



PHP 5

Almir Mendes de Araújo
m3nd3s@gmail.com
<http://www.almirmendes.net>



Informações

- Duração 1,5h
- Conhecimento prévio de Lógica de Programação
- Conhecimento básico de HTML

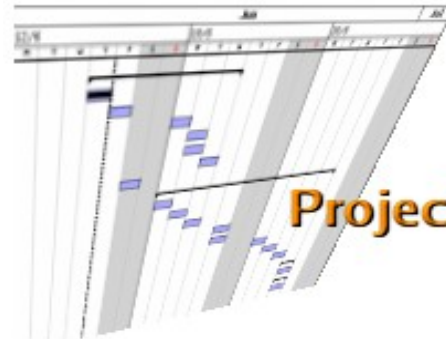


Por que usar PHP?

- PHP é gratuito, livre e de código aberto.
- Muita documentação disponível e comunidade grande e ativa.
- Módulos: DBs, Streams, File Systems, Imagens, Multimídia...
- Velocidade, Orientação a objeto, Multiplataforma, Sintaxe similar a Linguagem C/C++ e o PERL



Ferramentas



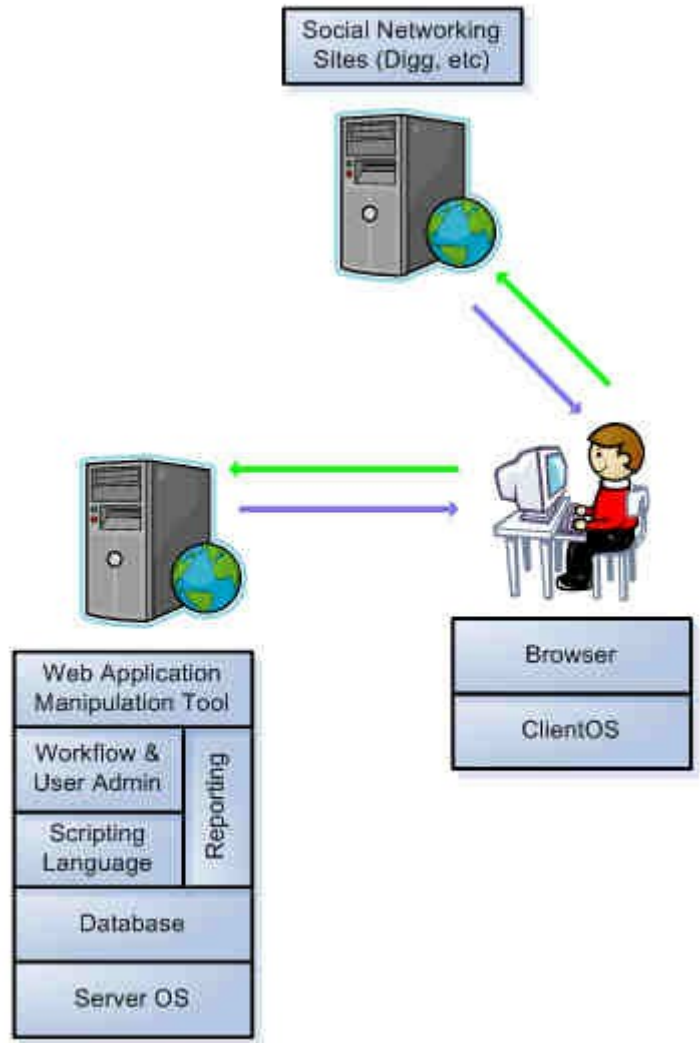


Introdução

- Client-side x Server-side
- Javascript x HTML x PHP
- Tipagem dinâmica
- PHP é embutido no HTML



Server Side





O que é necessário?

- Servidor Web que suporte PHP
 - **Apache**
 - IIS
 - Xitami
 - OmniHTTPd
 - lighttpd
- PHP



Sintaxe básica

- Tags do PHP (`<?php ?>`)
- PHP e HTML
- Separador de instruções(;
- Comentários
- Variáveis precedidas com sinal \$ (cifrão). Ex.:
`$nome`



Variáveis e Constantes

- Variáveis
 - Não precisam ser declaradas, apenas use!
 - Sempre iniciam com o sinal \$.
 - Seguem um padrão de nomenclatura assim como nas demais linguagens.
 - PHP é *case sensitive*
- Constantes
 - Referência direta, SEM o sinal \$



Tipos e Variáveis no PHP

Básicos:

boolean, integer, float (double), string

Compostos:

array, object

Especiais:

resource, NULL

Strings

- Apóstrofos (aspas simples)
 - Maneira mais fácil de especificar uma string.
- Aspas
 - Entre aspas ("), o PHP entende mais seqüências de escape para caracteres especiais como: `\n`, `\r`, `\t`, `\\`, `\$`, `\'`
 - Aceitam variáveis no seu conteúdo



Operadores Aritiméticos

- Exemplos

- $\$a + \b

- Adição** - Soma de $\$a$ e $\$b$.

- $\$a - \b

- Subtração** - Diferença entre $\$a$ e $\$b$.

- $\$a * \b

- Multiplicação** - Produto de $\$a$ e $\$b$.

- $\$a / \b

- Divisão** - quociente de $\$a$ por $\$b$.

- $\$a \% \b

- Módulo** - Resto de $\$a$ dividido por $\$b$.

Operadores Atribuição

- Atribuição é "=". NÃO é "é igual". Isto quer dizer, na verdade, que o operando da esquerda **recebe** o valor da expressão da direita.
- Além do operador básico de atribuição, há "*operadores combinados*" para todos os operadores aritméticos e de string.
 - Ex.: +=, -=, /=, *=, %=



Exercícios (10min)

- Faça um algoritmo que tendo a idade de uma pessoa expressa em anos, meses e dias e mostre-a expressa apenas em dias.
- Faça um algoritmo que tendo o tempo de duração de um evento em uma fábrica expressa em segundos e mostre-o expresso em horas, minutos e segundos.



Será que vai dar erro?!?!

```
<?php
```

```
$var1 = "5patos";
```

```
$var2 = "3galinhas";
```

```
$var3 = "1pardal";
```

```
$var4 = true;
```

```
echo $var1+$var2-$var3-$var4;
```

```
//O que você acha será impresso?
```

```
?>
```



Arrays

- Armazenam mais de um valor de forma indexada.
- Podem conter qualquer tipo de valor no mesmo array.
- O acesso ao valor é feito através do índice.
- Os índices também podem ser nominais.
- Podem ser unidimensional ou multidimensional.

- Exemplos

- $\$a == \b **Igual** - Verdadeiro (TRUE) se $\$a$ é igual a $\$b$.
- $\$a != \b **Diferente** - Verdadeiro se $\$a$ não é igual a $\$b$.
- $\$a < \b **Menor que** - Verdadeiro se $\$a$ é estritamente menor que $\$b$.
- $\$a > \b **Maior que** - Verdadeiro se $\$a$ é estritamente maior que $\$b$.
- $\$a <= \b **Menor ou igual** - Verdadeiro se $\$a$ é menor ou igual a $\$b$.
- $\$a >= \b **Maior ou igual** - Verdadeiro se $\$a$ é maior ou igual a $\$b$.



Estruturas de controle

- if
- switch
- for
- foreach
- while

```
<?php
```

```
if ($a > $b) {
```

```
    echo "a is greater than b";
```

```
} else {
```

```
    echo "a is NOT greater than b";
```

```
}
```

```
?>
```

```
switch ($i) {  
    case 0:  
        echo "i equals 0"; break;  
    case 1:  
        echo "i equals 1"; break;  
    case 2:  
        echo "i equals 2"; break;  
}
```



for

```
for ($i = 1; $i <= 10; $i++) {  
    echo $i;  
}
```



while

```
$i = 1;  
while ($i <= 10) {  
    echo $i++;  
}
```



Algumas funções úteis

- `empty($var)` → retorna TRUE se \$var é vazio
- `isset($var)` → retorna TRUE se \$var existe
- `require("file.php")`, `include("file.php")` → include arquivo no script atual
- `var_dump($var)` → faz o dump de \$var na tela
- `print_r($var)` → o mesmo que `var_dump`*

Formulários

- **\$_GET** - Variáveis postadas para o script via método HTTP GET (url).
 - Ex.: `.../remover.php?id=4`
- **\$_POST** - Variáveis postadas para o script via método HTTP POST (via post de formulários).
- **\$_POST** e **\$_GET** são arrays superglobais do PHP, estão disponíveis em qualquer nível.
- Cada campo de um formulário



Fontes

- <http://www.php.net/>
- http://br.php.net/manual/pt_BR/
- <http://www.almirmendes.net>
- <http://www.phpbrasil.com>



Algumas IDEs

- **Komodo (grátis)**
 - http://www.activestate.com/Products/komodo_edit/
- **Zend (pago)**
 - http://www.zend.com/products/zend_studio
- **PHPEclipse (grátis)**
 - http://phpeclipse.de/tiki-view_articles.php
- **Vim/Gvim (Livre)**
 - <http://www.vim.org/>