

# Replace Function

## Description

Replaces a field, a value, or a subvalue, in a dynamic array.

## Syntax

**Replace** (*expression, field, value, subvalue, new*)

## Parameters

The Replace function has the following parameters.

Parameter	Description
<i>expression</i>	Designates the dynamic array that is to be searched. The expression is not modified by the Replace function.
<i>field, value, subvalue</i>	These values specify the location within the array to replace. Their respective numeric values determine whether the new data will replace a field, a value, or a subvalue. Note these examples: F = Replace(A,2,0,0,NEW) This example replaces a field. When both the value and subvalue are 0 (zero), the new data replaces the second field (specified by the field argument) of dynamic array A. The variable F is assigned the new array. V = Replace(A,2,3,0,NEW) This example replaces a value. When only the subvalue is 0 (zero), the new data replaces the third value of the second field (specified by the field and value arguments) of dynamic array A. The variable V is assigned the new array. S = Replace(A,3,2,1,NEW) This example replaces a subvalue. When all three delimiter expressions have a non-zero value, the new data replaces the first subvalue of the second value of the third field (specified by the field, value, and subvalue arguments) of dynamic array A. S is assigned the new array. Field is the highest level delimiter, while subvalue is the lowest level delimiter. If a higher level delimiter has a 0 (zero) value while a lower level delimiter has a non-zero value, the 0 (zero) delimiter is assumed to be 1 (one). In the following example: Replace(A,0,0,2,B) is assumed to be: Replace(A,1,1,2,B) If the second, third, or fourth expression has a -1 (minus one) value, the new data is appended after the specified field, value, or subvalue.
<i>New</i>	new specifies the new data that is to replace the existing contents.

Replace is identical to using < >.

## See also

<> ([Angle Brackets operator](#)), [Delete\(\)](#), [FieldStore\(\)](#), [Insert\(\)](#)

## Example

```
/* The program builds an array (field mark-delimited) of names. The program searches the name array, and
replaces the old name with the new name. */
* multi-valued field with 4 names
Names = "Alan": @FM : "Brad": @FM : "Hal": @FM : "Mike"
NAME_TO_CHANGE = "Hal"
New_name = "Darcie"
Locate NAME_TO_CHANGE In Names Using @FM Setting Pos Then
  NEW_NAMES = Replace(Names, Pos, 0, 0, NEW_NAME)
End
```