

# Curso de PHP

## Aula 1 – Lógica de Programação I

### Ministrantes:

- Pessoa X
- Pessoa Y
- Pessoa Z

# Objetivos do Curso

- Ensinar noções de programação em PHP;
- Diferenciar uma linguagem de programação de uma linguagem de marcação;
- Prover conhecimento necessário para criação de pequenos sistemas web que acessem e modifiquem informações no servidor, utilizando para isso PHP e manipulação de arquivos;

# Objetivos da Aula 1

- Explicar o que é PHP;
- Explicar os tipos de dados (Inteiro, ponto flutuante, string, booleano) em PHP;
- Lidar com operações de atribuição, aritméticas, incremento e decremento;
- Entender operações lógicas e de comparação;

# O que é PHP

- Criada por volta de 1994;
- O objetivo era criar scripts para "automatizar" tarefas;
- A linguagem HTML não é capaz de fazer isso, visto que é executada no cliente;
- PHP é executada no servidor e devolve ao cliente uma página HTML a ser manipulada nele;

# Por que utilizar PHP?

- Popularidade no mercado de trabalho;
- Facilidade de aprendizado;
- Roda em vários Sistemas Operacionais (Multi-Plataforma);
- Licença Gratuita (e código-fonte disponível para modificação);

# Vantagens do PHP...

- É popular e tem espaço crescente no mercado de trabalho;
- Acesso aos principais tipos de banco de dados, tais com MySQL, MS-SQL, PostgreSql, Oracle, SQLite, entre outros;
- Velocidade e Robustez;
- Tipagem fraca;
- Eficiente e Segura;

# Primeiro script em PHP

- O primeiro script será o "hello world";

Código:

```
<?
    echo "hello world"; // mostra hello world na tela
?>
```

# Comentários no código

- Não serão manipulados pelo interpretador PHP, sendo de uso exclusivo do programador;
- Servem para definir o que cada parte do código faz;

Exemplo:

```
<?
    echo "hello world"; // mostra hello world na tela
?>
```



# Bloco de código no PHP: { e }

- Quando for necessária a utilização de mais de uma linha, deve-se utilizar bloco de códigos;

Exemplo:

```
<?
echo "hello world"; // mostra hello world na tela
if (CONDICAO) {
    echo "primeira linha do bloco";
    echo "segunda linha do bloco";
}
?>
```

# Tipos de Dados

- Booleano;
- Inteiro;
- Ponto Flutuante;
- String

# Tipos de Dados Booleano

- Assume dois valores: *True* ou *False*;
- Útil na utilização de expressões booleanas, armazenar valores de retorno de funções, etc;

# Tipos de Dados Inteiro

- Compreende o conjunto dos números inteiros, isto é: os negativos, nulo e os positivos;

# Tipos de Dados Ponto Flutuante

- Compreende o conjunto dos números fracionários e inteiros;
- Diferente da nossa realidade, esse tipo de número é *finito* em um computador;

# Tipos de Dados String

- Usado para formação de letras e palavras;
- Útil para armazenar informações alfanuméricas, dados de formulários, etc;

# Operações de Atribuição

- Atribui à uma variável algum valor;

## Exemplos:

```
<?
```

```
$num_LOST = "4 8 15 16 23 42"; // é uma palavra
```

```
$numero = 42;
```

```
$eh_verdade = false; // não é verdade
```

```
$num_PI = 3,1415; // numero PI
```

```
?>
```

- OBS: PHP é case-sensitive.

# Operações Aritméticas

- Os operadores são similares aos já utilizados na matemática do dia-a-dia;
- Podem ser feitos sobre variáveis;

## Exemplos:

```
$num_dez = 4 + 6;           // $num_dez recebe 10
$num_vinte = 30 - $num_dez; // e assim por diante...
$num_dois = $num_vinte / $num_dez; // a barra é a
divisão
$var_quinze = 3 * 5;
$resto = $num_vinte % 3;    // "%" é o resto da divisão
```



# Operações de Incremento

- Faz um incremento unitário em uma variável ordinal;
- Funciona como  $\$var = \$var + 1$ ;
- Economiza espaço no código;

Exemplo:

<?

```
$contador = 10;           // $contador recebe 10
$contador++;             // agora $contador vale 11
$contador = $contador + 1; // agora vale 12
```

?>

Programa Bolsa-Instrutor - <http://www.inf.ufpr.br/instrutores>  
© 2010 – Todos os direitos reservados



# Operações de Decremento

- Faz um decremento em uma variável ordinal;
- Funciona como  $\$var = \$var - 1$ ;
- Economiza espaço no código;

Exemplo:

<?

```
$contador = 10;      // $contador recebe 10
```

```
$contador--;        // agora $contador vale 9
```

```
$contador = $contador - 1; // agora vale 8
```

?>

# Operações de Comparação

- Comparam dois ou mais valores e retornam como resultado um valor booleano, ou seja, **verdadeiro** ou **falso**;

# Comparação: Igual

- Retorna **verdadeiro** se os dois valores forem *iguais* e **falso** caso contrário;

Exemplo:

```
<?  
    if ($idade == 18)  
        echo "Tem 18. É maior ;)";  
?>
```

# Comparação: Diferente

- Retorna **verdadeiro** se os dois valores forem *diferentes* e **falso** caso contrário;

Exemplo:

```
if ($idade != 18)
    echo "Não possui 18 anos.";
```

- Pode-se utilizar também  $\langle \rangle$  no lugar de  $\neq$ ;
- Exemplo:

```
if ($idade <> 18)
    echo "Não possui 18 anos.";
```

# Comparação: Idêntico

- Retorna **verdadeiro** se os dois valores forem **iguais** **E** de **mesmo tipo** e **falso** caso contrário;

Exemplo:

```
<?
```

```
$palavra = "10";           // palavra (tipo string)
$numero = 10;             // inteiro
if ($palavra == $numero)  // são iguais?
    echo "Sao iguais <br>";
if ($palavra === $numero) // são idênticos?
    echo "Sao iguais e de tipos iguais";
```

```
?>
```

# Comparação: Idêntico

- Exemplo 2:

```
<?
```

```
$palavra = 10;           // inteiro
$numero = 10;           // inteiro
if ($palavra == $numero)
    echo "Sao iguais <br>";
if ($palavra === $numero)
    echo "Sao iguais e de tipos iguais";
```

```
?>
```

# Comparação: Não Idêntico

- Retorna **verdadeiro** se os dois valores forem *diferentes* **OU** de *tipos diferentes* e **falso** caso contrário;



# Comparação: Não Idêntico

## Exemplo:

```
<?
```

```
$palavra = "10";
```

```
$numero = 10;
```

```
$numero2 = "15";
```

```
if ($palavra !== $numero) // tipos diferentes
```

```
    echo "Nao sao identicos <br>";
```

```
if ($palavra !== $numero2) // valores diferentes
```

```
    echo "Nao sao identicos.";
```

```
?>
```

# Comparação: Menor

- Retorna **verdadeiro** se o primeiro valor for *menor* e **falso** caso contrário;

Exemplo:

```
<?
    if ($idade < 18)
        echo "Menor de Idade. Não pode dirigir ;)";
?>
```

# Comparação: Maior

- Retorna **verdadeiro** se o primeiro valor for *maior* e **falso** caso contrário;

Exemplo:

```
<?
    if ($idade > 18)
        echo "Maior de Idade. Já pode dirigir ;)";
?>
```

# Comparação: Menor ou igual

- Retorna **verdadeiro** se o primeiro valor for *menor ou igual* e **falso** caso contrário;

Exemplo:

```
if ($altura <= 180)
    echo "Apenas crianças maiores de 1,80m podem
    andar nesse brinquedo ;)";
```

# Comparação: Maior ou igual

- Retorna **verdadeiro** se o primeiro valor for *maior ou igual* e **falso** caso contrário;

Exemplo:

```
<?
```

```
// Calcula o Índice de Massa Corporea
```

```
$IMC = $massa / ($altura * $altura)
```

```
if ($IMC >= 40)
```

```
    echo "Você está com obesidade de grau III";
```

```
?>
```

# Operações Lógicas

- São aquelas que, dada determinada sentença, resultam em um valor *booleano*, ou seja, **verdadeiro** ou **falso**;
- Podem ser unárias ou binárias;

# Operador Lógico "Negação": !

- Retorna **verdadeiro** se *falso* e falso se *verdadeiro*;

Exemplo:

```
<?
```

```
$valor = true; // ajusta $valor como verdadeira
```

```
$valor = !$valor; // agora $valor eh falsa
```

```
?>
```

# Operador Lógico "E": AND

- Retorna verdadeiro se *os valores forem verdadeiros* e falso caso contrário;

Exemplo:

```
<?
```

```
// Se $idade for maior que 18 _E_
```

```
// $possui_habilitacao
```

```
if (($idade >= 18) AND ($possui_habilitacao))
```

```
    echo "Pode dirigir";
```

```
?>
```

- Pode-se utilizar também **&&** no lugar de **AND**.



# Operador Lógico "OU": OR

- Retorna **verdadeiro** se *pele menos um* dos valores *for verdadeiro* e **falso** caso contrário;

Exemplo:

```
// Se $idade for menor que 18 _OU_  
// não $possui_habilitacao  
if (($idade < 18) OR !($possui_habilitacao))  
    echo "Nao pode dirigir";
```

- Pode-se utilizar também **||** no lugar de **OR**;

# Operador Lógico "OU EXCLUSIVO": XOR

- Retorna **verdadeiro** se *apenas um* dos valores *for verdadeiro* e **falso** caso contrário;

## Exemplo:

```
$quero_cachorro_quente = true;    // pedi um hot-dog
$quero_x_salada = true;          // pedi também um x-salada
if (($quero_cachorro_quente) XOR ($quero_x_salada))
    echo "OK. Vou preparar seu pedido";
else
    echo "Escolha uma, e apenas uma, opção.";
```