

PHP - Funções



Revisão

NA AULA PASSADA...

- Protocolos
 - HTTP x HTTPS
- Métodos
 - GET e POST
- Status Code
- Formulários
- Comunicação entre HTML e PHP



Funções

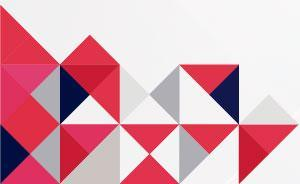


Funções

Permitem separar um “programa” em partes para:

- Organizar e identificar seções no código.
- Reutilizar funções em vez de repetir código.
- Corrigir, alterar e refatorar códigos de forma mais fácil.

Funções - Estrutura



Funções - Estrutura

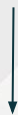
`function`



palavra reservada
imutável

Funções - Estrutura

`function` NomeFunção



opcional

Funções - Estrutura

`function NomeFunção ()`



uso obrigatório
para que
entenda-se que
=é uma função

Funções - Estrutura

`function` NomeFunção (parâmetro)



não-obrigatório
caso haja mais de
um, separá-los
por vírgula

Funções - Estrutura

```
function NomeFunção (parâmetro) {  
  
    //código  
    return  
  
}
```

uso obrigatório
- delimita o
espaço da minha
função -

Funções - Estrutura

```
function NomeFunção (parâmetro) {  
  
    //código  
    return  
  
}
```

Funções

Código

<?php

```
function saldoBancario($saldoAnterior,$debito) {  
    return $saldoAnterior - $debito;  
}
```

```
echo saldoBancario(110, 100);
```

?>

Funções

Código

```
<?php
```

```
function saldoBancario($saldoAnterior,$debito){  
    return $saldoAnterior - $debito;  
}
```

```
echo saldoBancario(110, 100);
```

```
?>
```

Gera

10

Funções - DEVO usar return ou não?



Funções

- Funções void

```
function totalFilmes ($assistidos, $naoAssistidos)
{
    $total = $assistidos + $naoAssistidos;
}

echo totalFilmes(10, 20);
```


Funções

- CUIDADOS AO USAR RETURN!

```
function finalizarCompra($nome)
{
    return "Olá $nome! Sua compra foi
aprovada!";
    echo "Deseja comprar mais produtos?";
}

echo finalizarCompra("Luiza");
```

Funções

- Funções com retorno

```
function finalizarCompra($nome)
{
    return "Olá $nome, sua compra foi
aprovada!";
}
```

```
echo finalizarCompra("Luiza");
```

Funções - Parâmetros, preciso?

→ Quando?



Funções

```
function finalizarCompra()  
{  
    echo "Olá \n";  
    echo "Sua compra foi aprovada!";  
}  
  
finalizarCompra();
```

Funções - Parâmetro Pré-determinado

<?php

```
function viagem($item1, $item2, $item3 = "shampoo") {  
    return "Preciso de $item1 e $item2 e $item3";  
}
```

//executando a função

```
echo viagem("escova", "roupas")
```

?>

Funções - Parâmetro Pré-determinado

<?php

```
function viagem($item1, $item2, $item3 = "shampoo") {  
    return "Preciso de $item1 e $item2 e $item3";  
}
```

//executando a função

```
echo viagem("escova", "roupas", "chinelos")
```

?>

Funções

Código

`<?php`

```
function nome($nome, $sobrenome = "") {  
    return $nome . " " . $sobrenome;  
}  
  
echo nome("Bon", "Jovi");  
echo "<br>";  
echo nome("João");
```

`?>`

Gera

Olá Bon Jovi
Olá João

Escopo (Scope)

O escopo de uma variável é o contexto em que ela está definida.
A maior parte das variáveis PHP só tem um escopo local.

O PHP tem 2 alcances possíveis:

- Global
- Local

Escopo (Scope) - Local



Escopo (Scope) - Revisão Variáveis

Breve Revisão:

→ Represento por **\$** e **nomeDaVariável**

◆ **\$a**

◆ **\$panela**

→ Determino valor

◆ **\$fastFood = "burger king";**

◆ **\$primeiro = 1;**

→ Case Sensitive

Escopo

Código

```
<?php
```

```
$a = 1; // âmbito global  
function test() {  
    echo $a; // âmbito local  
}  
test();
```

```
?>
```

Gera

Não produz saída!

Escopo - Global

Código

```
<?php
```

```
    $a = 1; //âmbito global
```

```
    function test() {
```

```
        global $a;
```

```
        echo $a + 1;
```

```
    }
```

```
    test();
```

```
?>
```

Gera

2

Escopo - Local

Código

```
<?php
```

```
function test() {
```

```
    $a = 0;
```

```
    echo $a;
```

```
}
```

```
test();
```

```
echo $a;
```

```
?>
```

Gera

0 e **erro**

Inclusão de arquivos

- **include**("arquivo.php") || **include** "arquivo.php";
Em caso de erro continua a execução e mostra um aviso

Inclusão de arquivos

- **require**("arquivo.php") || **require** "arquivo.php";
Em caso de erro para a execução e emite um erro

Inclusão de arquivos

- **include_once**("arquivo.php") || **include_once** "arquivo.php";
Evita inclusão repetida
- **require_once**("arquivo.php") || **require_once** "arquivo.php";
Evita inclusão repetida

Obrigado até a próxima

