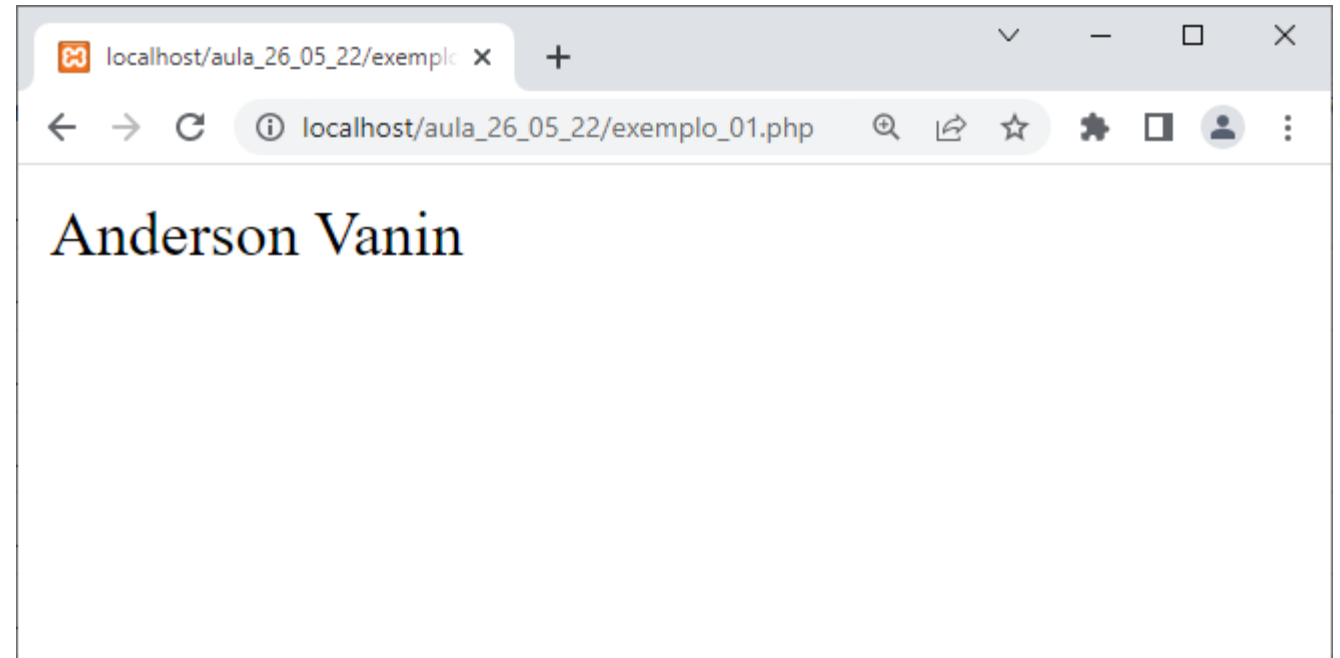


PHP – Funções

Prof. Anderson Vanin

Função sem Retorno e sem Parâmetros

```
exemplo_01.php
1  <?php
2      function mostraNome(){
3          echo "Anderson Vanin";
4      }
5
6      mostraNome();
7  ?>
```

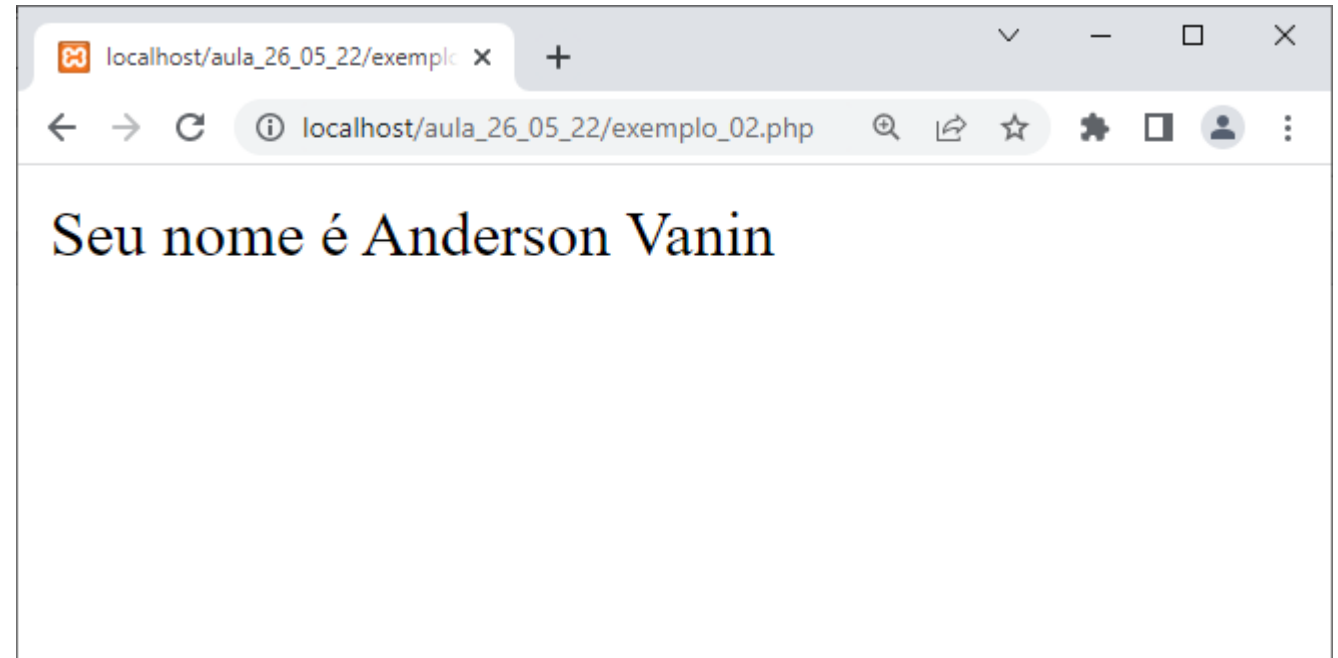


Para executar uma função, a mesma deve ser chamada.

Função sem Retorno e com Parâmetros

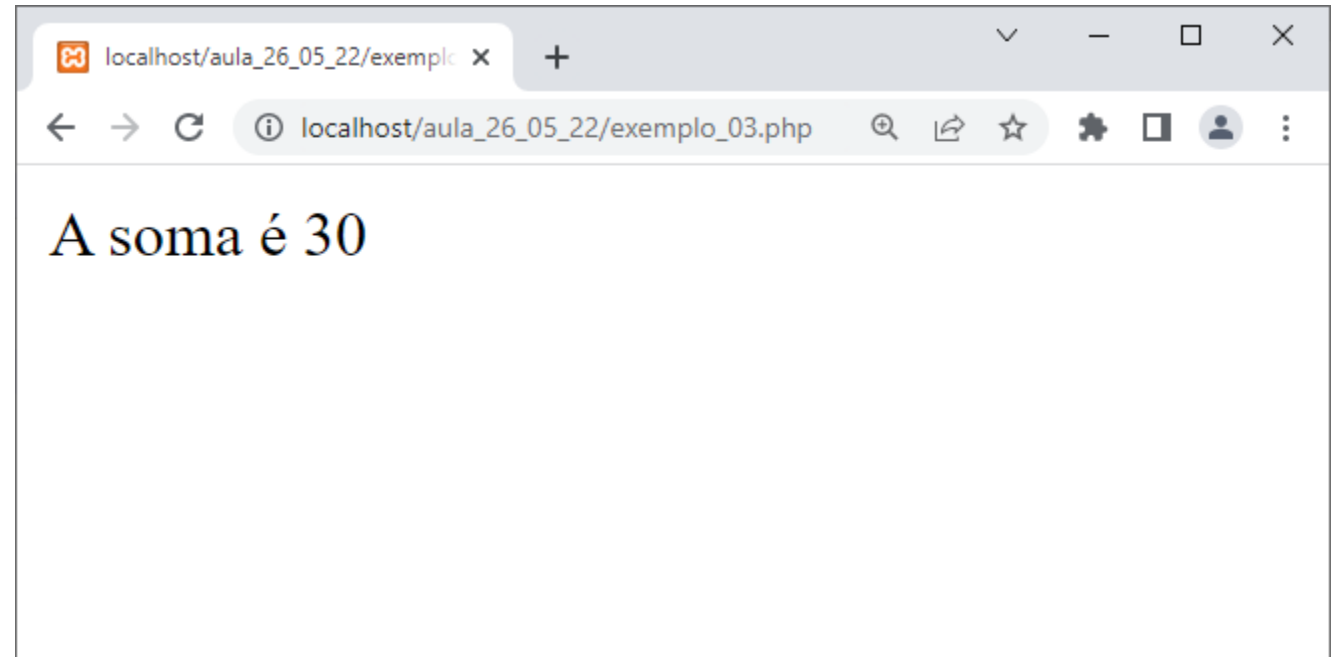
exemplo_02.php

```
1  <?php
2      $nome = "Anderson Vanin";
3
4      function mostraNome($a){
5          |   echo "Seu nome é $a";
6      }
7
8      mostraNome($nome);
9  ?>
```



Função sem Retorno e com Parâmetros

```
exemplo_03.php
1 <?php
2     $n1 = 10;
3     $n2 = 20;
4
5     function soma($a,$b){
6         $soma = $a + $b;
7         echo "A soma é $soma";
8     }
9
10    soma($n1,$n2);
11 ?>
```

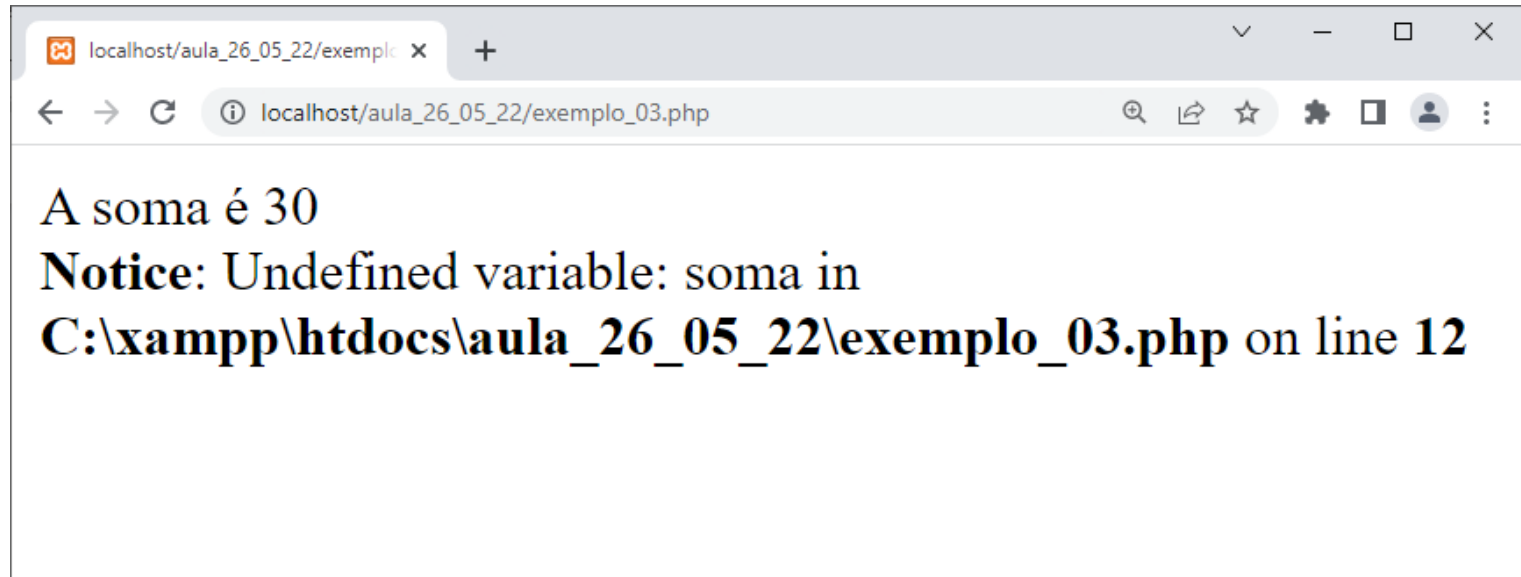


Um detalhe é que neste exemplo a variável \$soma só é acessada dentro da função. Fora da função, não é possível utilizá-la!

Função sem Retorno e com Parâmetros

exemplo_03.php

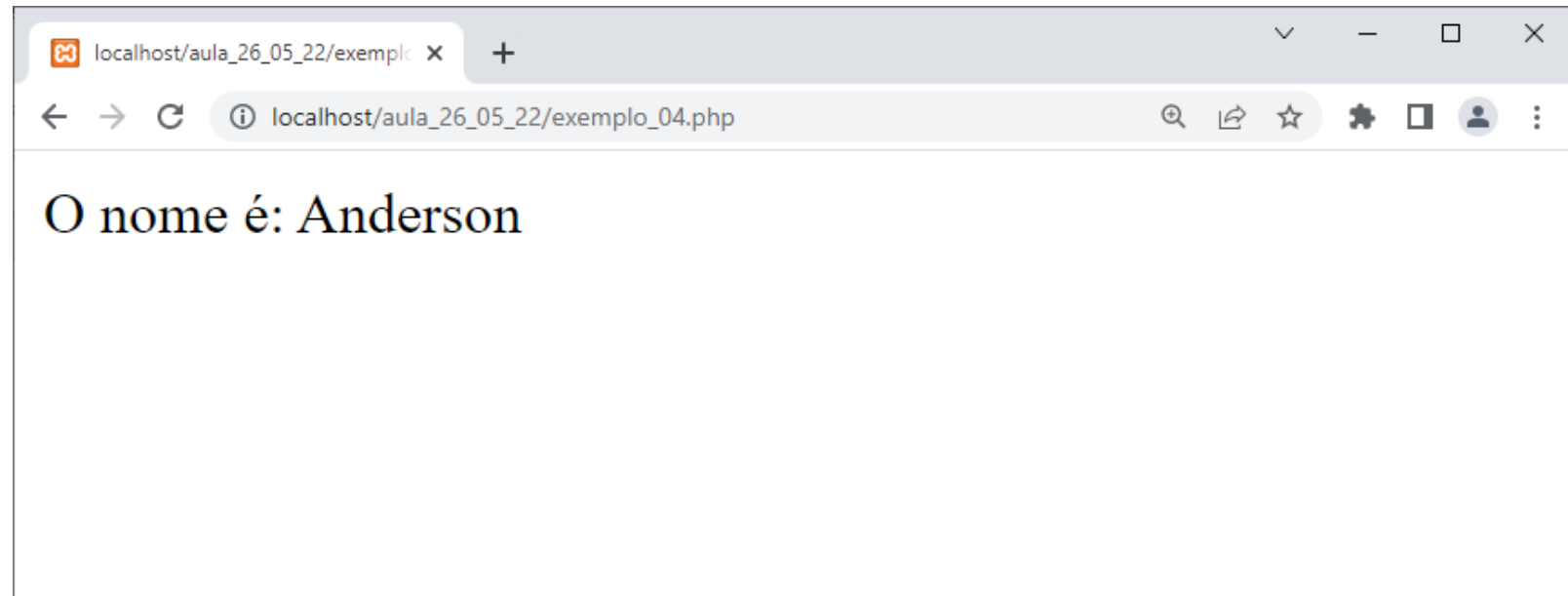
```
1  <?php
2      $n1 = 10;
3      $n2 = 20;
4
5      function soma($a,$b){
6          $soma = $a + $b;
7          echo "A soma é $soma";
8      }
9
10     soma($n1,$n2);
11     // a operação abaixo resulta um erro,
12     // pois $soma está dentro da função!
13     echo $soma;
14  ?>
```



Função com Retorno e sem Parâmetros

exemplo_04.php

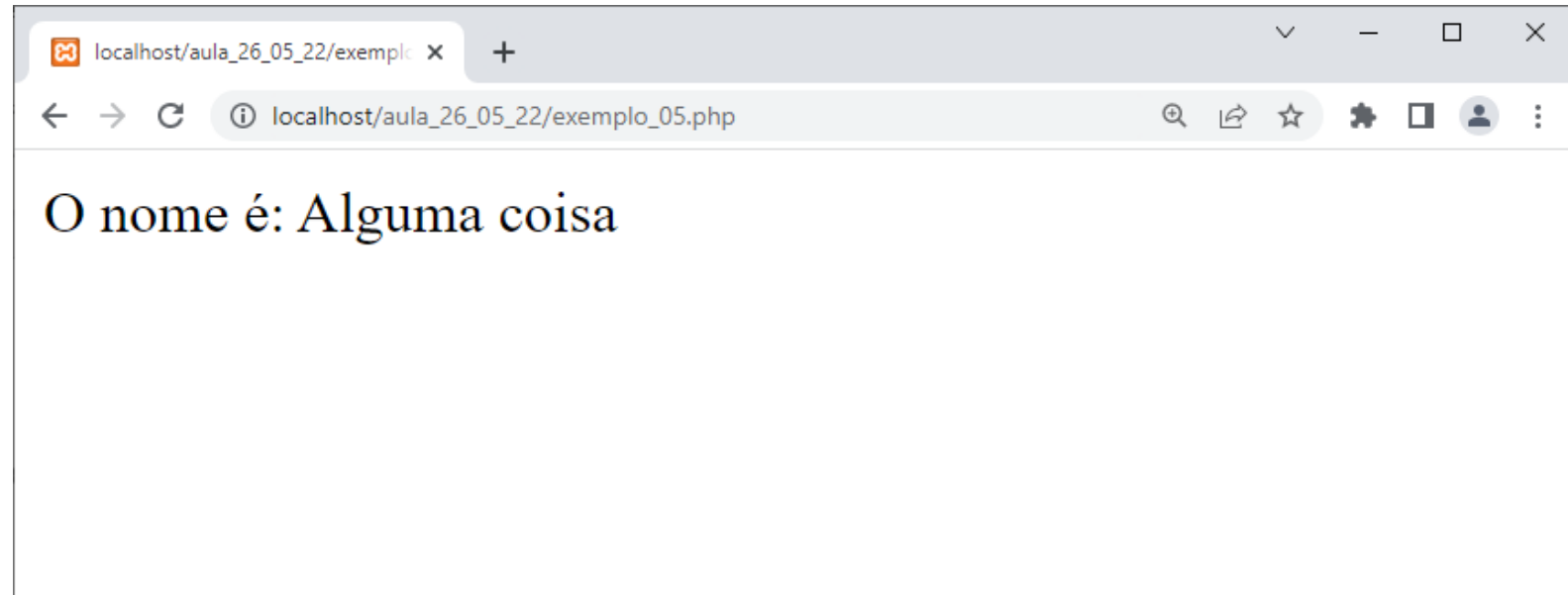
```
1  <?php
2
3      function nome(){
4          $n = "Anderson";
5          return $n;
6      }
7
8      $nome = nome();
9      echo "O nome é: $nome";
10 ?>
```



Função com Retorno e com Parâmetros

exemplo_05.php

```
1  <?php
2
3      function nome($abc){
4          $n = $abc;
5          return $n;
6      }
7
8      $nome = nome('Alguma coisa');
9      echo "O nome é: $nome";
10 ?>
```

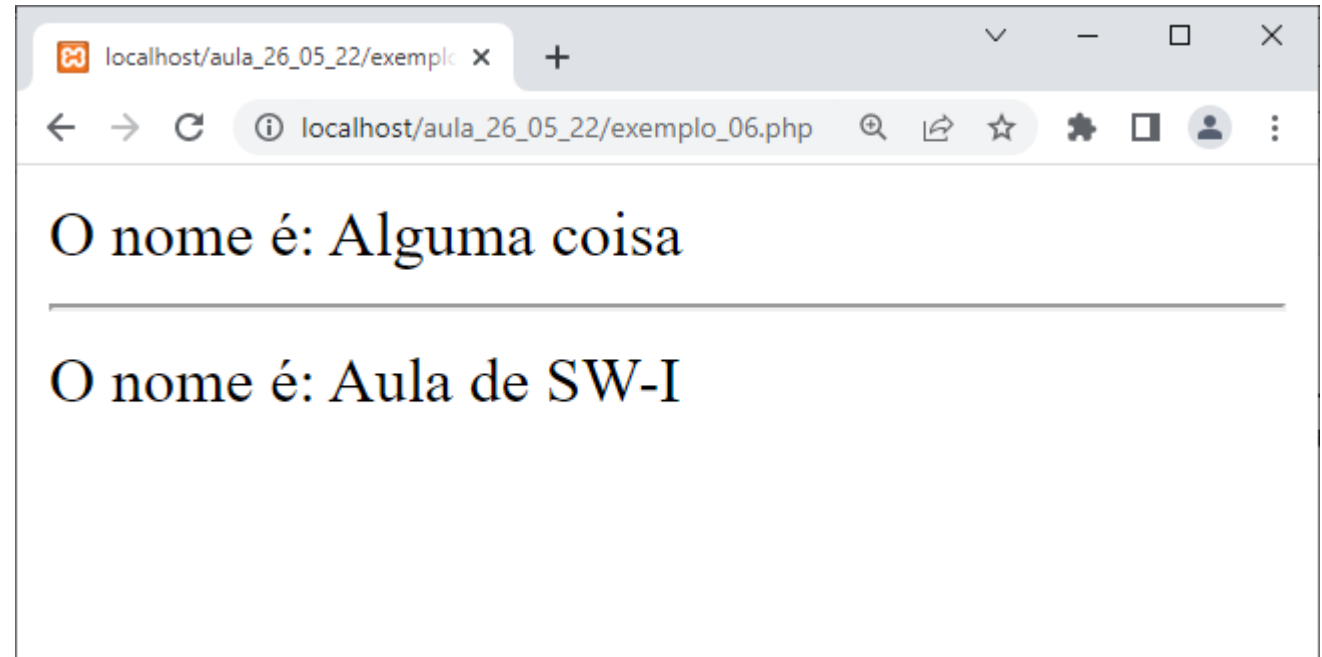


Uma das principais vantagens no uso de funções, é a sua reutilização em diferentes partes do programa.

Exemplo reutilização de funções

exemplo_06.php

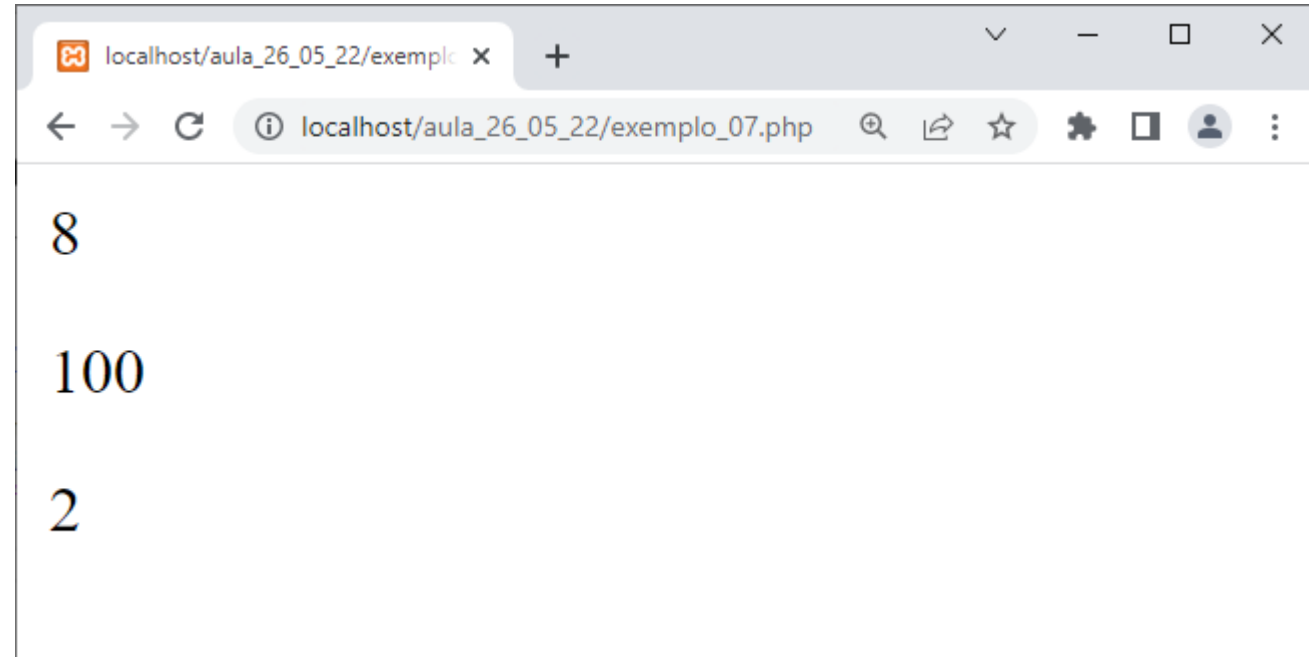
```
1  <?php
2
3  function nome($abc){
4      $n = $abc;
5      return $n;
6  }
7
8  $nome = nome('Alguma coisa');
9  echo "O nome é: $nome";
10
11 echo "<hr>";
12
13 // reutilizando a mesma função com valor
14 // diferente no parâmetro
15
16 $outro_nome = nome('Aula de SW-I');
17 echo "O nome é: $outro_nome";
18
19 ?>
```



Mais exemplos

exemplo_07.php

```
1  <?php
2
3      $vetor = [1,5,8,100,2];
4
5      function mostra_pares($entrada){
6          for($i=0;$i<count($entrada);$i++){
7              if ($entrada[$i]%2 == 0){
8                  echo "<p>$entrada[$i] </p>";
9              }
10         }
11     }
12
13     mostra_pares($vetor);
14  ?>
```



Exercícios

1. Escreva um script que pergunta ao usuário se ele deseja converter uma temperatura de grau Celsius para Farenheit ou vice-versa. Se ele digitar 1, é de Celsius para Farenheit, se digitar 2 é de Farenheit para Celsius, outro valor mostre uma mensagem de erro. Para cada conversão, chame a função correta.

$$F = C \cdot \frac{9}{5} + 32$$

$$C = (F - 32) \cdot \frac{5}{9}$$

Exercícios

2. Faça um programa para imprimir:

```
1
1 2
1 2 3
...
1 2 3 ... n
```

para um **n** informado pelo usuário. Use uma função que receba um valor **n** inteiro imprima até a ***n-ésima*** linha.

Referência para uso de funções em PHP

https://www.php.net/manual/pt_BR/funcref.php

Include, Require, Include_once, Require_once

Função INCLUDE

A função ***include()*** do PHP tem como objetivo incluir (como o próprio nome diz) um arquivo dentro do outro quando acessado. Caso ocorra algum problema na inclusão deste, será apresentado um ***Warning*** (aviso) que não foi possível incluir o arquivo e continuará a exibição da página normalmente sem a inclusão do arquivo. A função ***include()*** aceita parâmetros via ***GET*** quando chama um arquivo.

Função REQUIRE

A função **require()** do PHP tem a mesma funcionalidade da função **include()**, citada acima, com a diferença que se caso o arquivo que você está incluindo não exista (ou não seja encontrado), será gerado um ***Fatal Error*** (erro fatal), **paralisando** a execução de qualquer script que venha após a linha do ***require()***; outra divergência é o fato desta função **não aceitar parâmetros via GET** para o arquivo chamado. Caso você utilize este parâmetro, ele será ignorado.

Função `include_once()` e `require_once()`

As funções **`include_once()`** e **`require_once()`** do PHP tem as suas funcionalidades praticamente “idênticas” às funções **`include()`** e **`require()`**, respectivamente. Digo “idênticas” (entre aspas) pois a única diferença entre elas é o fato das funções que possuem o “**`_once`**” só permitirem a inclusão do arquivo uma única vez na página e ignorando caso você chame ela mais vezes sem notar.

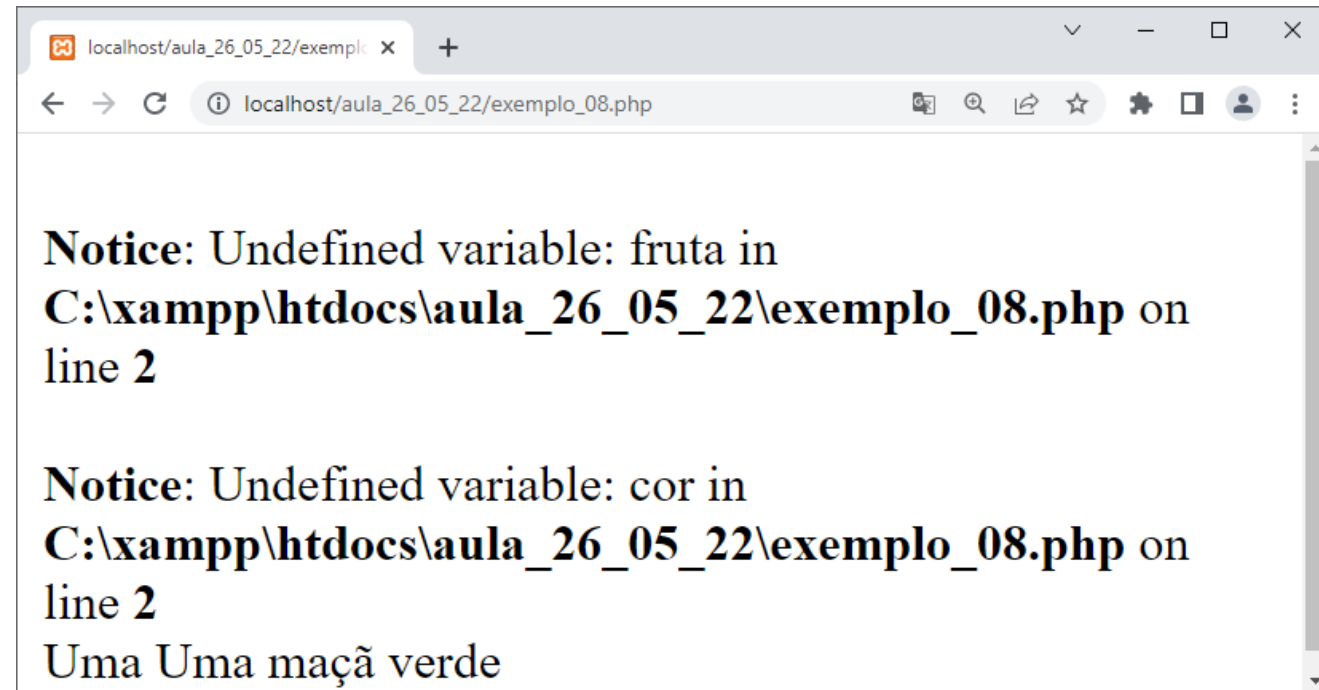
Exemplo include()

exemplo_08.php

```
1 <?php
2     echo "Uma $fruta $cor"; // Aqui é gerado um aviso!
3     // Mas o programa continua rodando!
4     include 'include_08.php';
5     echo "Uma $fruta $cor"; // Uma maçã verde
6 ?>
```

include_08.php

```
1 <?php
2     $cor = 'verde';
3     $fruta = 'maçã';
4 ?>
```



Exemplo require()

exemplo_08.php

```
1 <?php
2     echo "Uma $fruta $cor"; // Aqui é gerado um aviso!
3     // Mas o programa continua rodando!
4     include 'include_08.php';
5     echo "Uma $fruta $cor"; // Uma maçã verde
6 ?>
```

include_08.php

```
1 <?php
2     $cor = 'verde';
3     $fruta = 'maçã';
4 ?>
```

