# JSON Arrays as Lists

## This example template shows how to use either JSON string or object arrays to create lists. The example demonstrates how to sort, reverse, and filter arrays to make a list tailored to your specific use case. In the Simple List examples, we leverage HTML markup to add the line breaks we need to form a list without having to add a repeating section. The next set of examples uses an array of JSON objects as the input for the repeating section. We then use JSONata expressions to filter and sort the array, forming different lists from the same data.

## Examples:

See: [array-as-list.json](https://github.com/JoelGeraci/document-generation-cookbook/blob/main/mergeData/array-as-list.json) for for field reference

### Simple List

{{ $join(dogs, "<br>") }}

### Sorted Simple List

{{ $join($sort(dogs)), "<br>") }}

### Reverse Sorted Simple List

{{ $join($reverse($sort(dogs))), "<br>") }}

**Full Pet List ({{ $count(pets) }} total pets)**

{% repeating-section pets %}

{{name}}

{% end-section %}

**Filtered Full Pet List (Dogs)**

{% repeating-section pets[type='dog'] %}

{{name}}

{% end-section %}

**Sorted Filtered Full Pet List (Cats)**

{% repeating-section $sort(pets[type='cat'], function($l, $r) { $l.name > $r.name }) %}

{{name}}

{% end-section %}

**List of Pet Types in JSON**

{{$join($distinct(pets.type), '<br>') }}