

# Lenguaje de Formato CISIS

MFN function

[mfn mfn\(length\)](#)

Mode command

[mdl, mdu, mhl, mhu, mpl, mpu](#)

Field selector

[v](#)

Literals

["string"](#)

['string'](#)

[|string|](#)

Spacing commands

[x](#)

[c](#)

[/](#)

<#>

[%](#)

Dummy field selectors

[d](#)

[n](#)

Repeatable group

[\(format\)](#)

If command

[if ... then ... else ... fi](#)

Boolean functions

[p\(field selector\)](#)

[a\(field selector\)](#)

Comment specification

[/\\*string\\*/](#)

Select command

[select ... case ... elsecase ... endsel](#)

## Text file functions

[@](#)

[cat\(file\)](#)

## Settings functions

[lw\(number\)](#)

[newline\(string\)](#)

## Repeatable group functions

[iocc](#)

[break](#)

[continue](#)

## Numerical functions

[val\(string\)](#)

[l\(key\) l\(\[inverted file\],key\)](#)

[ravr\(string\)](#)

[rmax\(string\)](#)

[rmin\(string\)](#)

[rsum\(string\)](#)

[size\(string\)](#)

[nocc\(field\)](#)

[npost\(key\) npost\(\[inverted file\],key\)](#)

## String functions

[f\(num expr,length,decimals\)](#)

[ref\(mfn, format\) ref\(\[master file\]mfn, format\)](#)

[s\(expression\)](#)

[instr\(string1,string2\)](#)

[left\(string,length\)](#)

[mid\(string, start, length\)](#)

[right\(string, length\)](#)

[replace\(string1, string2, string3\)](#)

[mstname](#)

[type\(string\)](#)

[date date\(keyword\)](#)

[getenv\(expression\)](#)

## Putenv function

[putenv\(expression\)](#)

Field update function

[proc\(field update format\)](#)

System function

[system\(expression\)](#)

## Estructura del listado de referencia

<format specification>	<name>
<i>Soporte:</i>	Si Standard, significa que el comando/función tiene el mismo uso/resultado ambos en ISIS y CISIS. Si CISIS se especifica, significa que el comando/función sólo está disponible en CISIS. Comandos/funciones que se han mejorado en CISIS, se presenta con anotación de Standard/CISIS. (item siempre presente)
<i>Tipo de función:</i>	Especifica el tipo del valor de retorno de la función. Los posibles valores son: Boolean, String y Numeric. (item sólo aplicable a las funciones)
<i>Sintaxis:</i>	Notación formal del uso del comando/función. (item siempre presente)
<i>Definición:</i>	Exposición del uso del comando/función.
<i>Componentes:</i>	Exposición de características adicionales del comando/función.
<i>Notas:</i>	Aclara particularidades, restricciones y/o diferencias entre ISIS y CISIS.
<i>Exemplos:</i>	1 Provee uno o más ejemplos de uso del comando/función.
<i>Vea también:</i>	Lista comandos y funciones relacionadas.

## Comandos y funciones

#	unconditional newline
<i>Soporte:</i>	Standard
<i>Sintaxis:</i>	#
<i>Definición:</i>	Salta a la línea siguiente incondicionalmente.
<i>Exemplos:</i>	1 #,("address:"v3(9,9)+ ; #), 2 ( Name: v1^n,c20, Surname: v1^s/),###,

*Vea también:* / [conditional newline]  
% [reset blank line]

## % reset blank line

*Soporte:* Standard

*Sintaxis:* %

*Definición:* Cancela las líneas en blanco previas, si las hubiera.

*Exemplos:*

- 1 |Name: |v1^n,c20,|Surname:|v1^s,###,%/,
- 2 v10/#,v20/#,v30/#,%#,

*Vea también:* / [conditional newline]  
# [unconditional newline]

## "string" conditional literal

*Soporte:* Standard

*Sintaxis:* "<text>"<field selector >"<text>"  
"<text>"<dummy field (selector)>  
"<text>"<not present>

*Definición:* Muestra el texto que está entre comillas dobles solo si, <field selector>, <dummy selector> o <not present>, son evaluados como TRUE (verdadero). Los literales, prefijos y sufijos, pueden colocarse junto a un <field selector> de modo que el dato contenido en el campo también se visualizará. Si se asocia a un <dummy selector>, la salida es generada sólo si el campo tiene datos. Si se asocia a <not present>, la salida es generada sólo si el campo no tiene datos.

*Notas:* <text> se produce una sola vez, independientemente de la repetibilidad del campo.

*Exemplos:*

- 1 "Author: "v1^a,
- 2 "this text outputs if data field 10 is present"d10,
- 3 "this text outputs if data field 10 is absent"n5,

*Vea también:* '**string**' [unconditional literal]  
|**string**| [repeatable conditional literal]  
**d** [dummy field selector]  
**n** [not present]  
**v** [field selector]

<b>'string'</b>	<b>unconditional literal</b>
<i>Soporte:</i>	Standard
<i>Sintaxis:</i>	'<texto>'
<i>Definición:</i>	Muestra incondicionalmente el texto contenido entre comillas simples.
<i>Notas:</i>	Los literales incondicionales pueden ubicarse en cualquier lugar del formato, y pueden usarse para pasar parámetros a las funciones.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 this text will always output',</li> <li>2 'Name: ',v1/,</li> </ol>
<i>Vea también:</i>	<b>"string"</b> [conditional literal] <b> string </b> [repeatable conditional literal]

<b>(format)</b>	<b>repeatable group</b>
<i>Soporte:</i>	Standard
<i>Sintaxis:</i>	(<format>)
<i>Definición:</i>	Aplica el formato contenido entre paréntesis a cada una de las ocurrencias cada campo repetible, o una sola vez, en el caso de campos no repetibles.
<i>Notas:</i>	No está permitido el anidamiento de grupos repetibles
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 ( ;  +v2^s/),</li> <li>2 (v1,c15,v2,c35,v3/),</li> <li>3 (if iocc&lt;=3 then f(iocc,1,0)  -  v3/ else '-&gt; more than 3%', fi),</li> </ol>
<i>Vea también:</i>	<b> string </b> [repeatable conditional literal] <b>v</b> [field selector]

<b>/</b>	<b>conditional newline</b>
<i>Soporte:</i>	Standard
<i>Sintaxis:</i>	/
<i>Definición:</i>	Comienza una nueva línea siempre que no se encuentre al comienzo de una.
<i>Notas:</i>	Varios comandos de salto de línea condicional (/,/./,/.) producen el mismo efecto que uno solo.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 v1/,</li> <li>2 v1/,v3/,v10/,mfn/,</li> <li>3 s(v1,v3,v10)/,</li> </ol>

*Vea también:*           # [unconditional newline]  
                          % [reset blank line]

## **/\*string\*/** **comment**

*Soporte:*               CISIS

*Sintaxis:*              /\* <comment> \*/

*Definición:*          Encierra comentarios

*Notas:*                Los comentarios pueden ocupar varias líneas.

*Exemplos:*

- 1    /\* this is a single line comment \*/,
- 2    /\* this comment begins here  
      and ends here \*/,
- 3    if a(v10) /\*and p(v20) \*/ then v20/ fi,

## **@** **include format file**

*Soporte:*               CISIS

*Sintaxis:*              @<filename>

*Definición:*          Inserta en el formato corriente un formato que ha sido almacenado en un archivo externo.

*Notas:*                El <filename> puede incluir el dispositivo y la ruta del directorio en donde está alojado el formato. La sintaxis de los comandos contenidos en el archivo, es evaluada cuando el formato corriente es ejecutado. Es necesario encerrar entre comas el llamado a este comando (,@<filename>.).

*Exemplos:*

- 1    @test.pft,v20,
- 2    s(@c:\temp\test.pft,v3),
- 3    if v1='L' then @large.pft, fi,

## **a(field selector)** **absence check**

*Soporte:*               Standard

*Tipo de función:*      Boolean

*Sintaxis:*              a(<field selector>)

*Definición:*          Regresa el valor TRUE (verdadero) si el campo está ausente, y FALSE (false) en caso contrario.

*Notas:* Pueden usarse todos los componentes del selector de campo, excepto el comando de sangría

*Exemplos:*

- 1 if **a**(v12) then v13 else v12, fi,
- 2 if **a**(v20^b) and p(v30) then v40/, fi,

*Vea también:* **p** function  
**v** [field selector]

## average value of expression

*Vea:* ravr function

## break conditional branching/quitting

*Soporte:* CISIS

*Sintaxis:* **break**

*Definición:* Interrumpe la ejecución del formato de un grupo repetible. Cuando está fuera de un grupo repetible, sale del formato que se está ejecutando.

*Notas:* La ejecución continuará con los comandos posteriores al grupo repetibles. Cuando se utiliza dentro de una *función ref*, la ejecución del formato continú después de la función.

*Exemplos:*

- 1 (if iocc > 10 then '10+ occurrences',**break** else v5^n|-|,v5^s/, fi,)

*Vea también:* (format) [repeatable group]

## c column

*Soporte:* Standard

*Sintaxis:* **c**<int>

*Definición:* Desplazarse a una columna específica en la línea actual o en la línea siguiente.

*Exemplos:*

- 1 'Name: ',c10,v1^n/,
- 2 if p(v1^s) then c10,v1^s/, fi,

*Vea también:* **x** [spacing]

## cat(file) dump file

*Soporte:* CISIS

*Tipo de función:* String

*Sintaxis:* **cat**(<format>)

*Definición:* Muestra el contenido de un archivo cuyo nombre es generado por el <format>.

*Exemplos:*

- 1 mfn,cat('myfile.html'),
- 2 cat('current document'/, ,if v10='c' then 'firstdoc.txt' else 'default.doc' fi),
- 3 cat(v101),

*Vea también:* s(expression)

## comment

*Vea:* /\*string\*/

## conditional literal

*Vea:* "string"

## conditional newline

*Vea:* /

## continue

## repeatable conditional branching

*Soporte:* CISIS

*Sintaxis:* **continue**

*Definición:* Ejecuta la ocurrencia siguiente de un grupo repetible, si, existe al menos un campo con tal ocurrencia.

*Notas:*

*Exemplos:*

- 1 (if iocc = 1 then **continue** else v10/ fi),
- 2 (f(iocc,1,0),'=',v70,continue/),

*Vea también:* (format) [repeatable group]

## current date

*Vea:* date, date function

## d

## dummy field selector

*Soporte:* Standard

*Sintaxis:* **d**<field tag><subfield>

<i>Definición:</i>	Muestra el literal condicional prefijo si el campo o subcampo asociado contiene datos. Se usa asociado a un literal condicional.
<i>Notas:</i>	El selector de campo virtual no retorna un valor. <i>Bugs conocidos:</i> Cuando en un grupo repetible, el <subfield> es evaluado sólo para la primer ocurrencia del campo.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 "this text outputs if data field 10 exists"<b>d10</b>,</li> <li>2 "Name: "v20(5,5)/, ,"Name: "n20,v21(5,5)/,</li> </ol>
<i>Vea también:</i>	<b>"string" [conditional literal]</b> <b>n [not present]</b> <b>v [field selector]</b>

## **date date(keyword) current date**

<i>Soporte:</i>	CISIS
<i>Tipo de función:</i>	String
<i>Sintaxis:</i>	<b>date</b> <b>date(&lt;keyword&gt;)</b>
<i>Definición:</i>	Muestra la fecha actual del sistema. Usado sin parámetros, devuelve: <b>aaaammdd hhmmss d nnn</b> <i>dónde:</i> <ul style="list-style-type: none"> <li>aaaa = año</li> <li>mm = mes</li> <li>dd = día</li> <li>hh = hora</li> <li>mm = minuto</li> <li>ss = segundo</li> <li>d = día de la semana (0-6)</li> <li>nnn = cantidad de días transcurridos desde el 1<sup>ro</sup> de Enero.</li> </ul>
<i>Componentes:</i>	keywords <b>DATETIME</b> and <b>DATEONLY</b>
<i>Notas:</i>	<b>DATETIME</b> muestra la fecha del sistema en formato europeo y la hora actual (dd/mm/aa hh:mm:ss), mientras que <b>DATEONLY</b> muestra lo mismo pero sin la hora.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 'Today is ',<b>date</b>,</li> <li>2 'Current date: ',<b>date(DATEONLY)</b>/, 'Current time: ',mid(<b>date(DATETIME)</b>,10,8)/,</li> </ol>

## **display file**

*Vea:* cat function

## dummy field (selector)

*Vea:* d

## environment variable set

*Vea:* putenv function

## **f(num expr,length,decimals)** **format value**

*Soporte:* Standard

*Tipo de función:* String

*Sintaxis:* **f**(<format>,<expr-1>,<expr-2>)

*Definición:* Convierte un valor numérico en cadena de caracteres. <format> es la expresión numérica a ser convertida. <expr-1> y <expr-2> son opcionales y determinan la longitud mínima de la salida y la cantidad de decimales respectivamente.

*Notas:* Si el <formato> no es una expresión numérica válida, se reporta un error. Si presente <expr-2> también debe estar <expr-1> o se producirá un error de sintaxis. Si sólo se indica la <expr-1>, el resultado se muestra en notación científica exponencial. Si el número de caracteres requeridos para representar al <formato> es mayor que la <expr-1>, se agregan posiciones automáticamente. Si la <expr-1> está ausente, se asume una extensión de 16 caracteres.

*Exemplos:*

- 1 **f**(val(v1),2),
- 2 **f**((((3+5)/2)+1,4,2),
- 3 **f**(v2),

*Vea también:* **val** function

## field selector

*Vea:* v

## field update

*Vea:* proc function

**find string**

*Vea:* instr function

**format value**

*Vea:* f function

**getenv(expression)****get environment variable**

*Soporte:* CISIS

*Tipo de función:* String

*Sintaxis:* **getenv**(<format>)

*Definición:* Regresa el valor de una variable de ambiente.

*Notas:* Si el <formato> no genera el nombre de una variable de ambiente válido, no regresa ningún valor.

*Exemplos:*

- 1 'Current path: ',**getenv**('PATH'),
- 2 (v1|=|,**getenv**(v1)/),

*Vea también:* **putenv** function

**if ... then ... else ... fi****conditional flow control**

*Soporte:* Standard

*Sintaxis:* **if** <bool expr> **then** <format-1> [**else** <format-2> ] **fi**

*Definición:* Ejecuta un bloque de lenguaje de formateo (<formato-1>) si la evaluación de <bool expr> retorna TRUE (verdadero). Ejecuta otro bloque de lenguaje de formateo (<formato-2>) usando la cláusula **else** que se ejecuta cuando la evaluación de <bool expr> retorna FALSE (falso).

*Notas:* La cláusula **then** precede al primer bloque del formato. **else** es opcional y, si está presente, debe estar seguida de un bloque de lenguaje de formateo. La cláusula **fi** siempre termina el comando. Y, si está ausente, se reporta un error de sintáxis. El comando **if ... fi** puede ocupar varias líneas, en cuyo caso es recomendable usar sangría.

*Exemplos:*

- 1 **,ifinstr**(v5,'ab')>0 **then** ,v5/, **fi**,
- 2 **,if** p(v10) **then** ,|Title: |v3, **else** ,|Alternate title: |v4, **,fi**,

**include format file**

*Vea:* @

**instr(string1,string2)****find string**

- Soporte:* CISIS
- Tipo de función:* Numeric
- Sintaxis:* **instr**(<format-1>,<format-2>)
- Definición:* Regresa un número especificando la posición inicial del string generado por <format-2>, en el string generado por <format-1>. Si el string buscado (<format-2>) no se encuentra en <format-1> la función retorna cero.
- Notas:* Tanto <format-1> como <format-1> deben generar cadenas de caracteres (strings), en caso contrario se producirá un error de sintaxis. El uso de *la función s* puede ayudar en los casos donde se requiere un string complejo como parámetro.
- Exemplos:*
- 1 if **instr**(v5,'ab')>0 then v5/, fi,
  - 2 if **instr**(s('/v1/'),v5)>0 then v1, fi,
  - 3 left(v18,**instr**(v18,'.')->1),

**iocc****occurrence index**

- Soporte:* CISIS
- Tipo de función:* Numeric
- Sintaxis:* **iocc**
- Definición:* Regresa el número de orden (índice) de la ocurrencia en proceso (comenzando en 1), o cero en caso contrario.
- Exemplos:*
- 1 ("Author: "v1/, ,if **iocc** > 3 then 'et all',break, fi),
  - 2 (f(**iocc**,3,0),|.v10/),
- Vea también:* **nocc** function

**key lookup**

- Vea:* l function

**key postings**

- Vea:* npost function

**keyword**

*Vea:* v, date / date function

## **I(key) I([inverted file],key) key lookup**

*Soporte:* Standard/CISIS

*Tipo de función:* Numeric

*Sintaxis:* I(<format key>)  
I([<format ifname>]<format key>)

*Definición:* Regresa el **MFN** del primer posting (si lo hubiera) usando la clave generada por el formato <format key> para buscar en el archivo invertido actual. Puede también buscar en otro archivo invertido cuyo nombre se indica mediante el formato <format ifname>.

*Notas:* Las claves se convierten a mayúsculas antes de buscar la expresión. El modo de visualización por defecto es mpl. Si se especifica un modo diferente en la FST, debe tenerse en cuenta en <format key> ya que este genera la clave. Si no se encuentra la clave, la función retorna cero. El parámetro <format ifname> debe generar un string con un nombre de archivo invertido válido, de otra manera ocurrirá un error de sintaxis. Esta función también se usa frecuentemente en conjunción con *la función ref* para permitir la visualización de campos de otro registro.

*Exemplos:*

- 1 if I(v15)<> 0 then |Term: |v15, fi,
- 2 ref(I(['books']v1,'-',v2),v10/),

*Vea también:* **ref** function

## **left(string,length) left substring**

*Soporte:* CISIS

*Tipo de función:* String

*Sintaxis:* **left**(<format-1>,<format-2>)

*Definición:* Retorna un nuevo string, que contiene los n primeros caracteres del string original <format-1>, comenzando desde la izquierda, la cantidad n de caracteres es determinada por <format-2>.

*Notas:* Si el valor obtenido del string generado por <format-2> es mayor que el tamaño del string generado por <format-1>, la función retorna el string <format-1>. Si <format-2> es cero o un número negativo, retorna un string NULO.

*Exemplos:*

- 1 if **left**(v1^n,2)='Ma' then v1^n/, fi,
- 2 **left**(v1,instr(v1,'.')->1),

*Vea también:*           **right** function  
                              **mid** function

## **lw(number)** **set line width**

*Soporte:*                CISIS

*Tipo de función:*      Numeric

*Sintaxis:*              **lw**(<int>)

*Definición:*            Establece que el ancho de la línea de salida es de <int> caracteres.

*Notas:*                 El ancho por defecto de la línea de salida es de 76 caracteres.

*Exemplos:*

- 1    if size(v10) > 76 then **lw**(254), fi,
- 2    **lw**(70),v20/,**lw**(10),v30/,

## **master file name**

*Vea:*                    mstname function

## **maximum value of expression**

*Vea:*                    rmax function

## **mdl, mdu, mhl, mh, mpl, mpu** **mode**

*Soporte:*                Standard

*Sintaxis:*              **m**<mode><conv>

*Definición:*            Establece un nuevo modo de visualización para la salida actual.

*Notas:*                 El modo por defecto es mpl. <mode> representa el modo que se desea establecer. <conv> especifica si se establece la conversión a mayúsculas. MODE puede aparecer varias veces en un formato y su efecto sobre el formateo estará activo hasta que se establezca un nuevo modo.

<mode> puede especificarse de las siguientes maneras:

**p** = *proof*: los campos se visualizan tal como se almacenaron en los registros.

**h** = *heading*: los caracteres de control y los delimitadores de campo son ignorados, excepto los delimitadores de subcampo, que son reemplazados por signos de puntuación.

**d** = *dato*: similar al *modo encabezamiento*, solo que agrega un punto al final del campo, seguido de dos espacios.

*En*

<conv> se pueden establecer las siguientes opciones:

**u**: convierte los datos a mayúscula (*u* de upper case)

**l**: deja los datos en minúscula (*l* de lower case), en realidad como estaban .

*Exemplos:*

- 1 **mpl**, "First author: "v10[1]/,
- 2 **mpu**, "Second author: "v10[2]/,
- 3 **mdl**, "Third author: "v10[3]/,

<b>mfn mfn(length)</b>	<b>record number</b>
------------------------	----------------------

*Soporte:* Standard

*Tipo de función:* String or numeric

*Sintaxis:* **mfn**  
**mfn(<int>)**

*Definición:* Regresa el MFN (Master File Number) de un registro.

*Notas:* Un valor entero puede pasarse como parámetro para establecer la longitud del string que retornará la función MFN. **mfn** regresa un valor de tipo numérico ó cadena dependiendo de los requerimientos del formato.

*Exemplos:*

- 1 'Record: ',**mfn**(3)/,
- 2 if **mfn** > 2 then **mfn**/, fi,
- 3 ref(**mfn**-1,v2/),

*Vea también:*

**ref** function  
**l** function

<b>mid(string, start, length)</b>	<b>substring</b>
-----------------------------------	------------------

*Soporte:* CISIS

*Tipo de función:* String

*Sintaxis:* **mid**(<format-1>,<format-2>,<format-3>)

*Definición:* Regresa un nuevo string, conteniendo un número específico de caracteres del string original (<format-1>). <format-3> especifica la cantidad de caracteres a leer de <format-1> y <format-2> la posición a partir de la cual se extraen los caracteres de <format-1>.

*Notas:* Si <format-2> es mayor que el tamaño de <format-1>, la función regresa un string NULO. Si <format-2> es cero o un número negativo, el valor por defecto es 1.

*Exemplos:*

- 1 **mid**(v2,2,80),
- 2 **mid**(v1,instr(v1,'key'),size(v1))/,

*Vea también:*           **right** function  
                              **left** function

### minimum value of expression

*Vea:*                       rmin function

### mode

*Vea:*                       m...

### mstname

master file name

*Soporte:*                CISIS

*Tipo de función:*       String

*Sintaxis:*               **mstname**

*Definición:*           Regresa el nombre del archivo maestro actual/corriente.

*Exemplos:*

- 1   'Current database: ',**mstname**/,
- 2   ref(['names']|(['names']X39BJ), , 'Database now is ',**mstname**/),

### n

not present (dummy field selector)

*Soporte:*                Standard

*Sintaxis:*               **n**<field tag>

*Definición:*           Comprueba la ausencia de un campo. Se utiliza en conjunción con literales condicionales.

*Notas:*                 Como selector de campo virtual, no regresa ningun valor.

*Exemplos:*

- 1   "this text outputs if data field 10 is absent"**n**10,
- 2   "Author:"v10/, , "Author: "**n**10,v20/,

*Vea también:*         "**string**" [conditional literal]  
                          **d** [dummy field selector]  
                          **v** [field selector]

### newline(string)

set newline

*Soporte:*                CISIS

*Tipo de función:*       String

*Sintaxis:* **newline**(<format>)

*Definición:* Establece y/o restablece el par CR/LF por defecto con el/los caracter(es) generados por <format>.

*Notas:* <format>, puede también contener una secuencia de escape reservada como:

\r - es un salto de párrafo

\n - es un salto de línea

Los saltos de párrafo y línea \ subsecuentes serán reemplazados automáticamente por el string generado por <format> hasta que un nuevo llamado a la función **newline** establezca un nuevo par de strings (o caracteres) para salto de línea y párrafo.

*Exemplos:* 1 **newline**(if v151='unix' then '\n' else '\r\n' fi,

2 **newline**(v301),

3 **newline**('<BR>'),

*Vea también:* / [conditional newline]

# [unconditional newline]

## **nocc(field) number of occurrences**

*Soporte:* CISIS

*Tipo de función:* Numeric

*Sintaxis:* **nocc**(<field selector>)

*Definición:* Regresa el número de ocurrencias de un campo o subcampo. El campo o subcampo se indica mediante <field selector>.

*Notas:* Esta función sólo puede recibir como parámetro un <field selector> con campos o subcampos. Todos los demás componentes de la función <field selector>, produzcan un error de sintaxis si son utilizados.

*Exemplos:* 1 if **nocc**(v3)> 10 then 'Too many occurrences.', fi,

2 'There are ',f(**nocc**(v20),2,0),' authors.',

*Vea también:* **iocc** function

v [field selector]

## **not present**

*Vea:* n

## **npost(key) npost([inverted file],key) key postings**

<i>Soporte:</i>	CISIS
<i>Tipo de función:</i>	Numeric
<i>Sintaxis:</i>	<b>npost</b> (<format key>) <b>npost</b> ([<format>],<format key>)
<i>Definición:</i>	Regresa todos los postings de una clave (la cual es generada por <format key>) en un archivo invertido. Si <format> es indicado, este debe generar un string conteniendo el nombre del archivo invertido a ser utilizado. <format key> genera la clave a buscar en el archivo invertido.
<i>Exemplos:</i>	1 if <b>npost</b> (v1)> 1 then 'duplicate key ',v1,' found'/, fi, 2 'There are ',f( <b>npost</b> (v20),3,0),'keys for ',v20,'.', '/',
<i>Vea también:</i>	<b>I</b> function

### number of occurrences

*Vea:* nocc function

### occurrence index

*Vea:* iocc function

### p(field selector)

### presence check

<i>Soporte:</i>	Standard
<i>Tipo de función:</i>	Boolean
<i>Sintaxis:</i>	<b>p</b> (<field selector>)
<i>Definición:</i>	Retorna TRUE (verdadero) si el campo asociado está presente, retorna FALSE (falso) en caso contrario.
<i>Notas:</i>	Pueden utilizarse todos los componentes de la función field selector ( <b>v</b> ) excepto sangría.
<i>Exemplos:</i>	1 if <b>p</b> (v12) then v12 else v13, fi, 2 if <b>p</b> (v50^a) and <b>p</b> (v50^b) then v50^a/,v50^b/, fi,
<i>Vea también:</i>	<b>a</b> function <b>v</b> [field selector]

### proc(field update format)

### field update

*Soporte:* CISIS

<i>Tipo de función:</i>	String
<i>Sintaxis:</i>	<b>proc</b> (<fldupd format>)
<i>Definición:</i>	Añade o reemplaza campos en el registro actual. <fldupd format> es un formato que genera los comandos de actualización que especifican a la función las tareas a realizar.
<i>Notas:</i>	<p>Una especificación de actualización de campos es un string (cadena de caracteres) compuesto por los comandos <b>d</b> (borrar), <b>a</b> (agregar) y <b>h</b> (agregar) y las modificaciones que estos especifiquen. Las modificación se aplicará al registro corriente. Todos los comandos <b>d</b> (borrar) deben preceder a los comandos agregar (<b>a</b> y <b>h</b>).</p> <p>Especificación de los comandos:</p> <p><b>d*</b> - borra todos los campos del registro</p> <p><b>d</b>&lt;field tag&gt; - borra todas las ocurrencias del campo &lt;field tag&gt;</p> <p><b>d</b>&lt;field tag&gt;/&lt;occ&gt; - borra la ocurrencia &lt;occ&gt; del campo &lt;field tag&gt;</p> <p><b>a</b>&lt;field tag&gt;#&lt;string&gt;# - agrega la cadena &lt;string&gt; como una nueva ocurrencia del campo &lt;field tag&gt;</p> <p><b>h</b>&lt;field tag&gt; &lt;n&gt; &lt;string&gt; - agrega la cadena &lt;string&gt;, de &lt;n&gt; bytes de longitud, como una nueva ocurrencia del campo &lt;field tag&gt;</p> <p>El delimitador # puede ser cualquier caracter no numérico.</p> <p>Debe haber un espacio entre &lt;field tag&gt;, &lt;n&gt; y &lt;string&gt;, parámetros del comando <b>h</b>.</p>
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 <b>proc</b>('d70', a10# v70 #),</li> <li>2 <b>proc</b>(if v24*0.4 = 'Tech' then 'd*', fi),</li> </ol>

<b>putenv(expression)</b>	<b>environment variable set</b>
<i>Soporte:</i>	CISIS
<i>Tipo de función:</i>	String
<i>Sintaxis:</i>	<b>putenv</b> (<format>)
<i>Definición:</i>	Establece una variable de ambiente a el nivel de del sistema operativo con su valor correspondiente.
<i>Notas:</i>	La variable está disponible solo dentro del alcance del proceso actual.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 <b>putenv</b>('TEST=test'),getenv('TEST'),</li> <li>2 set CIPAR=somefile set mx null "pft=<b>putenv</b>('CIPAR=another'),getenv('CIPAR')/" set</li> </ol>
<i>Vea también:</i>	<b>getenv</b> function

<b>ravr(string)</b>	<b>average value of expression</b>
<i>Soporte:</i>	Standard
<i>Tipo de función:</i>	Numeric
<i>Sintaxis:</i>	<b>ravr</b> (<format>)
<i>Definición:</i>	Regresa el valor promedio de un formato dado. <format> debe generar una expresión alfanumérica.
<i>Notas:</i>	Puede usarse para computar el promedio de valores numéricos en campos repetibles.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 f(<b>ravr</b>(s(v8,x1,v1)),3,0),</li> <li>2 f(<b>ravr</b>(v1,x1,v2),5,2),</li> <li>3 f(<b>ravr</b>('8;15;16.73'),3,2),</li> <li>4 if <b>ravr</b>(v20 ;)&gt;=5 then 'pass'/ else 'fail'/, fi,</li> </ol>
<i>Vea también:</i>	<b>rmin</b> function <b>rmax</b> function <b>rsum</b> function

<b>record number</b>	
<i>Vea:</i>	mfn, mfn function

<b>ref(mfn, format) ref([master file]mfn, format)</b>	<b>record reference link</b>
<i>Soporte:</i>	Standard/CISIS
<i>Tipo de función:</i>	String
<i>Sintaxis:</i>	<b>ref</b> (<expr>,<format>) <b>ref</b> ([<format dbname>]<expr>,<format>)
<i>Definición:</i>	Ejecuta <format> en el registro seleccionado mediante <expr>. Si se indica <format dbname> se puede referenciar a otra (o la misma) base de datos.
<i>Notas:</i>	<expr> puede ser cualquier formato que regrese el MFN de un registro. La función <b>I</b> puede usarse para ejecutar una búsqueda y retornar el MFN del primer registro encontrado.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 <b>ref</b>(I(v3),v1/,v2/,v3/),</li> <li>2 if <b>ref</b>(['account']I(['user']v2),v4)='active' then  Name:  v10/, fi,</li> <li>3 (if p(v99) then <b>ref</b>([v99]1,v30/), fi),</li> </ol>

*Vea también:* **I function**

## literal condicional para grupos repetibles

*Vea:* |string|

## Grupos repetibles

*Vea:* (format)

## replace(string1, string2, string3)

**replace**

*Soporte:* CISIS

*Tipo de función:* String

*Sintaxis:* **replace**(<format-1>,<format-2>, <format3>)

*Definición:* Devuelve una nueva cadena de caracteres (string), reemplazando <format-2> con <format-3> en <format-1>

*Notas:* Si <format-2> es una cadena de caracteres (string) nula o no se encuentra en <format-1>, la función devuelve la cadena <format-1>.

Si <format-3> es nula, la cadena <format-2> será excluida de <format-1>.

**Replace** es una opción sensible para ambas cadenas de caracteres: la cadena a buscar (<format-2>) y la cadena de reemplazo (<format-3>).

*Exemplos:*

- 1 **replace**('Mary And John','And','and')/,
- 2 if **replace**(v1^a,'01x','01X')= '894501X' then v1^n/, fi,
- 3 **replace**(s(v304,v333),',', ' '),/,
- 4 **replace**(s(if v415='spanish' then v299 else 'none' fi),v1,v759)/,

## reset blank line

*Vea:* %

## right(string, length)

**right substring**

*Soporte:* CISIS

*Tipo de función:* String

*Sintaxis:* **right**(<format-1>,<format-2>)

<i>Definición:</i>	Retorna un nuevo string, que contiene los n últimos caracteres del string original <format-1>, comenzando desde la derecha; la cantidad n de caracteres es determinada por <format-2>.
<i>Notas:</i>	Si el valor de <format-2> es mayor que la longitud de <format-1>, la función devuelve la cadena de caracteres <format-1>. Si <format-2> es igual a cero o contiene un número negativo, no devuelve nada.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 if <b>right</b>(v1^n,1) = 'r' then v1^n/, fi,</li> <li>2 right(v65,4)/,</li> </ol>

<b>rmax(string)</b>	<b>maximum value of expression</b>
<i>Soporte:</i>	Standard
<i>Tipo de función:</i>	Numeric
<i>Sintaxis:</i>	<b>rmax</b> (<format>)
<i>Definición:</i>	Devuelve el valor máximo de un formato dado. <format> debe generar una cadena de caracteres (string).
<i>Notas:</i>	Puede ser utilizado para calcular el máximo entre los valores numéricos de un campo repetible.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 f(<b>rmax</b>('72,54,2'),2,0),</li> <li>2 f(<b>rmax</b>(v1,x1,v4,x1,(v8 , )),5,2),</li> <li>3 if <b>rmax</b>(v40 ; )&gt;val(v41) then 'Limit of ',v41,'exceeded.',/, fi,</li> </ol>
<i>Vea también:</i>	<b>rmin</b> function <b>ravr</b> function <b>rsum</b> function

<b>rmin(string)</b>	<b>minimum value of expression</b>
<i>Soporte:</i>	Standard
<i>Tipo de función:</i>	Numeric
<i>Sintaxis:</i>	<b>rmin</b> (<format>)
<i>Definición:</i>	Devuelve el valor mínimo del formato especificado. <format> debe generar una cadena de caracteres (string).
<i>Notas:</i>	De forma similar a la <i>función rmax</i> , <i>rmin</i> puede calcular el mínimo de valores numéricos en un campo repetible.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 f(<b>rmin</b>('10;2;5;4;-2'),2,0),</li> <li>2 f(<b>rmin</b>(v1,x1,v2,x1,'44'),4,2),</li> </ol>

3 if **rmin**(v80||,v90| |,v100| |) < 1990 then 'Wrong decade.', fi,

*Vea también:*

**rmax** function  
**ravr** function  
**rsum** function

## **rsum(string)** **sum of expression**

*Soporte:* Standard

*Tipo de función:* Numeric

*Sintaxis:* **rsum**(<format>)

*Definición:* Devuelve la suma de un formato dado. <format> debe generar una cadena de caracteres (string).

*Notas:* En forma similar a las *funciones rmax y rmin*, *rsum* calcula la suma de valores numéricos en un campo repetible.

*Exemplos:*

1 f(**rsum**('102,45,-37'),2,0),  
 2 f(**rsum**(v1,x1,v3,x1,f(val(v8)+2)),4,2),  
 3 if **rsum**(v20^d)>1000 then 'Aborted.'/ else 'OK'/, fi,

*Vea también:*

**rmax** function  
**ravr** function  
**rmin** function

## **s(expression)** **string**

*Soporte:* Standard

*Tipo de función:* String

*Sintaxis:* **s**(<format>)[*command component*]

*Definición:* Devuelve la concatenación de cadenas de caracteres (string) generadas por <format>.

*Componentes:* extraction

*extraction:* Extracts partial content of the resulting string. <offset int> is the first position to start extraction, while <length int> determines how many characters will be extracted. If <length int> is omitted or is greater than the resulting string, the default is the end of the resulting string.

*Notas:* Puede ser utilizadas por funciones que requieran una cadena de caracteres como parámetro.

*Exemplos:*

1 if **s**(v1,v2,v3):'ABCDE' then **s**(v1,v2,v3)\*0.50, fi,

2 if s(|\*|v5|\*|):s('\*E\*')then 'English'/, fi,

*Vea también:* v [field selector]

## select ... case ... elsecase ... endsel

## conditional branch control

*Soporte:* CISIS

*Sintaxis:* **select** <format expr>  
**case** <option-1>: <format-1>  
**case** <option-2>: <format-2>  
**case** <option-n>: <format-n>  
[**elsecase** <format-0>]  
**endsel**

*Definición:* Se evalúa <format expr> y se compara el resultado con cada opción **case** (<option-1>, <option-2>...<option-n>). Si una opción coincide con <format expr>, se ejecuta el bloque de instrucciones asociado (<format-1>, <format-2>...<format-n>); si ninguna opción es igual a <format expr> se ejecuta la cláusula **elsecase** (<format-0>), si fué definida.

*Notas:* <format expr> debe generar una cadena de caracteres (string) o un valor numérico. Si <format expr> genera una cadena, todos los valores de las opciones de las cláusulas **case** deben ser de tipo cadena de caracteres, de lo contrario. Si <format expr> genera un valor numérico, los valores de la opción también deben ser numéricos.

*Exemplos:*

- 1 **select** s(v5)  
**case** '1': ,f(val(v5)/2,2,2)/,  
**case** '2': ,v5/  
**case** '3': ,v6,'-',v1/  
**elsecase** ,|Error in field v5 = |v5/  
**endsel**,
- 2 **select** nocc(v7)  
**case** 0: 'absent'/,  
**case** 1: 'one occurrence'/,  
**case** 2: 'two occurrences'/,  
**elsecase** 'more than 2 occurrences'/,  
**endsel**,

*Vea también:* **if ... then ... else ... fi**

## set line width

*Vea:* lw function

**size(string)****string size**

<i>Soporte:</i>	CISIS
<i>Tipo de función:</i>	Numeric
<i>Sintaxis:</i>	<b>size</b> (<format>)
<i>Definición:</i>	Devuelve el tamaño de una cadena de caracteres (string).
<i>Notas:</i>	<format> debe devolver una cadena de caracteres (string) de lo contrario se producirá error de sintaxis.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 if <b>size</b>(v10)&gt; 76 then lw(254), fi,</li> <li>2 f(<b>size</b>(v10,v20),1,0),</li> </ol>

**spacing**

*Vea:* x

**string**

*Vea:* s function

**string size**

*Vea:* size function

**string to value**

*Vea:* val function

**string type**

*Vea:* type function

**substring**

*Vea:* mid function

**sum of expression**

*Vea:* rsum function

**system(expression)****system call**

<i>Soporte:</i>	CISIS
<i>Tipo de función:</i>	String
<i>Sintaxis:</i>	<b>system</b> (<format>)
<i>Definición:</i>	Ejecuta el argumento producido por <format> como un comando del sistema operativo.
<i>Notas:</i>	<format> debe generar una cadena de caracteres (string) que contenga el comando a ser ejecutado. La eventual salida generada por este comando será direccionada a la salida estandar del sistema.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 <b>system</b>('dir'),</li> <li>2 if p(v2) then <b>system</b>('type ',v2), fi,</li> </ol>

**type(string)****string type**

<i>Soporte:</i>	CISIS
<i>Tipo de función:</i>	String
<i>Sintaxis:</i>	<b>type</b> (<format>)
<i>Definición:</i>	Devuelve el tipo de una cadena de caracteres de la siguiente manera: A - si la cadena contiene solamente caracteres alfabéticos (conforme a una tabla de caracteres alfabéticos por defecto, como ISISAC.TAB) o espacios N - si la cadena contiene únicamente caracteres numéricos (0-9) X - para cualquier otro caso.
<i>Notas:</i>	<format> debe generar una cadena de caracteres (string) o un mensaje de que ha ocurrido un error de sintaxis.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 if <b>type</b>(v1)='N' then f(val(v1),3,2)/ else v1/, fi,</li> <li>2 if s(<b>type</b>(v1),<b>type</b>(v2),<b>type</b>(v3))&lt;&gt;'AAA' then 'Invalid character type detected'/, fi,</li> </ol>

**unconditional literal**

*Vea:* 'string'

**unconditional newline**

*Vea:* #

**v****field selector**

<i>Soporte:</i>	Standard
<i>Sintaxis:</i>	$v\langle\text{field tag}\rangle[\textit{command components}]$
<i>Definición:</i>	Contenido de los campos de salida de datos. El contenido puede ser seleccionado, restringido, extraído o sangrado mediante componentes del mismo comando (ver más abajo). $v$ significa <i>campo de longitud variable</i> .
<i>Componentes:</i>	<i>subfield, occurrence, extraction and indent</i>
<i>syntactic order:</i>	$\wedge\langle\text{subfield id}\rangle [\langle\text{index}\rangle[..\langle\text{upper index}\rangle]] * \langle\text{offset int}\rangle.\langle\text{length int}\rangle (\langle\text{first line int}\rangle,\langle\text{next line int}\rangle)$
<i>subfield:</i>	Restricts the output to the contents of a subfield. If data field exists but subfield is not present, no output is generated.
<i>occurrence:</i>	Narrows the output to one or a range of occurrences of a repeatable field. $\langle\text{index}\rangle$ and $\langle\text{upper index}\rangle$ refer to the first (or unique) and last occurrences, respectively. If the specified $\langle\text{index}\rangle$ is greater than the actual number of occurrences, no output is generated. The same occurs if data field is not repeatable and $\langle\text{index}\rangle$ is set to a number equal or greater than 2. However, if $\langle\text{index}\rangle$ is set to 1 and it is used in a non-repeatable field, content is normally output. This component must be used outside a repeatable group; otherwise, $\langle\text{upper index}\rangle$ is ignored. If <i>double dot</i> (..) is used and $\langle\text{upper index}\rangle$ is missing <b>LAST</b> is assumed. The <b>LAST</b> keyword is set with the value of total occurrences of a data field.
<i>extraction:</i>	Extracts partial content of a data field, <i>subfield</i> or <i>occurrence</i> . $\langle\text{offset int}\rangle$ is the first position to start extraction, while $\langle\text{length int}\rangle$ determines how many characters will be extracted. If $\langle\text{length int}\rangle$ is omitted or is greater than field length, the default is the end of data field.
<i>indent:</i>	Aligns the output of data field, <i>subfield</i> , <i>occurrence</i> or extracted content, according to $\langle\text{first line int}\rangle$ (alignment for the first line) and $\langle\text{next line int}\rangle$ (alignment for successive lines). Both values are numeric constants. If current line position differs from zero, indentation is disabled.
<i>Notas:</i>	El funcionamiento del comando $v$ depende de los componentes utilizados. No se generará salida alguna cuando el campo de datos esté ausente o cuando el componente ejecuta una restricción o una extracción que esté fuera de los límites.
<i>Exemplos:</i>	<ol style="list-style-type: none"> <li>1 <math>v2/,v3^a  -  ,v1/,</math></li> <li>2 <math>v1^n*0.3,</math></li> <li>3 <math>( ;  +v3^s)/,</math></li> <li>4 <math>v20[4],</math></li> <li>5 <math>v10[2..7]/,</math></li> </ol>

- 6 **v5[3..]/,\* equals to ,v5[3..LAST], \*/**
- 7 **v1[LAST]\*2.7/,**
- 8 **v1(5,5)/,**
- 9 **|Title: |v1^n(5,5)/,**

*Vea también:*

**"string"** [conditional literal]  
**d** [dummy field selector]  
**n** [not present]  
**|string|** [repeatable conditional literal]  
**(format)** [repeatable group]

## **val(string)** **string to value**

*Soporte:* Standard

*Tipo de función:* Numeric

*Sintaxis:* **val(<format>)**

*Definición:* Devuelve el valor numérico del argumento generado por <format>.

*Notas:* Si <format> produce solamente caracteres no numéricos, la función devuelve cero. Si encuentra más de un valor numérico, sólo el primero de ellos es devuelto.

*Exemplos:*

- 1 if **val(v2)>5** then 'Error'/ else 'OK'/, fi,
- 2 f(**val(v2)/3,4,2**),

## **x** **spacing**

*Soporte:* Standard

*Sintaxis:* **x<int>**

*Definición:* Inserta una cantidad <int> de espacios.

*Notas:* Si <int> es más grande que el espacio disponible en la línea en curso, pasa a la línea siguiente.

*Exemplos:*

- 1 'Name: ',**x5**,v1^n/,
- 2 (v1,**x3**,v2,**x8**,v3/),

*Vea también:* **c** [column]

## **|string|** **repeatable conditional literal**

*Soporte:* Standard

*Sintaxis:* |<text>|<+><field selector><+>|<text>|

*Definición:* Da salida al texto colocado entre las barras verticales para cada ocurrencia de un campo repetible, solamente si el selector de campo se considera TRUE (verdadero). Combinado con un campo repetible, el funcionamiento del comando puede ser ampliado mediante el uso del operador <+>. Cuando <+> está presente, el primer literal-prefijo y/o el último literal-sufijo no se visualizan.

**Ejm.:**

(|; |+v1), /\* literal-prefijo \*/

(v1+|; |), /\* literal-sufijo \*/

*Notas:* Los literales prefijo y sufijo pueden usarse conjuntamente, incluso con el operador <+>. Si un literal prefijo o sufijo es utilizado con <+> fuera de un grupo repetible, los contenidos del literal pueden no ser visualizados como se desea. Si el campo no es repetible, la visualización del literal ocurre para la primera y única ocurrencia del campo de datos.

*Exemplos:* 1 (|; |+v1^s,|,|v1^n\*0.1|,|),

2 (v10|: |, ,v11,| - |v12),

*Vea también:* **"string"** [conditional literal]  
**'string'** [unconditional literal]  
**(format)** [repeatable group]  
**v** [field selector]