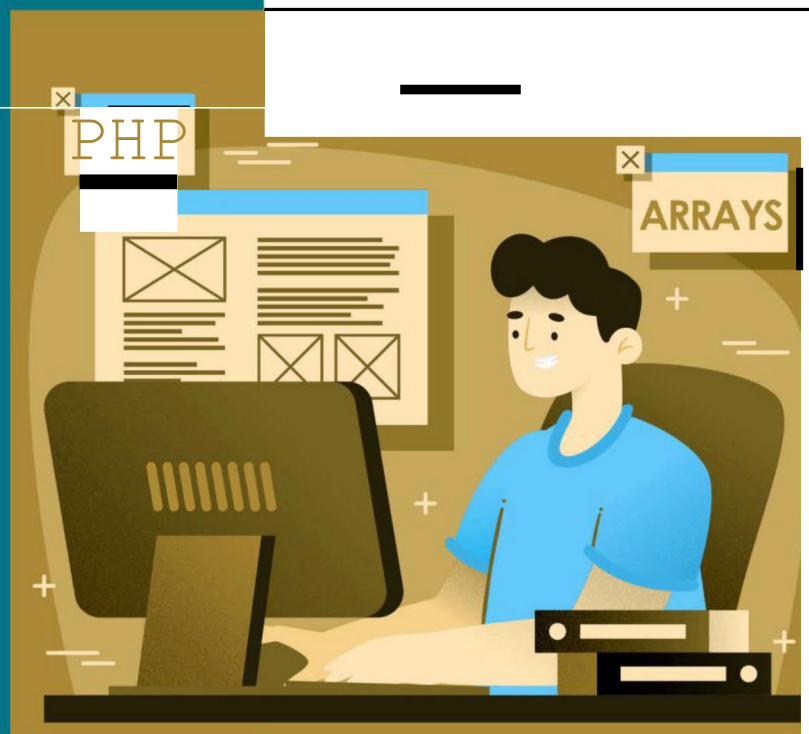

AGENDA 5

PHP - ARRAYS





Arrays

A definição correta para um array em PHP é de um mapa ordenado, ou seja, trata-se de um tipo de dado, assim como integer, float, string ou boolean. Porém, ele pode armazenar mais de um valor, relacionando-os a suas chaves. Simplificando, em um contexto geral, podemos dizer que:

Array é uma variável do php que nos fornece a possibilidade de atribuir diversos valores ao mesmo tempo.

É possível relacionar o array ao conceito e utilização de vetor e matriz (por meio do uso arrays multidimensionais). Como essa variável possui um identificador, torna-se capaz de armazenar mais de um valor na mesma estrutura, e de acordo com o índice é possível gerenciar esses conteúdos (valores). Então, cada array está associado a um índice, que indica a posição de armazenamento de um valor na memória do array.

Obs.: No PHP esse índice pode ser tanto um texto, quanto um número, e seu uso sempre está delimitado entre colchete ([]).

Declarando arrays

Para melhor entendimento vamos programar, então, no visual studio Code.

Crie um arquivo e o salve dentro da pasta root ou Agenda5, com o nome de “decArray.php”

– Neste arquivo vamos criar o primeiro array, utilizando o método construtor da linguagem array(). Então codique:

```
<?php
    $sudeste = array("São Paulo", "Minas Gerais", "Rio de Janeiro", "Espírito
    Santo");

?>
```

Para a declaração utilizamos uma variável normal, atribuindo o resultado da função array para cada valor informado, resultando em um índice para ser acessado que pode ser representado graficamente pela imagem a seguir.

| 0 | 1 | 2 | 3 |
|-----------|--------------|----------------|----------------|
| São Paulo | Minas Gerais | Rio de Janeiro | Espírito Santo |

Imagem 3. Representação gráfica dos valores ordenados em um array.

Podemos perceber pela representação gráfica (imagem 3), que os índices são gerados automaticamente, iniciando sempre em 0.

Obs.: É possível também realizar a declaração do array utilizando colchetes. Para o PHP, essas duas formas são iguais, conforme código a seguir:

```
<?php
    $sudeste = ["São Paulo", "Minas Gerais", "Rio de Janeiro", "Espírito Santo"];
?>
```

Outras maneiras para declarar o mesmo array seriam:

Diretamente:

```
$sudeste[] = "São Paulo";
$sudeste[] = "Minas Gerais";
$sudeste[] = "Rio de Janeiro";
$sudeste[] = "Espírito Santo";
```

Ou ainda você pode definir os índices, mesmo que fora de ordem:

```
$sudeste[1] = "Minas Gerais";
$sudeste[0] = "São Paulo";
$sudeste[2] = "Rio de Janeiro";
$sudeste[3] = "Espírito Santo";
```

Para acessar os valores do array, basta utilizar seu nome seguido do índice entre colchetes, como exemplo a seguir:

```
<?php
    echo $sudeste[0];
?>
```

O resultado no navegador para todos os exemplos anteriores seria:

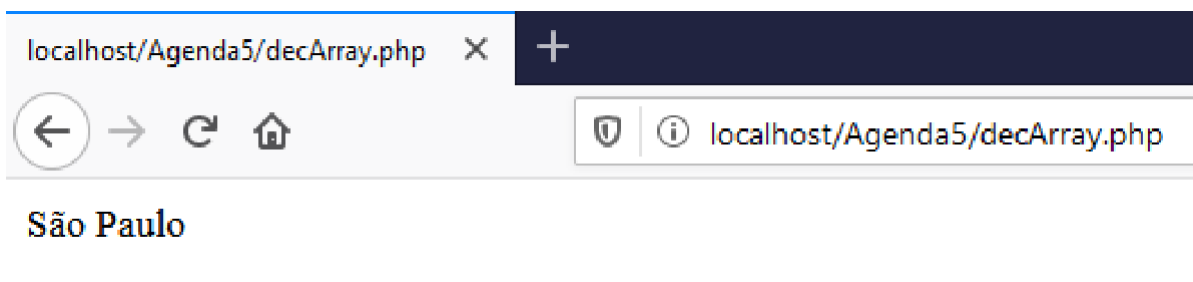


Imagem 4. Navegador o primeiro valor do array através do comando echo.html.

Declarando Array Associativo (arrays com índices textuais)

Você também pode definir índices (*keys* ou *chaves*) como *strings*. Esse tipo de array é denominado associativo e sua declaração precisa sempre “associar” o índice com o seu respectivo valor. Pode ser declarado por meio de um construtor:

```
$filme = array('titulo' => 'Uma mente Brilhante', 'duracao' => '135min', 'genero' => 'drama');
```

Ou até mesmo de forma direta:

```
$filme['titulo'] = 'Uma mente Brilhante';
$filme['duracao'] = '135min';
$filme['genero'] = 'drama';
```

Para melhor visualização da declaração desse array associativo, observe a representação gráfica a seguir:

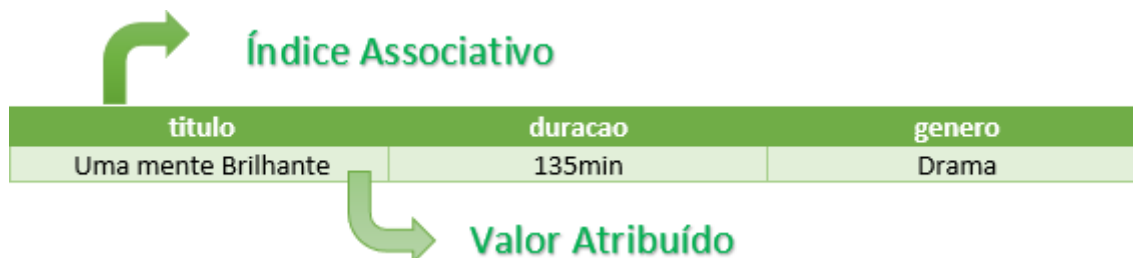


Imagem 5. Representação gráfica dos valores ordenados em um array.

Para acessar os valores do array, basta utilizar o mesmo padrão: nome do array, seguido do índice entre colchetes, que neste caso será uma string, como exemplo a seguir:

```
<?php
echo $filme['titulo'].'<br>';
echo $filme['duracao'].'<br>';
echo $filme['genero'].'<br>';

?>
```

Resultado no navegador:

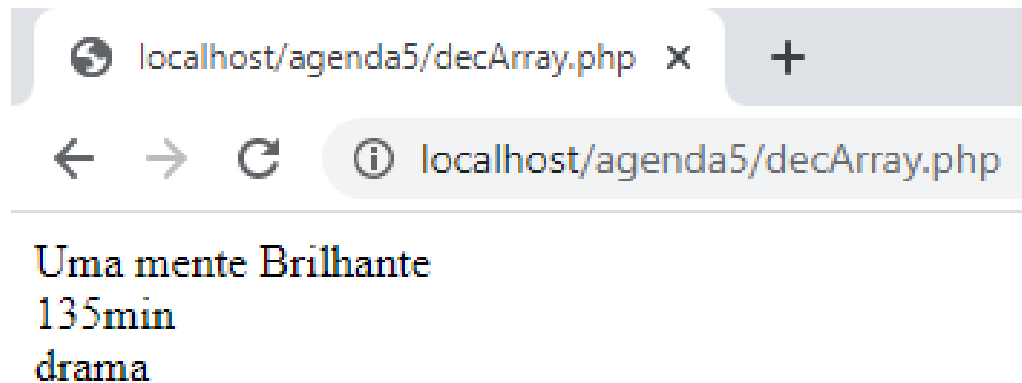


Imagem 6. Navegador exibindo todos os valores do array associativo “filme” por meio do comando echo.html.

Iterando um array com Estruturas de Repetição

Uma das principais vantagens na utilização de um array é a possibilidade de automatização de tarefas em alguma funcionalidade e proposta da web site de manipular as informações no array com utilização de algoritmos. Essas vantagens acontecem principalmente por meio da utilização de estruturas de repetição.

Para melhor entendimento, crie um arquivo no visual studio Code e o salve dentro da pasta root ou Agenda5, com o nome de “exemploForArray” - Neste arquivo vamos criar um formulário com dois inputs, uma combobox e um botão.

O mais importante para esse exemplo é mostrar que a partir de um array populado com todos os estados do país, por exemplo, podemos criar um combobox que ofereça possibilidades como a de um usuário escolher o estado para o envio de mercadoria. Veja a codificação:

```

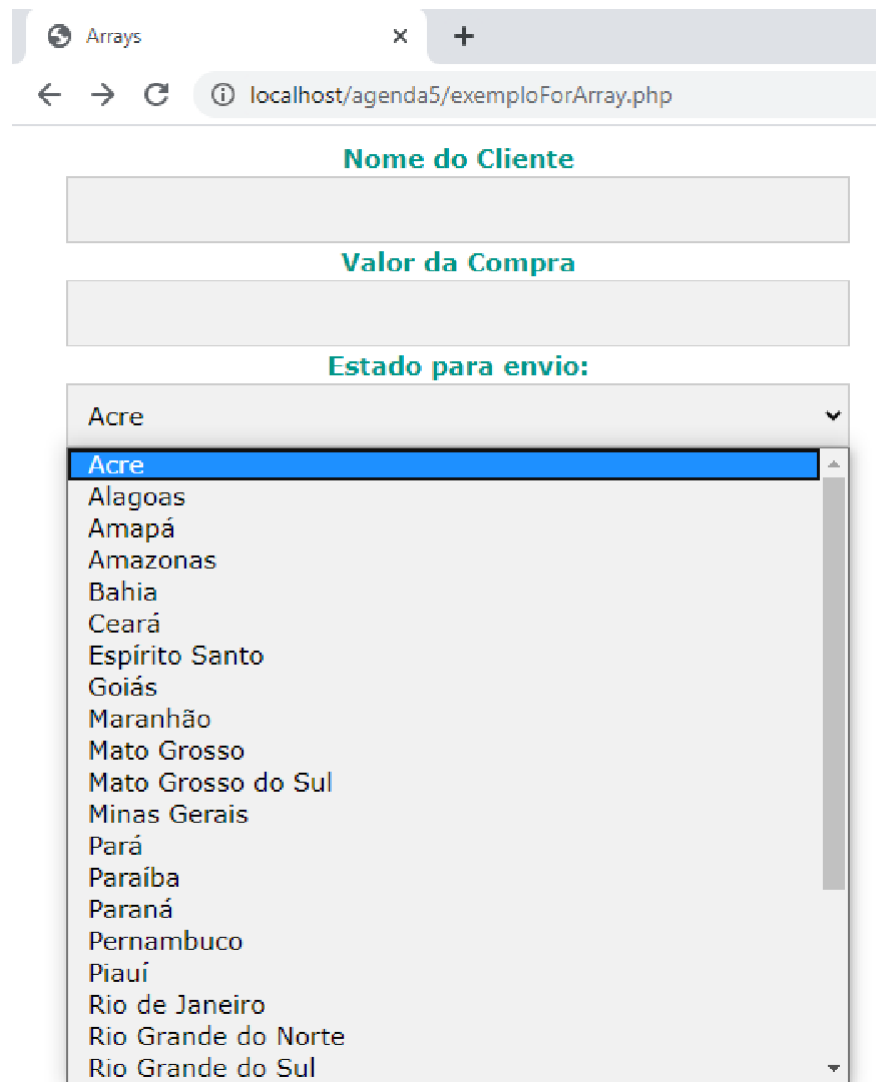
<!DOCTYPE html>
<html lang="pt-br">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
    <title>Arrays</title>
</head>
<body>
<div class=" w3-third w3-center w3-animate-top w3-padding">
    <form class="w3-
container" method="post" action="atividadeOnlineAction.php">
        <label class="w3-text-teal"><b>Nome do Cliente</b></label>
        <input class="w3-input w3-border w3-light-
grey" name="txtNome" type="text">
        <label class="w3-text-teal"><b>Valor da Compra</b></label>
        <input class="w3-input w3-border w3-light-
grey" name="txtValorCompra" type="number">
        <label class="w3-text-teal"><b>Estado para envio:</b></label>
        <select class="w3-input w3-border w3-light-grey" name = "cmbEstados">
            <?php
                $estados = array("Acre","Alagoas","Amapá","Amazonas","Bahia","
Ceará",
                                "Espírito Santo","Goiás","Maranhão","Mato Grosso","Mato Grosso
do Sul ",
                                "Minas Gerais","Pará","Paraíba","Paraná","Pernambuco","Piauí",

                                "Rio de Janeiro",
                                "Rio Grande do Norte","Rio Grande do
Sul","Rondônia","Roraima" ,
                                "São Paulo","Santa Catarina", "Sergipe", "Tocantins","Distrito
Federal");

                for($i = 0; $i < count($estados); $i++)
                {
                    echo '<option value="'. $i. '">'. $estados[$i]. '</option>';
                }
            ?>
        </select>
        <br>
        <button class="w3-btn w3-blue-grey">Enviar</button>
    </form>
</div>
</html>

```

O resultado em um navegador pode ser visualizado a seguir:



The screenshot shows a web browser window with the title 'Arrays' and the address bar displaying 'localhost/agenda5/exemploForArray.php'. The page contains three input fields and a dropdown menu. The first input field is labeled 'Nome do Cliente'. The second input field is labeled 'Valor da Compra'. The third input field is labeled 'Estado para envio:' and is a dropdown menu with 'Acre' selected. The dropdown menu lists the following states: Acre, Alagoas, Amapá, Amazonas, Bahia, Ceará, Espírito Santo, Goiás, Maranhão, Mato Grosso, Mato Grosso do Sul, Minas Gerais, Pará, Paraíba, Paraná, Pernambuco, Piauí, Rio de Janeiro, Rio Grande do Norte, and Rio Grande do Sul.

Imagem 7. Resultado no Navegador do combobox criado a partir do array estados com a estrutura de repetição for.

Porém em PHP, a estrutura mais comum para a iteração de arrays, certamente, é por meio da utilização da Estrutura de Repetição **foreach**. Para o mesmo resultado do exemplo anterior podemos codificar dessa forma:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
  <title>Arrays</title>
</head>
<body>
<div class=" w3-third w3-center w3-animate-top w3-padding">
```

```

    <form class="w3-
container" method="post" action="atividadeOnlineAction.php">
        <label class="w3-text-teal"><b>Nome do Cliente</b></label>
        <input class="w3-input w3-border w3-light-
grey" name="txtNome" type="text">
        <label class="w3-text-teal"><b>Valor da Compra</b></label>
        <input class="w3-input w3-border w3-light-
grey" name="txtValorCompra" type="number">
        <label class="w3-text-teal"><b>Estado para envio:</b></label>
        <select class="w3-input w3-border w3-light-grey" name = "cmbEstados">
            <?php
                $estados = array("Acre","Alagoas","Amapá","Amazonas","Bahia","
Ceará",
                                "Espírito Santo","Goiás","Maranhão","Mato Grosso","Mato Grosso
do Sul ",
                                "Minas Gerais","Pará","Paraíba","Paraná","Pernambuco","Piauí",
                                "Rio de Janeiro",
                                "Rio Grande do Norte","Rio Grande do Sul","Rondônia","Roraima"
,"Santa Catarina",
                                "São Paulo","Santa Catarina", "Sergipe", "Tocantins","Distrito
Federal");

                foreach($estados as $estado)
                {
                    echo '<option>'.$estado.'</option>';
                }
            ?>
        </select>
        <br>
        <button class="w3-btn w3-blue-grey">Enviar</button>
    </form>
</div>
</html>

```

Conforme visto, o resultado será exatamente o mesmo, mas perceba que a declaração dessa estrutura é um pouco mais fácil de entender. Podemos interpretá-la simplesmente fazendo a leitura da estrutura:

Para cada um dos **\$estados**, será tratado como um **\$estado**, ou seja, para cada iteração da estrutura de repetição, o valor da nova variável **\$estado** será de um valor do array identificado por meio de um índice diferente, começando do primeiro até o último, para maior consolidação da interpretação.

Observe representação a seguir:

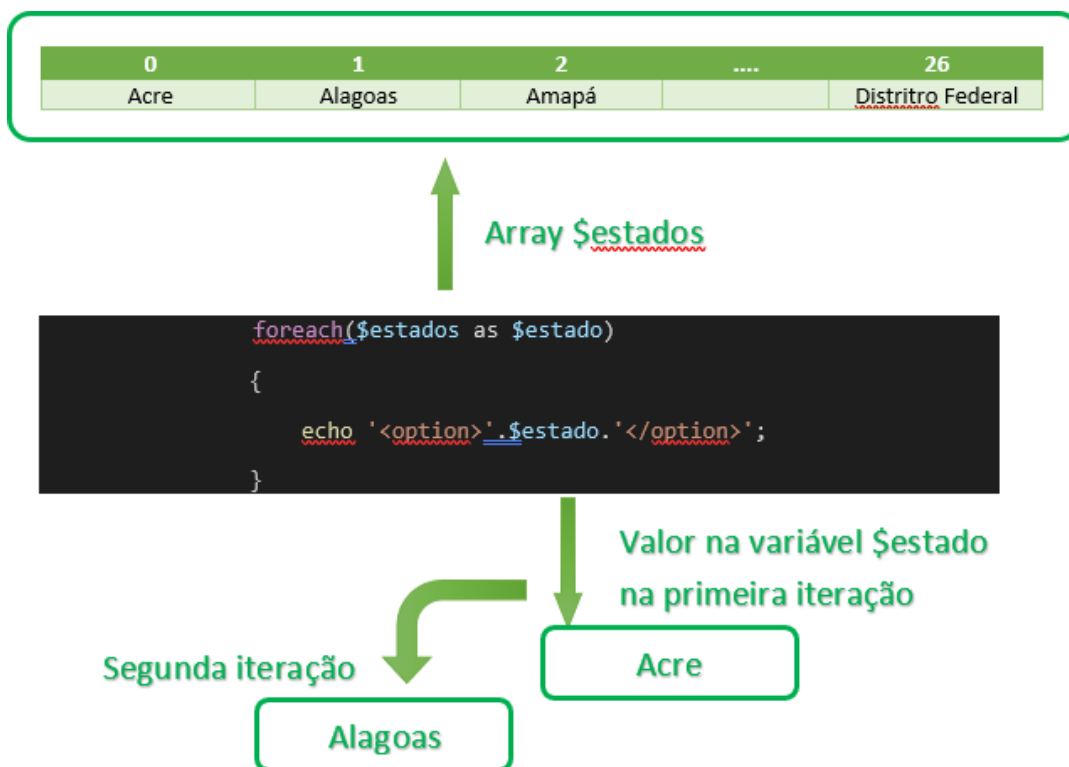


Imagem 8. Representação da execução da primeira e segunda iteração de um comando Foreach no array estado.

Funções para arrays

Existem diversas funções nativas do PHP que oferecem diversos recursos para serem utilizados em array. A seguir uma sequência de quatro vídeos que explica as principais funções e suas utilizações.

Funções de Arrays – Parte #1



Fonte: Node Studio Treinamentos - Curso de PHP 7 - Aula 15 - Funções de Arrays #1. Disponível em <https://www.youtube.com/watch?v=ys5KjCdvSAQ>. Acessado em 29/08/2025.

Funções de Arrays – Parte #2



Fonte: Node Studio Treinamentos - Curso de PHP 7 - Aula 16 - Funções de Arrays #2. Disponível em <https://www.youtube.com/watch?v=O5mKRnvoWbE>. Acessado em 29/08/2025.

Funções de Arrays – Parte #3



Fonte: Node Studio Treinamentos - Curso de PHP 7 - Aula 16 - Funções de Arrays #3. Disponível em <https://www.youtube.com/watch?v=4XcUwfCpshq>. Acessado em 29/08/2025.

Funções de Arrays – Parte #4



Fonte: Node Studio Treinamentos - Curso de PHP 7 - Aula 18 - Funções de Arrays #4. Disponível em https://www.youtube.com/watch?v=Zz-5WjcP8_q. Acessado em 29/08/2025.

Arrays Multidimensionais.

Para entender de forma bem simples, trata-se de um Array que tem vinculado um outro Array para cada uma de suas posições, ou seja, uma array dentro de outro array.

O PHP permite construir estruturas com quantas dimensões forem necessárias, mas é certo que, em quase sua totalidade o desenvolvimento será por meio do uso de estruturas que tenham até 2 dimensões, ou seja, estruturas em formato de tabelas ou planilhas.

Para melhor entendimento, no visual studio Code, crie um arquivo e o salve dentro da pasta root ou Agenda5, com o nome de “exemploDeclaracaoArrayMultidimensional” - Neste arquivo vamos criar um array bidimensional chamado produto. Nele deve constar o nome do produto e seu respectivo valor.

Para arrays multidimensionais também existem várias formas de declaração, então vamos codificar.

Obs.: O uso de strings para índices também são permitidos.

```
$produtos = array(
    array("Processador", "900"),
    array("Mouse", "15"),
    array("Teclado", "20"),
    array("Impressora", "500"),
    array("Monitor", "450"),
    array("Placa de Vídeo", "1500"),
    array("Memória RAM 8G", "500"),
    array("Placa Mãe", "600"),
    array("Mouse Pad", "25"),
    array("SSD", "245"),
);
```

Para essa declaração pode utilizar a imagem a seguir para sua apresentação gráfica.

| Índice | 0 | 1 |
|--------|----------------|------|
| 0 | Processador | 900 |
| 1 | Mouse | 15 |
| 2 | Teclado | 20 |
| 3 | Impressora | 500 |
| 4 | Monitor | 450 |
| 5 | Placa de Vídeo | 1500 |
| 6 | Memória RAM 8G | 500 |
| 7 | Placa Mãe | 600 |
| 8 | Mouse Pad | 25 |
| 9 | SSD | 245 |

Imagem 9. Representação gráfica dos valores em um array bidimensional.

Para acessar os valores do array bidimensional, basta utilizar seu nome seguido dos índices entre colchetes, como exemplo a seguir:

```
<?php
    echo $produtos[0][0];
?>
```

Resultado da imagem no navegador:

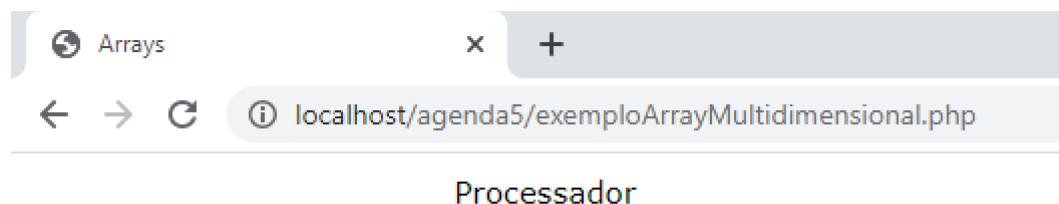


Imagem 10. Navegador exibindo o primeiro valor do array bidimensional produto por meio do comando echo.html.

Uma alternativa para a declaração, que inclusive, é muito utilizada para armazenamento de tabelas de banco de dados em array no php. Vamos colocar as “colunas” com índices associativos (strings) fazendo referência a seu conteúdo.

Exemplo:

```
$produtosAssociativo = array(
    array("nome"=> "Processador","valor"=> "900" ),
    array("nome"=> "Mouse","valor"=> "15" ),
    array("nome"=> "Teclado","valor"=> "20" ),
    array("nome"=> "Impressora","valor"=> "500" ),
    array("nome"=> "Monitor","valor"=> "450" ),
    array("nome"=> "Placa de Vídeo","valor"=> "1500" ),
    array("nome"=> "Memória RAM 8G","valor"=> "500" ),
    array("nome"=> "Placa Mãe","valor"=> "600" ),
    array("nome"=> "Mouse Pad","valor"=> "25" ),
    array("nome"=> "SSD","valor"=> "245" ),
);
```

Para esta declaração uma representação gráfica adequada seria:

| Índice | nome | valor |
|--------|----------------|-------|
| 0 | Processador | 900 |
| 1 | Mouse | 15 |
| 2 | Teclado | 20 |
| 3 | Impressora | 500 |
| 4 | Monitor | 450 |
| 5 | Placa de Vídeo | 1500 |
| 6 | Memória RAM 8G | 500 |
| 7 | Placa Mãe | 600 |
| 8 | Mouse Pad | 25 |
| 9 | SSD | 245 |

Para acessar os valores do array bidimensional, basta utilizar o nome seguido dos índices entre colchetes, conforme exemplo:

```
echo $produtosAssociativo[0]['nome'];
?>
```

O resultado seria o mesmo que o exemplo anterior exibido anteriormente por meio da imagem 10.

Iteração em Array Bidimensional.

De forma bem simples e objetiva, podemos tratar o array bidimensional como uma matriz, permitindo assim o uso de estruturas de repetição de diversas formas. Porém, o mais comum é usar a estrutura foreach.

Para melhor entendimento, no visual studio Code, crie um arquivo e o salve dentro da pasta root ou Agenda5, com o nome de “iteracaoArrayMultidimensional” - Neste arquivo vamos utilizar o array bidimensional “\$produtos” do exemplo anterior.

A partir de seu conteúdo, vamos criar uma tabela de produtos para serem exibidas no navegador, por meio de dois exemplos de uso da estrutura foreach.

Veja o código do primeiro exemplo:

```
echo '<table class="w3-table-all w3-hoverable w3-text-black">';
echo '<tr class="w3-teal ">';
echo '<th class="w3-center"> Nome</th>';
echo '<th class="w3-center"> Valor</th>';
echo '</tr>';
foreach($produtos as $produto)
{
    echo '<tr>';
    foreach($produto as $item)
    {
        echo '<td class="w3-center">'.$item.'</td>';

    }
    echo '</tr>';
}
echo '</table>';
```

Observe que neste exemplo estamos utilizando duas estruturas de repetição foreach. Para cada iteração o primeiro transforma o array bidimensional (**\$produtos**) em um array unidimensional (**\$produto**) e o segundo foreach transformando o array unidimensional (**\$produto**) em uma variável.

Veja a representação gráfica da primeira iteração de cada foreach:

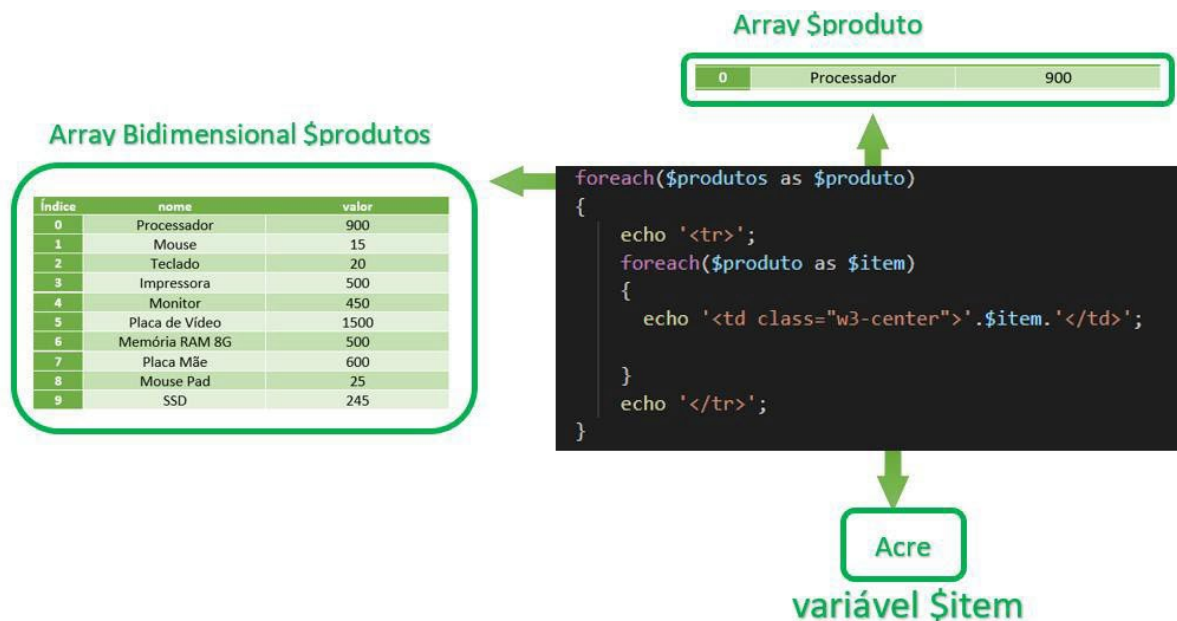


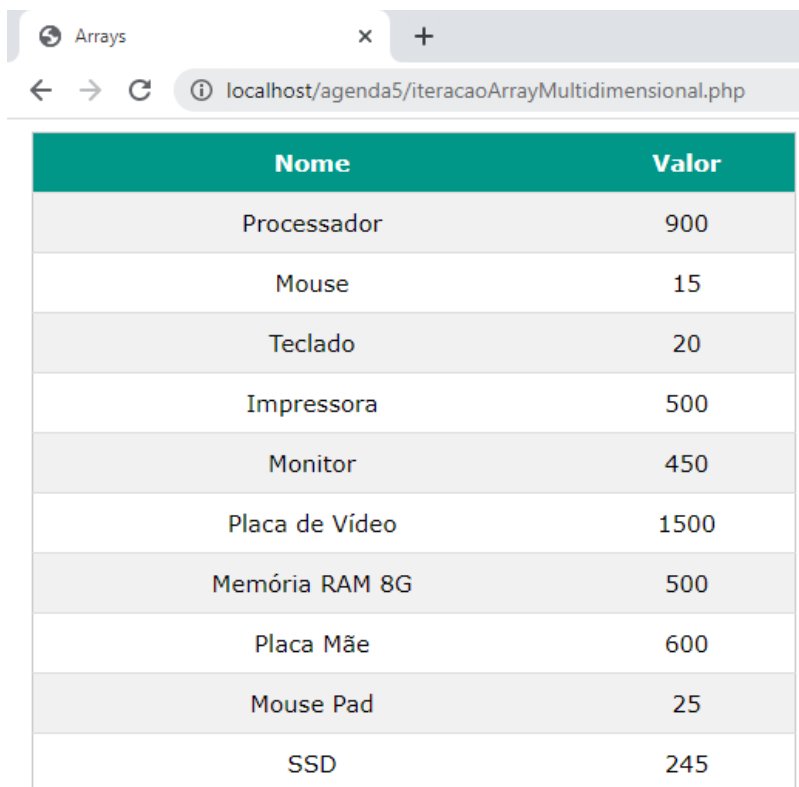
Imagem 12. Representação gráfica da primeira iteração dos foreach no array \$produto.

Para o segundo exemplo, deixamos um formatado que é usado de forma quase unânime quando os dados são oriundos de uma consulta a banco de dados.

```
echo '<table class="w3-table-all w3-hoverable w3-text-black">';
    echo '<tr class="w3-teal ">';
    echo '<th class="w3-center"> Nome</th>';
    echo '<th class="w3-center"> Valor</th>';
    echo '</tr>';
    foreach($produtosAssociativo as $produto)
    {
        echo '<tr>';
        echo '<td class="w3-center">'.$produto['nome'].'</td>';
        echo '<td class="w3-center">'.$produto['valor'].'</td>';
        echo '</tr>';
    }
    echo '</table>';
```

Observe que neste exemplo foi utilizado apenas um foreach, em que para cada iteração é transformado o array bidimensional (**\$produtosAssociativo**) em um array unidimensional (**\$produto**) e dentro desse, é acionada, para cada echo, uma posição diferente do array (nome e valor).

O resultado no navegador para ambos os exemplos pode ser observado na representação da imagem a seguir:



The screenshot shows a web browser window with the title 'Arrays' and the address bar displaying 'localhost/agenda5/iteracaoArrayMultidimensional.php'. The main content is a table with two columns: 'Nome' and 'Valor'.

| Nome | Valor |
|----------------|-------|
| Processador | 900 |
| Mouse | 15 |
| Teclado | 20 |
| Impressora | 500 |
| Monitor | 450 |
| Placa de Vídeo | 1500 |
| Memória RAM 8G | 500 |
| Placa Mãe | 600 |
| Mouse Pad | 25 |
| SSD | 245 |

Imagem 13. Navegador exibindo o resultado do uso da estrutura de repetição para a exibição do conteúdo de um array bidimensional em formato de tabela.html.



Joseph, recebeu de seu professor o desafio de criar uma tabela por meio de um array multidimensional, populado com todos os nomes e siglas dos estados brasileiros, inclusive o Distrito Federal. Essa tabela será exibida em um site de uma escola municipal de ensino fundamental.



Imagem 14. Joseph recebendo o desagio do professor (MACROVECTOR / FREEPIK,2020)

Utilizando o que foi visto até agora, crie a página solicitada:

- 1 - Crie um arquivo PHP na pasta root ou Agenda5.
- 2 - Crie uma tabela utilizando uma estrutura de repetição.

Dicas:

- Utilize a Estrutura de Repetição Foreach
- Os índices são: "estado" e "sigla".

Array disponibilizado pelo professor de Joseph

```
$estados = array(  
    array("estado" => "Acre", "sigla" => "AC"),  
    array("estado" => "Alagoas", "sigla" => "AL"),  
    array("estado" => "Amapá", "sigla" => "AP"),  
    array("estado" => "Amazonas", "sigla" => "AM"),  
    array("estado" => "Bahia", "sigla" => "BA"),  
    array("estado" => "Ceará", "sigla" => "CE"),  
    array("estado" => "Espírito Santo", "sigla" => "ES"),  
    array("estado" => "Goiás", "sigla" => "GO"),  
    array("estado" => "Maranhão", "sigla" => "MA"),  
    array("estado" => "Mato Grosso", "sigla" => "MT"),  
    array("estado" => "Mato Grosso do Sul", "sigla" => "MS"),  
    array("estado" => "Minas Gerais", "sigla" => "MG"),  
    array("estado" => "Pará", "sigla" => "PA"),  
    array("estado" => "Paraíba", "sigla" => "PB"),  
    array("estado" => "Paraná", "sigla" => "PR"),  
    array("estado" => "Pernambuco", "sigla" => "PE"),  
    array("estado" => "Piauí", "sigla" => "PI"),  
    array("estado" => "Rio de Janeiro", "sigla" => "RJ"),  
    array("estado" => "Rio Grande do Norte", "sigla" => "RN"),  
    array("estado" => "Rio Grande do Sul", "sigla" => "RS"),  
    array("estado" => "Rondônia", "sigla" => "RO"),  
    array("estado" => "Roraima", "sigla" => "RR"),  
    array("estado" => "Santa Catarina", "sigla" => "SC"),  
    array("estado" => "São Paulo", "sigla" => "SP"),  
    array("estado" => "Sergipe", "sigla" => "SE"),  
    array("estado" => "Tocantins", "sigla" => "TO"),  
    array("estado" => "Distrito Federal", "sigla" => "DF")  
);
```

A seguir, confira se você conseguiu resolver os desafios propostos!

Código para criação da Tabela

```
<table class="w3-table-all w3-hoverable w3-textblack">'
  <tr class="w3-teal ">'
    <th class="w3-center"> Estado</th>';
    <th class="w3-center"> Sigla</th>'
  </tr>
  <?php foreach ($estados as $estado) { ?>
    <tr>
      <td class="w3-center"><?php echo $estado['estado'] ?></td>
      <td class="w3-center"><?php echo $estado['sigla'] ?></td>
    </tr>
  <?php
  }
  ?>
</table>
```

Codificação Completa:

```
<!DOCTYPE html>
<html lang="pt-br">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
  <title>Arrays</title>
</head>

<body>
  <div class="w3-third w3-center w3-animate-top w3-padding">
    <?php
    $estados = array(
      array("estado" => "Acre", "sigla" => "AC"),
      array("estado" => "Alagoas", "sigla" => "AL"),
      array("estado" => "Amapá", "sigla" => "AP"),
      array("estado" => "Amazonas", "sigla" => "AM"),
      array("estado" => "Bahia", "sigla" => "BA"),
      array("estado" => "Ceará", "sigla" => "CE"),
      array("estado" => "Espírito Santo", "sigla" => "ES"),
      array("estado" => "Goiás", "sigla" => "GO"),
      array("estado" => "Maranhão", "sigla" => "MA"),
      array("estado" => "Mato Grosso", "sigla" => "MT"),
      array("estado" => "Mato Grosso do Sul", "sigla" => "MS"),
      array("estado" => "Minas Gerais", "sigla" => "MG"),
      array("estado" => "Pará", "sigla" => "PA"),
      array("estado" => "Paraíba", "sigla" => "PB"),
      array("estado" => "Paraná", "sigla" => "PR"),
      array("estado" => "Pernambuco", "sigla" => "PE"),
      array("estado" => "Piauí", "sigla" => "PI"),
      array("estado" => "Rio de Janeiro", "sigla" => "RJ"),
      array("estado" => "Rio Grande do Norte", "sigla" => "RN"),
      array("estado" => "Rio Grande do Sul", "sigla" => "RS"),
      array("estado" => "Rondônia", "sigla" => "RO"),
```

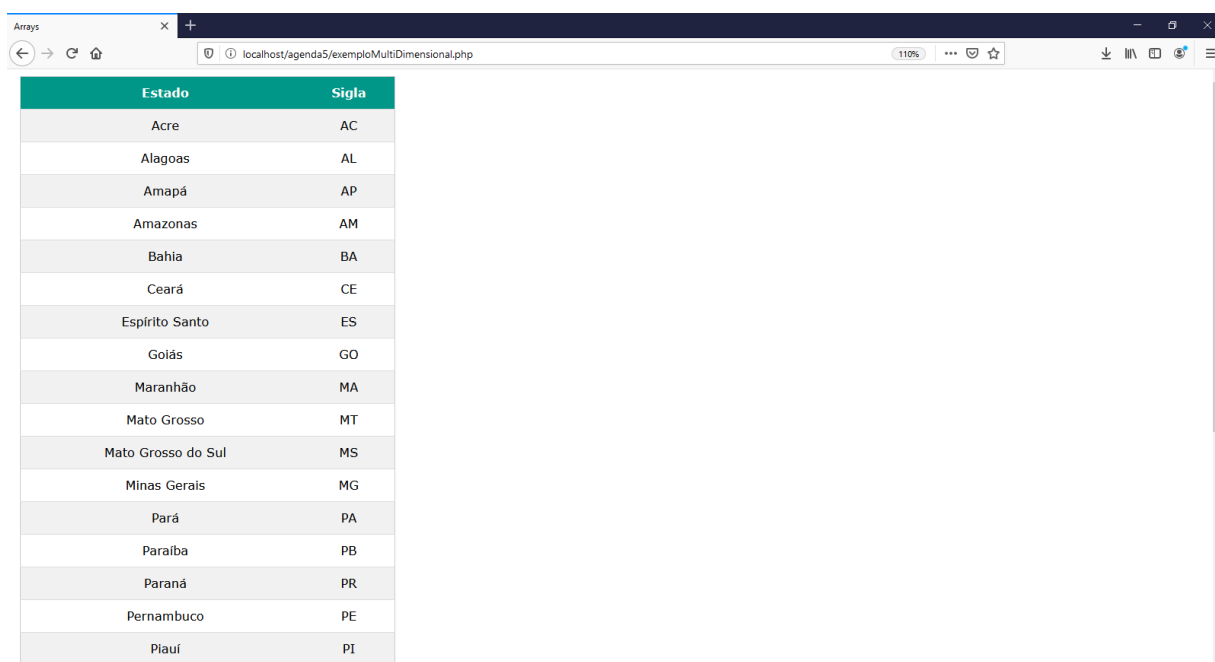
```

        array("estado" => "Roraima", "sigla" => "RR"),
        array("estado" => "Santa Catarina", "sigla" => "SC"),
        array("estado" => "São Paulo", "sigla" => "SP"),
        array("estado" => "Sergipe", "sigla" => "SE"),
        array("estado" => "Tocantins", "sigla" => "TO"),
        array("estado" => "Distrito Federal", "sigla" => "DF")
    ); ?>
<table class="w3-table-all w3-hoverable w3-textblack">'
    <tr class="w3-teal ">'
        <th class="w3-center"> Estado</th>';
        <th class="w3-center"> Sigla</th>'
    </tr>
    <?php foreach ($estados as $estado) { ?>
        <tr>
            <td class="w3-center"><?php echo $estado['estado'] ?></td>
            <td class="w3-center"><?php echo $estado['sigla'] ?></td>
        </tr>
    <?php } ?>
</table>
<br>
</div>
</body>

</html>

```

Resultado no Navegador:



A screenshot of a web browser window. The address bar shows the URL 'localhost/agenda5/exemploMultiDimensional.php'. The page displays a table with two columns: 'Estado' and 'Sigla'. The table lists 27 Brazilian states and their corresponding abbreviations. The browser interface includes standard navigation buttons (back, forward, home) and a scrollbar on the right side of the table.

| Estado | Sigla |
|--------------------|-------|
| Acre | AC |
| Alagoas | AL |
| Amapá | AP |
| Amazonas | AM |
| Bahia | BA |
| Ceará | CE |
| Espírito Santo | ES |
| Goiás | GO |
| Maranhão | MA |
| Mato Grosso | MT |
| Mato Grosso do Sul | MS |
| Minas Gerais | MG |
| Pará | PA |
| Paraíba | PB |
| Paraná | PR |
| Pernambuco | PE |
| Piauí | PI |

Imagem 15 – Possível resultado no navegador do Exercício Você no Comando.