

### Reconhecida pela Portaria nº 821/MEC – D.O. U de 01.06.94

## COORDENAÇÃO UVA / EVOLUTIVO

**DISCIPLINA:** CRIAÇÃO DE SITES NA WEB II

**PROFESSOR (A):** JOÃO CLÁUDIO

**AULA 2 (24/01/2008)**

Vamos a nosso primeiro programa php, um programa simples que escreve a palavra TESTE no browser, chamaremos este de teste.php e deve ser colocado no diretório public\_html no diretório (C:\Arquivos de programas\EasyPHP1-8\www\exercicios):

**Exercício 1.** Primeiro programa em PHP

<?

echo "<html>";

echo "<title>Título</title>";

echo "<body>";

echo "<br>TESTE";

echo "</body>";

echo "</html>";

?>

RESULTADO🡪 http://127.0.0.1:81/exercicios/teste.php

**Exercício 2.** Escreva 3 variáveis e exiba cada uma em 1 linha.

**Exercício 3.** Como a checagem de tipos em PHP é dinâmica, valores de tipos diferentes podem ser usados como índices de array, assim como os valores mapeados também podem ser de diversos tipos.

<?

$cor[1] = “vermelho”;

$cor[2] = “verde”;

$cor[3] = “azul”;

$cor[“teste”] = 1;

?>

Equivalentemente, pode-se escrever:

<?

$cor = array(1 => “vermelho, 2 => “verde, 3 => “azul”, “teste => 1);

?>

**Exercício 4.** O que o script abaixo irá mostrar?  
$frase = "O PHP é uma linguagem";  
function mostraValor() {  
global $frase;  
$frase .= " de programação";  
echo $frase;  
}  
mostraValor();

**Exercício 5.**Qual o valor de N?

$a = 1;  
$a++;  
$a = N

**Exercício 6.**Como iremos notar, o PHP converte string para integer ou double mantendo o valor. O sistema utilizado pelo PHP para converter de strings para números é o seguinte:

− É analisado o início da string. Se contiver um número, ele será avaliado. Senão, o valor será 0

(zero);

− O número pode conter um sinal no início (“+” ou “-“);

− Se a string contiver um ponto em sua parte numérica a ser analisada, ele será considerado, e o valor obtido será double;

− Se a string contiver um “e” ou “E” em sua parte numérica a ser analisada, o valor seguinte será considerado como expoente da base 10, e o valor obtido será double;

Exemplos:

$vivas = 1 + “10.5”; // $vivas == 11.5

$vivas = 1 + “-1.3e3”; // $vivas == -1299

$vivas = 1 + “teste10.5”; // $vivas == 1

$vivas = 1 + “10testes”; // $vivas == 11

$vivas = 1 + " 10testes"; // $vivas == 11

$vivas = 1 + "+ 10testes"; // $vivas == 1

Faça os testes acima.

**Exercício 7.**Exemplos de uso de arrays com PHP: Arrays multidimensionais, ordenação, busca, delete de itens, etc.

<?php

// Criando um Array   
$colorList = array("apple"=>"red",   
"grass"=>"green",   
"sky"=>"blue",   
"night"=>"black",   
"wall"=>"white");

// Mostrando um item do Array   
echo "O ceu esta ".$colorList["sky"]   
." e a grama está ".$colorList["grass"];

// Mostrando o tamanho do Array   
echo "<br/>O tamanho do Array é: ".sizeof($colorList);

// Remove um elemento do array   
unset($colorList["sky"]);   
echo "<br/>Novo tamanho do Array: ".sizeof($colorList);   
echo "<br>";   
  
// Checa a existência de um elemento   
if (isset($colorList["grass"])) echo "chave grass encontrado";   
else echo "chave grass não encontrado";   
if (isset($colorList["sky"])) echo "Chave sky encontrada";   
else echo "Chave key não encontrada";   
  
// Mostra o conteudo completo do Array   
echo "Array antes da ordenação:<br/>";   
print\_r($colorList);   
  
// Mostra o conteudo completo do Array após a ordenação   
echo "Array depois da ordenação:<br/>";   
print\_r (asort($colorList));   
echo "Array depois de ordenar:<br/>";   
print\_r (sort($colorList));   
  
// Criando uma array multidimensional   
$myLists['colors'] = array("apple"=>"red",   
"grass"=>"green",   
"sky"=>"blue",   
"night"=>"black",   
"wall"=>"white");   
  
$myLists['cars'] = array("BMW"=>"M6",   
"Mercedes"=>"E 270 CDI",   
"Lexus"=>"IS 220d",   
"Mazda"=>"6",   
"Toyota"=>"Avensis");   
  
// Mostrando um item do array   
echo "<br />O item demo é:".$myLists['cars']['Toyota'];   
  
// Criando um novo array   
$colorList2[] = "red";   
$colorList2[] = "green";   
$colorList2[] = "blue";   
$colorList2[] = "black";   
$colorList2[] = "white";   
  
// DUmp no conteudo   
echo "<br/>Dump colorList2 com print\_r";   
print\_r($colorList2);   
echo "<br/>Dump colorList2 com var\_dump";   
var\_dump($colorList2);   
echo "<br/><br/><br/>";   
  
// Mostrando elemntos do array com looping   
echo "Conteudo do array:<br/>";   
  
for ($i=0;$i<=4;$i++){   
echo $colorList2[$i]."<br/>";   
}   
  
// Mostrando so elementos do array com foreach   
echo "Conteudo do array:<br/>";   
foreach ($colorList2 as $value) {   
echo $value."<br/>";   
}   
?>

**OBSERVAÇÕES:**

Separador de instruções : Entre cada instrução em PHP é preciso utilizar o ponto-e-vírgula, assim como em C, Perl e outras linguagens mais conhecidas. Na última instrução do bloco de script não é necessário o uso do ponto-e-vírgula, mas por questões estéticas recomenda-se o uso sempre.