

Material 01 - PHP

- Windows, Apache e PHP





Nessa Aula:

- Introdução ao PHP,
- Histórico da Linguagem,
- Versões,
- Bases de dados, protocolos e linguagens,
- Características,
- Arquitetura Cliente X Servidor
- Instalação no Windows,



PHP

- PHP (um acrônimo recursivo para "PHP: Hypertext Preprocessor") é uma linguagem de script **Open Source** de uso geral, utilizada para o desenvolvimento de aplicações Web dentro do HTML.

Histórico da Linguagem

do PHP/FI ao PHP 7

PHP/FI PHP/FI 2.0 PHP 3.0 PHP 4.0 PHP 5.X PHP 7.X PHP 8



1995

ATUAL

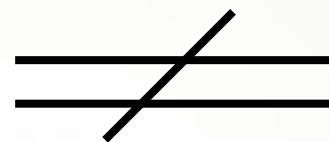
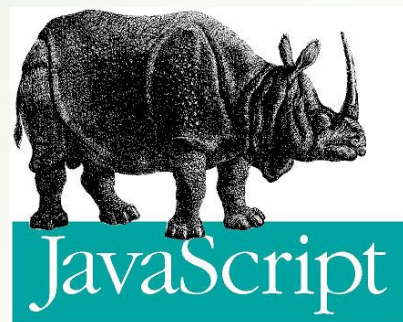
Histórico

- 1995:
 - Um subconjunto de scripts Perl criados por Rasmus Lerdof.
- 1997:
 - Com as adições de Zeev Suraski e Andi Gutmans (israelitas) reescreveram o parser, era assim lançada PHP 3, primeira versão estável e parecida com a linguagem atual. Ao reescrever o parser, foi criado o Zend Engine, que é mantido oficialmente pela empresa Zend em conjunto com a comunidade PHP.
- 2000:
 - 05/2000 lançada versão 4.
- 2004:
 - Lançada a versão 5, onde a principal mudança foi uma nova API para orientação a objetos provida pelo Zend Engine 2.
- 2015:
 - versão 7.x
- Atual
 - Versão 8.x



Versões

- Existem versões do PHP disponíveis para os seguintes sistemas operacionais:
 - Windows,
 - Linux,
 - Mac OS,
 - OS/2,
 - AS/400,
 - Novell Netware,
 - RISC OS, IRIX e Solaris.
- A Wikipedia funciona sobre um software inteiramente escrito em PHP, usando bases de dados MySQL: o MediaWiki.



- O que distingue o PHP de Javascript no lado do cliente é que o código é executado no servidor.
- O cliente recebe os resultados da execução de um script PHP, sem nenhuma condição de determinar como é o código fonte.



Bases de Dados

- Construir uma página dinâmica baseada em bases de dados é simples com PHP, este provê suporte a um grande número de bases de dados:
 - Oracle,
 - Sybase,
 - PostgreSQL,
 - InterBase,
 - MySQL,
 - SQLite,
 - MSSQL etc, podendo abstrair o banco com a biblioteca ADOdb, entre outras.



Open source

- Uma das vantagens do PHP é que ele é distribuído gratuitamente através do site <http://www.php.net>.
- No site oficial do PHP você encontra as versões mais recentes para downloads.
- Seu código-fonte é aberto, e você encontra toda a documentação do software também no site.



Cliente X Servidor

- O PHP → Baseado no servidor
 - Quando você executa uma página PHP no seu navegador, todo o código PHP é executado no servidor de origem da página, ou seja, o navegador apenas exibe a página processada.



Instalando o PHP

Instalando no Windows



Instalando no Windows

- Usaremos a versão 7.4.x (Estável)
- Passos para instalação:
- Acessar o link:
- <https://windows.php.net/download#php-7.4>

Baixar a versão correta para seu sistema operacional

- Apresento como realizar a instalação no Windows.
- Versão usada:
- **VC15 x64 Non Thread Safe (2021-Feb-02 23:05:42)**

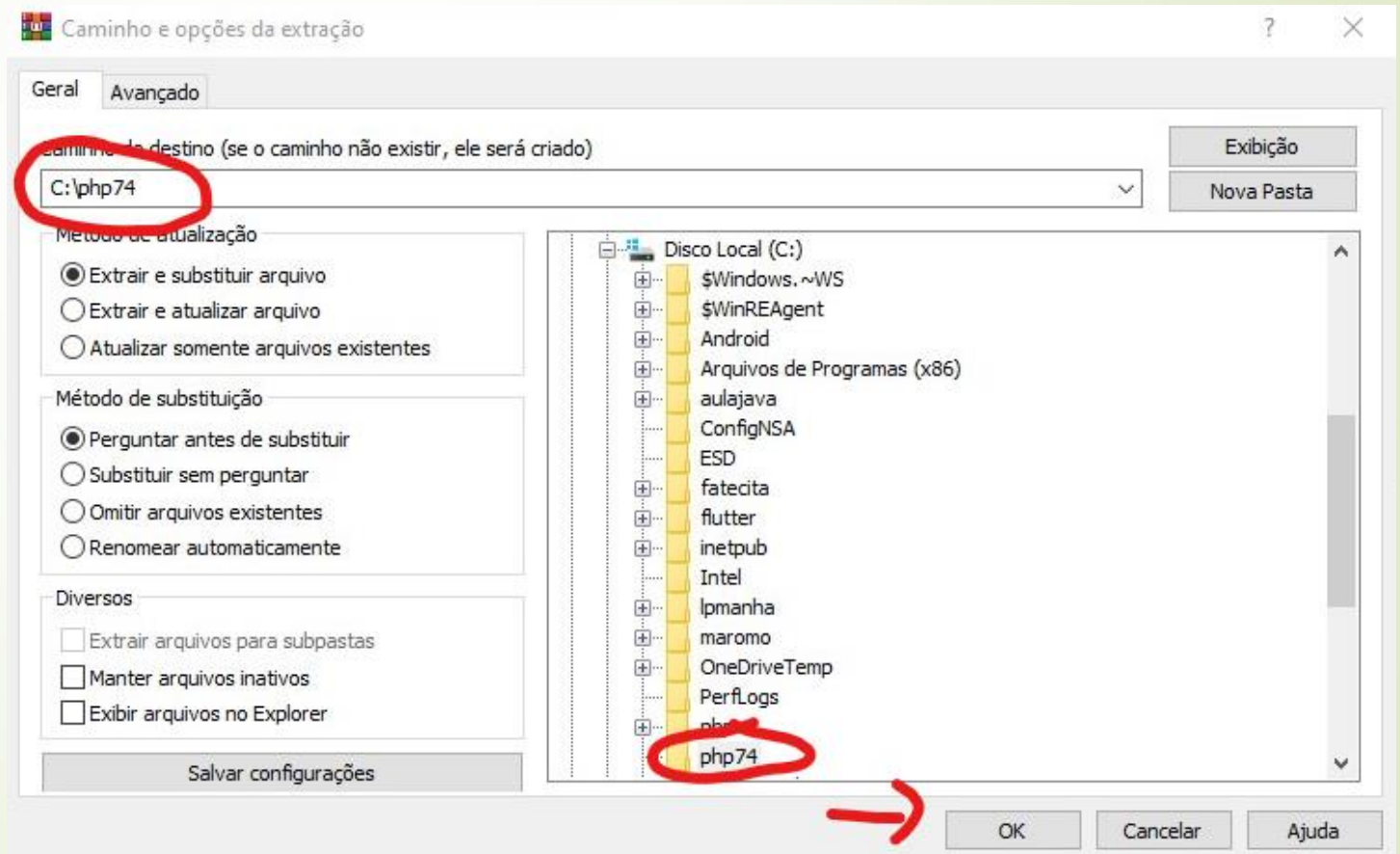
VC15 x64 Non Thread Safe (2021-Feb-02 23:05:42)

- [Zip](#) [24.85MB]

sha256: 25cce9343993e703597b92297f33a2337eceadd2fcdd329e1d823140c4c5778e

Descompactar

- Descompacte na localização:
- C:\php74\





Renomear e configurar

- Renomeie o arquivo **php.ini-development** para **php.ini**.



Visual Studio Code

- Use o Visual Studio Code para realizar as seguintes alterações:
- busque pela configuração **memory_limit = 128M** e mude de 128M para **1G**. [CTRL + F]
- busque também pela linha onde se encontra a configuração **extension_dir = "ext"** e descomente a mesma (Tire o caractere ';' do início da linha).

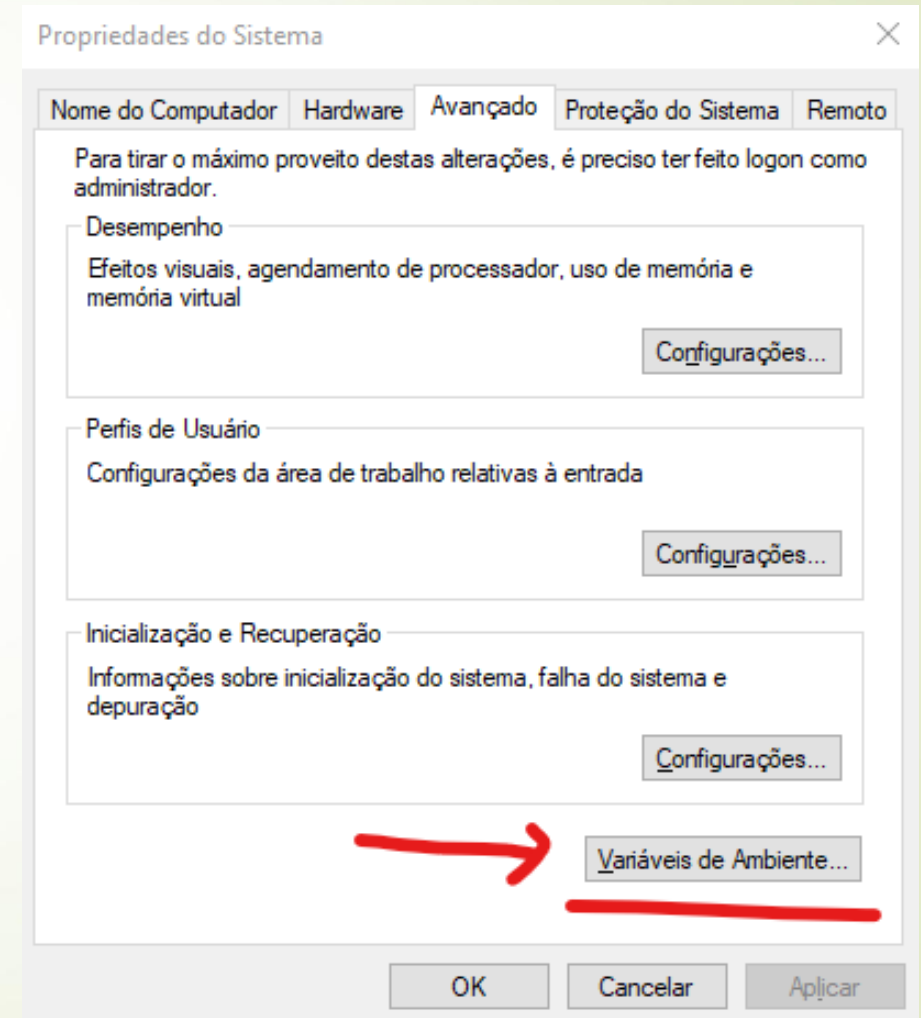


Visual Studio Code

- As linhas das extensões abaixo também devem ser descomentadas. (Tire o caractere ';' do inicio de cada uma das linhas onde constam as configurações:
 - `extension=gd2`
 - `extension= curl`
 - `extension=mbstring`
 - `extension=openssl`
 - `extension=pdo_mysql`
 - `extension=pdo_pgsql`
 - `extension=pdo_sqlite`
 - `extension=sockets`

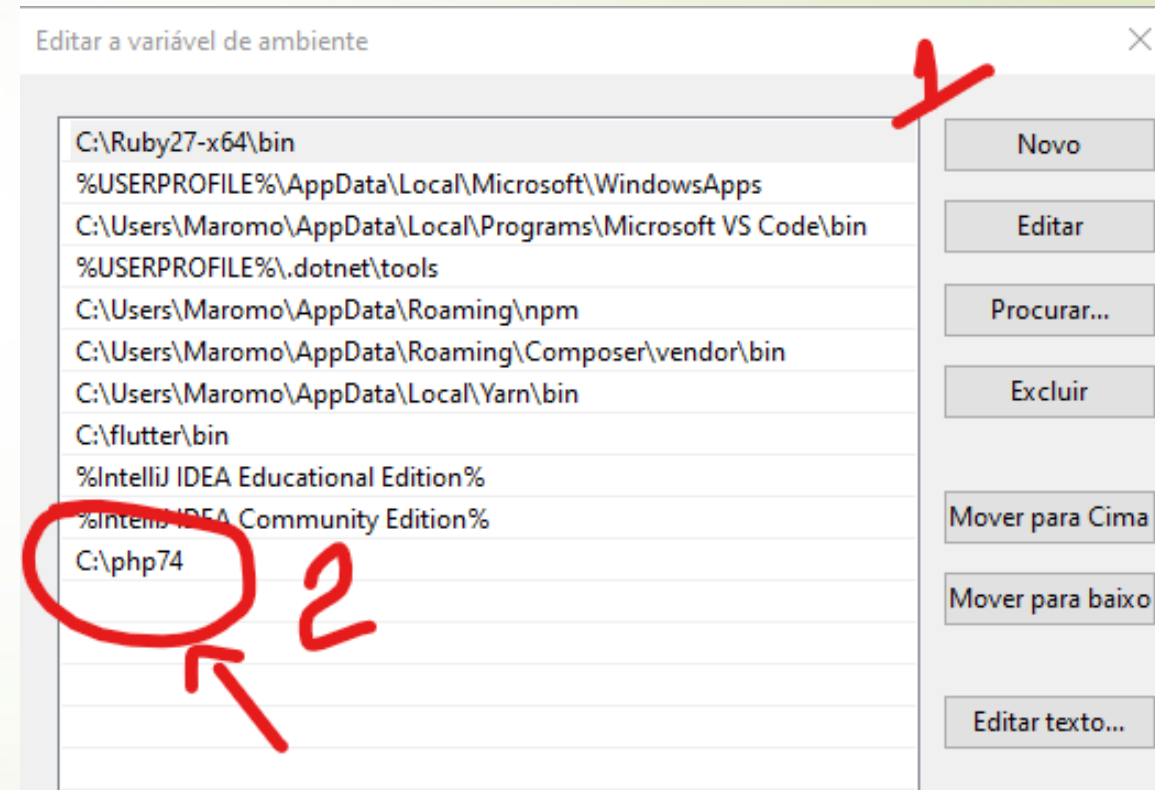
Editando as Variáveis de Ambiente do Windows 10

- No seu Painel de Controle do Windows, navegue até Sistema e Segurança > Sistema > Configurações avançadas do sistema. Clique em **Variáveis de Ambiente**.



Caminho para o php

- Na nova janela que se abriu localize a variável PATH, dê um duplo clique nela. Abrirá uma nova janela onde você deve clicar em **Novo** e digitar **C:\php74**



Verificando se tudo deu certo

- No prompt de comando, digite:

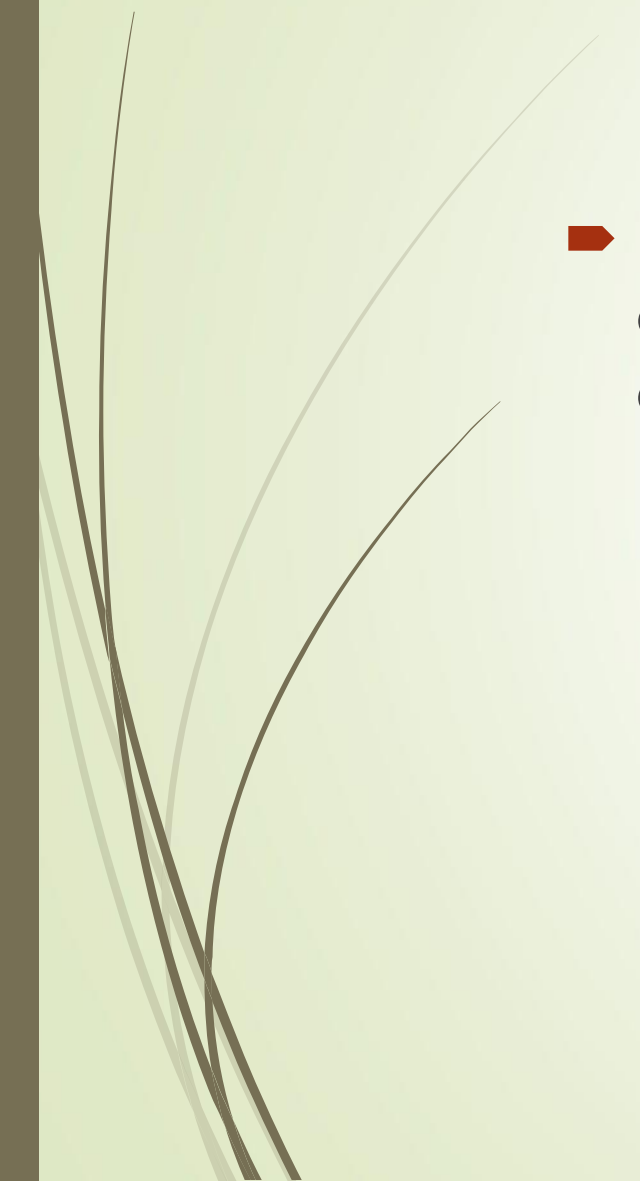


```
C:\Users\Maromo>php --version
PHP 7.4.15 (cli) (built: Feb  2 2021 20:47:36) ( NTS Visual C++ 2017 x64 )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies

C:\Users\Maromo>
```



Testando o PHP

- Para criar e editar scripts em PHP podemos utilizar qualquer editor de textos de preferência, ou até mesmo o bloco de notas. Usaremos o Visual Studio Code.
- 



hello.php

```
<?php  
    echo "Hello World!!";
```



Entendendo o código

- Um código php pode conter ou não tags Html, essas tags não são processadas pelo servidor, são simplesmente passadas ao browser solicitante.
- Normalmente utiliza-se Html para fazer a parte estática da página, sua estrutura e o php para a parte lógica, que exige processamento. “Deve-se salvar os códigos em PHP com extensão “.php”.
- Nesse material, considera-se que o aluno tenha conhecimentos básicos de HTML, quando necessário explicarei os códigos. 😊



TAGS PHP

- Há quatro conjuntos de tags que podem ser usadas para marcar blocos de código PHP. Delas, somente duas (`<?php...?>` e `<script language="php">...</script>`) são sempre disponíveis.
- As outras podem ser ativadas ou desativadas a partir do arquivo de configuração `php.ini`.

TAGS PHP

- Vamos usar no início a sintaxe configurado no arquivo php.ini abaixo:

```
<?php
```

Código php

- **Nota:** Em arquivos que possuem apenas código PHP o fechamento da tag é opcional. Aliás, considera-se que nesses casos a TAG php não deve ser fechada, pois assim você evita a inserção accidental de uma quebra de linha na resposta PHP gerada.



Notas sobre o echo

- O comando echo retorna uma string para o resultado em html, podemos passar esta string diretamente (como no exemplo) ou uma variável contendo uma string.
- Quando o PHP interpreta um arquivo, ele repassa o texto do arquivo até encontrar uma das tags especiais que lhe diz para começar a interpretar o texto como código PHP.
- O interpretador então executa todo o código que encontra, até chegar em uma tag de fechamento PHP, que novamente o coloca simplesmente repassando texto novamente.



Variáveis

- As variáveis no PHP são representadas por um **cifrão (\$)** seguido pelo nome da variável.
- Os nomes de variável no PHP fazem distinção entre maiúsculas e minúsculas.
- Um nome de variável válido se inicia com uma letra ou sublinhado, seguido de qualquer número de letras, algarismos ou sublinhados.

Exemplos válidos e inválidos

```
<?php
$var = "Bob";
$Var = "Joe";
echo "$var, $Var";
// exhibe "Bob, Joe"

$4site = 'not yet';
// inválido; começa com um número

$_4site = 'not yet';
// válido; começa com um sublinhado

$stätte = 'mansikka';
// válido; 'ä' é um caracter ASCII (extendido) 228
```



Segundo Exemplo em PHP

- O próximo exemplo mostra como realizar a declaração de variáveis no PHP. Para tanto execute os passos a seguir.


Arquivo exemplo2.php

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="UTF-8">
5      <title></title>
6  </head>
7  <body>
8      <?php
9          $nome = "Seu nome completo aqui";
10         $cidade = "Mogi Mirim";
11         $estado = "SP";
12         $junta = "$cidade - $estado";
13         echo $junta;
14         $num = 10;
15         $valor = "15 bolas";
16         echo "<br />";
17         echo $num . $valor;
18     ?>
19 </body>
20 </html>
```

Neste caso o código em php encontra-se dentro de um arquivo no formato HTML5. Lembrando que a extensão do arquivo deve ser **.php**



Comentários

- **Observação:** você pode acrescentar comentários para as linhas de código de várias formas, umas delas é a utilização da string “//” antes do comentário.
 - Quando o interpretador PHP encontra essa sequência ele ignora o restante da linha.
- 

Comentários 2

- O PHP suporta comentários do 'C', 'C++' e Unix shell. Por exemplo

```
<?php
echo "Isto é um teste";
//Comentário de uma linha estilo C++
/* Isto é um comentário de mais de uma linha
e aqui temos outra linha */
echo "Isto é um outro teste";
echo "O último teste"; #Comentário no estilo Unix shell
```




Terceiro Exemplo

- Continuando ainda o assunto sobre tipo de dados, vamos para mais um exemplo, dessa vez utilizaremos duas variáveis que receberão a data e hora atual respectivamente.

Terceiro Exemplo

```
<?php
    $data = date("d/m/y", time());
    $hora = date("h:i", time());

    echo "Hoje é $data e agora são $hora <br />" ;
```



Terceiro Exemplo - Resultado

Hoje é 01/03/16 e agora são 11:45

No exemplo acima as variáveis \$data e \$hora, receberam o valor da data e hora respectiva do sistema através da função date().



Dados Alfanuméricos (Strings)

- São sequência de caracteres, que podem ser delimitados:
 - Delimitadores
 - **Aspas Simples:** delimita qualquer dado alfanumérico;
 - **Aspas Duplas:** interpolação de variáveis;
 - **Aspas Invertidas:** interpolação de comandos do sistema operacional.



exemplo_04.php

Exemplo de aspas simples

```
<?php
    echo 'Exemplo de Aspas Simples';
    echo '<br>';
    echo 'Delimita qualquer texto alfanumérico';
    echo '<br>';
    echo 'Em casos especiais quando tivermos uma aspas simples no meio';
    echo '<br>';
    echo 'Podemos usar \\ uma contra barra antes';
    echo 'Exemplo de contra barra. Bem vindo ao Maromo\'s site';
```



exemplo_05.php

Exemplo de aspas duplas

```
<?php
    $nome="Maromo"; //Variável string
    echo "Exemplo de Aspas Duplas. <br>";
    echo "Delimita qualquer texto alfanumérico e permite interpolação<br>";
    echo "de variáveis.";
    echo "<br>";
    echo "Veja o exemplo abaixo:";
    echo "Bem vindo ao site do $nome";
```



exemplo_06.php

Exemplo de aspas invertidas

```
<?php
```

```
    echo "Versão do Windows: "; echo `ver`;
```



Fim

➤ Obrigado,

➤ Maromo