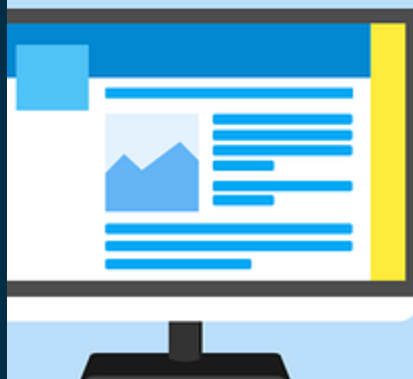


Programação Back End I

Introdução ao PHP

Prof. Me. Rafael Florindo



Proposta de cronograma dos encontros:

- ❖ Aula 01: Princípios de Desenvolvimento WEB com HTML

Unidade I

❖ Aula 02: Introdução ao PHP até estrutura condicional

Unidade II e Unidade III

- ❖ Aula 03: Continuação do PHP com aplicação de estrutura de repetição e array

Unidade III e Unidade IV

- ❖ Aula 04: Continuação do PHP com aplicação de estrutura de repetição e array e funções

- ❖ Unidade IV e V

❖ Introdução ao PHP

Unidade 02 e 03

- ✓ O que é PHP
- ✓ Onde encontramos o PHP
- ✓ Evolução do PHP
- ✓ Servidor Web
- ✓ Sintaxe Básica



❖ Introdução ao PHP

Unidade 02 e 03

- ✓ O que é PHP
- ✓ Onde encontramos o PHP
- ✓ Evolução do PHP
- ✓ Servidor Web
- ✓ Sintaxe Básica



❖ Introdução

O que é PHP

É uma linguagem interpretada e livre que atua do lado do servidor, tem seu melhor uso no desenvolvimento de **aplicações web dinâmicas**.

"PHP significa **Processador Hipertexto** (*Hypertext Preprocessor*), é uma linguagem de programação de ampla utilização e pode ser mesclada dentro do HTML" (MANUAL DO PHP, [2018], on-line).

O que a torna ideal para servidores web.

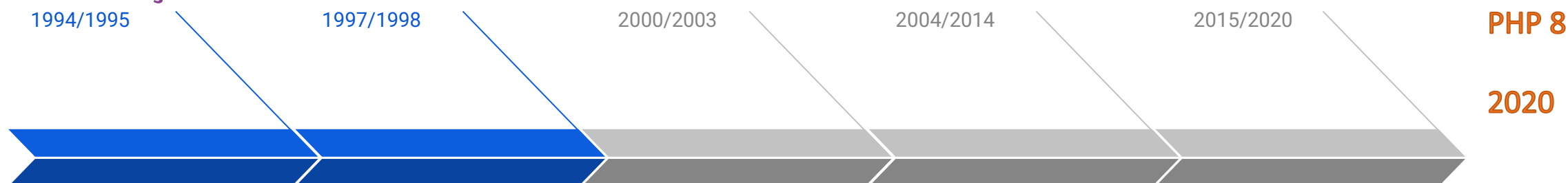
❖ Introdução

Onde encontramos o PHP

- O PHP é utilizado em aplicações como:
 - Facebook,
 - Drupal,
 - Joomla,
 - WordPress,
 - Magento,
 - E-commerce,
 - Laravel,
 - Symfony, Zend Framework, Aplicações entre outros.

❖ Introdução

Evolução do PHP



PHP 1 e 2

O PHP é criado por Rasmus para ser usado no desenvolvimento de sua página pessoal, era apenas um conjunto de scripts que foi chamado de Personal Home Page Tools.

PHP 3

Andi e Zeev entram no time para reescrever a base do PHP como uma linguagem completa de programação e vários desenvolvedores passam a colaborar com o desenvolvimento do PHP.

PHP 4

A versão 3 é totalmente abandonada e o PHP 4 é desenvolvido com recursos de OO.

PHP 5

Ocorre a ascensão do PHP com a versão 2 do zend engine, agora totalmente orientado a objetos, banco de dados com PDO, JSON nativo, namespaces, interfaces, traits e diversos outros recursos que solidificaram o PHP.

PHP 7

Com performance surpreendente e fortificação em recursos de OO, o PHP é visto como uma linguagem madura presente em mais de 83.5% dos sites na internet sendo a mais utilizada linguagem de programação do mundo.

❖ Introdução

O que é necessário para utilizar o PHP?

- Servidor WEB
- Editor de Código Fonte
- Navegador

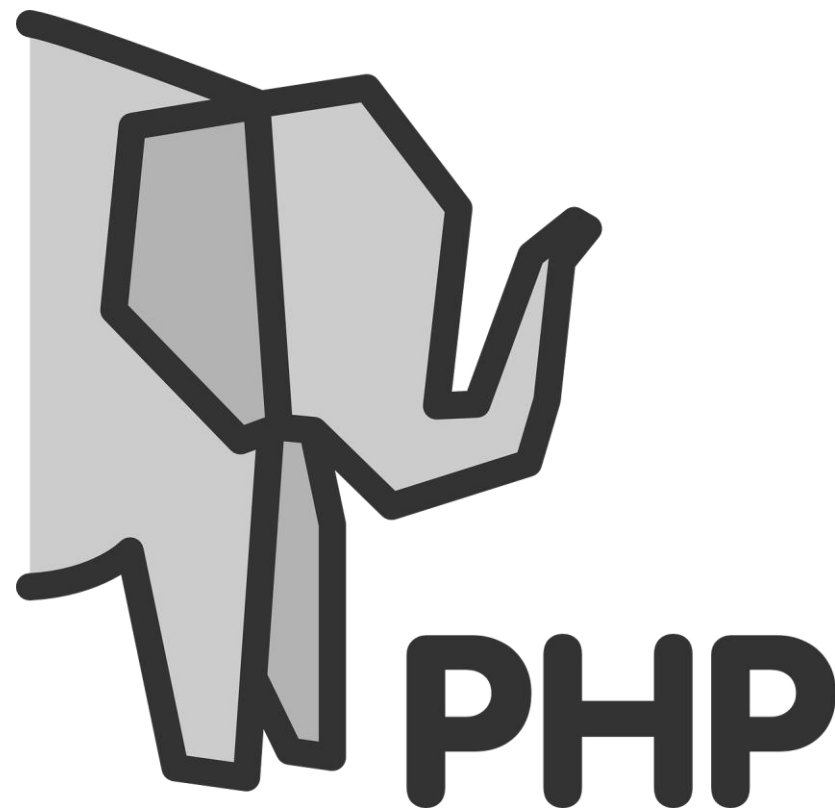


https://www.apachefriends.org/pt_br/index.html





Sintaxe Básica do PHP



Variáveis

Sintaxe: `$nova_da_variável = valor_da_variavel;`

Exemplos

`$number = 10; //inteiro`

`$number = 15.89; //float`

`$name = "Rafael"; //string`

`$status = true; //boolean`

`$status = false; //boolean`

`$numbers = []; //array`

`$client = new Cliente(); //objeto`

Tags de abertura

```
<?php
```

```
    echo "<h1>Olá seja bem-vindo</h1>";
```

```
?>
```

Comandos de saída

echo: Exibe uma ou mais strings

`<?= ;?`

print_r(): Imprime informação sobre uma variável de forma legível

var_dump(): mostrará uma representação estruturada sobre uma ou mais expressões, incluindo o tipo e o valor.

Exemplo de echo

```
<?php
```

```
$number = 10;
```

```
echo "Olá !!!";
```

```
echo $number;
```

```
echo "$number";
```

```
echo "<p>$number</p>";
```

```
echo "Número $number";
```

```
echo "Número " . $number;
```


Exemplo de print_r() e var_dump()

```
<?php
```

```
$number1 = 10;
```

```
$number2 = 15.89;
```

```
print_r($number1);
```

```
print_r($number2);
```

```
var_dump($number1);
```

```
var_dump($number2);
```

Operações aritméticas

```
<?php
```

```
$number1 = 10;
```

```
$number2 = 15.89;
```

```
$sum = $number1 + $number2;
```

```
echo $sum;
```

Estrutura de controle

Estruturas de controle de fluxo são responsáveis pela forma como o fluxo do programa será executado, ou seja, são nelas que colocamos **condições para uma ou outra rotina ser executada**, ou **rodarmos determinada rotina repetidas vezes** dada uma determinada condição de parada, isso permite dinamismo ao programa.

- **if, elseif, else**
- **switch**
- **Operador ternário**

Exemplo if-else-elseif

```
<?php
```

```
    $number1 = 10;
```

```
    if ($number1 > 10) {
```

```
        echo "Número maior que 10";
```

```
    }elseif ($number1 < 10) {
```

```
        echo "Número menor que 10";
```

```
    }else {
```

```
        echo "Número igual 10";
```

```
    }
```

O PSR-2 recomenda que funções e parênteses de estruturas de controle devem ser separadas por espaço antes da abertura e após o fechamento do parênteses, e sem espaço para o conteúdo interno.

Exemplo switch

```
<?php
    $month = 2;

    switch ($month) {
        case '1':
            echo "Mês: Janeiro";
            break;
        case '5':
            echo "Mês: Maio";
            break;
        default:
            echo "Mês: Inválido";
            break;
    }
```

O PSR-2 recomenda que funções e parênteses de estruturas de controle devem ser separadas por espaço antes da abertura e após o fechamento do parênteses, e sem espaço para o conteúdo interno.

Exemplo operador ternário

```
<?php
```

```
    $average = 5;
```

```
    echo $average >= 6 ? "Aprovado" : "Reprovado";
```

```
    $status = $average >= 6 ? "Aprovado" : "Reprovado";
```

```
    $status = $average >= 6 ? "Aprovado" : ($average >= 4 ? "Exame" :  
    "Reprovado");
```


Operadores de String

- **Ponto(.)** - realiza a concatenação dos argumentos presentes nos lados direito e esquerdo.

```
<?php
```

```
    $firstName = "Rafael Alves";
```

```
    $lastName = "Florindo";
```

```
    $name = $firstName . " " . $lastName;
```

```
    echo $name;
```

Operadores de String

- **Ponto e igual(.=)** realiza a concatenação utilizando uma variável declarada anteriormente com o argumento desejado, variável ou não.

```
<?php
```

```
    $firstName = "Rafael Alves";
```

```
    $lastName = "Florindo";
```

```
    $name = $firstName . " " . $lastName;
```

```
    echo $name;
```

Operadores de Atribuição

No caso de operadores de atribuição, o PHP fornece um operador básico e diversos derivados de atribuição, a operação é feita entre os dois operandos, sendo atribuído o resultado ao primeiro.

Atribuição simples(=)	Atribuição com adição (+=)	Atribuição com subtração (-=)
\$a = 10;	\$a += 15;	\$a -= 15;
Atribuição com multiplicação (*=)	Atribuição com divisão (/=)	
\$a *= 2;	\$a /= 2;	

Operadores relacionais

Operadores de comparação ou condicionais, são aqueles capazes de fazer comparações entre variáveis, com eles podemos saber se uma variável é maior que a outra, diferente, etc. Veja abaixo os operadores de comparação:

(==) Igual a	(===) Idêntico, e do mesmo tipo de dados	(!=),(<>) Diferente de
(!==) Não idêntico	(<) Menor que	(>) Maior que
(<=) Menor ou igual a	(>=) Maior ou igual a	

Prática



Materiais extras

<https://www.dicasux.com.br/figma/figma-tutorial-prototipo/>

<https://www.w3c.br/pub/Cursos/CursoHTML5/html5-web.pdf>



