

Estruturas de Controle em PHP

MBADSPHP - Tópico 2 / Aula 5

Objetivo

O objetivo desta aula é apresentar as estruturas de controle em PHP: de seleção ou condicional (*if* e *switch*), de repetição (*for*, *do...while*, *while*), e quebra de fluxo (*break* e *continue*).

Conteúdo

O comando de seleção *if* testa uma condição e caso seja verdade, executa uma ou várias linhas de código. Exemplo 1: Fique atento aos comentários no código.

```
<HTML>
<BODY>
<?php
    if ($teste > 10)
        echo "<h1> $teste";
    ?>
</BODY>
</HTML>
```

Exemplo 2: Fique atento aos comentários no código.

```
<HTML>
<BODY>
<?php
    if ($teste > 10) {
        echo "<h1> $teste";
        $teste = $teste + 10;
    } // quando o comando if executa mais de um
    comando, deve-se colocá-los entre chaves
    ?>
</BODY>
</HTML>
```

O comando de seleção *if* possui o comando opcional *else* que executa uma ou várias linhas de código caso a condição testada seja falso. Exemplo 1: Fique atento aos comentários no código.

```
<HTML>
<BODY>
<?php
    if ($teste > 10) {
        echo "<h1> $teste";
        $teste = $teste + 10;
    }
    else echo "<h2> $teste";
    ?>
</BODY>
</HTML>
```

Exemplo 2: Fique atento aos comentários no código.

```
<HTML>
<BODY>
<?php
    if ($teste > 10) {
        echo "<h1> $teste";
        $teste = $teste + 10;
    }
    else {
        echo "<h2> $teste";
        $teste = $teste + 20;
    }
    ?>
</BODY>
</HTML>
```

O comando de seleção *switch* trabalha da mesma forma que uma série de comandos *if* na mesma expressão. Exemplo: Fique atento aos comentários no código.

```
<HTML>
<BODY>
<?php
    switch ($teste) {
        case 0:
            echo "<h1> Igual a 0";
            break;
    }
```

```
case 1:
    echo "<h1> Igual a 1";
    break;
case 2:
    echo "<h1> Igual a 2";
    break;
    }
    ?>
</BODY>
</HTML>
```

Verifique que dentro de cada case, existe um comando *break* que não permite que se fique testando outras possibilidades, já que uma foi encontrada.

O comando de repetição *while* fica executando uma ou várias linhas de código enquanto a condição é satisfeita. Se a condição é falsa logo na primeira vez, nada é executado. Exemplo: Fique atento aos comentários no código.

```
<HTML>
<BODY>
<?php
    $var = 1;
    while ($var <= 10) {
        echo "<h1> Valor igual a $var ";
        $var = $var + 1;
    } // imprime o valor de $var dez vezes
    ?>
</BODY>
</HTML>
```

O comando de repetição *do ... while* fica executando uma ou várias linhas de código enquanto a condição é satisfeita. Se a condição é falsa logo na primeira vez, é executado o(s) comando(s) pelo menos uma vez. Exemplo: Fique atento aos comentários no código.

```
<HTML>
<BODY>
<?php
    $var = 1;
    do {
        echo "<h1> Valor igual a $var ";
        $var = $var + 1;
    }
    while ($var <= 10); // imprime o valor de
    $var dez vezes
    ?>
</BODY>
</HTML>
```

O comando de repetição *for* executa uma ou várias linhas de código enquanto a condição é satisfeita. Perceba no exemplo a seguir que o comando *for* possui a sintaxe *for*(<inicialização>, <condição>, <incremento/ decremento>). O comando de inicialização é executado apenas uma vez logo no começo. O comando de condição é executado ao final de todas as execuções das linhas de código. O comando incremento/decremento é executado ao final de toda verificação do comando de condição. Fique atento aos comentários no código.

```
<HTML>
<BODY>
<?php
    for ($var = 1; $var <= 10; $var = $var + 1){
        echo "<h1> Valor igual a $var ";
        // imprime o valor de $var dez vezes
    }
    ?>
</BODY>
</HTML>
```

A quebra de fluxo de uma estrutura de repetição pode ser feita com os comandos *continue* ou *break*.

Quando se executa o comando *break* dentro de uma estrutura de repetição, simplesmente a repetição acaba e vai-se para a próxima linha de instrução após a estrutura de repetição no programa. Por outro lado, quando se executa o comando *continue* dentro de uma estrutura de repetição, simplesmente o laço termina nesse comando e vai-se para o próximo laço.

Sendo assim, o comando *continue* não sai da estrutura, apenas passa para o próximo comando. Exemplo 1: Fique atento aos comentários no código.

```
<HTML>
<BODY>
  <?php
    for($var = 1; $var <= 10; $var = $var + 1){
      if ($var == 5)
        break;
      else echo "<h1> Valor igual a $var ";
      // imprime o valor de $var 4 vezes (1,2,3,4)
    }
  ?>
</BODY>
</HTML>
```

Exemplo 2: Fique atento aos comentários no código.

```
<HTML>
<BODY>
  <?php
    for ($var = 1; $var <= 10; $var = $var + 1){
      if ($var == 5)
        continue;
      else echo "<h1> Valor igual a $var ";
//imprime o valor de $var 9 vezes (1,2,3,4,6,7,8,9,10)
    }
  ?>
</BODY>
</HTML>
```

Exercícios

Faça um programa PHP que contenha todos os recursos aprendidos neste Tópico. Sempre apresente o resultado ao usuário com o comando `echo ""`. Não se esqueça de colocar o arquivo PHP no diretório `htdocs` do Apache HTTP Server.