Working With **PHP Arrays** in the Right Way

The Basics

array_values()
array_keys()
array_combine()
array_flip()

```php
$keys = ['sky', 'grass', 'orange'];
$values = ['blue', 'green', 'orange'];

$array = array_combine($keys, $values);
print_r($array);

// Array
// (
//     [sky] => blue
//     [grass] => green
//     [orange] => orange
// )
```

```php
print_r(array_keys($array)); // ['sky', 'grass', 'orange']
print_r(array_values($array)); // ['blue', 'green', 'orange']
print_r(array_flip($array));

// Array
// (
//     [blue] => sky
//     [green] => grass
//     [orange] => orange
// )
```

Make Your Code Shorter

list()
extract()
compact()

```php
// define array
$array = ['a', 'b', 'c'];

// without list()
$a = $array[0];
$b = $array[1];
$c = $array[2];

// with list()
list($a, $b, $c) = $array;
```

```php
$string = 'hello|wild|world';
list($hello, , $world) = explode('|', $string);
echo("$hello, $world"); // hello, world
```

```php
$arrays = [[1, 2], [3, 4], [5, 6]];

foreach ($arrays as list($a, $b)) {
    $c = $a + $b;
    echo($c . ', '); // 3, 7, 11,
}
```

```php
$array = [
    'clothes' => 't-shirt',
    'size'    => 'medium',
    'color'   => 'blue',
];

extract($array);

echo("$clothes $size $color"); // t-shirt medium blue
```

```php
$clothes = 't-shirt';
$size = 'medium';
$color = 'blue';

$array = compact('clothes', 'size', 'color');
print_r($array);

// Array
// (
//     [clothes] => t-shirt
//     [size] => medium
//     [color] => blue
// )
```

Filtering functions

array_filter()
array_unique()
array_column()

```php
$numbers = [20, -3, 50, -99, 55];

$positive = array_filter($numbers, function($number) {
    return $number > 0;
});

print_r($positive); // [0 => 20, 2 => 50, 4 => 55]
```

```php
$numbers = [-1, 0, 1];

$not_empty = array_filter($numbers);

print_r($not_empty); // [0 => -1, 2 => 1]
```

```php
$array = [1, 1, 1, 1, 2, 2, 2, 3, 4, 5, 5];
$uniques = array_unique($array);

print_r($uniques);

// Array
// (
//     [0] => 1
//     [4] => 2
//     [7] => 3
//     [8] => 4
//     [9] => 5
// )
```

```php
$array = [
    ['id' => 1, 'title' => 'tree'],
    ['id' => 2, 'title' => 'sun'],
    ['id' => 3, 'title' => 'cloud'],
];

$ids = array_column($array, 'id');

print_r($ids); // [1, 2, 3]
```

```php
$cinemas = Cinema::find()->all();
$cinema_ids = array_column($cinemas, 'id'); // php7 forever!
```

Walking Throught the Arrays

array_map()
array_walk()

```php
$cities = ['Berlin', 'KYIV', 'Amsterdam', 'Riga'];
$aliases = array_map('strtolower', $cities);

print_r($aliases); // ['berlin', 'kyiv, 'warsaw', 'riga']

$numbers = [1, -2, 3, -4, 5];
$squares = array_map(function($number) {
    return $number ** 2;
}, $numbers);

print_r($squares);  // [1, 4, 9, 16, 25]
```

```php
$model = ['id' => 7, 'name'=>'James'];

$callback = function($key, $value) {
    return "$key is $value";
};

$res = array_map($callback, array_keys($model), $model);
print_r($res);

// Array
// (
//     [0] => id is 7
//     [1] => name is James
// )
```

```php
$fruits = [
    'banana' => 'yellow',
    'apple' => 'green',
    'orange' => 'orange',
];

array_walk($fruits, function(&$value, $key) {
    $value = "$key is $value";
});

print_r($fruits);

// Array
// (
//     [banana] => banana is yellow
//     [apple] => apple is green
//     [orange] => orange is orange
// )
```

Joining Arrays

array_merge()
array_diff()
array_intersect()

```php
$array1 = ['a' => 'a', 'b' => 'b', 'c' => 'c'];
$array2 = ['a' => 'A', 'b' => 'B', 'D' => 'D'];

$merge = array_merge($array1, $array2);
print_r($merge);
// Array
// (
//     [a] => A
//     [b] => B
//     [c] => c
//     [D] => D
// )
```

```php
$array1 = [1, 2, 3, 4];
$array2 =       [3, 4, 5, 6];

$diff = array_diff($array1, $array2);
print_r($diff); // [0 => 1, 1 => 2]

$intersect = array_intersect($array1, $array2);
print_r($intersect);  // [2 => 3, 3 => 4]
```

Do the Math With Array Values

array_sum()
array_product()
array_reduce()
array_count_values()

```php
$numbers = [1, 2, 3, 4, 5];

echo(array_sum($numbers)); // 15
echo(array_product($numbers)); // 120

echo(array_reduce($numbers, function($carry, $item) {
    return $carry ? $carry / $item : 1;
})); // 0.0083 = 1/2/3/4/5
```

```php
$things = ['apple', 'apple', 'banana', 'tree', 'tree', 'tree'];
$values = array_count_values($things);

print_r($values);

// Array
// (
//     [apple] => 2
//     [banana] => 1
//     [tree] => 3
// )
```

Generating Arrays

array_fill()
range()
array_slice()

```php
$bind = array_fill(0, 5, '?');
print_r($bind); // ['?', '?', '?', '?', '?']
```

```php
$letters = range('a', 'z');
print_r($letters); // ['a', 'b', ..., 'z']

$hours = range(0, 23);
print_r($hours); // [0, 1, 2, ..., 23]
```

```php
$numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];
$top = array_slice($numbers, 0, 3);
print_r($top); // [1, 2, 3]
```

Sorting Arrays

sort()
and friends

	a	k	r	u
a	asort		arsort	uasort
k		ksort	krsort	
r	arsort	krsort	rsort	
u	uasort			usort

Combining Array Functions

Like a Boss

```php
$values = ['say  ', '  bye', ' ', ' to', ' spaces ', '    '];

$words = array_filter(array_map('trim', $values));
print_r($words); // ['say', 'bye', 'to', 'spaces']
```

```php
$models = [$model1, $model2, $model3];

$id_to_title = array_combine(
    array_column($models, 'id'),
    array_column($models, 'title')
);
```

```php
$letters = ['a', 'a', 'a', 'a', 'b', 'b', 'c', 'd', 'd', 'd', 'd',

$values = array_count_values($letters); // get key to count array
arsort($values); // sort descending preserving key
$top = array_slice($values, 0, 3); // get top 3

print_r($top);
// Array
// (
//     [d] => 5
//     [a] => 4
//     [b] => 2
// )
```

```php
$order = [
    ['product_id' => 1, 'price' => 99, 'count' => 1],
    ['product_id' => 2, 'price' => 50, 'count' => 2],
    ['product_id' => 2, 'price' => 17, 'count' => 3],
];

$sum = array_sum(array_map(function($product_row) {
    return $product_row['price'] * $product_row['count'];
}, $order));

print_r($sum); // 250
```

Thanks!

full tutorial:
https://goo.gl/VQ7QWb

email me:
tony@dontgiveafish.com

envatotuts+ DON'TGIVEAFISH