

Estudo sobre o PHP

1221194773 - João Vitor Rezende Moura

1 Introdução

PHP (Preprocessador de Hypertexto) é uma linguagem de programação amplamente utilizada para desenvolvimento web, principalmente dentro do HTML, embutido, e é uma linguagem de servidor, que também pode ter outros usos por ser de propósito geral. Um exemplo de código PHP é o seguinte:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Página estudando php</title>
  </head>

  <body>
    <?php
      echo "Hi, im a script!";
    ?>
  </body>
</html>
```

Em vez de um monte de comandos, podemos usar o PHP para usar código HTML, e dentro dele colocar o php embutido, para que assim ele faça o processamento das informações igualmente, mas de forma muito mais arrumada, e eficiente. PHP é diferente de outras linguagens, e de tecnologias do client-side, como o JavaScript, porque ele é executado do lado do servidor. gerando o HTML completo que será enviado ao cliente. Podemos inclusive configurar o servidor web para processar todas as páginas html com php, e não será possível detectar sua presença pelo lado do usuário.

1.1 Principais áreas de uso do php

1.2 Pré-requisitos

Para podermos fazer uso do php, precisamos ter um servidor o qual suporte essa linguagem, para que o desenvolvimento de aplicações possa ser feito com ela de forma correta. Precisa-se instalar um web server com **Apache**, php e de banco de dados, costuma-se utilizar o **MySQL**. Essas ferramentas possibilitam rodar o php, e possibilitar o armazenamento de dados que pode ser utilizado durante a construção da nossa aplicação, respectivamente.

1.3 Primeira aplicação PHP

```
<!DOCTYPE html>
<html>
  <head>
    <title>PHP Test</title>
  </head>
  <body>
    <?php echo '<p>Hello World</p>'; ?>
  </body>
</html>
```

Esse código, permite que criemos um site HTML, e que dentro dele pré-processemos um código php que permite a criação de um parágrafo html, o arquivo deve possuir o nome de `hello.php`, e podemos com isso, acessar ele dentro do nosso navegador, que quando tiver o servidor rodando, pode ser acessado por meio de: `http://localhost/hello.php`

2 Conceitos Iniciais

2.1 Tags PHP

As tags do PHP, são a forma de indicarmos para o processador do arquivo o qual estamos escrevendo, onde está o fim e o início daquele determinado código php, pois elas representam a abertura e o fechamento, e são respectivamente `<?php` e `?>`. O PHP também inclui uma tag encurtada, que pode ser utilizada quando o único comando o qual for utilizado dentro daquele determinado espaço é o `echo`, e essa tag se dá por `<?=`. Essas tags curtas estão habilitadas por padrão, mas podemos desabilitar elas por meio da diretiva **short_open_tag**, presente dentro do arquivo de configuração *php.ini* com a configuração **—disable-short-tags**

2.2 Escapando HTML

Tudo fora do par de tags do PHP é ignorado pelo interpretador do php, e é entendido como HTML, possibilitando assim a mistura dessas duas linguagens dentro do mesmo arquivo, o que pode ser muito útil na hora de criarmos templates. Isso fará com que o interpretador comece a repassar o conteúdo que encontre posteriormente à tag de fechamento, como html, a menos que a instrução passada dentro do PHP seja uma condicional, como no exemplo a seguir:

```
<?php if ($expression == true): ?>
    Isso irá aparecer se a expressão for verdadeira.
<?php else: ?>
    Senão isso irá aparecer.
<?php endif; ?>
```

2.3 Separação de Instruções

Como em outras linguagens da mesma época que o PHP foi criado, as instruções devem ser divididas entre si por meio de ponto e vírgula. A tag de fechamento de um bloco de código PHP implica automaticamente em um ponto e vírgula. Um tipo especial de instrução são os comentários, os quais permitem que nos escrevamos informações úteis dentro do código, mas que não necessariamente vão ser processadas pelo interpretador e serem mostradas ao usuário, e esses comentários podem ser feitos de três formas:

- `//` - Comentários de linha única
- `/**/` - Comentários multilinha
- `#` - Comentários de linha única

2.4 Tipos primitivos dentro do PHP

Os tipos primitivos são os tipos de dados que alguma variável ou informação dentro do código PHP pode assumir para si, sendo os tipos primitivos da linguagem os seguintes:

- | | | |
|--------|----------|------------|
| • null | • float | • object |
| • bool | • string | • callable |
| • int | • array | • resource |

O PHP é linguagem com tipagem dinâmica, ou seja, por padrão não existe a necessidade de especificar o tipo da variável, já que isso será determinado em tempo de execução. Entretanto, é possível de restringir estaticamente os tipos através da declaração desses tipos primitivos. Essas tipagens restringem determinadas operações de serem feitas dentro do nosso código. Entretanto, uma expressão ou uma variável utilizada numa operação que o tipo não dê suporte, faz com que o interpretador tente transformar o valor em algum que permite a solução da operação.

Para verificar o valor e o tipo de alguma determinada expressão, podemos usar a função `var_dump()`, e para extrair o tipo de uma variável, podemos usar `get_debug_type()`.

```
<?php
$a_bool = true;    // um valor booleano
$a_str  = "foo";   // um texto
$a_str2 = 'foo';   // um texto
$a_int  = 12;      // um inteiro

echo get_debug_type($a_bool), "\n";
echo get_debug_type($a_str), "\n";
```

```
// Se essa variável conter um inteiro, aumento o número por quatro
if (is_int($an_int)) {
    $an_int += 4;
}
var_dump($an_int);

// Se $a_bool for um texto, imprima
if (is_string($a_bool)) {
    echo "String: $a_bool";
}
?>
```

As variáveis podem ter comportamentos e atributos específicos conforme o tipo primitivo o qual elas pertencem, sendo assim, podemos entender de uma determinada maneira para cada tipo que elas podem possuir

2.4.1 Características e atributos dos tipos primitivos

Cada tipo primitivo possui sua peculiaridade específica, que pode ou não ser compartilhada com outros tipos, e temos que são:

- **null:** É o tipo unitário do PHP, pode apenas assumir o valor de nulo, e ele é designado para variáveis indefinidas e variáveis que passam pelo método `unset()`.
- **booleanos:** São variáveis que representam valores lógicas, ou seja, de verdadeiro ou falso.
- **inteiros:** São variáveis as quais permite guardamos valores numéricos que não possuem casas decimais
- **float:** São variáveis que assim como os inteiros, guardam valores numéricos, mas nesse tipo específico, elas podem assumir uma parte que é decimal dentro da sua informação.
- **string:** É uma série de caracteres o qual um caractere é representado por um byte. Atualmente dentro do php, podem ser utilizados os dois tipos de aspas para definir uma string, e além disso, temos a sintaxe heredoc e temos a sintaxe nowdoc.
- **strings numéricas:** Strings numéricas são strings que podem ter seu conteúdo
- **array:** Um array pode ser entendido como um mapa ordenado, que define relação entre valores e suas respectivas chaves
- **object:** São objetos, que são elementos presentes do paradigma orientado à objetos

- **callable:**
- **resource:** É uma variável especial que mantém uma referência a um recurso externo

2.4.2 Valores lógicos e numéricos

Valores Booleanos Dentro do PHP, podemos ter valores lógicos, ou seja, determinar uma variável com valor de verdadeiro ou falso, comparar alguma variável com esse valor, ou modificar expressões lógicas por meio desses valores, uma forma de ver isso dentro do PHP, é da seguinte forma:

```
<?php
    if($action == "mostrar_versao"){
        echo "Versão 1.23";
    }
    if($exibir_comentários == TRUE){
        echo "<hr>\n";
    }
?>
```

Nesse código podemos ver, que uma determinada parte dele compara o valor de uma variável com um valor lógico booleano, vendo se é verdadeiro ou falsa necessidade de exibir os comentários ad página. Entretanto, vemos que a estrutura condicional verifica se uma comparação é verdadeira, com isso, não precisamos necessariamente fazer essa comparação, podemos apenas colocar dentro da condicional o valor da variável em si.

Valores Inteiros