1. `array()` - Create an array.
2. `count()` - Count the number of elements in an array.
3. `sizeof()` - Alias of `count()`, also used to count elements in an array.
4. `array_push()` - Add one or more elements to the end of an array.
5. `array_pop()` - Remove and return the last element from an array.
6. `array_shift()` - Remove and return the first element from an array.
7. `array_unshift()` - Prepend one or more elements to the beginning of an array.
8. `array_merge()` - Merge two or more arrays into a single array.
9. `array_slice()` - Extract a slice of an array.
10. `array_splice()` - Remove a portion of the array and replace it with something else.
11. `array_reverse()` - Return an array with elements in reverse order.
12. `array_search()` - Search an array for a given value and return the corresponding key if found.
13. `array_keys()` - Return all the keys of an array.
14. `array_values()` - Return all the values of an array.
15. `array_unique()` - Remove duplicate values from an array.
16. `array_filter()` - Filter elements of an array using a callback function.
17. `array_map()` - Apply a callback function to each element of an array and return the modified array.
18. `array_reduce()` - Apply a callback function to each element of an array, reducing it to a single value.
19. `array_walk()` - Apply a user-defined function to each element of an array.
20. `array_intersect()` - Return the values that are present in multiple arrays.
21. `array_diff()` - Compare arrays and return the differences.
22. `sort()` - Sort an array in ascending order.
23. `rsort()` - Sort an array in descending order.
24. `asort()` - Sort an associative array in ascending order, according to the values.
25. `arsort()` - Sort an associative array in descending order, according to the values.
26. `ksort()` - Sort an associative array in ascending order, according to the keys.
27. `krsort()` - Sort an associative array in descending order, according to the keys.
28. `array_rand()` - Pick one or more random keys from an array.
29. `shuffle()` - Randomly shuffle the elements of an array.
30. `in_array()` - Check if a value exists in an array.
31. `array_chunk()` - Split an array into chunks of specified size.
32. `array_column()` - Return the values from a single column in a multidimensional array.
33. `array_combine()` - Create an array by using one array for keys and another for its values.
34. `array_diff_assoc()` - Compare arrays with additional index check and

return the differences.
35. `array_fill()` - Fill an array with values, specifying a range.
36. `array_flip()` - Exchanges all keys with their associated values in an array.
37. `array_key_exists()` - Checks if a specified key exists in an array.
38. `array_map()` - Apply a callback function to the elements of one or more arrays.
39. `array_merge_recursive()` - Merge two or more arrays recursively.
40. `array_pad()` - Pad an array to a specified length with a specified value.
41. `array_product()` - Calculate the product of values in an array.
42. `array_push()` - Add one or more elements to the end of an array.
43. `array_rand()` - Pick one or more random keys from an array.
44. `array_reduce()` - Reduce an array to a single value using a callback function.
45. `array_replace()` - Replaces the values of the first array with the values from the second array.
46. `array_reverse()` - Return an array with elements in reverse order.
47. `array_search()` - Search an array for a value and return the corresponding key if found.
48. `array_shift()` - Shift an element off the beginning of an array.
49. `array_sum()` - Calculate the sum of values in an array.
50. `array_unique()` - Remove duplicate values from an array.
51. `array_values()` - Return all the values of an array.
52. `array_walk_recursive()` - Apply a user-defined function recursively to every element in an array.
53. `array_key_first()` - Get the first key of an array.
54. `array_key_last()` - Get the last key of an array.
55. `array_intersect_assoc()` - Compute the intersection of arrays with additional index check.
56. `array_replace_recursive()` - Recursively replaces the values of the first array with the values from the following arrays.
57. `array_reverse()` - Return an array with elements in reverse order.
58. `array_slice()` - Extract a slice of an array.
59. `array_intersect_key()` - Computes the intersection of arrays using keys for comparison.
60. `array_diff_key()` - Computes the difference of arrays using keys for comparison.
61. `array_walk()` - Apply a user-defined function to each element of an array.
62. `array_walk_recursive()` - Apply a user-defined function recursively to every element in an array.
63. `array_uintersect()` - Computes the intersection of arrays, comparing the values using a user-defined callback function.
64. `array_udiff()` - Computes the difference of arrays, comparing the values using a user-defined callback function.
65. `array_uintersect_assoc()` - Computes the intersection of arrays with additional index check, comparing the values using a user-defined callback

function.

66. `array_udiff_assoc()` - Computes the difference of arrays with additional index check, comparing the values using a user-defined callback function.
67. `array_uintersect_uassoc()` - Computes the intersection of arrays with additional index check, compares the values and the keys separately using user-defined callback functions.
68. `array_udiff_uassoc()` - Computes the difference of arrays with additional index check, compares the values and the keys separately using user-defined callback functions.
69. `array_multisort()` - Sort multiple arrays or multi-dimensional arrays.
70. `array_count_values()` - Counts the number of occurrences of each value in an array.
71. `array_key_exists()` - Checks if a specified key exists in an array.
72. `array_fill_keys()` - Fill an array with values, specifying keys.
73. `array_chunk()` - Split an array into chunks of specified size.
74. `array_unique()` - Remove duplicate values from an array.
75. `array_push()` - Add one or more elements to the end of an array.
76. `array_pop()` - Remove and return the last element from an array.
77. `array_shift()` - Remove and return the first element from an array.
78. `array_unshift()` - Prepend one or more elements to the beginning of an array.
79. `array_key_first()` - Get the first key of an array.
80. `array_key_last()` - Get the last key of an array. .