Reduce Subroutine

Description

Creates a cursor using a filter or applies a filter to an existing cursor.

Syntax

Reduce *(reduce_script, sort_list, mode, table_name, cursorvar, flag)*

Parameters

The Reduce subroutine has the following parameters.

Parameter	Description

reduce_script	Used to pass an expression containing the formatted reduction criteria. The information passed in reduce_script must be formatted as follows. · All column names must be enclosed in braces { }. · Column names must not have spaces in them. · All constants must be on the right side of the comparison operator and must be enclosed in either single or double quotes. · All comparison operators must be reduced to the primitive set of operators and must be separated by spaces.

Operator	Primitive
Between	Between
Equal to	EQ
Less than	LT
Greater than	GT
Less than or equal to	LE
Greater than or equal to	GE
Not	Not
Starting with	]
Ending with	[
Containing	[ ]
Matches	Match
Range	From To
Sunday	Sun
Monday	Mon
Tuesday	Tue
Wednesday	Wed
Thursday	Thu
Friday	Fri
Saturday	Sat
Today	Today
Yesterday	Yst
Tomorrow	Tom
Next	Next
Last	Last

Example Assignments for reduce_script:
```
reduce_script = "WITH NOT {DATE} EQ '9-23-59'"
reduce_script = "WITH {CITY} EQ 'Boston' OR WITH {STATE} EQ 'CA'"
reduce_script = 'WITH {GRADE_POINT} FROM "2.0" TO "4.0"'
reduce_script = 'WITH {MAJOR} EQ ':  Quote(major_var)
reduce_script := ' AND WITH {MINOR} EQ ' :  Quote(minor_var)
reduce_script = 'WITH {INV_DATE} LT ' :  Quote(Date() - 30)
reduce_script := ' OR WITH {invoice_amt} GE "1000"'
reduce_script = 'WITH {PHONE} Match "3N-4N"'
```

Sort_list	Pass the sort criteria in sort_list. The sort criteria requires the same format as when used in Select ... By. If the program intends to sort, both the call to Reduce and the Select statement must use the same sort_list. Example assignments for sort_list:

```
sort_list = "CITY"
sort_list = "ST" :  @FM :  "city"
sort_list = "#INV_TOTAL"
sort_list = "ST" :  @FM :  "#INV_TOTAL"
```

Mode	Used to pass one of three values:
	Value / **Description** table below
table_name	Used to pass the name of the table to be searched.
cursorvar	Used to pass a cursor already initialized (Modes 0 and 2) or a cursor to be initialized (Mode 1).
flag	Returns true if the subroutine was successful and false if unsuccessful.

Value	Description
0	Indicates that these are new reduction criteria for an existing cursor.
1	Indicates that the subroutine should return the results of the reduction in the next available cursor.
2	Indicates that these are additional criteria for an existing cursor. The new criteria will be joined with a logical And to any previous criteria.

See also

Select...By

Example

```
Function Reduce_Example(param1)
* The following code provides reduction criteria to a selection of keys
* from CUSTOMERS using Select...By.

Declare Subroutine Reduce, Msg, FsMsg
$Insert Logical

Equ new_exist$ To 0 ; * Reduce Mode 0
Equ next_cur$ To 1
Equ add_exist$ to 2

table_name = "CUSTOMERS"

flag = ""
done = False$
CursorVar = ""

* Clears all cursors
For counter = 0 To 8
  ClearSelect counter
Next counter

sort_list = "STATE"
Reduce_Script = "WITH {STATE} EQ 'NY' OR WITH {STATE} EQ 'CA'"
mode = NEXT_CUR$

Reduce(reduce_script, sort_list, mode, table_name, Cursorvar, flag)
If flag then
  Select table_name By sort_list Using Cursorvar then
    Open table_name To file_var then
      ctr = 0

      Loop
        ReadNext key Using Cursorvar By AT Else done = TRUE$
      Until done
        Read rec From file_var, key then
          Gosub Processing
        end
      Repeat
    End Else
      FsMsg()
    End
  End Else
    FsMsg()
  end
End Else
  FsMsg()
End
Return ctr

Processing:
  ctr += 1
return
```