

SRP_Join_Arrays

Logically combines two arrays.

Syntax

```
ToArray = SRP_Join_Arrays(LeftArray, RightArray, Delim, Operation)
```

Returns

The logical combination of both arrays.

Parameters

Parameter	Description
LeftArray	The first dynamic array to be merged. Required.
RightArray	The second dynamic array to be merged. Required.
Delim	The delimiter that separates elements in both arrays. Optional. If you omit this parameter, @FM will be assumed. This is also the delimiter used to create the new array.
Operation	Determines how the arrays are merged. Options are AND, OR, NOT, and XOR. Optional. If you omit this parameter, OR will be assumed.
CaseSensitive	Determines if the merging is case sensitive. Optional. If you omit this parameter, 1 will be assumed.

Remarks

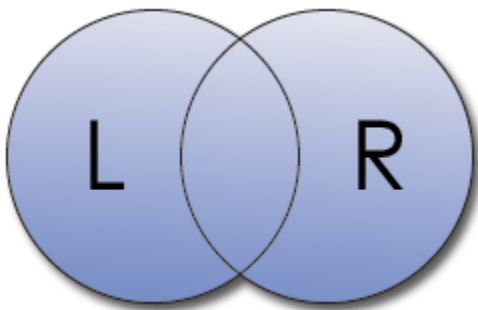
It's not unusual to have at your disposal two arrays of unique values, especially arrays of keys. SRP_Join_Arrays takes two arrays and returns either the union or intersection of those arrays. For example, imagine an array of Employee keys that references all managers and another array that references all female employees. SRP_Join_Arrays can quickly produce the intersection of these two arrays, giving you an array of keys for all female managers.

To be specific, SRP_Join_Arrays can join two arrays using for logical operations: AND, OR, NOT, and XOR. Each operation is described in detail below.

IMPORTANT: *Keep in mind that no matter which operation you choose, SRP_Join_Arrays only returns unique values.*

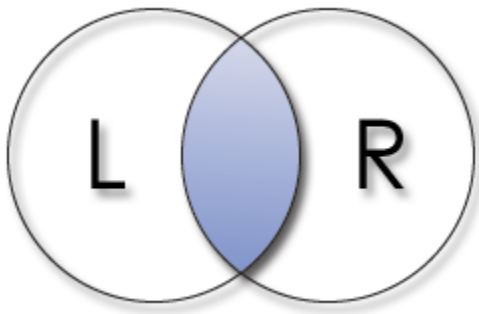
OR

The logical OR operation tells SRP_Join_Arrays to return all items that appear in either LeftArray or RightArray. This creates the union of two lists. It's effectively the same as append the two arrays, with the exception that SRP_Join_Arrays still only returns unique values. Here is the Venn diagram for the OR operations:



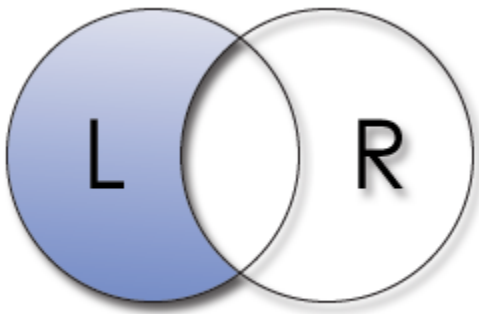
AND

The logical AND operation tells SRP_Join_Arrays you want all items that appear in both LeftArray and RightArray. This is what we call the intersection of the two lists as demonstrated by the following Venn diagram:



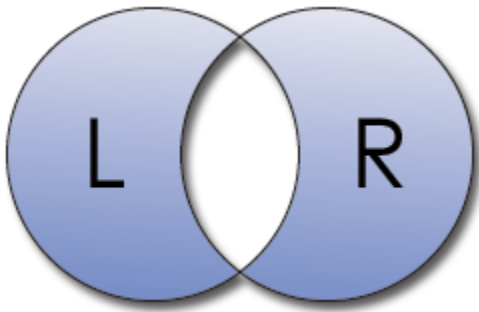
NOT

The logical NOT operation tells SRP_Join_Arrays you want all items that appear in LeftArray but not in RightArray. In other words, think of RightArray as an exclusion list. Here's the Venn diagram for the NOT operation:



XOR

The logical XOR operation is shorthand for Exclusive OR. It tells SRP_Join_Arrays to return all values that appear in LeftArray or RightArray, but not both. The following Venn diagram illustrates how this is the inverse of the AND operation:



Note: SRP_Join_Arrays is case sensitive. That means that "John" and "JOHN" are considered different values. Since SRP_Join_Arrays was designed for arrays of keys, we felt this approach was appropriate.

The Delim parameter is the delimiter for both arrays. It is also the delimiter used to build the resulting array. By default, this parameter is set to @FM.

Deprecation Note: SRP_Join_Arrays used to perform only two operations: Intersection (AND) and Union (OR). The operation was indicated by passing a boolean 1 or 0 to the operation parameter. For backwards compatibility, passing 1 will perform the AND operation and passing 0 will perform the OR operation.

Examples

```

// Make two big arrays, the first is every two numbers, the second is every three numbers
LeftArray = ""
RightArray = ""
For iRow = 1 to 10000
    LeftArray := iRow * 2:@FM
    RightArray := iRow * 3:@FM
Next iRow
LeftArray[-1, 1] = ""
RightArray[-1, 1] = ""

// Get the intersection, only those numbers that appear in both arrays, without duplicates
ArrayIntersect = SRP_Join_Arrays(LeftArray, RightArray, @FM, "AND")

// Remove all items in the left array that also appear in the right array
ArrayNot = SRP_Join_Arrays(LeftArray, RightArray, @FM, "NOT")

// Get an array of those numbers do not appear in both lists
ArrayXor = SRP_Join_Arrays(LeftArray, RightArray, @FM, "XOR")

// Get the union, all numbers from both lists, without duplicates
ArrayUnion = SRP_Join_Arrays(LeftArray, RightArray, @FM, "OR")

// We can also get the union of @FM delimited arrays by omitting parameters
ArrayUnion = SRP_Join_Arrays(LeftArray, RightArray)

```