Array Formulas | WeWeb Documentation

|  |  |
| --- | --- |
| URL source | <https://docs.weweb.io/formulas/array.html> |
| Date de scraping | 2025-04-08T15:54:40.398218+00:00 |

## Array formulas ​

These formulas are very useful to manipulate arrays , like changing their content, filtering them, etc.

## 💡 Good to know 💡 ​

## add ​

This formula will add one or multiple values at the end of an existing array. It's the JavaScript equivalent of push .

## Example ​

In this example, we have a variable countries which equals to ["France", "USA"] , so an array of strings containing the countries where WeWeb employees live. But we're missing some countries! To solve this, we'll use the add formula to add two new countries.

### Images

<https://docs.weweb.io/assets/array-add1.B_g-gqSH.png>

## contains ​

This formula will check if a value exist in a given array. It always returns a boolean , meaning it will either return true or false .

## Example ​

Let's get back to our countries array variable, which now equals ["France", "USA", "Belgium", "Croatia"] . If we use the contains formula with this array and "France" , it will return true . With "Canada" , it returns false , as Canada isn't in the list of countries.

### Images

<https://docs.weweb.io/assets/array-contains1.D7QAdnvv.png>

<https://docs.weweb.io/assets/array-contains2.Dhq_v65T.png>

## createArray ​

The createArray formula is useful to create a whole array from scratch, without any code. It's the equivalent of writing the whole array in JavaScript.

## Example ​

Let's say we want to create the countries array (containing the places where WeWeb employees live, meaning ["France", "USA", "Belgium", "Croatia"] ) in a workflow, without needing to type any JavaScript. To do this, we would write this formula:

### Images

<https://docs.weweb.io/assets/array-createarray1.DLGGgNV-.png>

## compare ​

This formula will compare that two arrays are equals. Meaning they both have the same values and that their values are in the same order (same index for each). This formula always returns a boolean , which means true or false .

## Example ​

If we take our countries array which contain ["France", "USA", "Belgium", "Croatia"] , compare will return true or false in these use-cases:  
same values, same order -> true  
  
same values, different order -> false  
  
different values -> false

### Images

<https://docs.weweb.io/assets/array-compare1.DqjA5Z0g.png>

<https://docs.weweb.io/assets/array-compare2.D7KdggcN.png>

<https://docs.weweb.io/assets/array-compare3.zt2Gu8_r.png>

## distinct ​

distinct iterates over the values in an array, and will return a new array with only unique values. Meaning that if a value is present twice or more in the array, it will be returned only once.

## Example ​

Let's say we have an array with WeWeb employees names, but we wrote one of them twice by mistake. By using the distinct formula, we're able to correct this mistake.

### Images

<https://docs.weweb.io/assets/array-distinct.DEeTt_rB.png>

## filterByKey ​

This formula will iterate over an array of objects, and return an array where only the object's with a certain key have a certain value.  
It seems hard to understand like that, I know... But check the example and it'll become super clear 😉

## Example ​

Let's say we have an array of users with objects containing their names and role in the web app:  
We want to filter this array to only have users who are admins. We can do this easily with the filterByKey formula:  
  
Learn more about the filterByKey formula in this video tutorial .

### Code

Langage: unknown

[  
 {"name": "Quentin", "role": "admin"},  
 {"name": "Joyce", "role": "normal"},  
 {"name": "Aurélie", "role": "admin"},  
 {"name": "Kévin", "role": "normal"}  
]

### Images

<https://docs.weweb.io/assets/array-filterbykey.chhXykAU.png>

## findIndex ​

findIndex , when given and array and a value, will find the index in the array of the first element which equals the given value. If the value isn't found, it will return -1 . It's equivalent to JavaScript's indexOf .  
🔥 Pro tip 🔥

## Example ​

If we take the countries array, which includes ["France", "USA", "Belgium", "Croatia"] , it will return 2 when we use findIndex for "Belgium" , -1 for "Germany" as it's not in the array.

### Images

<https://docs.weweb.io/assets/array-findindex1.BqbyZPML.png>

<https://docs.weweb.io/assets/array-findindex2.B8WflzgX.png>

## findIndexByKey ​

This formula is equivalent to the findIndex formula but for arrays of objects. It'll return the index of the first object in the array whose key is equal to value.

## Example ​

Let's get back to our array of users with objects containing their names and role in the web app:  
If we try to find the first object whose "role" equals to "normal" , we'll get the index 1 (the second object). We'll get the index 2 to match on "name" equals "Aurélie" and -1 when matching on "John" as this name doesn't exist in the array.

### Code

Langage: unknown

[  
 {"name": "Quentin", "role": "admin"},  
 {"name": "Joyce", "role": "normal"},  
 {"name": "Aurélie", "role": "admin"},  
 {"name": "Kévin", "role": "normal"}  
]

### Images

<https://docs.weweb.io/assets/array-findindexbykey1.CGTFr-b2.png>

<https://docs.weweb.io/assets/array-findindexbykey2.ZDjWWMww.png>

<https://docs.weweb.io/assets/array-findindexbykey3.CDnbtqSv.png>

## getByIndex ​

This formula will return an element in a array at a specific index.

## Example ​

Let's get back to our array of users with objects containing their names and role in the web app:  
If we use getByIndex on the second index, we'll get back Aurélie's object.

### Code

Langage: unknown

[  
 {"name": "Quentin", "role": "admin"},  
 {"name": "Joyce", "role": "normal"},  
 {"name": "Aurélie", "role": "admin"},  
 {"name": "Kévin", "role": "normal"}  
]

### Images

<https://docs.weweb.io/assets/array-getbyindex.DIoesFwd.png>

## groupBy ​

groupBy , when given an array of objects, will return a new array where objects are grouped by their value for a given key.

## Example ​

Again, we go back to our array of users with objects containing their names and role in the web app:  
Let's say we want to regroup them by role type (admins together and normal users together). We can use the groupBy formula applied on the "role" key for this.

### Code

Langage: unknown

[  
 {"name": "Quentin", "role": "admin"},  
 {"name": "Joyce", "role": "normal"},  
 {"name": "Aurélie", "role": "admin"},  
 {"name": "Kévin", "role": "normal"}  
]

### Images

<https://docs.weweb.io/assets/array-groupby.CeS7eFhN.gif>

## join ​

This formula will join all the elements of an array into a string, separating them by a separator (optional, , by default). It's equivalent to JavaScript's join .

## Example ​

Say we want to write a string telling people where WeWeb employees live, using the countries array which equals to ["France", "USA", "Belgium", "Croatia"] . We can easily do this with string concatenation and join .

### Images

<https://docs.weweb.io/assets/array-join.Df1zF7Cx.png>

## length ​

It will return the length of an array, meaning the number of items inside it. Same as JavaScript's length .

## Example ​

Using it again on countries which is ["France", "USA", "Belgium", "Croatia"] , length will return 4 as there's four countries in this array.

### Images

<https://docs.weweb.io/assets/array-length.DWP2vzx_.png>

## lookup ​

Given an array of objects, this formula will return the first object where the key equals to a value. The key is optional, and equals "id" by default. It's equivalent of using getByIndex and findByIndex together.  
💡 Good to know 💡

## Example ​

Let's say, that in our users array, we want to find back the one for Aurélie. We can do this using the lookup formula.  
As a reminder, here the array:  
  
Also, small example with getByIndex used with findByIndex .

### Code

Langage: unknown

[  
 {"name": "Quentin", "role": "admin"},  
 {"name": "Joyce", "role": "normal"},  
 {"name": "Aurélie", "role": "admin"},  
 {"name": "Kévin", "role": "normal"}  
]

### Images

<https://docs.weweb.io/assets/array-lookup1.BRmBiyJ3.png>

<https://docs.weweb.io/assets/array-lookup2.DhR0ZsHY.png>

## lookupArray ​

Same as lookup , but will match an array of values and return all the objects where the value is in the array of values for a given key.  
🔥 Pro tip 🔥

## Example ​

Let's say we improve our users array with a new member, Raphael, who's a "superAdmin" :  
If we wanted to get only users that are either admins or super admins, we would use the lookupArray formula.

### Code

Langage: unknown

[  
 { "name": "Quentin", "role": "admin" },  
 { "name": "Joyce", "role": "normal" },  
 { "name": "Aurélie", "role": "admin" },  
 { "name": "Kévin", "role": "normal" },  
 { "name": "Raphael", "role": "superAdmin" }  
]

### Images

<https://docs.weweb.io/assets/array-lookuparray.CfOWi4Vs.gif>

## map ​

Given an array of objects, this formula will return an array with only the selected key(s) from each object.

## Example ​

Again, we take our users array of objects:  
If we wanted to get an array of the users' names only, we would use the map formula like so:

### Code

Langage: unknown

[  
 { "name": "Quentin", "role": "admin" },  
 { "name": "Joyce", "role": "normal" },  
 { "name": "Aurélie", "role": "admin" },  
 { "name": "Kévin", "role": "normal" },  
 { "name": "Raphael", "role": "superAdmin" }  
]

### Images

<https://docs.weweb.io/assets/array-map.B4YKwz0_.png>

## merge ​

Given two or more arrays, the merge formula will return a single array containing all the former arrays.

## Example ​

Let's say we have two arrays with these values:  
The result of merging those two arrays will give a bigger array containing all the countries:

### Code

Langage: unknown

// countries array  
[  
 "France",  
 "USA",  
 "Belgium",  
 "Croatia"  
]  
  
// countries2 array  
[  
 "Germany",  
 "India",  
 "Japan"  
]

### Images

<https://docs.weweb.io/assets/array-merge.BtlcpO5j.png>

## prepend ​

The prepend formula adds one or more values at the beginning of a given array.

## Example ​

Let's say we have a countries array with the following values:  
By prepending "Germany" to it, we'll get this result:

### Code

Langage: unknown

// countries array  
[  
 "France",  
 "USA",  
 "Belgium",  
 "Croatia"  
]

### Images

<https://docs.weweb.io/assets/array-prepend.CS7MI3-q.png>

## remove ​

This formula will remove a given value from an array.

## Example ​

Still with our countries array as an example:  
Here's what happens when removing "USA" from it:

### Code

Langage: unknown

// countries array  
[  
 "France",  
 "USA",  
 "Belgium",  
 "Croatia"  
]

### Images

<https://docs.weweb.io/assets/array-remove.NaDJ3aNF.png>

## removeByIndex ​

removeByIndex does the same thing as the remove formula , except it does it by removing the element at a certain index in an array, rather than a value.

## Example ​

Same example and return as the example for remove , but with the index:

### Images

<https://docs.weweb.io/assets/array-removebyindex.DZ5lbWxr.png>

## removeByKey ​

Given an array of objects, this formula will remove all the objects from the array where the given key is equal to a given value.

## Example ​

Here's an array called users with the following value:  
If we were to remove all the admin users, we would do it like so:

### Code

Langage: unknown

[  
 { "name": "Quentin", "role": "admin" },  
 { "name": "Joyce", "role": "normal" },  
 { "name": "Aurélie", "role": "admin" },  
 { "name": "Kévin", "role": "normal" },  
 { "name": "Raphael", "role": "superAdmin" }  
]

### Images

<https://docs.weweb.io/assets/array-removebykey.DuYQ-umi.png>

## reverse ​

Given an array, this formula will reverse its values order.

## Example ​

Given back our countries array:  
Reversing it will return this value:

### Code

Langage: unknown

[  
 "France",  
 "USA",  
 "Belgium",  
 "Croatia"  
]

### Images

<https://docs.weweb.io/assets/array-reverse.BZKsZR5s.png>

## rollup ​

Given an array of objects, this formula will return all the values for a given key. If the distinct parameter is set to true , it'll return unique values only ( false by default).

## Example ​

Here's an array called users which equals to:  
If we were to get the possible roles, we would use the rollup formula like this:

### Code

Langage: unknown

[  
 { "name": "Quentin", "role": "admin" },  
 { "name": "Joyce", "role": "normal" },  
 { "name": "Aurélie", "role": "admin" },  
 { "name": "Kévin", "role": "normal" },  
 { "name": "Raphael", "role": "superAdmin" }  
]

### Images

<https://docs.weweb.io/assets/array-rollup.CL9gx8EK.png>

## slice ​

This formula will return the values from an array from the startIndex up to the endIndex ( endIndex not included). It's the exact same function as the JavaScript's slice .

## Example ​

Given a countries array:  
Slicing it to get back only ["USA", "Belgium"] would it be possible by using the slice formula:

### Code

Langage: unknown

[  
 "France",  
 "USA",  
 "Belgium",  
 "Croatia"  
]

### Images

<https://docs.weweb.io/assets/array-slice.CMenaNMu.png>

## sort ​

Given an array, this formula will sort it in ascending ( "asc" ) or descending ( "desc" ) mode. It can be applied on a given key for arrays of objects.

## Example ​

Taking back our array called users which equals to:  
To sort it by user's names in ascending order (alphabetically), we would use the sort formula this way:

### Code

Langage: unknown

[  
 { "name": "Quentin", "role": "admin" },  
 { "name": "Joyce", "role": "normal" },  
 { "name": "Aurélie", "role": "admin" },  
 { "name": "Kévin", "role": "normal" },  
 { "name": "Raphael", "role": "superAdmin" }  
]

### Images

<https://docs.weweb.io/assets/array-sort.C64TYZ6v.png>