

Funções

- ▶ Uma função é um pedaço de código com objetivo específico
- ▶ Pode receber um conjunto de parâmetros e retornar um dado
- ▶ Uma vez declarada, pode ser utilizada inúmeras vezes
- ▶ É uma das estruturas mais básicas para promover o reuso

```
<?php
//exemplo de função

function nome_da_funcao($arg1, $arg2, $arg3){
    $valor = $arg1 + $arg2 + $arg3;
    return $valor;
}
?>
```

Funções

Exemplo

► Calcular o IMC

```
<?php
    function calcula_IMC ($peso, $altura) {
        return $peso / ($altura * $altura);
    }

    echo calcula_IMC(70, 1.85);
?>
```

Funções

Passagem de parâmetros - por valor

- ▶ É possível fazer a passagem de parâmetros de duas formas
 - ▶ Por valor (*by value*)
 - ▶ Por referência (*by reference*)
- ▶ Por padrão, os parâmetros são por valor
 - ▶ O parâmetro que a função recebe é tratado como local
 - ▶ Não altera seu valor externo

```
<?php
//exemplo de passagem por valor
function incrementa($var, $valor) {
    $var += $valor;
}

$a = 10;
incrementa($a, 20);
echo $a;
?>
```



Resultado:

10

Funções

Passagem de parâmetros - por referência

- ▶ Para realizar a passagem por referência, basta utilizar o operador & na frente do parâmetro
- ▶ As modificações realizadas no conteúdo da variável são válidas no contexto externo à função

```
<?php
//exemplo de passagem por referência
function incrementa(&$var, $valor) {
    $var += $valor;
}

$a = 10;
incrementa($a, 20);
echo $a;
?>
```



Resultado:

30

Funções

Passagem de parâmetros - valor padrão

- ▶ É possível definir valores padrão para parâmetros
- ▶ Se a função for chamada sem especificar o parâmetro, o valor padrão será assumido

```
<?php
//exemplo de passagem por referência com valor padrão
function incrementa(&$var, $valor = 40) {
    $var += $valor;
}

$a = 10;
incrementa($a);
echo $a;
?>
```



Resultado:

50

Funções

Recursão

- ▶ É possível criar funções recursivas
- ▶ A função a seguir calcula o fatorial de um número

```
<?php
function fatorial($numero) {
    if($numero == 1){
        return $numero;
    }else{
        return $numero * fatorial($numero - 1);
    }
}

echo fatorial(5) . "<br>";
echo fatorial(7) . "<br>";
?>
```



Resultado:

120

5040

Observação

Para cada um dos exercícios, fornecer a função e pelo menos 3 testes de seu uso.

Exercícios 18, 19 e 20

- 18) Crie uma função que receba um número inteiro e verifique e retorne se ele é par ou ímpar

Salve seu arquivo como `ex18.php`

19. Crie uma função que receba dois números e verifique e retorne o maior

Salve seu arquivo como `ex19.php`

- 20) Crie uma função que receba o salário de uma pessoa, calcule e retorne o valor do imposto de renda relativo devido conforme a tabela abaixo

Base de cálculo mensal em R\$	Alíquota %	Parcela a deduzir do imposto em R\$
Até 1.787,77	<i>isento</i>	<i>isento</i>
De 1.787,78 até 2.679,29	7,5	134,08
De 2.679,30 até 3.572,43	15,0	335,03
De 3.572,44 até 4.463,81	22,5	602,96
Acima de 4.463,81	27,5	826,15

Exemplo para um salário de R\$ 3000:

$$(3000 * 15\%) - 335,03 = 114,97$$

Salve seu arquivo como `ex20.php`

Observação 2

Tecnicamente, o cálculo do Imposto de Renda Retido na Fonte não é exatamente assim, mas iremos considerar conforme especificado para fins didáticos.