

Técnico em Desenvolvimento de Sistemas

Programação WEB 2

Linguagem PHP com banco de dados MySQL

Prof. Laércio Silva

laercio.silva31@etec.sp.gov.br

Indsilva@Hotmail.com

Sintaxe básica do PHP:

Array

Também tradicionalmente conhecidos como vetores, são variáveis que servem para guardar vários valores de forma uniforme na memória.

É um grupo de itens que compartilham certas características, como similaridade e tipo de dados.

Até o momento, se quiséssemos armazenar o nome de 05 pessoas, teríamos que declarar 05 variáveis (\$nome1, \$nome2, \$nome3 ,..., \$nome5).

Mas utilizando vetor, bastaríamos declarar apenas 1 variável do tipo array e armazenar os 05 nomes nesta variável.

Sintaxe básica do PHP:

Exemplo dos tipos de declarações

```
exemplo_vetor.php x
<html>
<body>

<p>Exemplo declaração de variável sem array</p>

<?php
    $nome1 = "Maria";
    $nome2 = "João";
    $nome3 = "Pedro";
    $nome4 = "Paulo";
    $nome5 = "Lucas";

    echo("$nome1<br/>$nome2<br/>$nome3 <br/>$nome4<br/>$nome5<br/>");

?>
<p>Exemplo declaração de variável com array</p>

<?php
    $nome = array("Maria", "João", "Pedro", "Paulo", "Lucas");

    echo "Os nomes no array<br/>";

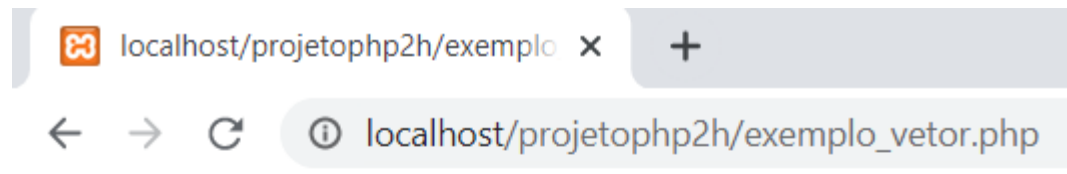
    foreach ($nome as $nomes) {
        echo "<br/>".$nomes;
    }

?>

</body>
</html>
```

Sintaxe básica do PHP:

Resultado



Exemplo declaração de variável sem array

Maria
João
Pedro
Paulo
Lucas

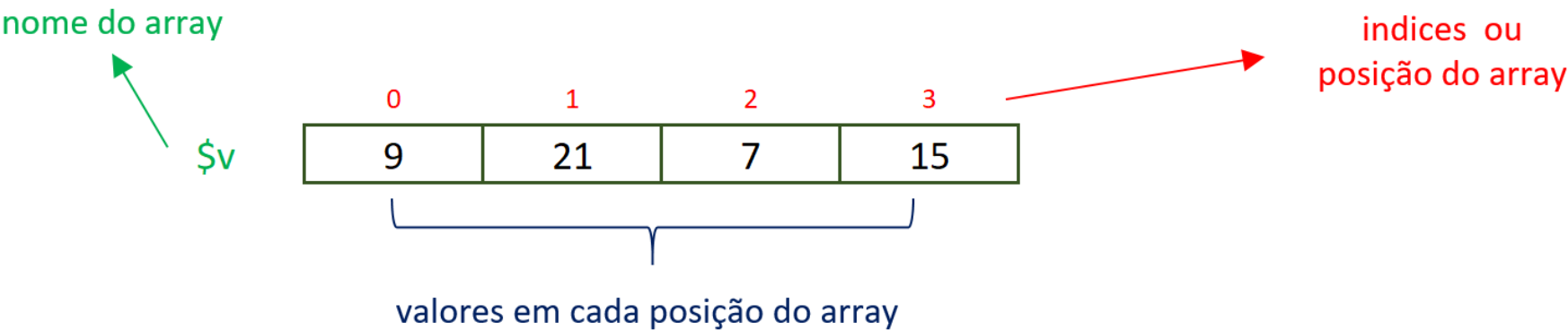
Exemplo declaração de variável com array

Os nomes no array

Maria
João
Pedro
Paulo
Lucas

Sintaxe básica do PHP:

Estrutura de um Array



Criando um Array

Em PHP não precisa que se defina o tamanho do array no momento da criação, nem menos que você declare o array antes de usá-lo.

Declaração básica de um array:

```
$vetor = array();
```

Sintaxe básica do PHP:

Array

Declaração de um array com inicialização de valores:

```
$vetor = array{ "Maria" , "José" , "Pedro" };
```

Sintaxe básica do PHP:

Array

Podemos também inicializar cada valor, em cada posição do vetor:

```
$vetor[0] = "Maria";
```

```
$vetor[1] = "José";
```

```
$vetor[2] = "Pedro";
```


Array

Para exibir os dados do vetor, temos que indicar a posição do vetor que queremos acessar no caso de utilizar o comando:

```
echo $vetor[0];
```

```
echo $vetor[1];
```

```
echo $vetor[2];
```

Array

Uma outra forma de atribuir, é não precisando indicar o índice/posição:

```
$vetor[] = "Maria";  
$vetor[] = "José";  
$vetor[] = "Pedro";
```

Matriz Numérica

Esses arrays podem armazenar números, strings e qualquer objeto, mas seu índice será representado por números. Por padrão, o índice da matriz começa do zero.

Sintaxe básica do PHP:

Matriz Numérica

```
array_numerica.php x
<html>
  <body>

    <?php

      $num = array( 1, 2, 3, 4, 5);

      foreach( $num as $valor ) {
        echo "Valor é $valor <br />";
      }

      $num[0] = "um";
      $num[1] = "dois";
      $num[2] = "três";
      $num[3] = "quatro";
      $num[4] = "cinco";

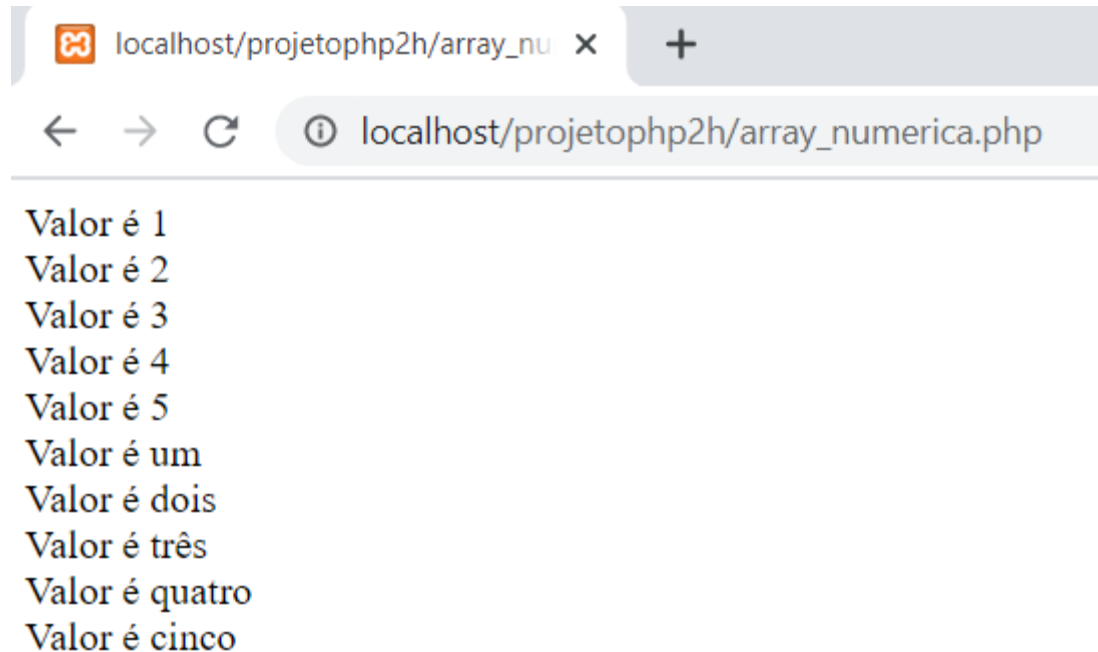
      foreach( $num as $valor ) {
        echo "Valor é $valor <br />";
      }

    ?>

  </body>
</html>
```

Sintaxe básica do PHP:

Resultado:



Array associado

Podemos também criar um array associado, ou seja, capaz de fazer associações por meio dos índices, que geralmente em vez de números, são usadas strings para definir a posição no vetor.

```
$nome["MA"] = "Maria";
```

```
$nome["ZE"] = "José";
```

```
$nome["PE"] = "Pedro";
```

Sintaxe básica do PHP:

Array associado

Podemos também inicializar um array associativo da seguinte forma:

```
$vetor = array ("MA" => "Maria", "ZE" => "José", "PE" => "Pedro");
```

Importante:

```
$vetor = array ("MA" => "Maria", "ZE" => "José", "PE" =>
```



Símbolo de associação

Sintaxe básica do PHP:

Exemplo Array associativa

```
array_associativa.php x
<html>
  <body>

    <?php

      $salarios = array("João" => 2000, "Pedro" => 1000, "Maria" => 500);

      echo "Salário de João é ". $salarios['João'] . "<br />";
      echo "Salário de Pedro é ". $salarios['Pedro'] . "<br />";
      echo "Salário de Maria é ". $salarios['Maria'] . "<br />";

      $salarios['João'] = "Alto";
      $salarios['Pedro'] = "Médio";
      $salarios['Maria'] = "Baixo";

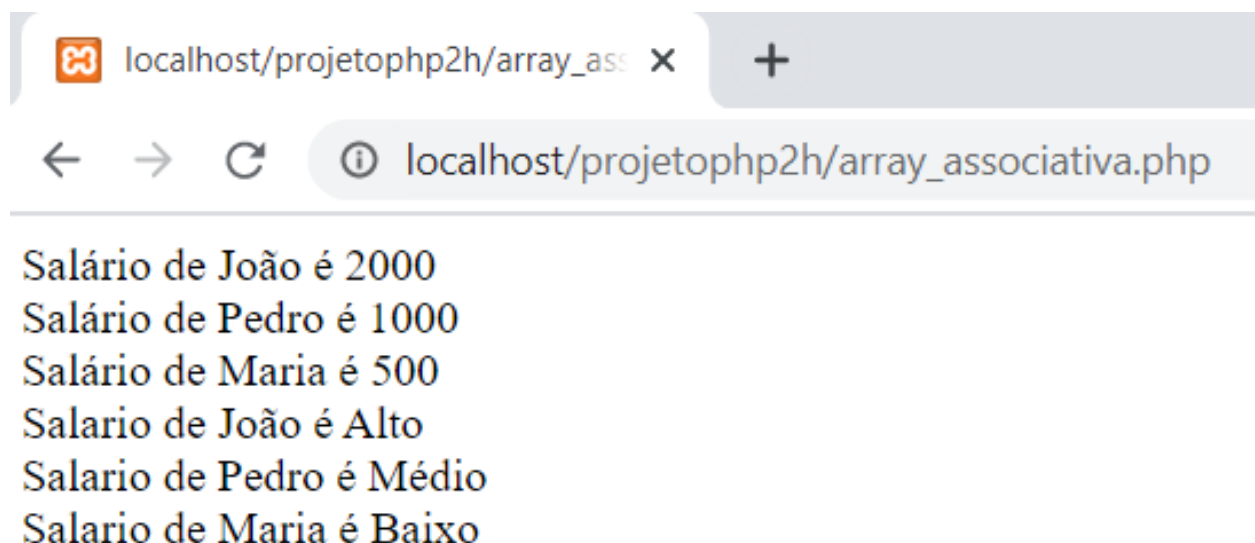
      echo "Salario de João é ". $salarios['João'] . "<br />";
      echo "Salario de Pedro é ". $salarios['Pedro'] . "<br />";
      echo "Salario de Maria é ". $salarios['Maria'] . "<br />";

    ?>

  </body>
</html>
```


Sintaxe básica do PHP:

Resultado:



Sintaxe básica do PHP:

Função is_array()

Esta função serve para fazer um teste e saber se a variável é do tipo array.

```
funcao_is_array.php x
<?php

$variavel = "Ana";

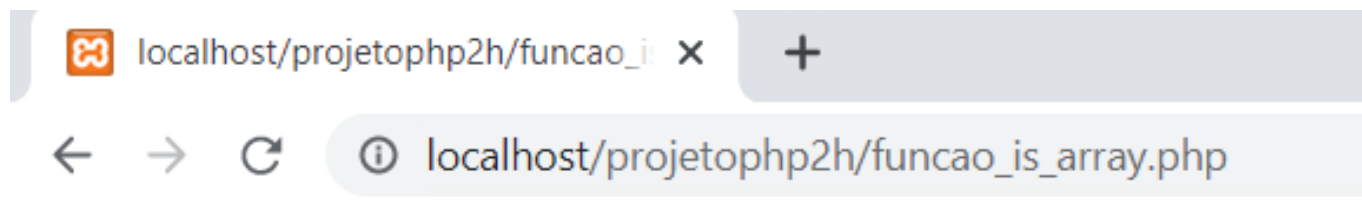
$nomes = array("MA" => "Maria", "ZE" => "José", "PE" => "Pedro");

if (is_array($variavel)) {
    echo "<br/> Variável é um array";
} else
{
    echo "<br/>Variável não é um array";
}
if (is_array($nomes)) {
    echo "<br/>Nomes é um array";
}
else{
    echo "</br>Nomes não é um array";
}

?>
```

Sintaxe básica do PHP:

Resultado:



Variável não é um array

Nomes é um array

Sintaxe básica do PHP:

Adicionando valores no vetor

Função `array_unshift()`

Esta função tem a finalidade de adicionar elementos na frente (início) do array.

array_unshift.php

<?php

```
$nomes = array("Maria", "José", "Pedro");  
array_unshift($nomes, "Ana");
```

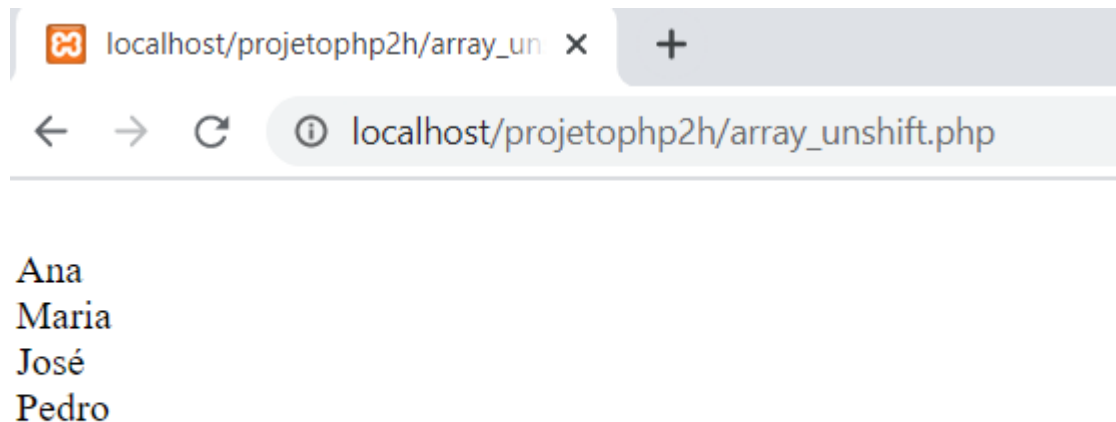
```
foreach ($nomes as $item) {  
    echo "<br/>".$item;  
}
```

?>

Sintaxe básica do PHP:

Resultado:

Podemos notar, que o nome Ana, foi inserido na primeira posição do vetor.



Sintaxe básica do PHP:

Função array_push()

Esta função tem a finalidade de adicionar elementos no final do array.

```
array_push.php x
<?php

$nomes = array("Maria", "José", "Pedro");
array_push($nomes, "Ana");

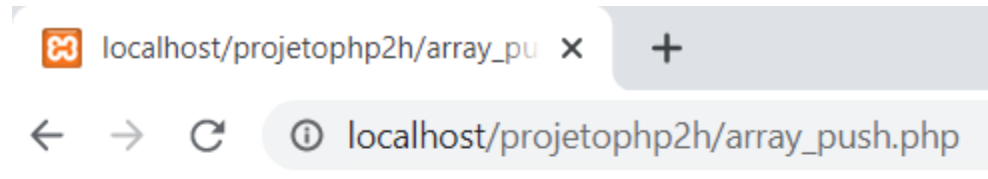
foreach ($nomes as $item) {
    echo "<br/>".$item;
}

?>
```

Sintaxe básica do PHP:

Resultado:

Podemos notar, que o nome Ana, foi inserido na última posição do vetor.



Maria
José
Pedro
Ana

Sintaxe básica do PHP:

Removendo valores do vetor

Função array_shift()

Esta função remove e retorna o item encontrado na primeira posição do vetor.

```
array_shift.php x
<?php

$nomes = array("Maria", "José", "Pedro");
$excluido = array_shift($nomes);

foreach ($nomes as $item) {
    echo "<br/>".$item;
}

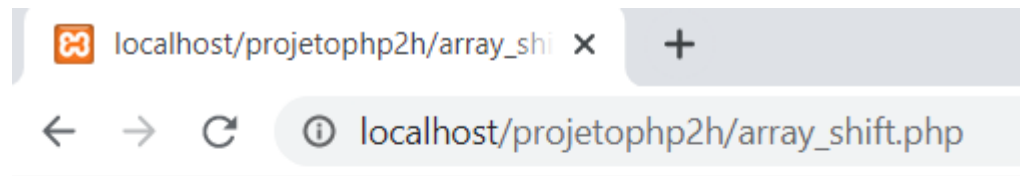
echo "<br/><br/> Nome removido: ".$excluido;

?>
```


Sintaxe básica do PHP:

Resultado:

Podemos notar, que o nome Maria que constava na primeira posição do vetor, foi removido.



José
Pedro

Nome removido: Maria

Sintaxe básica do PHP:

Função array_pop()

Esta função remove e retorna o item encontrado no final do vetor.

array_pop.php

×

```
<?php
```

```
$nomes = array("Maria", "José", "Pedro");  
$excluido = array_pop($nomes);
```

```
foreach ($nomes as $item) {  
    echo "<br/>".$item;  
}
```

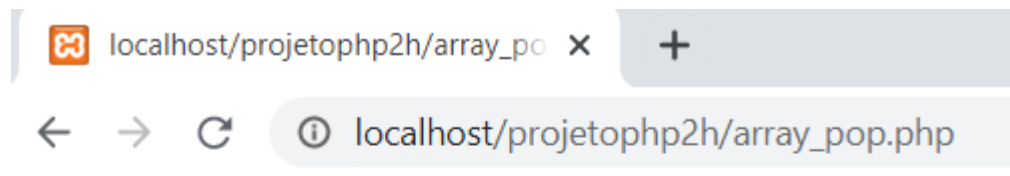
```
echo "<br/><br/> Nome removido: ".$excluido;
```

```
?>
```

Sintaxe básica do PHP:

Resultado:

Podemos notar, que o nome Pedro que constava na última posição do vetor, foi removido.



Maria

José

Nome removido: Pedro

Sintaxe básica do PHP:

Função array_key_exists()

Esta função permite a busca de um elemento usando como parâmetro a chave de associação, sendo que retorna *TRUE* se a chave for localizada.

```
array_key_exists.php x
<?php

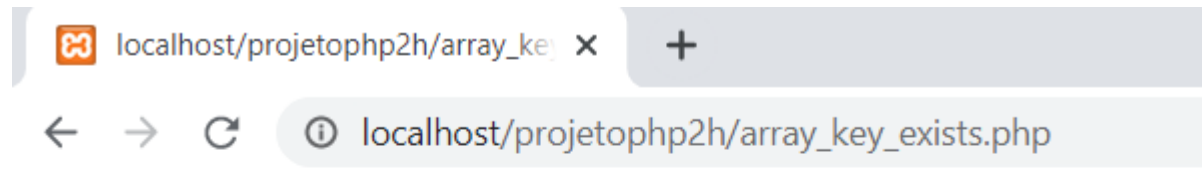
$nomes = array("MA"=>"Maria", "ZE"=>"José", "PE"=>"Pedro", "AN"=>"Ana");

if (array_key_exists("ZE", $nomes)) {
    echo "<br/> Nome encontrado no vetor, através da chave.";
}

?>
```

Sintaxe básica do PHP:

Resultado:



Nome encontrado no vetor, através da chave.

Sintaxe básica do PHP:

Função array_search()

Esta função busca um valor no array e retorna a chave se for localizado este valor.

array_search.php

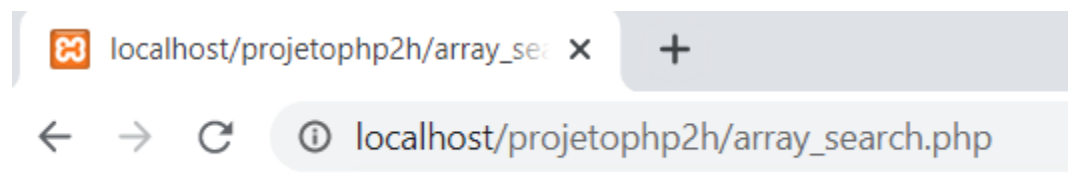
<?php

```
$nomes = array("MA"=>"Maria", "ZE"=>"José", "PE"=>"Pedro", "AN"=>"Ana");  
$chave = array_search("Ana", $nomes);  
  
echo "<br/> Chave de busca realizada: ".$chave;
```

?>

Sintaxe básica do PHP:

Resultado:



Chave de busca realizada: AN

Sintaxe básica do PHP:

Função array_keys()

Esta função retorna um array formado por todas as chaves localizadas de outro array.

```
array_keys.php x
<?php

$nomes = array("MA"=>"Maria", "ZE"=>"José", "PE"=>"Pedro", "AN"=>"Ana");

$chaves = array_keys($nomes);

foreach ($chaves as $item) {
    echo "<br/>".$item;
}
echo "<br/>";

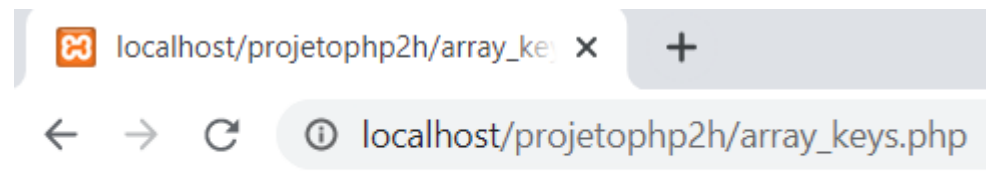
//função para exibir dados de um vetor

print_r($chaves);

?>
```


Sintaxe básica do PHP:

Resultado:



MA

ZE

PE

AN

Array ([0] => MA [1] => ZE [2] => PE [3] => AN)

Sintaxe básica do PHP:

Função array_values()

Esta função retorna um array com todos os valores localizados de outro array (sem as chaves associadas).

array_values.php

×

<?php

```
$nomes = array("MA"=>"Maria", "ZE"=>"José", "PE"=>"Pedro", "AN"=>"Ana");
```

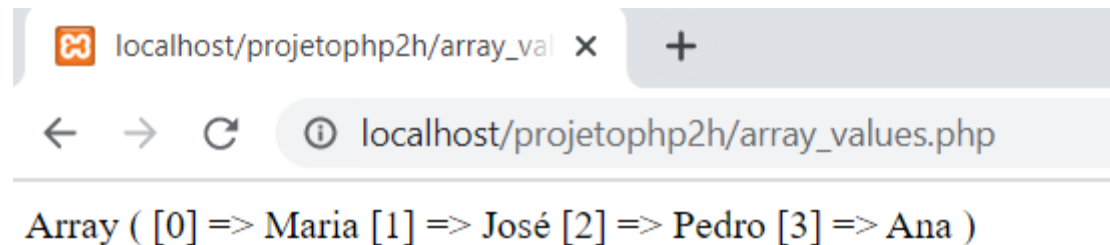
```
$valores = array_values($nomes);
```

```
print_r($valores);
```

?>

Sintaxe básica do PHP:

Resultado:



Movendo o ponteiro de um array

Existem algumas funções para tornar possível o deslocamento do ponteiro de um vetor.

As funções são:

- **next()** retorna o valor do próximo elemento do array na posição seguinte ao ponteiro.
- **prev()** retorna o valor anterior da atual posição do ponteiro.
- **reset()** volta ao começo do array.
- **end()** vai para o final do array.

Sintaxe básica do PHP:

Movendo o ponteiro de um array

array_movendo_ponteiro.php x

```
<?php
```

```
$nomes = array("MA"=>"Maria", "ZE"=>"José", "PE"=>"Pedro", "AN"=>"Ana");
```

```
echo "<br/> Next: ".next($nomes);
```

```
echo "<br/> End: ".end($nomes);
```

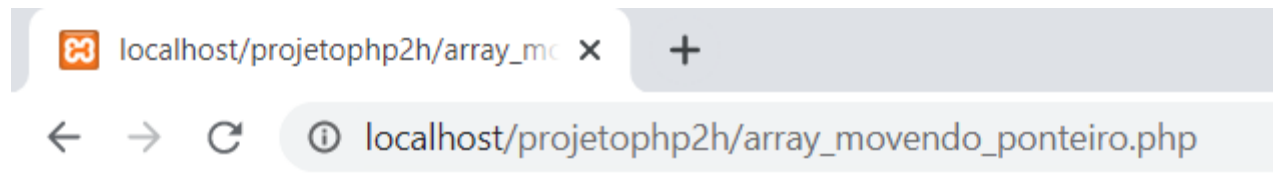
```
echo "<br/> Prev: ".prev($nomes);
```

```
echo "<br/> Reset: ".reset($nomes);
```

```
?>
```

Sintaxe básica do PHP:

Resultado:



Next: José
End: Ana
Prev: Pedro
Reset: Maria

Sintaxe básica do PHP:

Recuperando a chave de um ponteiro

A função **key()** retorna a chave da posição atual do ponteiro.

array_recuperando_chave_ponteiro.php x

```
<?php
```

```
$nomes = array("MA"=>"Maria", "ZE"=>"José", "PE"=>"Pedro", "AN"=>"Ana");
```

```
while ($chave = key($nomes)) {
```

```
    echo "<br/>".$chave;
```

```
    next($nomes);
```

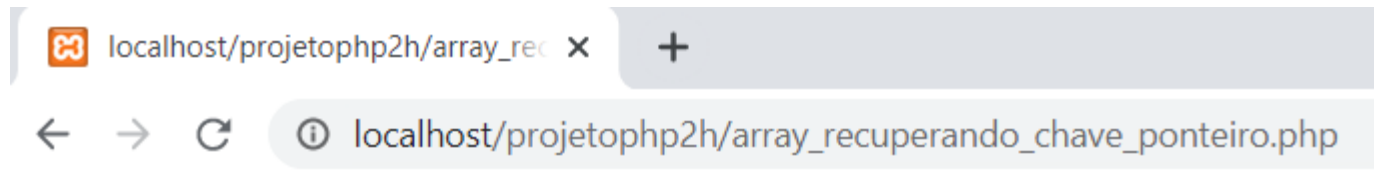
```
}
```

```
?>
```

Sintaxe básica do PHP:

Resultado:

O resultado apresentado no navegador, serão as chaves de cada elemento do vetor.



MA
ZE
PE
AN

Muito Obrigado

Até a Próxima Aula

Prof. Laércio Silva
Email: Indsilva@hotmail.com