



ANÁLISE E DESENVOLVIMENTO SISTEMAS

DESENVOLVIMENTO WEB E BANCO DE DADOS

Eder Pazinato
Juliano Tonezer da Silva

PHP – Introdução

1



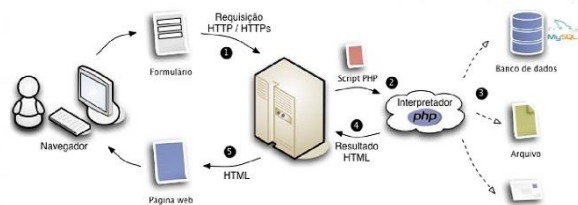
PHP

O que é PHP (Hypertext Preprocessor) ?



- ❑ O PHP é uma Linguagem de script open source de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML.

Cliente-Servidor-Php →



fonte(<https://www.fredericomarinho.com>)

2



PHP - Características

- ☐ Linguagem interpretada, utilizada na geração de conteúdo dinâmico
- ☐ Código executado em um servidor *web*
- ☐ Baseada nas linguagens Perl e C
- ☐ Programação estruturada e orientada a objetos
- ☐ Tipagem dinâmica
- ☐ Material de apoio: <http://www.php.net>

3



PHP - Como executar códigos PHP?

- ☐ Projeto deve estar hospedado em um *Servidor Web*, com suporte ativado para PHP
 - ☐ Exemplos de servidores *web*
 - ☐ **Apache Web Server**
 - ☐ Apache Tomcat
 - ☐ Microsoft Internet Information Services (IIS)
 - ☐ Nginx *web server*
 - ☐ Lighttpd
 - ☐ <https://www.w3schools.com/php> (*web*)

4



PHP - Como executar códigos PHP?

- ☐ O servidor *web* também pode ser executado localmente (*localhost*)
 - ☐ Opções de pacotes pré-configurados:
 - ☐ XAMPP (*multiplataforma*)
 - ☐ https://www.apachefriends.org/pt_br/download.html
 - ☐ WAMP (*windows*)
 - ☐ <http://www.wampserver.com/>
 - ☐ ApacheTriad
 - ☐ <https://apache2triad.soft112.com>
 - ☐ VertrigoServ
 - ☐ <https://www.vswamp.com>

5



PHP

- ☐ As soluções PHP também podem interagir com bases de dados
- ☐ Suporte a diferentes bancos de dados, entre eles:

<input type="checkbox"/> MySQL	<input type="checkbox"/> SQLite
<input type="checkbox"/> PostgreSQL	<input type="checkbox"/> IBM DB2
<input type="checkbox"/> Oracle	<input type="checkbox"/> Firebird
<input type="checkbox"/> SQL Server	<input type="checkbox"/> NoSQL
	<input type="checkbox"/>

6



PHP – Comandos e Estruturas

❑ Operadores e Expressões

❑ Precedência similar a linguagem C/C++

❑ Atribuição: =

❑ Aritméticos: + – * / %

❑ Incrementadores: x++ --y ++x y--

❑ Comparação: == >= <= != <> > <

❑ Atribuição combinada: += -= *= /= %=

❑ Lógicos: !m m && n m || n

7



PHP – Comandos e Estruturas

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Exemplo PHP</title>
</head>
<body>
  <?php
    echo "Hello World, PHP!";
    // echo = Exibe uma ou mais strings
    /* Comentários em duas ou mais linhas */
  ?>
</body>
</html>
```

Outras formas do interpretador identificar código PHP:

```
<script language="php">
  comandos
</script>
```

```
<?
  comandos
?>
```

```
<%
  comandos
%>
```

8



PHP – Comandos e Estruturas

☐ Função *Date*

```
<!DOCTYPE html>
<html>
<head> <title>Exemplo PHP</title>
</head>
<body>
  As linhas a seguir serão processadas pelo servidor web, e os
  resultados enviados ao navegador do usuário.</br>
  Data e hora atuais do Servidor Web:</br></br>
  <strong> <?php
      date_default_timezone_set('America/Sao_Paulo');
      echo date("d/m/Y");   echo"<br/>";
      echo date("h:i:s");   echo"<br/>";
  ?></strong>   </br>
  HTML resultante, exibindo o código-fonte da página através do
  navegador!
</body>
</html>
```

9



PHP – Comandos e Estruturas

☐ Variáveis

- ☐ *Case sensitive*
- ☐ Tipagem dinâmica
- ☐ Nomes iniciam com o símbolo \$

```
<!DOCTYPE html>
<html>
<body>
  <?php
    $valor= 45.50;   $semestres= 6;
    $nome= "ADS";   $ativo= true;
    echo "Meu curso => $nome"; echo"<br/>";
    echo $semestres;
  ?>
</body>
</html>
```

10



PHP – Comandos e Estruturas

☐ Variáveis e Função *nl2br* (funções: https://www.php.net/manual/pt_BR/indexes.functions.php)

```
<!DOCTYPE html>
<html>
<head> <title>Exemplo PHP</title> </head>
<body>
    TEXTO qualquer <strong>formatado</strong>.
    <?php
        $A = 5;  $B = 10;
        $C = $A + $B;
    ?>
        <br /> Uma nova linha, e o resultado...
    <?php
        echo nl2br("$C\n");
    ?>
    com quebra de linha usando nl2br (insere quebras de linha HTML
    (<br />) com ("\n") em uma string).
</body> </html>
```

11



PHP – Comandos e Estruturas

☐ Constantes

☐ *Uso da função **define**("nome da constante", "valor")*

```
<!DOCTYPE html>
<html>
<head> <title>Exemplo PHP</title>
<body>
    <?php
        define("PI", 3.14159265359);
        $resultado = 2 * PI;
        echo "$resultado"
    ?>
</body>
</html>
```

12



PHP – Comandos e Estruturas

☐ Formatação com uso de aspas e apóstrofes

```
<!DOCTYPE html>
<html>
  <head> <title>Exemplo PHP</title> </head>
  <body>
    <?php
      echo "<strong><font color=\"#FF0000\">Com aspas, é preciso
utilizar a contrabarra.</font> <br /> <br />";

      echo "<strong><font color='#0000FF'>Com apóstrofes, fica mais
simples.</font></strong> ";
    ?>
  </body>
</html>
```

13



PHP – Comandos e Estruturas

☐ Comandos de Seleção/Condicionais

if (expressao) comandos

```
if(expressao) {
  comandos
}

if(expressao) {
  comandos
}
else // ou
elseif(expressao) {
  comandos
}
```

switch (operador) casos

```
switch(operador) {
  case valor1:
    bloco de comandos
    break;
  case valor2:
    bloco de comandos
    break;
  default:
    bloco de comandos
    break;
}
```

14



PHP – Comandos e Estruturas

❑ Comandos de Repetição *while, do...while, for, foreach*

```
while(expressao) {  
    comandos  
}
```

```
do{  
    comandos  
}  
while(expressao)
```

```
for(inicializacao; condicao; operador) {  
    comandos  
}
```

```
foreach($array as $valor) {  
    comandos  
}
```

```
/* OU */  
foreach($array as $key => $valor) {  
    bloco  
}
```

15



PHP – Comandos e Estruturas

Vetores e Matrizes

❑ Vetor: \$variavel [idx]

❑ Matriz: \$variável [lin][col]

Vetor

```
$vet = array(elem1, elem2, elem3);
```

Exemplo:

```
for($x = 0; $x < 3; $x++)  
    echo "$vet[$x] <br/>";
```

16



PHP – Comandos e Estruturas

☐ Concatenação de strings

☐ Operador . (ponto)

Exemplo:

☐ \$str1 = "Hello World, "; \$str2 .= "Olá pessoal!";

☐ \$junta = \$str1 . \$str2;

// com espaço em branco

☐ \$completo = \$nome . " " . \$sobrenome;

☐ Captura parte da string

☐ Operador substr

Exemplo:

☐ \$data = "02/09/2024";

☐ \$dia = substr(\$data,0,2);

// "0": começa na posição 0 – "2": pega as duas próximas posições

☐ Mais funções: https://www.php.net/manual/pt_BR