
# Introdução

Prof. Anderson Henrique

TECH
soluções em ti

# Capítulo 3

# Arrays

É, sem dúvida, um dos recursos mais poderosos da linguagem. O programador que assimilar bem esta parte terá muito mais produtividade no seu dia-a-dia. Isto porque os arrays no PHP servem como verdadeiros contêineres, servindo para armazenar números, strings, objetos, dentre outros, de forma dinâmica. Além disso, o PHP nos oferece uma gama enorme de funções para manipulá-los, as quais serão vistas a seguir.

# Criando um array

Arrays são acessados mediante uma posição, como um índice numérico. Para criar um array, pode-se utilizar a função array([chave] => valor,...)

```
    $cores = array('vermelho','azul','verde','amarelo');
    $cores = array(0 => 'vermelho', 1 => 'azul', 2 => 'verde',
20                                                  3 => 'amarelo');
```

Outra forma de criar um array é simplesmente adicionando-lhe valores com a seguinte sintaxe:

```
25  $nomes[] = 'maria';
26  $nomes[] = 'joão';
27  $nomes[] = 'carlos';
28  $nomes[] = 'josé';
```

De qualquer forma, para acessar o array indexado, basta indicar o seu índice entre colchetes:

```
35    echo $cores[0] . "<br>";
36    echo $cores[1] . "<br>";
37    echo $cores[2] . "<hr>";
38
39    echo $nomes[0] . "<br>";
40    echo $nomes[1] . "<br>";
41    echo $nomes[2] . "<hr>";
```

**Arrays associativos**

Os arrays no PHP são associativos pois contêm uma chave de acesso para cada posição. Para criar um array, pode-se utilizar a função array([chave] => valor).

```
48    $hex = array('vermelho' => 'FF0000', 'verde' => '00FF00',
49                                          'azul' => '0000FF');
```

Outra forma de criar um array associativo é simplesmente adicionando-lhes valores com a seguinte sintaxe:

```
56    $pessoa['nome'] = 'Maria da Silva';
57    $pessoa['rua'] = 'São João';
58    $pessoa['bairro'] = 'Cidade Alta';
59    $pessoa['cidade'] = 'Porto Alegre';
```

De qualquer forma, para acessar o array, basta indicar sua chave entre colchetes:

```
64    echo $hex['vermelho'] . "<br>";
65    echo $hex['azul'] . "<br>";
66    echo $hex['verde'] . "<hr>";
67
68    echo $pessoa['nome'] . "<br>";
69    echo $pessoa['rua'] . "<br>";
70    echo $pessoa['bairro'] . "<hr>";
```

TECH
soluções em ti

# Iterações

Os arrays podem ser iterados no PHP pelo operador FOREACH, percorrendo cada uma das posições do array. Exemplo:

```
78    $frutas['cor'] = 'vermelha';
79    $frutas['sabor'] = 'doce';
80    $frutas['formato'] = 'redonda';
81    $frutas['nome'] = 'maçã';
82
83    foreach ($frutas as $chave => $fruta){
84        echo "$chave => $fruta <br>";
85    }
```

# Acesso

As posições de um array podem ser acessadas a qualquer momento, e sobre elas operações podem ser realizadas.

```php
$minha_multa['carro'] = 'Pálio';
$minha_multa['valor'] = 178.00;

$minha_multa['carro'] .= ' ED 1.0';
$minha_multa['valor'] += 20.00;

print "<pre>";
var_dump($minha_multa);

$comidas[] = 'Lazanha';
$comidas[] = 'Pizza';
$comidas[] = 'Macarrão';

$comidas[1] = 'Pizza Calabreza';

var_dump($comidas);
```

# Arrays multidimensionais

Também conhecidas como matrizes são arrays nas quais algumas de suas posições podem conter outros arrays de forma recursiva. Um array multidimensional pode ser criado pela função array().

```
117   $veiculos = array(
118       'Palio' => array(
119           'cor'=>'azul',
120           'potência'=>'1.0',
121           'opcionais'=>'Ar Cond.'
122           ),
123       'Corsa' => array(
124           'cor'=>'cinza',
125           'potência'=>'1.3',
126           'opcionais'=>'MP3'
127           ),
128       'Gol' => array(
129           'cor'=>'branco',
130           'potência'=>'1.0',
131           'opcionais'=>'Metálica'
132           )
133   );
```

TECH
soluções em ti

Para realizar iterações em um array multidimensional é preciso observar quantos níveis ele possui. No exemplo a seguir, realizamos a iteração para o primeiro nível do array(veiculos) e, para cada iteração, realizamos uma nova iteração, para imprimir suas características.

```
142 foreach ($veiculos as $modelo => $caracteristicas){
143     echo "=> Modelo: $modelo <br>";
144     foreach ($caracteristicas as $caracteristica => $valor){
145         echo "característica $caracteristica => $valor <br>";
146     }
147 }
```

A seguir veremos uma série de funções utilizadas exclusivamente para manipulação de arrays, funções de ordenação, intersecção, acesso, dentre outras.

TECH
soluções em ti

**array_push**

Adiciona elementos ao final de um array.

```
158    $a = array('verde','azul','vermelho');
159    array_push($a, 'amarelo');
160    print_r($a);
```

**array_pop**

Remove um valor do final de um array.

```
166    array_pop($a);
167    print_r($a);
```

TECH
soluções em ti

## array_shift

Remove um elemento do início de um array.

```
173    array_shift($a);
174    print_r($a);
```

## array_unshift

Adiciona um elemento no início de um array.

```
180    array_unshift($a, 'laranja');
181    print_r($a);
```

## array_pad

Preenche um array com um dado valor, determinada quantidade de posições.

```
    $cli = array('ana','marcos','simone');
    $cli = array_pad($cli, 6, 'luiza');
189 print_r($cli);
```

## array_reverse

Recebe um array e retorna-o na ordem reversa.

```
195 $num[0] = 1;
196 $num[1] = 2;
197 $num[2] = 3;
198 $num[3] = 4;
199 $rev = array_reverse($num,TRUE);
200 print_r($rev);
```

TECH
soluções em ti

## array_merge

Mescla dois ou mais arrays. Um array é adicionado ao final de outro. O resultado é um novo array. Se ambos arrays tiverem conteúdo indexado pela mesma chave, o segundo irá sobrepor ao primeiro.

```
208    $times_paulistas = array('Palmeiras','Corinthians');
209    $times_cariocas = array('Vasco da Gama','Botafogo');
210    $times_juntos = array_merge($times_paulistas,$times_cariocas);
211    print_r($times_juntos);
```

## array_keys

Retorna as chaves de um array. Se o segundo parâmetro for indicado, a função retornará apenas índices que apontam para um conteúdo igual ao parâmetro.

```
218    $aluno = array('matricula'=>'0012','nome'=>'liliane','idade'=>14);
219    $indices = array_keys($aluno);
220    print_r($indices);
```

## array_values

Retorna um array contendo os valores de um outro array.

```
226    $valores = array_values($aluno);
227    print_r($valores);
```

## array_slice

Extrai uma porção de um array.

```
233    $color[0] = 'green';
234    $color[1] = 'yellow';
235    $color[2] = 'red';
236    $color[3] = 'blue';
237    $color[4] = 'gray';
238    $color[5] = 'white';
239    $fatia = array_slice($color, 2, 3);
240    print_r($fatia);
```

TECH
soluções em ti

**count**

Retorna a quantidade de elementos de um array.

```
246    $bebidas = array('refrigerante','suco','vinho','cerveja','vodka');
247    echo 'O array $bebidas contém '.count($bebidas).' elementos <hr>';
```

**in_array**

Verifica se um array contém um determinado valor.

```
253  if(in_array('café', $bebidas)){
254      echo 'café foi encontrado <hr>';
255  }else{
256      echo 'café não foi encontrado <hr>';
257  }
```

TECH
soluções em ti

**sort**

Ordena um array pelo seu valor, não mantendo associação de índices.

```
263    sort($bebidas);
264    print_r($bebidas);
```

**rsort**

Ordena um array em ordem reversa pelo seu valor, não mantendo a associação de índices.

```
271    rsort($bebidas);
272    print_r($bebidas);
```

TECH
soluções em ti

**asort**

Ordena um array pelo seu valor, mantendo a associação de índices. Para ordenar de forma reversa, use o arsort().

```
279    asort($bebidas);
280    print_r($bebidas);
```

**ksort**

Ordena um array pelos seus índices. Para ordem reversa, utilize o krsort().

```
286    $automovel['potência'] = '1.0';
287    $automovel['cor'] = 'branco';
288    $automovel['modelo'] = 'celta';
289    $automovel['opcionais'] = 'ar quente';
290    ksort($automovel);
291    print_r($automovel);
```

**explode**

Converte uma string e um array, separando os elementos por meio de um identificador.

```
297    $data = "31/12/2018";
298    print_r(explode("/", $data));
```

**Implode**

Converte um array em uma string, unindo os elementos da string por meio de um identificador.

```
305    $padrao = array('Maria','Paulo','Joana','Loiane');
306    $resultado = implode(' - ', $padrao);
307    print_r($resultado);
```

TECH
soluções em ti