

Instituto Federal Sul-rio-grandense – Campus Jaguarão
Disciplina: Programação para Internet II – Turma 2022
Assunto: Lista de exercícios de revisão de PHP
Professor: Vinícius Alves Hax

Exercício 1:

Observe o código abaixo em PHP

```
<?php
$arr = array(1, 20, 3, 4, 55);

foreach($arr as $valor) {
    if ($valor < 5)
        echo "<br />" . $valor;
}

?>
```

O que será exibido na tela?

Resposta:

Serão exibidos os elementos do vetor (separados por uma quebra de linha)

1
20
3
4
55

Exercício 2:

Implemente uma função em PHP que receba os coeficientes de uma equação de segundo grau, e retorne 1 se a concavidade da mesma é volta para cima, ou -1 se a concavidade da mesma é voltada para baixo.

Exemplo de uso da função:

```
// Suponha que os valores abaixo se refiram a equação  $2x^2 + 20x - 30 = 0$ 
$resultado = concavidade(2, 20, -30);
echo $resultado;
```

```
// Deverá mostrar, nesse caso, 1
```

// Resposta

```
function concavidade($a, $b, $c) {
```

```
// Na verdade a concavidade de uma função de segundo grau é definida somente pelo  
parâmetro $a
```

```
if ($a > 0)
```

```
    // Concavidade para cima  
    return 1;
```

```
else
```

```
    // Concavidade para baixo
```

```
        return -1;
    }
```

Exercício 3:

Faça uma função em PHP que retorne um array com todos os divisores de um número

```
$divisores = calcula_divisores(20);
```

//Deve retornar um array com os valores 10, 5, 2 e 1. Os divisores não precisam estar nessa ordem necessariamente

Dica: Consulte a documentação da função `array_push`, que adiciona um elemento em um array. Para isso dentro do site `php.net` faça uma busca por `array_push`

//Resposta:

```
function calcula_divisores($numero) {
    $divisores = array();
    // Percorre todos os possíveis divisores começando em 1 e indo até o próprio número
    for($i = 1; $i <= $numero; $i++)
    {
        // Se o resto da divisão for zero então é um divisor
        if (($numero % $i) == 0) {
            // Não foi pedido mas apenas para fins didáticos vamos mostrar os divisores
            echo "<br />$i é um divisor";
            array_push($divisores, $i);
        }
    }
    return $divisores;
}
```

Exercício 4:

Faça uma função em PHP que diga se um número é primo ou não

Exemplo:

```
testa_primo(40);
// Mostra "Não é primo"
```

```
testa_primo(7);
// Mostra "É primo"
```

Resposta:

```
//Reutilizamos a mesma função do exercício anterior
function calcula_divisores($numero) {
    $divisores = array();
    // Percorre todos os possíveis divisores começando em 1 e indo até o próprio número
    for($i = 1; $i <= $numero; $i++)
    {
        // Se o resto da divisão for zero então é um divisor
        if (($numero % $i) == 0) {
            // Não foi pedido mas apenas para fins didáticos podemos mostrar os divisores
```

```

        // Como já mostramos anteriormente vamos comentar aqui para mostrar menos informações
        //echo "<br />$i é um divisor";
        array_push($divisores, $i);
    }
}

return $divisores;
}

function testa_primo($possivel_primo) {
    $divisores = calcula_divisores($possivel_primo);
    $numero_divisores = count($divisores);
    // Um número primo tem dois divisores: 1 e ele mesmo
    if ($numero_divisores == 2)
        echo $possivel_primo . " é primo <br />";
    else
        echo $possivel_primo . " não é primo <br />";
}

testa_primo(12);
testa_primo(7);
testa_primo(20);
testa_primo(11);

```

Exercício 5:

Faça um programa em PHP que receba como parâmetro um array e que retorne um outro array. O array retornado deve contar somente os valores cujas chaves são maiores que zero. Por exemplo se

\$a = array(-1 => “Esse não”, -5 => “Esse também não”, 10 => “Esse sim”, 20 => “Este também”)

ao usarmos a função (por exemplo com o nome filtra)

\$b = filtra(\$a);

deve retornar o array abaixo

\$a = array(10 => “Esse sim”, 20 => “Este também”)

//Resposta

```

function filtra($array_entrada) {
    $array_saida = array();
    foreach($array_entrada as $chave => $valor) {
        if ($chave > 0) {
            $array_saida[$chave] = $valor;
        }
    }

    return $array_saida;
}

```

```
$a = array(-1 => "Esse não", -5 => "Esse também não", 10 => "Esse sim", 20 => "Este também");
```

```
$b = filtra($a);
```

```
print_r($b);
```