

# *Programação Orientada a Objetos com PHP*

```
...mirror_mod.mirror_object  
operation == "MIRROR_X":  
mirror_mod.use_x = True  
mirror_mod.use_y = False  
mirror_mod.use_z = False  
operation == "MIRROR_Y":  
mirror_mod.use_x = False  
mirror_mod.use_y = True  
mirror_mod.use_z = False  
operation == "MIRROR_Z":  
mirror_mod.use_x = False  
mirror_mod.use_y = False  
mirror_mod.use_z = True
```

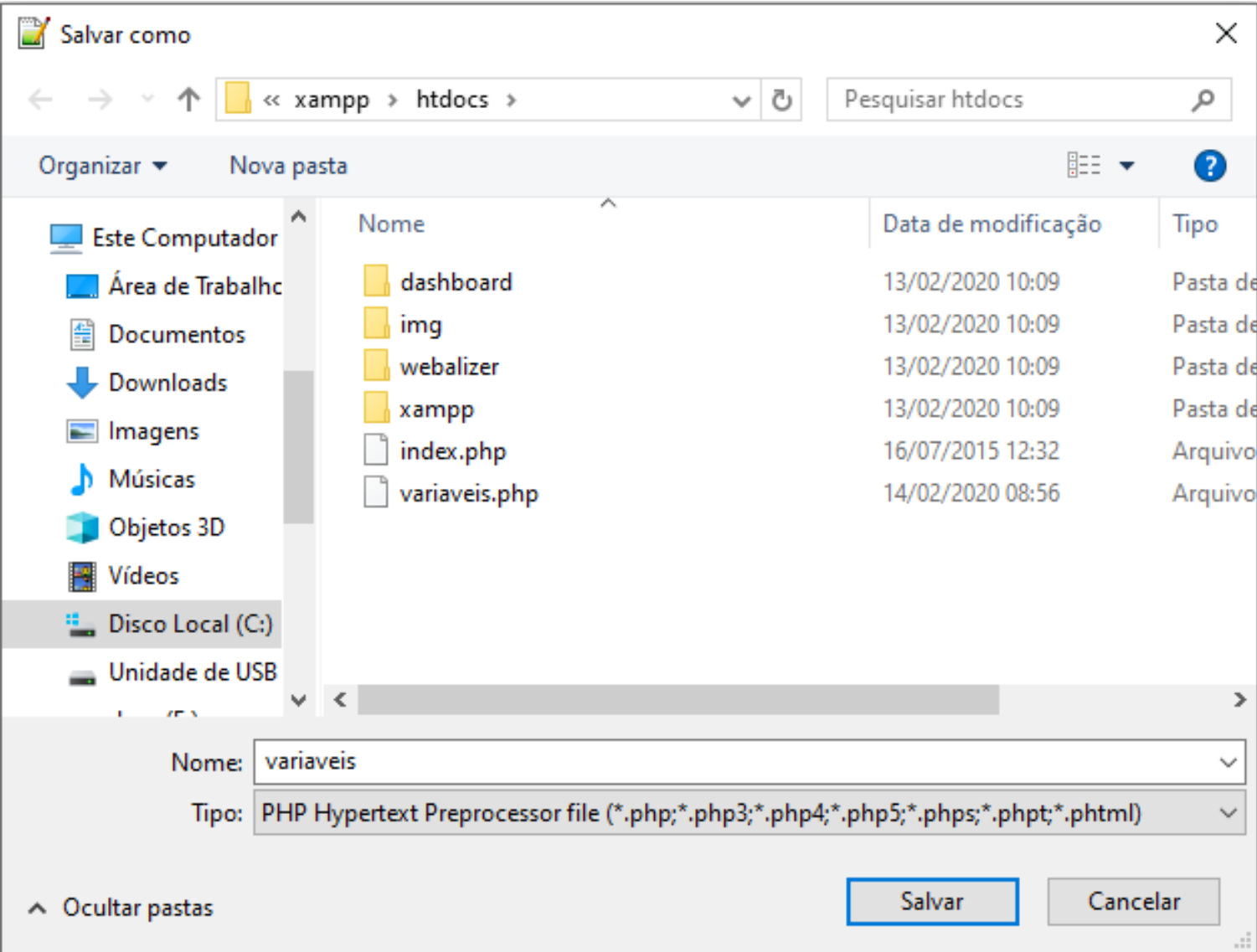
```
...selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
("Selected" + str(modifier_ob.name))  
mirror_ob.select = 0  
= bpy.context.selected_object  
data.objects[one.name].select  
print("please select exactly one mirror")
```

--- OPERATOR CLASSES ---

```
...types.Operator):  
X mirror to the selected  
object.mirror_mirror_x"  
mirror X"
```

*profº Mauricio Conceição Mario*

Abrir editor de texto (notepad++), salvar arquivo com extensão .php no diretório htdocs do XAMPP; editar o código a seguir.



## *Manipulação de variáveis: strings, variáveis variantes e variável atribuída a outra.*

```
variaveis.php x
1 <?php
2 /*atribuição de uma variável à outra:
3 as duas variáveis apontam para a
4 mesma região de memória*/
5 $a = 5;
6 $b = $a;
7 //echo: imprime no console
8 echo $b;
9
10 /*conceito de variáveis variantes:
11 variáveis que podem mudar de acordo com a necessidade ->
12 usam-se dois sinais "$$" precedendo o nome de uma variável
13 para o PHP referenciar uma variável representada pelo conteúdo da
14 primeira.*//
15
16 //nome da variável = c
17 $c = 'nome';
18
19 //criação de variável identificada pelo conteúdo da variável c
20 $$c = 'eliana';
21
22 //resultado da referência a $nome = eliana:
23 /*print: imprime uma string no console
24 -> nl2br indica que \n é para pular de linha
25 e "." concatena pular de linha + variável $nome*/
26 print nl2br("\n".$nome);
27 ?>
```

 xampp\_start

 xampp\_stop

No diretório do XAMPP utilizar os comandos start e stop para iniciar e desativar o servidor.

localhost/variaveis.php x

localhost/variaveis.php

5  
eliana

## Conditional if:

```
exemplo_2.php x  testa_função.php x  condicionais.php x
1  <?php
2  $salario = 800.70;
3  $tempo_servico = 25;
4  $folga_fim_semana = false; //variável booleana
5
6  if (($tempo_servico >= 20)and($salario > 1000.88)and($folga_fim_semana != true))
7  {print_r("dados errados do funcionário: ");}
8  else if (($tempo_servico == 25)and($salario == 800.70)and($folga_fim_semana != true))
9  {print_r("dados corretos do funcionário: ");}
10 print nl2br("\ntempo de serviço: ".$tempo_servico."\nsalário R$ = ".$salario."\nfolga sábado e domingo: não ".$folga_fim_semana);
11 ?>
```

← → ↻ ⓘ localhost/condicionais.php

dados corretos do funcionário:  
tempo de serviço: 25  
salário R\$ = 800.7  
folga sábado e domingo: não

## *Condiciona while:*

```
1  <?php
2  $a = 1;
3  while ($a <= 10) {
4      print nl2br($a."\n");
5      $a++;
6  }
7  ?>
```

← → ↻ ⓘ localhost/enquanto.php

1  
2  
3  
4  
5  
6  
7  
8  
9  
10

*foreach: laço de repetição para arrays ou matrizes*

```
1  <?php
2  $frutas = array("maçã", "laranja", "mamão", "banana");
3  foreach ($frutas as $valor)
4  print "$valor\t";
5  ?>
6
```

← → ↻ ⓘ localhost/arranjos.php

maçã laranja mamão banana

C:\xampp\htdocs\exemplo\_2.php - Notepad++

Arquivo Editar Localizar Visualizar Formatar Linguagem C

exemplo\_2.php x testa\_função.php x

```
1 <?php
2 /*
3  *função quadrado
4  *retorna o quadrado
5  *de um número
6  */
7
8 function quadrado($numero)
9 {
10     return $numero * $numero;
11 }
12 ?>
```

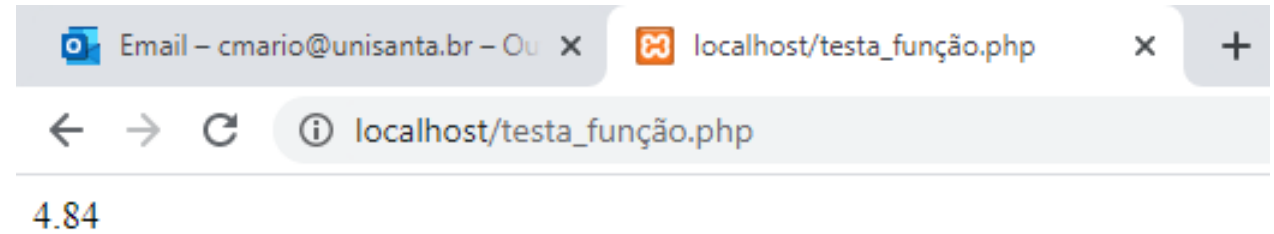
C:\xampp\htdocs\testa\_função.php - Notepad++

Arquivo Editar Localizar Visualizar Formatar Linguag

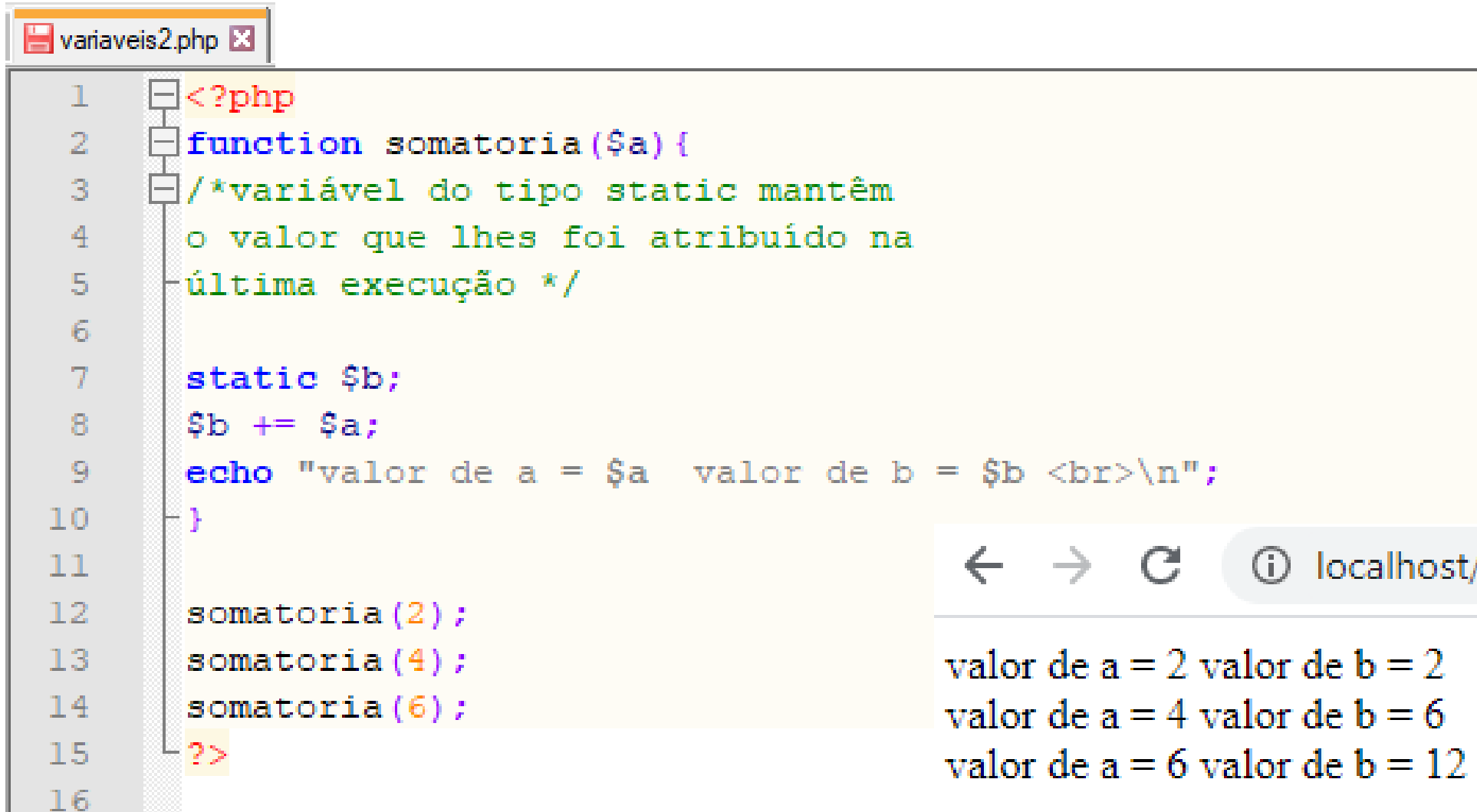
exemplo\_2.php x testa\_função.php x

```
1 <?php
2 //carrega biblioteca
3 include 'exemplo_2.php';
4
5 //chama função quadrado()
6
7 echo quadrado(2.2);
8 ?>
9
10
```

*Uso de função:*



## Uso de função e variável de forma estática:



The image shows a web browser window with a single tab titled 'variaveis2.php'. The browser's address bar shows 'localhost/variaveis2.php'. The main content area displays the output of a PHP script, which consists of three lines of text: 'valor de a = 2 valor de b = 2', 'valor de a = 4 valor de b = 6', and 'valor de a = 6 valor de b = 12'. The text is displayed in a monospaced font with color coding: 'valor de a' is blue, 'valor de b' is red, and the numbers are black. The background of the browser window is light gray.

```
1 <?php
2 function somatoria($a){
3     /*variável do tipo static mantêm
4     o valor que lhe foi atribuído na
5     última execução */
6
7     static $b;
8     $b += $a;
9     echo "valor de a = $a  valor de b = $b <br>\n";
10 }
11
12 somatoria(2);
13 somatoria(4);
14 somatoria(6);
15 ?>
```



## Exercícios propostos (sugestão de consultar funcionalidades do PHP):

1. Escrever código que possua uma função que receba como argumentos o nome e o salário de funcionários de uma empresa. A aplicação deve imprimir em cada linha o nome e o respectivo salário dos funcionários (pelo menos 3).
2. Escrever outra versão do código proposto acima, sendo que tanto os nomes quanto os salários estejam originalmente em arranjos.
3. Dada uma relação de 10 pares funcionário – salário, escrever versão do código em que sejam impressos os pares onde o salário seja maior que R\$1000.00.

## Referências Bibliograficas

- *php* Programando com Orientação a Objetos  
Pablo Dall'Oglio - editora Novatec - 2014
- Desenvolvimento web com PHP e MySQL  
Evaldo Junior Bento – editora Casa do Código -  
2018