------------------------
TEST 1: PHP
------------------------

Bearing in mind that PHP 5.3 introduced anonymous functions and the Closure class, please implement a function called array_build().
The function must return an associative array, where keys and values are given as input, and eventually filtered & handled by a closure. But you are free to organise the quantity of input/output.

The logic is: given any inputs, always return an associative array, so that it may be used in a SELECT html tag and relative OPTIONs
One possible method signature may be:

```
/*
* @param array $associative
* @param Closure $filter
* @return array
*/
```

Tips:
- where possible, avoid to assign temporary variables (i.e.: $sum = $a + $b; return $sum; may just be return $a + $b;)
- write as few lines of code as possible: think refactoring after you made your first version
- use proper indentation


------------------------
TEST 2: MYSQL
------------------------

Bearing in mind we deal with Hoteliers and Hospitality sector, please implement a set of database tables, best if already in Laravel 5 standard naming convention, where the following relations appear:

- a booking may have more than one reservation (ex: a wholesaler make a booking, composed of 3 reservations: one single, one double, and one triple room)
- each reservation may have more than one invoice (ex: the room is paid by the wholesaler, the extra services by the guest)
- each invoice must be in relation with the entity supposed to pay it: this could be a Guest, a Company or a Trader

You are free to add the relevant fields and/or extra fields you may require.


------------------------
TEST 3: JQUERY
------------------------

Logic: assign any words to the proper grammar entity container, by choosing the words and selecting the destination container

Abstract:

- "Words" is a set of checkboxes, labelled such as: slow, fast, cat, dog, think, write
- "Grammar entities" is a set of 3 paragraphs, such as:
  &lt;div id="adjective"&gt;&lt;/div&gt;
  &lt;div id="noun"&gt;&lt;/div&gt;
  &lt;div id="verb"&gt;&lt;/div&gt;

Having in the page a set of checkboxes, a select, and 3 paragraphs implement the following:

- the options of the select are the names of the 3 paragraphs
- the select is dynamically made, according to the paragraphs present in the html
- to assign the words to the container, it is needed to click on the relevant checkboxes and choose the container through the select
- upon submitting the select (via an event or a button), the chosen checkboxes must then appear under the proper grammar entity (i.e.: NOUN -> cat dog

Feel free to organise the html and the relevant jquery logic of the page