

# Sum Function

## Description

In a dynamic array, adds numeric data at the lowest delimiter level to numeric data at the next higher level.

## Syntax

*result* = **Sum** (*expression*)

## Parameters

The Sum function has the following parameters.

Parameter	Description
<i>Expression</i>	Designates the dynamic array in which the numeric data is to be summed. All elements must contain numeric data. The computer searches the specified array to find its lowest level delimiter. The numeric data of that lowest level is then added to the numeric value of the next higher level, and that sum is stored there.

## Remarks

In the following example:

```
array = 2:@SVM:1:@VM:2:@VM
result = Sum(array)
```

subvalue 1 (one) is the lowest level, and its value would be added to and stored in the next value. The result would be "3ý2ý".

Each single Sum function manipulates data only to one level higher in the existing string each time. Therefore, the Sum function needs to be executed repeatedly to raise a multi-level array to a single numeric value. If the preceding example is modified by the addition of another Sum operation:

```
array = 2:@SVM:1:@VM:2:@VM
result = Sum(Sum(array))
```

the result is "5"

Sum() recognizes seven levels of delimiters: ASCII character 248 through ASCII character 254.

## See also

[Delete\(\)](#), [Extract](#), [Field\(\)](#), [FieldStore\(\)](#), [Insert\(\)](#), [Locate](#), [Locate...By](#), [Remove](#), [Replace\(\)](#)

## Example

```
/* Y is assigned 5^7^6, where "^" represents a field mark (ASCII 254). */
X = 5:@FM:4:@VM:3:@FM:6
Y = Sum(X)
* Z is assigned 18.
Z = Sum(Sum(Y))
```