

StructCmp

%StructCmp function

| Function | The function compares the defined parts of local structures. | | | | | | | | |
|--|--|--|--|---|--|--------------------|-------------------------|-------------------|---|
| Declaration | <pre>BOOL %StructCmp(in _recA[fromRowA]^fromColA, IN _recA[toRowA]^toColA, in _recB[fromRowB]^fromColB, INT in _cmpOptions [,INT in _roundPrec])</pre> | | | | | | | | |
| Parameters | <table><tr><td>_recA [fromRowA] ^fromColA, _recA[toRowA] ^toColA</td><td>Reference to the value of local structured variable field.</td></tr><tr><td>_recB [fromRowB] ^fromColB</td><td>Reference to the value of local structured variable field.</td></tr><tr><td>_cmpOptions</td><td>Way of comparison.</td></tr><tr><td>_roundPrec</td><td>Optional parameter. It specifies how many decimal places the compared values (they must be of the <i>Real</i> type) are rounded to. The value must be >= 0.</td></tr></table> | _recA [fromRowA] ^fromColA, _recA[toRowA] ^toColA | Reference to the value of local structured variable field. | _recB [fromRowB] ^fromColB | Reference to the value of local structured variable field. | _cmpOptions | Way of comparison. | _roundPrec | Optional parameter. It specifies how many decimal places the compared values (they must be of the <i>Real</i> type) are rounded to. The value must be >= 0. |
| _recA [fromRowA] ^fromColA, _recA[toRowA] ^toColA | Reference to the value of local structured variable field. | | | | | | | | |
| _recB [fromRowB] ^fromColB | Reference to the value of local structured variable field. | | | | | | | | |
| _cmpOptions | Way of comparison. | | | | | | | | |
| _roundPrec | Optional parameter. It specifies how many decimal places the compared values (they must be of the <i>Real</i> type) are rounded to. The value must be >= 0. | | | | | | | | |
| Description | <p>The value of parameter _cmpOptions determines which part of value will be compared as follows:</p> <table><tr><th>Value</th><th>Compare the part of value</th></tr><tr><td>1</td><td>value</td></tr><tr><td>2</td><td>time of value formation</td></tr><tr><td>4</td><td>users flags</td></tr></table> <p>The value of parameter _cmpOptions can be obtained by addition of the above mentioned values (for example, if user wants to compare only the formation time of a value and user flags - the value of parameter is 2+4).</p> <p>The comparison is done as follows:</p> <p>A=B</p> <p>where:</p> <p>Symbol A represents an area of local structured variable values which is given by its left upper corner (field _recA[fromRowA]^fromColA) and right lower corner (field _recA[toRowA]^toColA). The structured variable which is used in the first and second parameter must be the same. The dimensions (number of rows and columns) are also defined by this way. They are used in definition of B area that is defined only by its right upper corner in the parameters.</p> <p>The comparison is done among the individual appropriate fields of A and B areas. The function gets back the value @TRUE if all fields, that are compared, accomplish a condition of equality.</p> | Value | Compare the part of value | 1 | value | 2 | time of value formation | 4 | users flags |
| Value | Compare the part of value | | | | | | | | |
| 1 | value | | | | | | | | |
| 2 | time of value formation | | | | | | | | |
| 4 | users flags | | | | | | | | |
| Note | <p>Two invalid values are considered to be equal during comparison. Valid and invalid values are considered to be different values.</p> <p>A flag <i>value validity</i> does not influence the comparison of users flags or time of value formation.</p> | | | | | | | | |
| Example | <p>In the following example there is shown the comparison of two matrixes.</p> <p>The first one is delimited by fields _A[1]^_firstColNr (where _firstColNr has value =1) and _A[2]^_lastColNr (where _lastColNr is set on consecutive number of the last column of relevant structure).</p> <p>The second one is defined by the field _B[1]^_firstColNr and consists of two rows and of the same number of columns like matrix in variable _A.</p> | | | | | | | | |

```

BEGIN
  BOOL _bEqual
  RECORD (SD.BIG) _A
  RECORD (SD.BIG) _B
  REDIM _A[20]
  REDIM _B[20]

  INT _firstColNr, _lastColNr

  _firstColNr := 1
  _lastColNr := _A\CNR
  _A[1]^R1 := 1
  _A[2]^R1 := 2
  DELAY 1[s]
  _B[1]^R1 := 1
  _B[2]^R1 := 2
  _bEqual := %StructCmp(_A[1]^_firstColNr,_A[2]^_lastColNr,_B[1]
^_firstColNr,1)
  ;result is @TRUE

  _bEqual := %StructCmp(_A[1]^_firstColNr,_A[2]^_lastColNr,_B[1]
^_firstColNr,2)
  ;result is @FALSE, because the time of values are different

END

```



Related pages:

[ESL extended functions](#)
[Function arguments - types](#)