

TEMA 5 - Conceito sobre Array

Vamos agora falar um pouco sobre Array, sendo uma rotina de tarefas muito usada em qualquer linguagem de programação, dentro do PHP vamos ver a importância desse comando ou função.

A finalidade de um Array é guardar vários dados economizando linhas de códigos, porque ao declarar uma variável do tipo Array ela pode guardar muitos dados e organizar por índice.

5.1 Introdução sobre tipos e Array

No PHP, os arrays podem ser criados por meio do construtor de linguagem `array()`, sendo composto por parênteses e chaves, na sintaxe.

Os arrays são estruturas de dados que permitem armazenar e manipular vários valores em uma única variável. Eles são muito úteis para organizar e processar informações de forma dinâmica e eficiente. No PHP, existem **três tipos de arrays: indexados, associativos e multidimensionais**.

Os **arrays indexados** são aqueles que usam índices numéricos para acessar seus elementos. Eles são criados usando a função `array()` ou a sintaxe de colchetes `[]`. Por exemplo:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exemplo de Arrays Indexados</title>
  <style>
    /* Estilos CSS para a página */
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
    }
    .container {
      max-width: 600px;
      margin: 50px auto;
      background-color: #fff;
      padding: 20px;
      border-radius: 8px;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    h2 {
      color: #333;
    }
    ul {
      list-style-type: none;
      padding: 0;
    }
    li {
      margin-bottom: 10px;
    }
  </style>
</head>
```

```

<body>
  <div class="container">
    <h2>Exemplo de Arrays Indexados em PHP</h2>
    <?php
      // Criando um array usando a função array()
      $frutas = array("Maçã", "Banana", "Laranja", "Pera");

      // Criando um array usando a sintaxe de colchetes []
      $cores = ["Vermelho", "Verde", "Azul", "Amarelo"];
    ?>

    <h3>Lista de Frutas</h3>
    <h4>Criando um array usando a função array()</h4>
    <ul>
      <?php
        // Iterando sobre o array de frutas e exibindo cada uma delas como itens de lista HTML
        foreach ($frutas as $fruta) {
          echo "<li>$fruta</li>";
        }
      ?>
    </ul>

    <h3>Lista de Cores</h3>
    <h4>Criando um array usando a sintaxe de colchetes []</h4>
    <ul>
      <?php
        // Iterando sobre o array de cores e exibindo cada uma delas como itens de lista HTML
        foreach ($cores as $cor) {
          echo "<li>$cor</li>";
        }
      ?>
    </ul>
  </div>
</body>
</html>

```

Os arrays associativos são aqueles que usam chaves alfanuméricas para acessar seus elementos. Eles são criados usando a função `array()` ou a sintaxe de colchetes `[]`, mas especificando as chaves e os valores de cada elemento. Por exemplo:

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exemplo de Arrays Associativos</title>
  <style>
    /* Estilos CSS para a página */
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
    }
  </style>

```

```

        margin: 0;
        padding: 0;
    }
    .container {
        max-width: 600px;
        margin: 50px auto;
        background-color: #fff;
        padding: 20px;
        border-radius: 8px;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    h2 {
        color: #333;
    }
    ul {
        list-style-type: none;
        padding: 0;
    }
    li {
        margin-bottom: 10px;
    }
</style>
</head>
<body>
<div class="container">
    <h2>Exemplo de Arrays Associativos em PHP</h2>
    <?php
    // Criando um array associativo especificando chaves e valores para cada elemento
    $pessoa = array(
        "nome" => "João",
        "idade" => 25,
        "cidade" => "São Paulo"
    );
    ?>

    <h3>Detalhes da Pessoa</h3>
    <ul>
        <?php
        // Iterando sobre o array associativo e exibindo cada chave e valor
        foreach ($pessoa as $chave => $valor) {
            echo "<li><strong>$chave:</strong> $valor</li>";
        }
        ?>
    </ul>
</div>
</body>
</html>

```

Os arrays multidimensionais são aqueles que contêm outros arrays como elementos. Eles são criados usando a função `array()` ou a sintaxe de colchetes `[]`, mas aninhando outros arrays dentro deles. Por exemplo:

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exemplo de Arrays Associativos - Múltiplas Pessoas</title>
  <style>
    /* Estilos CSS para a página */
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
    }
    .container {
      max-width: 600px;
      margin: 50px auto;
      background-color: #fff;
      padding: 20px;
      border-radius: 8px;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    h2 {
      color: #333;
    }
    ul {
      list-style-type: none;
      padding: 0;
    }
    li {
      margin-bottom: 10px;
    }
  </style>
</head>
<body>
  <div class="container">
    <h2>Exemplo de Arrays Associativos em PHP - Múltiplas Pessoas</h2>
    <?php
      // Array multidimensional com informações de múltiplas pessoas
      $pessoas = array(
        array(
          "nome" => "João",
          "idade" => 25,
          "cidade" => "São Paulo"
        ),
        array(
          "nome" => "Maria",
          "idade" => 30,
          "cidade" => "Rio de Janeiro"
        ),
      );
    </?php>
  </div>
</body>
</html>

```

```

        array(
            "nome" => "Pedro",
            "idade" => 28,
            "cidade" => "Belo Horizonte"
        )
    );
?>

<?php foreach ($pessoas as $pessoa): ?>
    <h3>Detalhes da Pessoa</h3>
    <ul>
        <?php foreach ($pessoa as $chave => $valor): ?>
            <li><strong><?php echo $chave ?>:</strong> <?php echo $valor ?></li>
        <?php endforeach; ?>
    </ul>
<?php endforeach; ?>
</div>
</body>
</html>

```

Para acessar os elementos de um array, basta usar o operador [] e informar o índice ou a chave correspondente. Por exemplo:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Agenda de Contatos</title>
    <style>
        /* Estilos CSS para a página */
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            margin: 0;
            padding: 0;
        }
        .container {
            max-width: 600px;
            margin: 50px auto;
            background-color: #fff;
            padding: 20px;
            border-radius: 8px;
            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        }
        h2 {
            color: #333;
        }
        ul {
            list-style-type: none;
            padding: 0;
        }
    </style>

```

```

    }
    li {
        margin-bottom: 10px;
    }
</style>
</head>
<body>
<div class="container">
    <h2>Agenda de Contatos</h2>
    <?php
    // Array multidimensional representando a agenda de contatos
    $agenda = array(
        array(
            "nome" => "João",
            "telefone" => "(11) 1234-5678",
            "cidade" => "São Paulo"
        ),
        array(
            "nome" => "Maria",
            "telefone" => "(21) 9876-5432",
            "cidade" => "Rio de Janeiro"
        ),
        array(
            "nome" => "Pedro",
            "telefone" => "(31) 4567-8901",
            "cidade" => "Belo Horizonte"
        )
    );
    ?>

    <h3>Contatos na Agenda:</h3>
    <ul>
        <?php foreach ($agenda as $contato): ?>
            <li>
                <strong>Nome:</strong> <?php echo $contato["nome"]; ?><br>
                <strong>Telefone:</strong> <?php echo $contato["telefone"]; ?><br>
                <strong>Cidade:</strong> <?php echo $contato["cidade"]; ?><br>
            </li>
        <?php endforeach; ?>
    </ul>

    <h3>Exemplos de Acessos aos Elementos do Array:</h3>
    <ul>
        <li>Acessando o telefone de Maria: <?php echo $agenda[1]["telefone"]; ?></li>
        <li>Acessando a cidade de Pedro: <?php echo $agenda[2]["cidade"]; ?></li>
    </ul>
</div>
</body>
</html>

```

5.2 Criando um Array e seus tipos

Array do tipo vetor:

Os dados guardados em um Array do tipo vetor são organizados por colunas e chamados por índice, todos os tipos de Array tem uma sequência lógica iniciado por zero, então para chamar um valor dentro de um Array, chamamos por esse número que é rotulado de índice.

Variável \$cor está recebendo três valores veja como fica a alocação desse tipo de variável.

| | |
|---|---------|
| 0 | amarelo |
| 1 | azul |
| 2 | rosa |

Cada valor alocado nessa variável, vai ser organizado em índice que inicia com zero, para chamar cada um desses valores vamos chamar eles por números que chamamos de índice ou chave.

```
<?php
$cor = array("Amarelo", "Azul", "Rosa");

// Exibir todos os dados
print_r($cor);
echo "<br>";
// Exibir somente o índice zero que é "Amarelo"
echo $cor[0];
?>
```

Array do tipo matriz:

Os dados guardados em um Array do tipo matriz, são guardados e organizados também pelo índice, só que nesse caso guardamos em uma tabela fictícia baseada em linhas e múltiplas colunas. Veja o exemplo:

```
<?php
$cadastro = array(
    array("Anselmo", "Renata", "Lucas"),
    array("Ana", "Ticlane", "Roberta"),
    array("Daniela", "Adriana", "Maria")
);

// Exibir Roberta (linha 0, coluna 1)
echo $cadastro[0][1];

echo "<br>";

// Exibir Daniela (linha 2, coluna 0)
echo $cadastro[2][0];
?>
```

5.3 Modificando e mesclando

Para modificar um array, é pela sua configuração explícita de valores neles. Para isso, basta inserirmos valores no array, com a chave específica para cada dado entre colchetes. Veja o exemplo:

```
<?php
```

```
// Criar um array
$frutas = array("Maçã", "Banana", "Laranja");

// Exibir o array antes da modificação
echo "Array antes da modificação: ";
print_r($frutas);

// Modificar um elemento do array
$frutas[1] = "Abacaxi";
echo "<br>";
echo "Trocando Banana por Abacaxi";
echo "<br>";

// Exibir o array após a modificação
echo "Array após a modificação: ";
print_r($frutas);
?>
```

Para mesclar valores dentro de um array temos que usar uma função específica que é `array_merge`, mesclar é unir duas ou mais array em uma só, uma junção de dados.

```
<?php
// Array de alunos
$alunos = array("João", "Pedro", "Lucas");
echo "<strong>Alunos</strong>". "<br>";
print_r($alunos);
echo "<br>";

// Array de alunas
$alunas = array("Maria", "Ana", "Carla");
echo "<strong>Alunas</strong>". "<br>";
print_r($alunas);
echo "<br>";

// Mesclar os dois arrays
$alunos_e_alunas = array_merge($alunos, $alunas);
echo "<strong>Mesclagem</strong>". "<br>";
// Exibir o array resultante
print_r($alunos_e_alunas);
?>
```

RESUMO:

Neste capítulo sobre Arrays, exploramos os conceitos fundamentais desse elemento crucial na programação. Os Arrays são estruturas de dados que permitem armazenar e manipular múltiplos valores em uma única variável. No PHP, existem três tipos principais de Arrays: indexados, associativos e multidimensionais. Os indexados usam índices numéricos para acessar elementos, enquanto os associativos utilizam chaves alfanuméricas. Os multidimensionais contêm outros Arrays como elementos.

Demonstramos como criar e acessar elementos em Arrays, destacando a importância dos índices e chaves. Além disso, apresentamos exemplos de Arrays do tipo vetor e matriz, ilustrando como

organizar dados em índices e como modificar e mesclar Arrays quando necessário. Entender esses conceitos é essencial para o desenvolvimento back-end em PHP, pois Arrays são ferramentas poderosas para organizar e manipular informações de forma eficiente.

1. Explique como os Arrays indexados são estruturados em PHP, incluindo a importância dos índices numéricos. Dê um exemplo simples.
2. Imagine que você está criando um Array que armazena nomes de cidades. Cite pelo menos três cidades e seus índices correspondentes no Array.
3. Em um site de receitas, como você usaria um Array associativo para armazenar informações sobre ingredientes e suas quantidades? Dê um exemplo simples.
4. Qual é a principal vantagem de usar Arrays em programação web, especialmente no desenvolvimento back-end? Explique usando conceitos abordados no capítulo.
5. Descreva o papel das chaves (ou índices) em Arrays associativos e explique por que elas são úteis na organização e recuperação de dados em PHP.
6. Descreva o processo de modificação de um Array em PHP, fornecendo um exemplo prático em que você adiciona ou altera valores em um Array.
7. Qual é a diferença entre vetor e matriz?
8. O que é índice?
9. Qual é a função do laço de repetição na programação?
10. O vetor e a matriz iniciam sua organização com o número 1 ou 0?

1. Arrays indexados em PHP:

- Em PHP, os arrays indexados são estruturados como coleções ordenadas de elementos, onde cada elemento é acessado através de um índice numérico.

- Os índices numéricos em arrays indexados começam com 0 e aumentam sequencialmente.

- Um exemplo simples de um array indexado em PHP:

```
```php
$frutas = array("Maçã", "Banana", "Laranja");
```
```

2. Exemplo de Array de cidades com índices correspondentes:

- Cidades: São Paulo, Rio de Janeiro, Belo Horizonte

- Índices correspondentes no array: 0, 1, 2

```
```php
$cidades = array("São Paulo", "Rio de Janeiro", "Belo Horizonte");
```
```

3. Uso de um Array associativo para armazenar ingredientes e quantidades:

- Um array associativo em PHP é útil para armazenar informações relacionadas, como ingredientes e suas quantidades em uma receita.

- Exemplo:

```
```php
$ingredientes = array(
 "farinha" => "200g",
 "açúcar" => "100g",
 "ovos" => "2 unidades"
);
```
```

4. Vantagens de usar Arrays em programação web:

- Os arrays permitem armazenar dados de forma estruturada e acessá-los eficientemente por

meio de índices ou chaves.

- São úteis para manipulação e organização de dados em aplicativos web, como armazenar informações de formulários, resultados de consultas a bancos de dados, entre outros.

5. Papel das chaves em Arrays associativos:

- As chaves em arrays associativos são usadas para associar um valor específico a uma chave única.
- Elas são úteis na organização e recuperação de dados, pois permitem acessar rapidamente um valor específico usando uma chave conhecida.

6. Processo de modificação de um Array em PHP:

- Para modificar um array em PHP, você pode usar índices para acessar e alterar elementos existentes ou adicionar novos elementos.

- Exemplo de adição de um novo elemento:

```
```php
$frutas = array("Maçã", "Banana", "Laranja");
$frutas[] = "Abacaxi"; // Adiciona "Abacaxi" ao final do array
```
```

7. Diferença entre vetor e matriz:

- Um vetor é um array unidimensional que armazena uma lista de elementos.
- Uma matriz é um array multidimensional que armazena dados em linhas e colunas, formando uma grade ou tabela.

8. Índice:

- Um índice é um valor usado para acessar um elemento específico em um array.
- Em PHP, os índices podem ser numéricos (em arrays indexados) ou alfanuméricos (em arrays associativos).

9. Função do laço de repetição na programação:

- O laço de repetição é uma estrutura de controle que permite executar um bloco de código várias vezes.

- É usado para automatizar tarefas repetitivas e processar conjuntos de dados de forma eficiente.

10. Início da organização do vetor e da matriz:

- Em PHP, assim como na maioria das linguagens de programação, a organização de vetores e matrizes começa com o índice 0 para a primeira posição. No entanto, é importante notar que algumas linguagens de programação, como o MATLAB, podem começar a indexação a partir de 1.