

# PHP

Uma pequena introdução

# Quem sou eu?

- Jonatas Oliveira
- Programador PHP desde 2005
- Programador e desenvolvedor Web da Diginet
- Entusiasta do movimento software livre
- Membro do PSL-RN e +web

# O que é o PHP?

- Linguagem scripting de propósito geral
- Server-side (executada no lado do servidor)
- Voltada para criação de conteúdo dinâmico para web
- Tipagem fraca e dinâmica
- Último release 5.2.6 em Maio de 2008

# Conteúdo dinâmico

- Conteúdo produzido sob demanda
- A página final não existe no servidor
- Exemplo de conteúdo dinâmico
  - Busca no google

# Funcionamento interno (simplificado) do PHP

- Requisição do usuário
- Servidor Web
- PHP
- HTML (opcional)
- Navegador do usuário

# Áreas de atuação do PHP

- **Server-side Scripting**
  - Foco principal
  - Criar páginas dinâmicas para web
  - Necessita do PHP instalado, de um servidor web e de um navegador para funcionar

# Áreas de atuação do PHP

- **Command line scripting**
  - Executar pequenas tarefas
  - Necessita somente do parser PHP
- **Aplicação desktop**
  - Possível (mas não recomendado)
  - Aplicações gráficas e multi-plataforma com PHP + GTK

# Por que PHP?

- Sintaxe básica simples
- Começo fácil
- Boa documentação em português

# Por que PHP?

- Disponível em várias plataformas
  - Linux
  - Unix
  - Mac OS X
  - Windows
  - Outros

# Por que PHP?

- Suporte de muitos servidores web
  - Apache
  - Microsoft IIS
  - Outros, se não existir módulo, ainda pode funcionar como processador CGI

# Por que PHP?

- Suporte a muitos banco de dados
  - MySQL
  - PostgreSQL
  - ODBC
  - SQLite
  - MS-SQL
  - E muitos outros

# Por que PHP?

- Comunicação com outros serviços
  - LDAP
  - IMAP
  - SNMP
  - NNTP
  - POP3
  - HTTP

# O que o PHP pode fazer

- Qualquer coisa que um programa CGI pode fazer
  - Coletar dados de um formulário
  - Criar conteúdo dinâmico
  - Enviar e receber cookies
  - Criar sessão com navegador

# O que o PHP pode fazer

- Ainda mais...
  - Criar imagens dinamicamente (gd)
  - Gerar arquivos PDF
  - Criar filmes em Flash dinamicamente
  - Gerar saída de qualquer tipo de texto (não só HTML)

# O que o PHP pode fazer

- E ainda mais...
- Pode salvar qualquer tipo de saída no sistema de arquivos, ao invés de mostrar, para criar cache do conteúdo
- Acessar arquivos do sistema de arquivos
- Pode salvar qualquer tipo de saída no sistema de arquivos, ao invés de mostrar, para criar cache do conteúdo

# Alguns projetos que usam PHP

- Wikipedia (mediawiki)
- Wordpress
- Drupal
- Existem muito mais, lógico

# Introdução a linguagem

Hello World!

```
<?php  
    echo "<p>Hello World!</p>";  
?>
```

# Introdução a linguagem

- Hello World!
  - Delimitadores para o parser
    - `<?php ?>`
    - `<? ?>`
    - `<?= ?>`, atalho para `<? echo ...; ?>`
  - Todo comando termina com ;

# Introdução a linguagem

- Hello World!
  - Comentários
    - #
    - //
    - /\* multiline \*/

# Introdução a linguagem

- Variáveis
  - Representadas por \$ mais o nome da variável
  - Diferencia maiúsculas e minúsculas
    - `$variavel` é diferente `$Variavel`
  - Nomes de variáveis válidos iniciam com uma letra ou underscore e seguem com letras, números ou underscore

# Introdução a linguagem

- Variáveis
  - Não requer declaração
  - O tipo é alterado dinamicamente
  - Os resultados são passados por valor, para ser passado por referência usa `&$variavel`

# Introdução a linguagem

- Tipos
  - Boolean
  - Números
  - String
  - Arrays
  - Objects
  - Resources
  - NULL

# Introdução a linguagem

- Boolean
  - True, False
  - São considerados falso
    - FALSE
    - inteiro 0 ou float 0.0
    - string vazia e string "0"
    - array vazio
    - NULL

# Introdução a linguagem

- Boolean
  - Operadores
    - ==

# Introdução a linguagem

- Números
  - Inteiro (decimal, octal, hexadecimal), Real, Float, Notação científica
  - Tamanho depende da plataforma

# Introdução a linguagem

- Números
  - Operadores
    - +
    - -
    - / (retorna float, para inteiro usa-se `round(1/2)`)
    - \*
    - %

# Introdução a linguagem

- String
  - ' ' (single-quoted)
    - Literal, não expande nenhum código interno
  - "" (double-quoted)
    - Expande qualquer código dentro da string
  - <<<'EOT' ... EOT; nowdoc

# Introdução a linguagem

- String
  - Suporte a UTF-8 via funções `utf8_encode()` e `utf8_decode()`
  - Operadores
    - `.` (ponto, para concatenar strings)

# Introdução a linguagem

- Arrays
  - Funções auxiliares
    - `unset()`
    - `array_values()` (reindex)
    - `print_r()`

# Introdução a linguagem

- Objects
  - Convertendo para objeto
    - De objeto para objeto não acontece nada
    - Qualquer outro tipo para objeto, nova instância de stdClass

# Introdução a linguagem

- Se valor NULL, a instância será criada vazia, se for um array, as chaves serão propriedades da instância com seus respectivos valores
- Para qualquer outro tipo será criado o método "scalar" que conterà o valor

# Introdução a linguagem

- Resources
  - Variáveis especiais com referência externa
  - Exemplo: conexão com banco de dados

# Introdução a linguagem

- NULL
  - Tipo especial para indicar que não tem valor

# Introdução a linguagem

- Funções verificadoras de tipo
  - `is_bool()`
  - `is_numeric()`
  - `is_float()`
  - `is_int()`
  - `is_string()`
  - `is_object()`
  - `is_array()`

# Estrutura de controle

## Condicional

```
<?php
    $nome = 'PHP';
    if ($nome == 'PHP') {
        // Faz alguma coisa.
    }
?>
```

# Estrutura de controle

- if
- else
- elseif

# Estrutura de controle

## Sintaxe alternativa

```
<?php if (5 > 1): ?>
```

```
5 é maior que 1.
```

```
<?php elseif (5 > 6) ?>
```

```
5 é maior que 6.
```

```
<?php endif; ?>
```

# Estrutura de controle

## Switch

```
<?php
    $i = 0;
    switch ($i) {
        case 0:
            echo "i igual a 0";
            break;
        case 1:
            echo "i igual a 1";
            break;
        default:
            echo "i não é nem igual a 0 nem igual a 1";
    }
?>
```

# Laço

## While e do-while

```
<?php
    $i = 0;
    while ($i <= 10) {
        echo $i++;
    }

    $i = 0;
    do {
        echo $i++;
    } while ($i <= 10);
?>
```

# Laço

## For

```
<?php
    for ($i = 0; $i <= 10; $i++) {
        echo $i;
    }

    for ($i = 0; $i <= 10; print $i++);
?>
```

# Laço

## Foreach

```
<?php
  for ($i = 0; $i <= 10; $i++) {
    echo $i;
  }

  for ($i = 0; $i <?php
    $arr = array(1, 2, 3, 4, 5);
    foreach ($arr as $value) {
      echo $value;
    }

    $arr = array("nome" => "PHP", "versao" => 5.2.6);
    foreach ($arr as $key => $value) {
      echo "Chave: $key; Valor: $value<br/>\n";
    }
?>
```

# Laço

- break
  - Pára a execução do laço
- continue
  - Passa para a próxima iteração do laço

# Função

```
<?php
    function soma($valor1, $valor2=1) {
        return $valor1 + $valor2;
    }

    $resultado = soma(10, 15);
    echo $resultado; // Saída: 25

    $funcao = 'soma';
    $resultado = $funcao(20, 1);
    echo $resultado; // Saída: 21
?>
```

# Orientação a objetos

- Classes
  - new
    - Cria uma nova instância de uma classe
  - \$this
    - Referência ao objeto que chamou
  - parent
    - Referência a classe estendida

# Orientação a objetos

- Classes
  - ->
    - Acesso aos membros da instância
  - ::
    - Acesso aos membros da classe
- extends
  - Faz uma classe herdar propriedades e métodos de outra

# Orientação a objetos

- Classes
  - `__construct`
    - Construtor da classe, chamado quando se usa `new Classe()`
  - `__destruct`
    - Destrutor da classe
- Autoloading
  - É chamado sempre que uma classe é usada e ela não está disponível no namespace local

# Orientação a objetos

- SPL - Standard PHP Library
  - Série de classes e interfaces para ajudar na resolução de problemas
    - Iteradores
    - Diretórios e Arquivos
    - XML
    - Arrays
    - Contagem
    - Exceções

# Segurança

- Recordista em CVE (Common Vulnerabilities and Exposures) pelo National Vulnerability Database
- 12% in 2003, 20% in 2004, 28% in 2005, 43% in 2006, 36% in 2007, and 33.8% for the first quarter of 2008

# Segurança

- Fatores que aumentam a vulnerabilidade
  - Programadores com maus hábitos
  - Dados manipulados sem checagem
  - `register_globals` (deprecated)

# Debug

- `error_reporting(E_ALL);`
- `ini_set('display_errors', true);`
- Log em arquivo (via php.ini)
  - `display_errors = Off`
  - `log_errors = On`
  - `error_log = /arquivo/de/log`

# Contato

jonatas.oliveira@digi.com.br

<http://blog.jonatasoliveira.com>