From a number to text

In TI processes:

### NUMBERTOSTRING

* Arguments:
  + A real value
* Input: the decimal separator for the current user locale!
* Output: similar to the 'general' number format: no thousands separators and - for negative numbers

### NUMBERTOSTRINGEX

* Arguments:
  + A real value
  + A format string
  + Decimal separator
  + Thousands separator

In TI processes and rules:

### STR

* Arguments:
  + A real value
  + The string length
  + The number of decimals
* Input: you must use . (period) as the decimal separator and , (comma) as the thousands separator
* Output: if necessary, the function inserts leading blank spaces to attain this length. Hence, use TRIM

From text to a number

In TI processes:

### STRINGTONUMBER

* Arguments:
  + A string
* Input: the decimal separator for the current user locale!

### STRINGTONUMBEREX

* Arguments:
  + A string
  + Decimal separator
  + Thousands separator

In TI processes and rules:

### NUMBR

* Arguments:
  + A string
* Input: you must use . (period) as the decimal separator and , (comma) as the thousands separator
* Output: All characters other than '0' through '9', '+', '-', ',', '.' and 'E' are ignored

Examples

Adding 1 to a period (string) as in 01, 02, 03, …, 12 (Rules):

IF(NUMBR(!Rp\_Period)<9, '0', '') | TRIM(STR(NUMBR(!Rp\_Period)+1, 2, 0))

Adding a Text attribute is by far the better option here.

Loop through a dimension with 24 forecast scenarios (spanning 2 years), Fct\_01 until Fct\_24:

f = 1;

WHILE( f <= 24 );

vFctScenario = 'Fct\_' | NUMBERTOSTRINGEX( f, '00');

# other code

f = f + 1;

END;

Reworking negative values with parentheses to a minus sign (Turbo Integrator):

vAmount = '(123,45)';

vAmount = Trim(vAmount);

IF(SCAN('(', vAmount) = 1 & SCAN(')', vAmount) = LONG(vAmount) );

vAmount = '-' | TRIM(SUBST(vAmount, 2, LONG(vAmount) - 2));

ENDIF;

IF(SCAN('(', vAmount) > 0 % SCAN(')', vAmount) > 0 );

ITEMREJECT('Unknown data format for ' | vAmount);

ENDIF;

vAmount\_N = STRINGTONUMBER(vAmount);

IF(vAmount\_N <> 0);

# ...

ENDIF;