

JS Transformations: How Do I Use the transformedObject Variable?

Last Modified on 02/20/2019 7:40 am EST

When doing a JavaScript transformation the `transformedObject` always represents the final state of the object when you are POSTing or GETting.

IMPORTANT: Do not define in your object definition or map in your transformation any fields that you are setting with custom JS.

For example, let's say I am starting with this originalObject:

```
{
  "Email": "test@cloud-elements.com",
  "phone": [
    {
      "number": "5555551234"
    },
    {
      "number": "5558492927"
    }
  ],
  "FirstName": "hello",
  "LastName": "world"
}
```

My mapping looks like:

Common Resources Name and URL

myContacts Salesforce for Docs (Salesforce Sales Cloud) /Lead

Sort Fields By: alphabetical

Organization Level Fields

Field Name	Type	Display Name	+

Account Level Fields

Field Name	Type	Display Name	+
<input type="button" value="v"/> <input type="button" value="^"/> <input type="button" value="trash"/> Email	string		Email
<input type="button" value="v"/> <input type="button" value="^"/> <input type="button" value="trash"/> FirstName	string		FirstName
<input type="button" value="v"/> <input type="button" value="^"/> <input type="button" value="trash"/> ID	string		ConvertedAccountId
<input type="button" value="v"/> <input type="button" value="^"/> <input type="button" value="trash"/> LastName	string		LastName

Instance Level Fields

Field Name	Type	Display Name	+

Now, in my transformation, I want to take the first phone number in the array and use it as the Phone field. Your JS should look something like this:

```
if (!fromVendor) {
  if (originalObject.phone.length > 0) {
    transformedObject.Phone = originalObject.phone[0].number
  }
}
done(transformedObject);
```

Then let's say I want to go the other way. When I call GET /MyContacts I want the phone numbers in an array.

Your JS should look like this:

```
if (fromVendor) {
  if (originalObject.Phone) {
    transformedObject.phone = [];
    transformedObject.phone.push({number: originalObject.Phone});
  }
}
done(transformedObject);
```

Putting it all together to account for both directions:

```
if (!fromVendor) {
  if (originalObject.phone.length > 0) {
    transformedObject.Phone = originalObject.phone[0].number
  }
} else {
  if (originalObject.Phone) {
    transformedObject.phone = [];
    transformedObject.phone.push({number: originalObject.Phone});
  }
}
done(transformedObject);
```
