

[Glossário](#)

set 03, 2021

Carlos E.

8min de leitura

O Que é PHP? Guia Básico de Programação PHP



Numa explicação de poucas palavras, **PHP** é uma [linguagem de programação](#) utilizada por programadores e desenvolvedores para construir sites dinâmicos, extensões de integração de aplicações e agilizar no desenvolvimento de um sistema.

Essa linguagem é mundialmente conhecida e uma das mais utilizadas pela facilidade em aprendê-la, manuseá-la, além de ser compatível com quase todos os sistemas operacionais que existem – o que torna seu custo menor. Neste artigo nós vamos mostrar a importância dessa sigla, além da estrutura e vantagens dessa linguagem.

Vamos lá?

Conteúdo

[Linguagens de Scripts](#)[Quando Surgiu e Como Funciona](#)[O Que é PHP?](#)[Por Que Usar PHP?](#)[PHP vs JavaScript](#)[Conclusão](#)

Linguagens de Scripts

PHP é um acrônimo recursivo para **PHP: Hypertext Preprocessor** (Pré-Processador de Hipertexto), que originalmente se chamava Personal Home Page (Página Inicial Pessoal).

Ele também é um subconjunto de linguagens de scripts como [JavaScript](#) e [Python](#). A diferença é que PHP costuma ser mais usado para comunicação do lado do servidor (*back-end*).

Parece confuso? É por isso que precisamos primeiramente entender sobre linguagens de script antes de mergulhar de maneira profunda no PHP.

Uma linguagem de script serve para automatizar a execução de tarefas num ambiente de tempo de execução especial. Isso inclui dizer para uma página estática (construída com [HTML](#) e [CSS](#)) para executar ações específicas com regras que você definiu anteriormente.

Por exemplo, você pode usar um script para validar uma forma de garantir que todos os campos de um formulário foram preenchidos antes que eles sejam enviados de volta para o servidor.

O script, então, iria rodar e checar todos os campos quando um usuário enviar o formulário. Se um deles estiver vazio, um texto de alerta seria exibido para informar o usuário disso.

Outros usos comuns de linguagens de script incluem mostrar um efeito de menu drop-down quando o cursor do mouse é passado sobre o menu principal, exibir botões de rolagem e animações, abrir caixas de diálogo, e assim em diante.

Quando Surgiu e Como Funciona

Nos dias atuais, o PHP, é uma das linguagens de programação mais usadas no mundo. O termo PHP foi criado com apenas um aglomerado de códigos CGI – um elemento que torna a ligação física ou lógica entre dois sistemas ou servidores, descritos em uma linguagem C.

A ideia inicial era acompanhar o tráfego do site pessoal do criador. Os anos passaram e o criador desenvolvia scripts, o que aumentava os recursos que o site dele tinha.

O sucesso dessa linguagem foi tão grande que o criador, Rasmus Lerdof, transformou o aglomerado de códigos CGI em uma linguagem de programação. Com isso, a grande maioria dos sites e aplicações passaram a utilizar o PHP como linguagem principal.

Versões PHP

Em 1997, dois programadores reformularam os códigos do **PHP** e lançaram o **PHP 3**, mudando o nome de *Personal Home Page*, para *PHP Hypertext Preprocessor*, como falamos acima. Essa versão contou com a primeira forma de recursos de orientações a objetos.

Com essa atualização, os programadores podiam implementar métodos e códigos. Testes públicos foram feitos e em junho do ano seguinte ele foi lançado de maneira oficial. Dois anos depois, a versão **PHP 4** foi lançada, dando mais recursos para a linguagem.

Essa versão apresentou um problema: a possibilidade de criar cópias de objetos. Isso porque a linguagem não trabalhava com handlers ou apontadores. Isso foi corrigido na versão **PHP 5.0**, a mais utilizada pelos programadores.

Atualmente, o **PHP** está na versão **7.2**. E não, não houve um salto da versão 5.0 para a 7.2, apenas a preferência de programadores pela versão 5.0 ou 5.3 entre as versões 6.0 e 7.0. É válido lembrar que as versões do PHP disponibilizadas podem variar de uma provedora, dependendo da [hospedagem PHP](#) oferecida.

Mas chega de história, pois para entender como utilizar o php na criação de um projeto, você precisa entender a linguagem que ela trabalha, ou melhor dizendo, os primeiros passos para programar algum sistema web.

Lado do Servidor (backend) e Lado do Cliente (frontend)

Linguagens de script podem rodar tanto do lado do servidor (backend) quanto do lado do cliente (frontend).

Scripts do tipo frontend são processados pelos navegadores. Quando o seu browser – também conhecido como o cliente – solicita uma página contendo scripts do lado do cliente, o servidor responde enviando os códigos-fontes que são executáveis pelo navegador.

Por outro lado, linguagem de scripts do tipo backend significam que esses scripts são executados nos servidores antes de serem enviados ao navegador. Então ao invés de mandar o código-fonte, os servidores da web processam (parse) os códigos antes ao transformá-los num formato HTML puro.



O Que é PHP?

Como mencionamos acima, trata-se de uma linguagem de script criada para comunicações do lado do servidor. Consequentemente, ela é capaz de lidar com várias funções de backend como coletar formulários de dados, gerenciar arquivos do servidor, modificar bases de dados e muito mais.

Apesar do PHP ser considerado uma linguagem de scripts de propósito geral, ela é mais usada para [desenvolvimento na web](#). Isso acontece por causa de um de seus recursos mais notáveis: a habilidade de ser integrado num arquivo HTML.

Vamos dizer que você não quer que as pessoas vejam o seu código-fonte. Para resolver isso, basta escondê-lo com essa linguagem de scripts. Simplesmente escreva o código num arquivo PHP, integre-o no HTML e, então, as pessoas nunca saberão qual é o conteúdo original.

Outro benefício desse recurso em particular é quando você tem que usar marcações HTML repetidamente. Ao invés de reescrevê-los de novo e de novo, apenas implemente o código num arquivo PHP. Quando precisar usar o HTML, insira o arquivo PHP e está tudo pronto.

Um exemplo de uma página PHP pode ser encontrado no Facebook. Abra a página inicial e você pode ver que a URL termina com .php (facebook.com/home.php).

Em outras palavras, essa página é criada com um arquivo php (home.php) que contém uma mistura de código de pré-processador de hipertexto e tags HTML.

Se você quer aprender mais sobre como um arquivo PHP é escrito, você pode ver um exemplo abaixo:

```
1. <html>
2. <head>
3. <title>Teste PHP</title>
4. </head>
5. <body>
6. <?php echo '<p>Saudações da Hostinger</p>'; ?>
7. </body>
8. </html>
```

Se você colocar esse arquivo PHP escrito no diretório raiz do seu site, iria encontrá-lo ao acessar seusite.com/saudacoes.php.

A saída desse arquivo PHP em HTML iria ficar assim:

```
1. <html>
2. <head>
3. <title>Teste PHP</title>
4. </head>
5. <body>
6. <p>Saudações da Hostinger</p>
7. </body>
8. </html>
```

Por Que Usar PHP?

PHP não é a única linguagem de script server-side disponível – existem muitas outras disponíveis. Entretanto, ela tem uma vantagem sobre seus competidores se você está rodando um site WordPress.

Como mencionamos anteriormente, o WordPress é construído usando PHP. Então ao usar essa linguagem, ele pode aumentar drasticamente a quantidade de customização que você pode ter no seu site.

Você pode usá-lo para modificar os [plugins](#) e [temas](#) disponíveis ou até mesmo criar o seu próprio. Se você quer se tornar um desenvolvedor [WordPress](#), aprender PHP é uma obrigação.

Com ele, você consegue criar aplicações para efetuarem alguma tarefa que o usuário determinar. Essas aplicações são compiladas dentro de um servidor, chamado de **server-side**, ou, script do lado do servidor – termo tradicional e muito utilizado pelos programadores quando se fala em **PHP**.

Para você conseguir fazer esses scripts funcionarem, vai precisar dos seguintes itens: um navegador, o interpretador do **PHP** e um servidor web. Você pode aprender a instalar esses itens no [guia de instalação e configuração do php](#).

Ainda falando em scripts, com o PHP você pode criar um script e utilizá-lo sem a necessidade de ter um navegador. Você precisa somente do interpretador. Essa função é muito útil para programadores que desejam executar o script para criação de APIs. O recurso é chamado de [execução de script via linha de comando](#).

Você também pode criar aplicações para desktop. Não é muito comum, mas se você é um programador que gosta de desafios, conseguirá utilizar os serviços mais avançados. Para fazer isso, você precisa utilizar o **PHP-GTK**, uma extensão do **PHP** que vai te ajudar a criar essas aplicações.

Caso você ainda não esteja convencido, aqui estão alguns outros motivos para entender que o PHP é uma ótima linguagem de script para se utilizar:

- **Fácil de aprender** — você pode aprendê-la facilmente já que ela tem [ótima documentação](#) sobre suas funções junto de exemplos.
- **Amplamente utilizada** — ela é utilizada para criar qualquer tipo de [plataformas](#) como [ecommerce](#), blogs, redes sociais e assim em diante. Estatísticas mostram que [79% de todos os sites](#) usam PHP!
- **Baixo custo** — ela é de código-aberto então você pode usá-la gratuitamente.
- **Grande comunidade** — se você encontrar qualquer problema com ela, não é necessário se preocupar porque há muitos blogs sobre PHP na internet.
- **Integração com bases de dados** — alguns exemplos são [MySQL](#), [Oracle](#), [Sybase](#), [DB2](#), etc.

Dicas de Aplicações Para o PHP

Desenvolver um site ou algum projeto na internet não é nada fácil. Existem coisas que você deve se preocupar bastante para que tudo ocorra perfeitamente, como a segurança.

Afinal, é necessário garantir a confiabilidade do seu sistema e dos usuários que o visitam. Por isso, os programadores precisam ficar cientes e atentos às ameaças que existem no mundo da internet.

1 – Cookies Diferentes

A primeira dica é evitar que ocorra roubo de sessão. Qualquer invasor tem conhecimento do nome padrão utilizado pelo **PHP** para salvar IDS, que é **PHPSESSID**. Ao mudar esse nome, você dificulta a entrada ou ação de algum invasor que queira prejudicar a aplicação.

2 – Acesse Cookies via HTTP

Os IDS são armazenados de maneira padrão, como acabamos de mostrar acima. Porém, a propriedade **session.cookie_httponly** não é padrão.

Dentro do **PHP**, existe uma configuração, onde é possível acessar os valores das variáveis salvas no navegador através do JavaScript, o que traz um risco para sua aplicação.

Portanto, é recomendável que seja habilitado o **session.cookie_httponly** no arquivo **php.ini**. Fazendo isso, você torna possível acessar as informações somente via HTTP.

Isso impediria o famoso **Cross-site scripting (XSS)**, um ataque que se aproveita da vulnerabilidade de um sistema e códigos JavaScripts são injetados para roubar informações do site que foi atacado.

Esse método não é exclusivo de páginas que foram programadas com linguagem PHP. É uma acesso básica na configuração do servidor web, porém importante.

Dentro do **PHP** você encontra diversas configurações armazenadas em arquivos **XML**, **txt**, **JSON**, **ini** e muitas outras. Quando você acessa esses arquivos pelo navegador, todos ficam visíveis. Se houverem informações ou arquivos confidenciais, você precisa se proteger.

Porém, essa configuração é somente para servidor web Apache, e que o **mod_rewrite** esteja habilitado para que o arquivo **.htaccess** funcione corretamente com as instruções abaixo:

```
1. <Files>
2.   order allow, deny
3.   deny from all
4. </Files>
```

Isso irá evitar o acesso.

PHP vs JavaScript

Nós mencionamos anteriormente que você pode fazer várias coisas com PHP. Contudo, ele é mais amplamente utilizado para gerar conteúdos dinâmicos.

Se você está familiarizado com o JavaScript, pode saber que ele também é usado para fazer com que uma página estática se torne mais dinâmica e interativa. Portanto, nós temos duas linguagens de script que possuem funções similares.

Por que precisamos de ambas? Quando se trata de criar um website dinâmico, de fato é possível projetar um que rode apenas em JavaScript. Contudo, a funcionalidade pode variar consideravelmente – não necessariamente no bom sentido.

Com o JavaScript, você pode criar um site dinâmico que permite interações simples como efeito de rolagem do mouse, autocorreção e galerias de fotos. Mas ele não pode criar conteúdo gerado por usuários como redes sociais e ecommerce (sites que mudam de acordo com o comportamento do usuário). Para fazer isso, você precisa de PHP.

Vamos dizer que você está fazendo o login numa rede social. Para mostrar o seu perfil, o PHP recupera os seus dados da base de dados e envia o resultado para o seu navegador.

Quando você promove uma alteração no seu perfil, o PHP irá guardar essa nova informação na base de dados para recuperações futuras. O processo total faz com que seja possível para uma rede social exibir diferentes conteúdos para usuários distintos.

No final das contas, isso significa que JavaScript e PHP não são competidores. Ao invés disso, eles complementam um ao outro para criar um site belo e dinâmico.

Conclusão

PHP é uma linguagem de script do tipo server-side com diversos propósitos. Porém, ela é principalmente utilizada para gerar conteúdos dinâmicos num site.

Trata-se de uma linguagem altamente popular devido à sua natureza de código aberto e suas funcionalidades versáteis. Ela é simples o suficiente para novatos, mas programadores profissionais podem também usá-la para recursos mais avançados.

Aprender e entender PHP indubitavelmente irá melhorar as suas habilidades de desenvolvedor – você poderá criar qualquer tipo de website que desejar, se tornar um desenvolvedor WordPress habilidoso e até mesmo criar uma aplicação para web um dia!

Então, como você pode ver, há todos os motivos para você aprender PHP, não é mesmo?



O AUTOR

Carlos E.

Carlos Estrella é formado em jornalismo pela UFSC e tem as funções de redator, tradutor e SEO na Hostinger Brasil. Já trabalhou com jornalismo de games e tecnologia e hoje aplica essa experiência escrevendo posts e tutoriais no blog da Hostinger. Suas paixões incluem games, dar rolês com a namorada e amigos e ler artigos aleatórios da Wikipédia de madrugada.

[Mais de Carlos E.](#)

 **HOSTINGER** TUTORIAIS

Domine o WordPress em 7
Dias com um Curso Grátis



Tutoriais relacionados

16 ago • [GLOSSÁRIO](#)

O Que é SEO? Conheça as Melhores Práticas para Chegar ao Topo dos Rankings em 2021

SEO significa search engine optimization, que representa a prática de otimizar as páginas do seu site para ranquear melhor nas páginas de...

Por Carlos E.

18 mar • [GLOSSÁRIO](#)

O Que É Um Blog? Uma Introdução ao Blogging

Inicialmente visto como uma maneira de compartilhar histórias com usuários da internet, nos anos recentes os blogs têm se tornado uma das...

Por Andrei L.

15 fev • [GLOSSÁRIO](#)

O Que é Brainstorming e Como Fazer uma Tempestade de Ideias

Brainstorming (ou tempestade de ideias) é uma técnica criativa para grupos que serve para tentar encontrar uma solução para um problema...

Por Carlos E.

Comentários

[Faça um comentário](#)



Danilo
maio 15 2020

[RESPONDER](#)

Muito boa a matéria!

Deixe uma resposta

Comentário*

Nome*

Email*

☐ Nós - e nossos parceiros - usamos cookies para oferecer nossos serviços e mostrar propagandas baseadas em seus interesses. Ao usar nosso website você concorda ao uso de cookies conforme explicado em nossa Política de Cookies. * Esse site utiliza o Akismet para reduzir spam. Aprenda como seus dados de comentários são processados.

Enviar



Nossa missão é facilitar a vida de desenvolvedores de sites e seus clientes. E fazemos isso oferecendo serviços de hospedagem fáceis de usar, rápidos e confiáveis.



HOSPEDAGEM

- [Hospedagem de Sites](#)
- [Hospedagem VPS](#)
- [Hospedagem Cloud](#)
- [Hospedagem WordPress](#)
- [Hospedagem de Email](#)
- [Hospedagem para CMS](#)
- [Hospedagem de Loja Virtual](#)
- [Hospedagem Grátis](#)
- [Criador de Sites](#)

DOMÍNIOS

- [Verificador de Domínio](#)
- [Transferência de Domínio](#)
- [Domínio Grátis](#)
- [Domínio XYZ](#)
- [Certificado SSL Barato](#)
- [Pesquisar WHOIS](#)

AJUDA

- [Reportar Abuso](#)

INFORMAÇÃO

- [Status do Servidor](#)
- [Mention Hostinger](#)
- [Reviews](#)
- [Métodos de Pagamento](#)
- [Programa de Afiliados](#)

LEGAL

- [Política de Privacidade](#)
- [Termos de Serviço](#)

Hostinger ePanel

HOSTINGER TUTORIAIS

Central de hospedagem

Base de Conhecimento

Tutoriais

Blog

Formulário de Migração

Termos de Migração

Carrinho

EMPRESA

Sobre a Hostinger

Fale Conosco

Carreiras

© 2004-2021 hostinger.com.br - Hospedagem de Sites, Cloud e VPS premium e Serviços de Registro de Domínio.

Os preços estão listados sem VAT

https://www.hostinger.com.br/tutoriais/o-que-e-php-guia-basico

9/9