

Operadores em PHP

Aritméticos

+	Adição
-	Subtração
*	Multiplicação
/	Divisão
%	Módulo (Resto da divisão)

De Strings

.	concatenação
---	--------------

Incremento

++	Incrementa
--	Decrementa

De Atribuição

=	Atribuição simples
+=	com adição
-=	com subtração
*=	com multiplicação
/=	com divisão
%=	com módulo
.=	com concatenação

Ex:

```
$a = 7;
```

```
$a += 2; // $a=9
```

Igual a

```
$a = $a + 2;
```

Comparação

==	Igual a
!=	Diferente de
<	Menor que
>	Maior que
<=	Menor ou igual que
>=	Maior ou igual que

Lógicos

AND &&	"e" lógico
Or 	"ou" lógico
!	"inversao"
XOR	"ou" exclusivo

Operadores em PHP

► Exemplos:

```
<?php
    $a = 10;
    $b = 20;
    $c = 30;
    echo "a+b = " . ($a+$b) ;
    $d = $c/$a;
    echo "<br/>c/a = " . $d;
?>
```

Operador ternário

- ▶ Forma abreviada de utilização do comando *if*
 - ▶ \$variavel = <cond> ? <expCasoVerdadeiro> : <expCasoFalso>;

```
<?php
$idade = 19;
#condição ? exp1 : exp2
$res = $idade >= 18 ? "Maioridade" : "Menoridade";
echo $res;
?>
```

Estrutura condicional: if..else

```
if (condicao1 é verdadeira){  
    #Bloco 1 de comandos  
}  
elseif (condicao2 é verdadeira){  
    #Bloco 2 de comandos  
}  
#condicao1 e condicao2 são falsas  
else{  
    #Bloco 3 de comandos  
}
```

if..else - exemplo

```
<?php
    $idade = 17;

    if ($idade >= 18)
        print "Maioridade";
    elseif ($idade < 12) {
        echo "Criança";
    }
    else{
        print "Adolescente";
    }
?>
```

Estrutura condicional: switch

```
switch(variavelAvaliada)
{
    case 1:
        <comandos>
        break;
    case 2:
        <comandos>
        break;
    default:
        <variavelAvaliada é diferente de 1 e de 2>
}
```

switch - Exemplo

```
<?php
    $dia = 1;

    switch($dia){
        case 1:
            echo "domingo";
            break;
        case 2:
            echo "segunda";
            break;
        default:
            echo "dia inválido";
    }
?>
```

while

- ▶ É composta por uma expressão e um bloco de comandos
- ▶ Enquanto a expressão for verdadeira, o bloco de comandos é executado

```
while (condicao_VERDADEIRA) {  
    <executa comandos>  
}
```


while - exemplo

```
<?php
/*Exemplo 1*/
$i = 1;
while ($i <= 10) {
    echo "i = ".$i++."<br>";
}

/*Exemplo 2 */
$j = 1;
while ($j <= 10) :
    echo "j = ".$j."<br>";
    $j++;
endwhile;
?>
```

do...while

- ▶ Avalia a expressão no final do laço, ou seja, o laço será executado pelo menos uma vez

```
do {  
    <executa comandos>  
}  
while (condicao_TRUE) ;
```

do...while

- ▶ Avalia a expressão no final do laço, ou seja, o laço será executado pelo menos uma vez

```
do {  
    <executa comandos>  
}  
while (condicao_TRUE) ;
```

do...while - exemplo

```
<?php
    $num = 1;
    do {
        echo "num = $num"."<br>";
        $num++;
    }
    while ($num <= 5) ;
?>
```

for

- ▶ Permite a execução de instruções num determinado número de vezes

```
for (inicialização; condição; oper){  
    executa comandos;  
}
```

for - exemplos

```
<?php
    for ($c=2; $c<=10; $c+=2) {
        echo "c = $c <br>";
    }
?>
```

break

- ▶ Termina a execução de um **if**, **switch**, **for**, **while** ou **do..while**
 - ▶ Quando encontrado dentro de algumas dessas estruturas, o fluxo de execução passa para o primeiro comando após o término da estrutura de controle
 - ▶ **break** aceita um argumento numérico opcional que diz o número de estruturas que devem ser finalizadas. Por exemplo **break 3**;

break - Exemplo 1

```
<?php
    echo "***** Exemplo 1 ***** <br>";
    $i = 1;
    $k = 1;
    while($k <= 10){
        while ($i <= 10){
            if ($i == 5){
                echo "Aborta o laço interno -> k = $k<br>";
                break;
            }
            $i++;
        }
        $k++;
        echo "Último comando do laço externo.<br><br>";
    }
?>
```


break - Exemplo 2

```
<?php
    echo "<br><br>***** Exemplo 2 ***** <br>";
    $i = 1;
    $k = 1;
    while($k < 10){
        while ($i < 10){
            if ($i == 5){
                echo "Aborta 2 laços. <br>";
                break 2;
            }
            $i++;
        }
        echo "i = $i <br> k = $k <br>";
        $k++;
    }
    echo "Comando seguinte ao final do laço externo.<br>";
?>
```

Gerando HTML dinamicamente

- Números de 1 a 10 dentro de uma tabela:


```
<html>
<head>
  <style>
    table{
      border-collapse: collapse;
    }
    td{
      border: 1px solid black;
    }
  </style>
</head>
<body>
  <?php

  echo "<table><tr>";
  for ($i=1; $i<=10; $i++){
    echo "<td>$i</td>";
  }
  echo "</tr></table>";
  ?>
</body>
</html>
```

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Exercícios

► Algoritmos em PHP

- 1) Escreva os números pares de 1 e 10. Coloque os valores em uma coluna de tabela. 
- 2) Escreva apenas os números entre 1 e 100 que são múltiplos de 3. Alterne a cor da plano de fundo da tabela entre verde e amarelo.

3	6	9	12	15	...				
---	---	---	----	----	-----	--	--	--	--

1
2
3
4
5
6
7
8
9
10

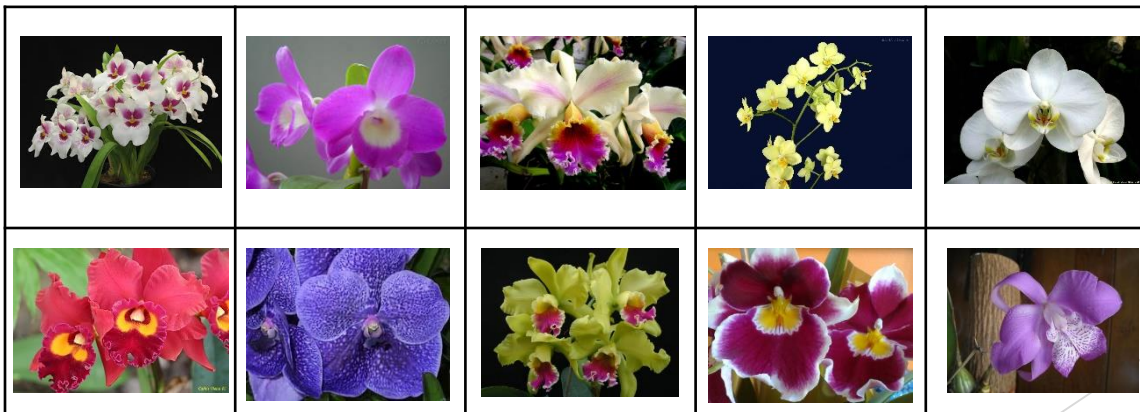
Exercícios

- 3) Crie uma tabela para representar um mês do ano. Coloque os valores de 1 a 30 dentro da tabela, tal como mostrado abaixo:

D	S	T	Q	Q	S	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

Exercícios

- 4) Crie uma galeria de imagens dinamicamente em PHP. Considere 10 imagens com o nome **foto##.jpg**, onde o **##** deve ser substituído pelo número da foto. As fotos devem ser numeradas de 01 a 10. Apresente as imagens dentro de uma tabela com 2 linhas e 5 colunas. Crie um CSS de modo a definir um tamanho padrão para todas as imagens.



Referências

- ▶ Fábio Procópio; João Maria. Estruturas de Controle. Notas de Aula. Disponível em: <https://sites.google.com/site/fabiooprocopio/programacao-web>
- ▶ Pedro Baesse. Linguagens de Script e PHP. Notas de Aula. Disponível em: <http://docente.ifrn.edu.br/pedrobaesse>
- ▶ CITI. Curso PHP - Aula 01. Disponível em: http://www.cin.ufpe.br/~vbom/arquivos/PHP/PHP%20-%20Aula_01.pdf