

# Programação Web I

# Resumo da aula

- Comparar as vantagens e desvantagens do PHP em relação a outras linguagens.
- Instalar e configurar o ambiente de desenvolvimento web.
- Conhecer a sintaxe da linguagem PHP.
- Construir páginas web com PHP.

# O que é PHP?

PHP é uma linguagem que permite criar sites web dinâmicos, fundamentada nos dados submetidos pelo usuário e derivada dos dados contidos no banco de dados, que são alterados frequentemente.



Vamos pegar o exemplo de uma loja virtual. Os produtos estão sempre sofrendo alterações, seja no preço, na quantidade em estoque, nos produtos em promoções, nos lançamentos, etc. Hoje, quando você entra em uma loja virtual, verá alguns produtos em promoção, outros em lançamentos, com um determinado preço. Na próxima semana que você visitar o site, pode ser que os preços estejam mais baixos, por causa de novas promoções, ou aquele produto que você tinha visto não se encontre mais em estoque, já tenha sido vendido.

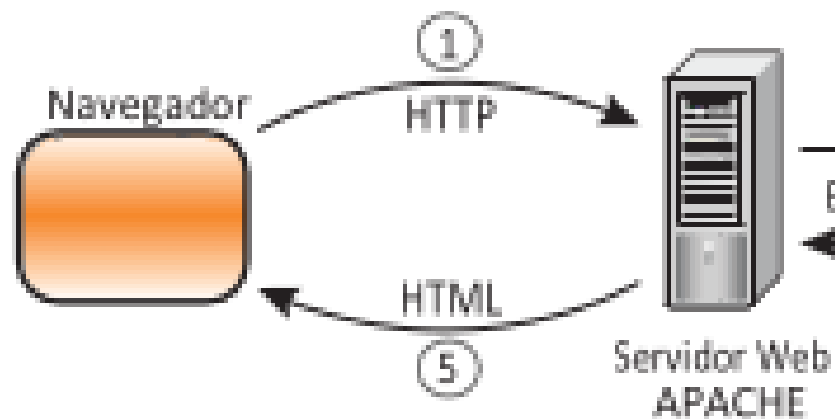




Passaremos a programar para web sob a visão do servidor. Para isso, utilizaremos a linguagem PHP. Como pré-requisito, é necessário o conhecimento de HTML, principalmente de formulário, que será utilizado para enviar dados para o servidor. Portanto é extremamente importante que você revise o conteúdo visto em Fundamentos do Desenvolvimento Web.

# Executado no servidor

O código PHP é executado no servidor, sendo enviado para o cliente apenas HTML. Dessa maneira é possível interagir com bancos de dados e aplicações existentes no servidor, com a vantagem de não expor o código fonte para o cliente.



# Instalação

Para testar as páginas PHP, não basta dar um duplo clique nos “arquivos.php”, como se faz com os .htm ou .html. É necessário ter um servidor web configurado para isso.

Um dos servidores web mais utilizados é o Apache.



# Testando o ambiente

No **Windows**, o diretório **www** fica dentro do diretório de instalação do produto. Por exemplo, se você usou o EasyPHP, o diretório é:

**C:\Arquivos de programas\EasyPHP-5.3.2\www**

O diretório “EasyPHP” pode mudar de nome de acordo com a versão instalada.

No **Linux**, o diretório é:  
**/var/www**



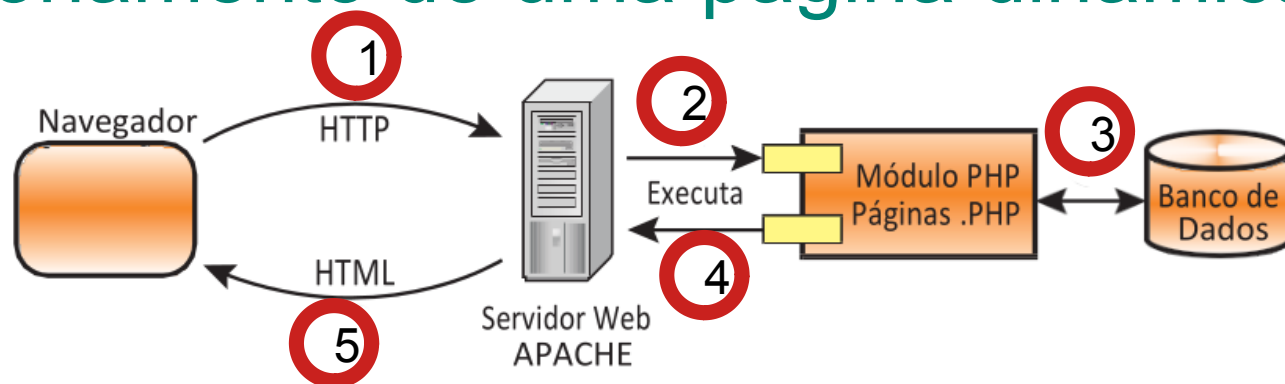
# Sintaxe básica

O código do PHP é embutido dentro de um arquivo HTML, quando for necessário algum processamento pelo servidor. Depois que o servidor processar o código PHP, apenas o que for gerado em HTML será enviado de volta para o usuário; assim, o usuário não conseguirá ver o código em PHP, que ficará apenas no servidor.

# Funcionamento de uma página dinâmica PHP

O código do PHP é embutido dentro de um arquivo HTML, quando for necessário algum processamento pelo servidor. Depois que o servidor processar o código PHP, apenas o que for gerado em HTML será enviado de volta para o usuário; assim, o usuário não conseguirá ver o código em PHP, que ficará apenas no servidor.

# Funcionamento de uma página dinâmica PHP



**Passo 1** – O usuário em seu navegador solicita uma página em PHP, por exemplo, consulta.php. Essa solicitação é enviada pelo protocolo HTTP ao servidor web da empresa (por exemplo, o Apache).

**Passo 2** – O Apache chama a página PHP que foi solicitada e a executa.

**Passo 3** – A página PHP pode ou não fazer acesso ao banco de dados.

**Passo 4** – Ao final da execução do programa PHP, uma página de resposta em HTML é enviada ao Apache.

**Passo 5** – O Servidor web Apache repassa a página de resposta para o navegador que a solicitou, que a exibe.

# Funcionamento de uma página dinâmica PHP

1) <?php Comandos; ?>	2) <? Comandos; ?>
3) <script language="php"> Comandos; </script>	4) <% Commandos; %>

**Figura 1.5: Delimitadores de código em PHP**

Fonte: Elaborada pelo autor

O primeiro delimitador da Figura é o padrão. O segundo é uma simplificação do primeiro. Esses são os mais usados. O terceiro segue o estilo de scripts em HTML. O quarto segue o estilo do ASP.

# Funcionamento de uma página dinâmica PHP

Aula 7 > 1 - code.php

```
1  <!DOCTYPE html>
2  <html lang="pt-BR">
3  <head>
4      <meta charset="UTF-8">
5      <title>Estrutura PHP</title>
6  </head>
7  <body>
8      <?php
9          // Coloque seu código aqui!
10     ?>
11 </body>
12 </html>
```

A mostra a estrutura de uma página HTML com o código PHP embutido.

# Gerando código em HTML

Aula 7 > 2-gerando\_html.php

```
1  <!DOCTYPE html>
2  <html lang="pt-BR">
3  <head>
4      <meta charset="UTF-8">
5      <title>Estrutura PHP</title>
6  </head>
7  <body>
8      <?php
9          echo "<H1>Página de teste do ECHO</H1>"
10         echo "<P>Parágrafo em HTML</P>"
11      ?>
12 </body>
13 </html>
```

Como o código PHP é processado no servidor e apenas o HTML é enviado como resposta, utilizaremos o comando *echo* para gerar esse HTML. Veja o exemplo da Figura.

# Gerando código em HTML

Agora é sua vez

Crie um arquivo **atividade3.php** que gere o seguinte resultado usando o php para gerar código HTML:

Curso de WEB I

Bem vindo João!



3 minutos



Universidade  
Estadual de Goiás



# Variáveis e tipos

Curso de WEB I

Bem vindo <?\$user\_name ?>!

As variáveis em PHP **não** precisam ser declaradas.

Ex. declarada:

```
string user_name;  
user_name="Marco";
```



Curso de WEB I

Bem vindo Marco!

Ex. não declarada:

```
user_name="Marco"
```





# Variáveis e tipos

Curso de WEB I

Bem vindo <?\$user\_name ?>!

Curso de WEB I

Bem vindo Marco!

Quando atribuímos algum valor para elas, o tipo é automaticamente reconhecido. Os tipos suportados são:

- Inteiros (integer ou long), Ex: 1
- Reais (float ou double), Ex: 1.5
- Strings, Ex: “Marco Antônio”
- Array (vetores), Ex: [1, 1.5, “Marco”]
- Objetos\*, Ex: Usuário

# Variáveis e tipos

Os nomes das variáveis devem ser criados com um '\$' seguido de uma *string* que deve ser inicializada por uma letra ou '\_'. Exemplos:

```
$x = 10.4;
```

```
$frase = "Exemplo de variável string";
```

```
$_cont = 0;
```

# Variáveis e tipos



O PHP é *case sensitive*, ou seja, letras maiúsculas são diferentes de minúsculas. Portanto, para facilitar, criem as variáveis sempre em minúsculo.

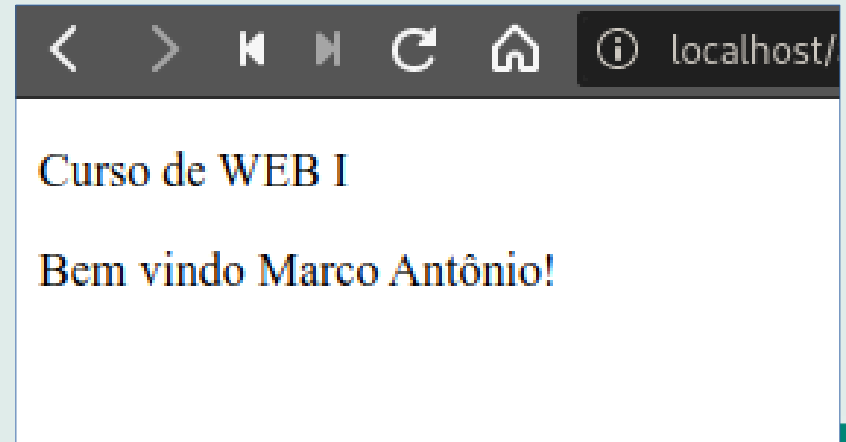
Os comentários podem ser de uma linha, utilizando o símbolo `//`, ou de mais linhas, delimitado pelos símbolos `/*` e `*/`. Exemplos:

```
$cont = 0; // Exemplo de comentário de uma linha
/* Exemplo de comentário com mais de uma linha. Preste atenção nos
    símbolos delimitadores. */
```

# Agora é sua vez

```
1 <!DOCTYPE html>
2 <html lang="pt-BR">
3 <head>
4     <meta charset="UTF-8">
5     <title>Estrutura PHP</title>
6 </head>
7 <body>
8     <?php
9         $user_name="Marco Antônio";
10    ?>
11    <p> Curso de WEB I </p>
12
13    Bem vindo <?php echo $user_name; ?>!
14 </body>
15 </html>
```

No arquivo **atividade3.php** crie uma variável que receba o seu nome e faça ecoar na tela:



5 minutos



## 1.6 Operadores

Os principais operadores em PHP estão descritos nos quadros a seguir: os **operadores aritméticos** podem ser vistos no Quadro 1.4, os **operadores lógicos** estão descritos no Quadro 1.5, os **operadores de comparação** são mostrados no Quadro 1.6 e demais operadores importantes estão no Quadro 1.7 a seguir.

Quadro 1.4: Operadores aritméticos

+	Adição
-	Subtração
*	Multiplicação
/	Divisão
%	Módulo (resto da divisão)

Fonte: Elaborado pelo autor

Quadro 1.5: Operadores lógicos

And	E
Or	Ou
Xor	Ou exclusivo
!	Negação
&&	E
	Ou

Fonte: Elaborado pelo autor

## 1.6 Operadores

Os principais operadores em PHP estão descritos nos quadros a seguir: os **operadores aritméticos** podem ser vistos no Quadro 1.4, os **operadores lógicos** estão descritos no Quadro 1.5, os **operadores de comparação** são mostrados no Quadro 1.6 e demais operadores importantes estão no Quadro 1.7 a seguir.

Quadro 1.6: Operadores de comparação

==	Igual a
!=	Diferente de
<	Menor que
>	Maior que
<=	Menor ou igual a
>=	Maior ou igual a

Fonte: Elaborado pelo autor

Quadro 1.7: Outros operadores importantes

•	Concatenação de <i>strings</i>
=	Atribuição
++	Incremento (soma 1)
--	Decremento (subtrai 1)

Fonte: Elaborado pelo autor

## 1.6 Operadores

Os principais operadores em PHP estão descritos nos quadros a seguir: os **operadores aritméticos** podem ser vistos no Quadro 1.4, os **operadores lógicos** estão descritos no Quadro 1.5, os **operadores de comparação** são mostrados no Quadro 1.6 e demais operadores importantes estão no Quadro 1.7 a seguir.

Quadro 1.6: Operadores de comparação

==	Igual a
!=	Diferente de
<	Menor que
>	Maior que
<=	Menor ou igual a
>=	Maior ou igual a

Fonte: Elaborado pelo autor

Quadro 1.7: Outros operadores importantes

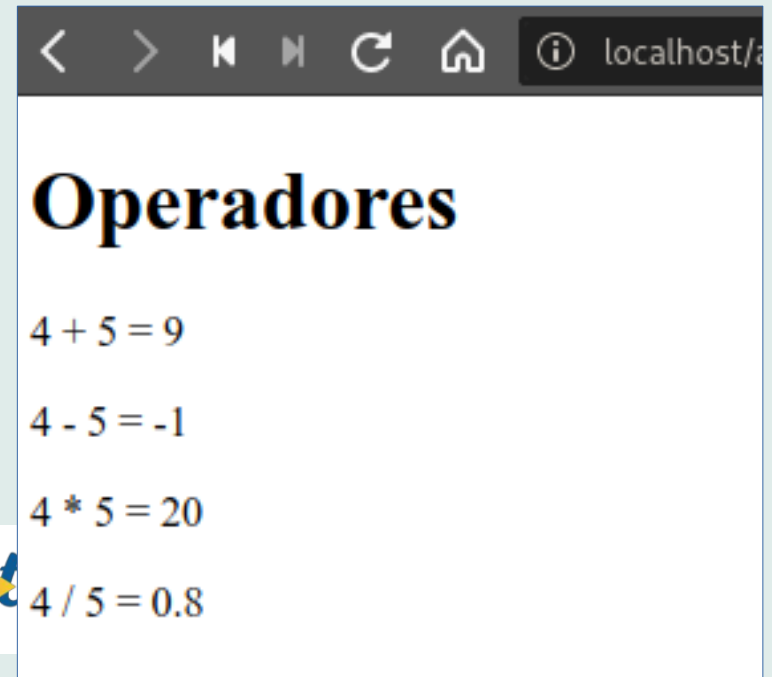
•	Concatenação de <i>strings</i>
=	Atribuição
++	Incremento (soma 1)
--	Decremento (subtrai 1)

Fonte: Elaborado pelo autor

# Agora é sua vez

```
1 <!DOCTYPE html>
2 <html lang="pt-BR">
3 <head>
4     <meta charset="UTF-8">
5     <title>Operadores</title>
6 </head>
7 <body>
8     <h1>Operadores</h1>
9     <?php
10         echo "</p> 4 + 5 = ".(4+5);
11         echo "</p> 4 - 5 = ".(4-5);
12         echo "</p> 4 * 5 = ".(4*5);
13         echo "</p> 4 / 5 = ".(4/5);
14     ?>
15 </body>
16 </html>
```

Crie **atividade4.php** e execute as operações de adição, subtração, divisão dos números 4 e 5:





# Referências

[https://www.php.net/manual/pt\\_BR/](https://www.php.net/manual/pt_BR/)

