si son negativos, se cuenta al revés.

El modo abreviado es:

s:sub(i, j)

Ejemplos:

s = "El Sol"

string.sub(s, i, j) s:sub(1, -1) = "El sol"

s:sub(1, 1) = "E"

s:sub(-3, -1) = "Sol"

s:sub(2, -1) = "l Sol"

El parámetro **j** es opcional

s:sub(-2) = "o"

s:sub(4) = "S"

Retorna al string **s** con todas las letras minúsculas en él convertidas en mayúsculas.

El modo abreviado es:

s:upper()

Ejemplo 1:

string.upper(s) s = "La Noche"

s:upper() = "LA NOCHE"

Ejemplo 2:

s = "¿Qué?, no puedes!"

s:upper() = "¿QUÉ?, NO PUEDES!"

De esta Librería también omití un par de funciones que quizás, al necesitarlas más adelante, las explique a cada una de ellas, aunque al ser una biblioteca de **LUA** se hace un poco más simple hallar información en la web sobre ellas.

Terminada esta librería, es momento para ver las formas abreviadas de las funciones comúnmente usadas en el **Kara Effector**:

ABREVIATURA FUNCIÓN

pi	math.pi
sin	math.sin
cos	math.cos
tan	math.tan
asin	math.asin
acos	math.acos
atan	math.atan
sinh	math.sinh
cosh	math.cosh
tanh	math.tanh
log	math.log10
ln	math.log
abs	math.abs
floor	math.floor
ceil	math.ceil
deg	math.deg
rad	math.rad
r	math.random
R	math.R
Rf	math.Rfake
rand	math.random
format	string.format
F	string.format

Todas las anteriores abreviaturas de las funciones aplican tanto en lenguaje **LUA** como en **Automation Auto-4**.