



**INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
GOIÁS**  
**Câmpus Luziânia**

# Desenvolvimento de Sistemas Web Introdução ao PHP

# O QUE É O PHP?

- Uma linguagem de programação;
- Foi criada em 1994 por Rasmus Lerdorf;
- Permite criar facilmente aplicações para Web;
- Em 1995 foi lançada a versão PHP/FI (Personal Home Page/Form Interpreter);
- Em 1997 foi lançada a segunda versão do PHP;
- No mesmo ano dois estudantes Andi Gutmans e Zeev Suraski, resolveram cooperar com Rasmus e lançaram a versão PHP3.

# O QUE É O PHP?

- Em 1998, o PHP já estava presente em 10% dos domínios da Internet;
- O significado da sigla PHP mudou para PHP: Hypertext Preprocessor;
- Em 2000 foi lançado o PHP4 e a linguagem passou a estar presente em 20% dos domínios da Internet;
- Em 2004 foi lançado o PHP5 com suporte a Orientação a Objetos.

# Um programa PHP

- Extensão de arquivos

Extensão	Significado
.php	Arquivo PHP contendo um programa.
.class.php	Arquivo PHP contendo uma classe.
.inc.php	Arquivo PHP a ser incluído, pode incluir constantes ou configurações.

# Um programa PHP

- Delimitando o código

```
<?php  
//código;  
//código;  
//código;  
?>
```

# Um programa PHP

- Comentários
  - Para comentar uma única linha

```
// echo "a";
```

# echo "a";
  - Para comentar duas linhas

```
/* echo "a";
```

echo "a"; \*/

# Um programa PHP

- Comandos de saída (output)
  - **echo** é um comando que imprime uma ou mais variáveis no console. Exemplo:

```
echo 'a', 'b', 'c';
```

## Resultado

abc

# Um programa PHP

- Comandos de saída (output)
  - **print** é uma função que imprime uma string no console.  
Exemplo:

```
print( 'abc' );
```

**Resultado**

abc

# Um programa PHP

- Comandos de saída (output)
  - **var\_dump** imprime o conteúdo de uma variável de forma explanativa. Exemplo:

```
$vetor = array('Palio', 'Gol', 'Fiesta',  
               'Corsa');  
var_dump($vetor);
```

# Um programa PHP

- Comandos de saída (output)
  - `print_r` imprime o conteúdo de uma variável de forma explanativa, assim como o `var_dump()`. Exemplo:

```
$vetor = array('Palio', 'Gol', 'Fiesta',
'Corsa');

print_r($vetor);
```

# Variáveis

- Variáveis são identificadores utilizados para representar valores mutáveis e voláteis que só existem durante a execução do programa;
- Elas são armazenadas na memória RAM e seu conteúdo é destruído após a execução do programa.

# Variáveis

- Para criar um variável em PHP, precisamos atribuir-lhe um nome de identificação, sempre precedido pelo caractere cífrão (\$).

```
<?php  
$nome = "João";  
$sobrenome = "da Silva";  
echo "$sobrenome, $nome";  
?>
```

# Variáveis

- **Resultado**  
da Silva, João

# Variáveis

- Nunca inicie a nome da variável com números;
- Nunca utilize espaços em branco no meio do identificador da variável;
- Nunca utilize caracteres especiais (!@#%^&\*/|[]{}) no nome das variáveis;
- Evite criar variáveis com mais de 15 caracteres;
- Nomes de variáveis devem ser significativos;
- Utilize preferencialmente palavras em minúsculo.

# Variáveis

- **Exemplos**

```
<?php  
$codigo_cliente  
$codigoCliente  
?>
```

# Variáveis

- **Exemplos**

```
<?php  
//define o nome da variável  
$variavel = 'nome';  
//cria variavel identificada pelo conteúdo da $variavel  
$$variavel = 'maria';  
// exibe variável $nome na tela  
echo $nome; //resultado = maria  
?>
```

# Variáveis

- **Exemplos**

```
<?php  
  
$a = 5;  
  
$b = $a;  
  
$b = 10;  
  
echo $a; //resultado = 5  
  
echo $b; //resultado = 10  
  
?>
```

# Variáveis

- **Exemplos**

```
<?php  
  
$a = 5;  
  
$b = &$a;  
  
$b = 10;  
  
echo $a; //resultado = 10  
echo $b; //resultado = 10  
  
?>
```

# Variáveis (Tipos)

- **Tipo booleano**

Um booleano expressa um valor lógico que pode ser **verdadeiro** ou **falso**.

No exemplo a seguir, declaramos a variável booleana **\$exibir\_nome**, cujo conteúdo é **TRUE** (verdadeiro).

Em seguida, testaremos o conteúdo desta variável para verificar se ela é realmente verdadeira.

Caso positivo, será exibido na tela o nome 'José da Silva'.

# Variáveis (Tipos)

```
<?php  
//declara variável com valor TRUE  
$exibir_nome = TRUE;  
//testa se $exibir é TRUE  
if ($exibir_nome)  
{  
echo 'José da Silva';  
}  
?>
```

# Variáveis (Tipos)

```
<?php
// declara variável numérica
$umidade = 91;

// testa se é maior que 90. Retorna um boolean
$vai_chover = ($umidade > 90);

// testa se $vai_chover é verdadeiro
if ($vai_chover)
{
    echo 'Está chovendo';
}
?>
```

# Variáveis (Tipos)

- **Tipo numérico**

Números podem ser especificados em notação **decimal** (base 10), **hexadecimal** (base 16) ou **octal** (base 8).

```
<?php  
// número decimal  
$a = 1234;  
// número negativo  
$a = -1234;  
?>
```

# Variáveis (Tipos)

- **Tipo string**

Uma string é uma cadeia de caracteres alfanuméricos.

```
<?php  
$variavel = 'Isto é um teste';  
$variavel = "Isto é um teste";  
?>
```

# Variáveis (Tipos)

- **Tipo array**

**Array** é uma lista de valores armazenados na memória, os quais podem ser de tipos diferentes (**números, strings, objetos**) e podem ser acessados a qualquer momento pois cada valor é relacionado com uma **chave**.

```
<?php  
  
$carros = array( 'Palio' , 'Corsa' , 'Gol' );  
  
echo $carros[ 1 ]; //resultado = 'Corsa'  
  
?>
```

# Variáveis (Tipos)

- **Tipo objeto**

Um objeto é uma entidade com determinado comportamento definido por seus **métodos** (ações) e **propriedades** (dados);

Para se cria um objeto deve-se utilizar o operador **new**.

# Variáveis (Tipos)

```
<?php
class Computador
{
    public $cpu;
    function ligar()
    {
        echo 'Ligando computador a ($this->cpu)...';
    }
}

$obj = new Computador;
$obj->cpu = "500Mhz";
$obj->ligar();
?>
```

# Referências

- DALL`OGLIO, Pablo. **PHP : Programando com orientação a objetos, 2<sup>a</sup> edição.** São Paulo : Novatec Editora, 2009.