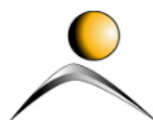


Referencia al lenguaje

Por Luis López Lezcano



mitosoftware
Pasión Por el software

Agradecimientos:

Debido a la gran cantidad de peticiones que me llegaban al correo electrónico, para la creación de una guía que encierre todos los comandos de Ninox Database, me decidí a crear este manual de referencia que espero que sea de gran ayuda.

Ha supuesto un gran trabajo traducir cada comando e incluirlos en Español, además de trabajos de investigación para poder ofrecerle el producto final con la calidad que este merece.

Le agradezco la adquisición de este manual de referencia.

Espero sinceramente que le sirva de gran ayuda en sus proyectos.

Atentamente.

Luis López Lezcano.

Indice

Indice.....	3
Aritmética básica.....	4
Operaciones de cadena.....	4
Trabajar con campos.....	5
Trabajar con referencias de tablas.....	6
Trabajar con funciones.....	6
Expresiones lógicas.....	7
Toma de decisiones.....	8
Filtrar valores.....	8
Las variables.....	9
Matrices.....	9
Funciones de matrices.....	10
Modificar datos.....	11
Creación de registros.....	11
Seleccionar registros.....	11
Eliminar registros.....	12
Bucles.....	13
Conversiones.....	13
Funciones matemáticas.....	14
Funciones de texto.....	15
Funciones de fecha.....	16
Formatos para la función format().....	18
Funciones de creación de calendarios.....	19
Funciones de color.....	20
Funciones de iconos.....	20
Funciones de localización.....	20
Automatización de la interface de usuario.....	20
Automatización de gestión de usuarios.....	21
Otras funciones.....	22
Iconos 1-80.....	24
Iconos 81-160.....	25
Iconos 161-240.....	26
Iconos 241-250.....	27

Aritmética básica

Ninox dispone de aritmética comparable al resto de lenguajes de programación.

+	Sumar	$1 + 2 = 3$
-	Resta	$3 - 2 = 1$
*	Multiplica	$2 * 3 = 6$
/	Divide	$10 / 5 = 2$

Los símbolos utilizados son siempre los mismos.

Recuerde que las operaciones respetan la precedencia típica de los operadores.

$1 + 2 * 3$ ó $1 + (2 * 3)$ Resultado de ambas 7

Las expresiones pueden ir entre paréntesis y estos se ejecutarán antes.

$(1 + 2) * 3$ Resultado 9

Tenga en cuenta que Ninox eliminará automáticamente los paréntesis insignificantes.

Ejemplo:

Si ingresa	$1 + (2 * 3)$
Ninox almacenará la expresión como	$1 + 2 * 3$

Operaciones de cadena

Se puede ingresar una cadena de texto, esta irá siempre entre comillas dobles.

Ejemplo:

Cadena de texto	"Hola mundo"
-----------------	--------------

Se pueden concatenar cadenas de texto, esto quiere decir qué suma dos cadenas de texto.

Ejemplo:

Cadena de texto	"Pedro" + "Delgado"
Resultado	PedroDelgado

Observe que en el ejemplo anterior, el resultado nos devuelve: PedroDelgado (todo junto).

Veamos otro ejemplo

Ejemplo:

Cadena de texto	"Pedro" + " " + "Delgado"
Resultado	Pedro Delgado

En esta ocasión hemos sumado un carácter de tipo espacio y el resultado ya es correcto.

Cuando concatenamos cadenas con valores, Ninox nos devolverá siempre texto.

Ejemplo:

Cadena de texto	"Pedro" + " " + "Delgado " + 42
Resultado	Pedro Delgado 42

Observe que en esta ocasión hemos incorporado otro espacio a la derecha del apellido Delgado y este ha provocado que mantenga ese espacio en la suma, además el número 42 se ha concatenado a la cadena de texto.

Por último para poder colocar a nuestro texto las comillas dobles será necesario repetirlas. doblemente.

Ejemplo:

Cadena de texto	"Mi nombre es ""Pedro""!"
Resultado	Mi nombre es "Pedro"!

Trabajar con campos

La mayoría de las funciones de Ninox operan con los registros de la base de datos. Ninox accede a todos campos del registro seleccionado o activo en ese momento.

Imagine una tabla Cliente que consta de los siguientes campos:

Nombre , Apellido, Calle , Código Postal , Ciudad .

Para generar la línea de dirección completa, puede escribir:

Calle + "," + ZIP + "-" + Ciudad

Algunos nombres de campo pueden contener caracteres especiales como espacios, puntos, dos puntos etc., también pueden chocar con una palabra clave reservada de Ninox, como if , then , else . Para solucionar esto, Los nombres de los campos se pueden escribir entre comillas simples (').

Aquí veremos varios ejemplos, ambos son nombres de campo.

'Nombre' + " " + 'Apellido'
'Ingresos totales del año pasado'

Tenga en cuenta que Ninox es independiente de los cambios de nombre de campo.

Es decir, si ha introducido una expresión que incluya un nombre de campo y cambia el nombre de ese campo, después en la tabla, no tendrá volver a modificar la expresión. Esto se debe a que Ninox almacena internamente la expresión con una referencia que identifica al campo (y que no se puede modificar dicha referencia). Por lo tanto, si abre el editor de funciones nuevamente, reflejará el nombre nuevo asignado al campo modificado.