

# PHP

Aula 5

# Manipulação de funções

- Sintaxe:

```
function nome_da_função([arg1, arg2, arg3]) {  
    Comandos;  
    ... ;  
    [return <valor de retorno>];  
}
```

- Valor de retorno

- Toda função pode opcionalmente retornar um valor, ou simplesmente executar os comandos e não retornar valor algum.
- Não é possível que uma função retorne mais de um valor, mas é permitido fazer com que uma função retorne um valor composto, como listas ou arrays.

# Trabalhando com funções

- Argumentos
  - É possível passar argumentos para uma função. Eles devem ser declarados logo após o nome da função, entre parênteses, e tornam-se variáveis pertencentes ao escopo local da função.
  - Exemplo:

```
<?php
function imprime($texto) {
    echo $texto;
}
imprime("teste de funções");
?>
```

# Trabalhando com funções

- Exemplo:

```
<?php
function soma($v1, $v2) {
    $soma = $v1+$v2;
    return soma;
}
$a = 10;
$b = 15;
$res = soma($a, $b);
?>
```

# Requisição de arquivos

- Usado para incluir em nossos programas outros arquivos com definições de funções, constantes, entre outros.
- Podemos fazer isso de diferentes formas:
  - Include
  - Include\_once
  - Require
  - Require\_once

# Requisição de arquivos

- Include
  - Inclui o arquivo informado.
  - O código é incluído no programa e pode ser utilizado a partir do ponto onde foi incluído.
  - Se o arquivo não existir será produzida uma mensagem de advertência.
  - Exemplo:

## **biblioteca.php**

```
<?php
function quadrado ($n){
    return $n * $n;
}
?>
```

## **teste.php**

```
<?php
include 'biblioteca.php';
echo quadrado (4);
?>
```

# Requisição de arquivos

- Require
  - Inclui o arquivo informado.
  - O código é incluído no programa e pode ser utilizado a partir do ponto onde foi incluído.
  - Se o arquivo não existir será produzida uma mensagem de erro fatal, diferentemente do include, que produz uma mensagem de advertência..

- Exemplo:

## **biblioteca.php**

```
<?php
function quadrado ($n){
    return $n * $n;
}
?>
```

## **teste.php**

```
<?php
    require 'biblioteca.php';
    echo quadrado (4);
?>
```

# Requisição de arquivos

- `include_once`
  - Funciona da mesma maneira que o `include`, a não ser que o arquivo informado já tenha sido incluído. Neste caso não refaz a operação.
- `require_once`
  - Funciona da mesma maneira que o `require`, a não ser que o arquivo informado já tenha sido incluído. Neste caso não refaz a operação.



# Requisição de arquivos

## cabecalho.php

```
<html>
<head><title>Teste PHP</title></head>
<body>
  <?php
    $meses = array(1 => "Janeiro",2 => "Fevereiro",3 => "Março",4 => "Abril",5 => "Maio",6 =>
    "Junho",7 => "Julho", 8 => "Agosto",9 => "Setembro",10 => "Outubro",11 => "Novembro",12
    => "Dezembro");
    $hoje = getdate();
    $dia = $hoje["mday"];
    $mes = $hoje["mon"];
    $nomeMes = $meses[$mes];
    $ano = $hoje["year"];
    echo "Olá. Hoje é dia $dia de $nomeMes de $ano."
  ?>
</body>
</html>
```

# Requisição de arquivos

```
<html>
<head><title>Página PHP</title></head>
<body>
  <h1>
    <?php
      include("cabecalho.php");
    ?>
  </h1>
</body>
</html>
```

# Exercício 1

- Crie um programa que leia 3 números e calcule a soma, a multiplicação e exiba o maior destes números. Uso de função é obrigatório.

# Exercício 1 - Resposta

```
<?php
function soma($v1, $v2, $v3) {
    $soma = $v1 + $v2 + $v3;
    return $soma;
}
function maior($v1, $v2, $v3) {
    if (($v1>$v2) && ($v1>$v3)){
        return $v1;
    } elseif (($v2>$v1) && ($v2>$v3)){
        return $v2;
    } elseif (($v3>$v1) && ($v3>$v2)){
        return $v3;
    }
}
function mult($v1, $v2, $v3) {
    $mult = $v1 * $v2 * $v3;
    return $mult;
}
$a = 2;
$b = 5;
$c = 3;
echo "Soma ". soma($a, $b, $c) . "<br/>";
echo "Maior ". maior($a, $b, $c) . "<br/>";
echo "Multiplicação ". mult($a, $b, $c). "<br/>";
?>
```

## Exercício 2

- Faça um programa em php que leia três notas de um aluno, calcule e escreva a média final deste aluno. Considerar que a média é ponderada e que o peso das notas é 2, 3 e 5. Uso de função é obrigatório

# Exercício 2 - Resposta

```
<?php
function media($n1, $n2, $n3) {
    $media = (($n1*2) + ($n2*3) + ($n3*5))/10;
    return $media;
}
$a = 8;
$b = 7;
$c = 3;
echo "Média Final ". media($a, $b, $c) . "<br/>";
?>
```

# Variáveis Globais

- Todas as variáveis declaradas dentro do escopo de uma função são locais.
- Para acessar uma variável externa ao contexto sem passá-la como parâmetro é necessário declará-la como global.
- Exemplo:

```
<?php
    $curso = "PHP";
    function mostra() {
        global $curso;
        echo $curso;
    }
    mostra();
?>
```

# Exercício 3

- Crie um programa que leia 3 números e calcule a soma destes números.
- Usar variáveis globais e função.



# Exercício 3 -Resposta

```
<?php
$a=2;
$b=5;
$c=3;
function soma() {
    global $a,$b,$c;
    $soma = $a + $b + $c;
    return $soma;
}
echo "Soma ". soma() . "<br/>";
?>
```