Create_Symbolic Routine

Description

Creates or redefines calculated (symbolic) columns in a native table dictionary.

Syntax

Create_Symbolic *(tablename, columnname, formula, datatype, columnheading, multivalueflag, conversion, justification, length, description)*

Parameters

The Create_Symbolic routine has the following parameters.

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>tablename</td><td>Name of table in which the calculated column is to appear.</td></tr>
<tr><td>columnname</td><td>Specifies the name of the symbolic column to create or redefine. Column names must comply with rules for key values.</td></tr>
<tr><td>Formula</td><td>The formula to execute when columnname is referenced. The formula is written using any valid BASIC+ expressions. The symbolic formula does not conform to rules defined for creating stored procedures.
Note: Multiline formulae should be delimited by either @vm or @fm.</td></tr>
<tr><td>datatype</td><td>The data type assigned to the column. If datatype is provided, the information associated with the data type, conversion, justification, and length, are automatically provided. If null, conversion, justification, and length must be provided in the appropriate arguments.</td></tr>
<tr><td>columnheading</td><td>The heading for the symbolic column. If null, the default is columnname.</td></tr>
<tr><td>multivalueflag</td><td>Specifies whether this column is single valued or multi-valued. If null, the default is single valued (S). Specify one of the codes listed below.
<table><tr><th>Value</th><th>Description</th></tr><tr><td>S</td><td>Single valued.</td></tr><tr><td>M</td><td>Multi-valued.</td></tr></table></td></tr>
<tr><td>conversion</td><td>The output format for the return data. If conversion and datatype are null, then no conversion is performed. If datatype is provided, the default is dependent on the data type assigned. See also, Oconv function.</td></tr>
<tr><td>justification</td><td>The justification for the return data. If justification and datatype are null, the default is left (L) justified. If datatype is specified, justification defaults to the justification for the data type. Specify one of the values below.
<table><tr><th>Value</th><th>Description</th></tr><tr><td>L</td><td>Left justified.</td></tr><tr><td>R</td><td>Right justified.</td></tr><tr><td>T</td><td>Left justified, text mark delimiters inserted at length intervals.</td></tr></table></td></tr>
<tr><td>length</td><td>The length for the return data. If length and datatype are null, length defaults to a length of 20. If datatype is specified, length defaults to the length for the data type.</td></tr>
<tr><td>description</td><td>Text describing the symbolic column.</td></tr>
</table>

See also

List_Dict

Example

```
* This code example creates a symbolic field named "TAX" in the ORDERS table within the EXAMPLES application
Declare Subroutine Create_Symbolic, Set_Status, FsMsg
Declare Function Get_Status
table = "ORDERS"
column = "TAX"
formula = "@ans = ''" : @vm : "@ans = {SUB_TOTAL} * .07" @fm
dType = "DOLLARS"
colHead = "Tax"
mvFlag = ""
conv = ""
just = "R"
length = 15
descr = "Calculate tax on Orders"
errCodes = ""
Set_Status(0)
Create_Symbolic(table, column, formula, dType, colHead, mvFlag, conv, just, length, descr)
If Get_Status(errCodes) Then
* Error Handling
FSMsg(errCodes)
end
```