1 Adicionando Vida às Suas Páginas Estáticas



Você tem criado ótimas páginas web com HTML, com recursos de

CSS mas tem percebido que os visitantes do seu site não podem fazer muita coisa além de olhar passivamente o conteúdo das páginas. A comunicação apresenta-se de forma unidirecional e você gostaria de mudar isso. Na verdade, o que você realmente gostaria é de saber o que o seu público está pensando. Mas para isso precisa permitir que os usuários digitem informações em um formulário web, para que possa saber o que eles têm em mente. E seu site precisa ser capaz de processar essas informações, para que elas lhe sejam entregues. Está começando a parecer que você irá precisar de mais do que HTML para levar seu site para o próximo nível.

HTML é estático e chato

O HTML é ótimo para se criar páginas web, até aí você já sabe. Mas, e quando você precisa que as páginas realizem alguma ação? Suponha que seja necessário fazer uma busca em um banco de dados ou enviar um email... como fazer? O HTML deixa a desejar, porque é uma linguagem sem vida, elaborada para exibir informações que nunca se modificam. Alô? O HTML é ótimo quando a intenção é apenas compartilhar uma foto do seu bicho de estimação... mas Estas pessoas não tanto quando deseja estão em busca interagir com os visitantes de interação. do seu site. O código HTML destas Páginas é determinado Servidor web quando o desenvolvedor web cria as páginas. Páginas HTML estáticas só se modificam quando o desenvolvedor web edita um arquivo html e o envia para o seu servidor web. O servidor web limita-se a servir uma página HTML Navegador estática atrás da web cliente outra. Com páginas HTML puras, o servidor O servidor web representa uma grande parte do problema com o HTML estático, uma simplesmente serve vez que ele funciona como nada mais que um mecanismo de entrega sem graça. O

navegador requisita uma página, o servidor responde com HTML, fim da história. Para transformar sites em **aplicações** web interativas, o servidor precisa assumir um papel novo, mais dinâmico... um papel possibilitado pelo PHP.

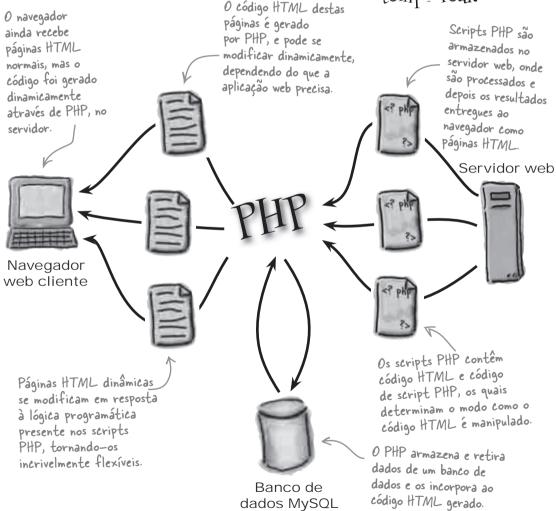
HTML estático, que só é capaz de exibir conteúdo.

Com uma pequena — ajuda do servidor!

PHP dá vida às páginas web

O PHP lhe permite manipular o conteúdo das páginas web *no servidor*, imediatamente antes de a página ser enviada ao navegador cliente. Funciona da seguinte maneira: um script PHP é executado no servidor, e pode alterar ou gerar código HTML à vontade. Uma página HTML então é entregue ao navegador, que não sabe e nem liga se há PHP envolvido na produção ou ajuste do HTML, no lado do servidor.

Com PHP na parada, o servidor web é capaz de gerar páginas HTML de forma dinâmica, em tempo real.



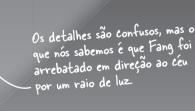
Cães no espaço

Este é o Owen. Ele perdeu seu cachorro, Fang. Mas encontrar seu cão não se restringe a uma simples tarefa de procurar pelo bairro. Acontece que Fang foi abduzido por alienígenas, o que expande a busca de Owen por todo o universo. Owen sabe um pouco de HTML e CSS e acha que um site personalizado poderia ajudá-lo a resolver seu problema, ao permitir que outras pessoas compartilhem suas próprias experiências de abdução alienígena.

Mas, para obter informações das pessoas, Owen precisará de um formulário web capaz de receber input (entrada de dados) dos usuários e notificá-lo sobre esse input. Sem problema – o HTML possui várias tags para a criação de formulários web.





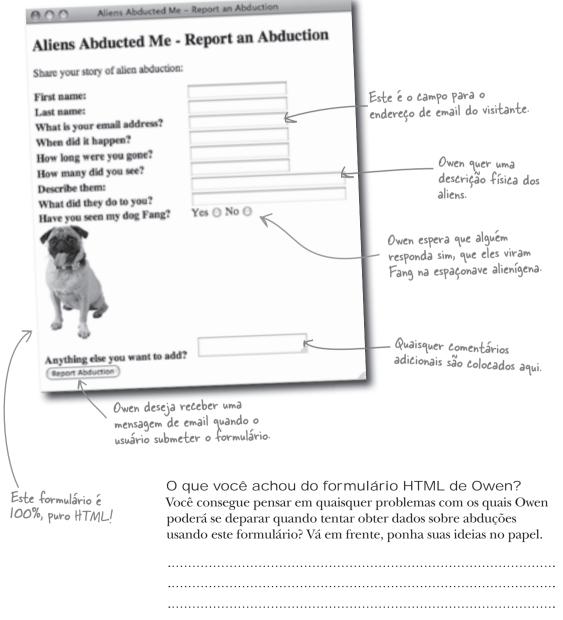


Owen sabe um pouco de HTML e CSS e acha que poderá usar a web como forma de ajudá-lo a descobrir o paradeiro de seu cão, Fang.



Um formulário ajuda Owen a descobrir toda a história

O novo site de Owen, AliensMeAbducteme.com, tem por objetivo conectá-lo com abduzidos que possam ajudar a esclarecer um pouco sobre o desaparecimento de Fang. Owen sabe que precisa de um formulário HTML para requisitar histórias de abdução dos visitantes e que precisa descobrir se por acaso, eles viram Fang durante as suas jornadas interestelares. Mas Owen precisa de ajuda para conseguir montar o site e colocá-lo no ar. Eis o que ele tem em mente para o formulário.



Formulários são feitos de HTML

O formulário "Relate uma Abdução de Owen" (baixe este código através do site da altabooks.com.br) se compõe, inteiramente, de tags e atributos HTML. Há campos de texto para a maioria das perguntas, botões de rádio para descobrir se o visitante viu Fang e uma área de texto para comentários adicionais. E o formulário está configurado para enviar os Owen irá receber o conteúdo dados recebidos para o endereço de email de Owen.

Se precisar de um refresco de memória

sobre como criar formulários HTML, dê

uma olhada no Capítulo

14 de Use a Cabeça!

HTML com CSS &

XHTML.

```
deste formulário enviado
 "mailto" é um protocolo que permite
                                                                   Para ele neste endereço de
  que dados de formulários sejam
                                                                   email - ponha aqui o seu
  enviados por email.
                                                                   próprio endereco para testar
Share your story of alien abduction:
                                                                   o formulário.
<form method="post" action="mailto:owen@aliensabductedme.com">
                                                                      Este valor diz ao servidor
  <label for="firstname">First name:</label>
                                                                      como enviar os dados.
                                                                      Será "post" ou "get". Nós
  <input type="text" id="firstname" name="firstname" /><br />
                                                                      explicaremos a diferença
  <label for="lastname">Last name:</label>
  <input type="text" id="lastname" name="lastname" /><br />
                                                                      mais à frente.
  <label for="email">What is your email address?</label>
                                                                     As tags input dizem ao
  <input type="text" id="email" name="email" /><br />
                                                                     formulário para esperar
  <label for="whenithappened">When did it happen?</label>
                                                                     informações, aqui.
  <input type="text" id="whenithappened" name="whenithappened" /><br />
  <label for="howlong">How long were you gone?</label>
  <input type="text" id="howlong" name="howlong" /><br />
                                                                  O atributo type diz à ação do
  <label for="howmany">How many did you see?</label>
                                                                  formulário para esperar texto.
  <input type="text" id="howmany" name="howmany" /><br />
  <label for="aliendescription">Describe them:</label>
  <input type="text" id="aliendescription" name="aliendescription" size="32" /><br />
  <label for="whattheydid">What did they do to you?</label>
  <input type="text" id="whattheydid" name="whattheydid" size="32" /><br />
  <label for="fangspotted">Have you seen my dog Fang?</label>
  Yes <input id="fangspotted" name="fangspotted" type="radio" value="yes" />
  No <input id="fangspotted" name="fangspotted" type="radio" value="no" /><br />
  <img src="fang.jpg" width="100" height="175"</pre>
    alt="My abducted dog Fang." /><br />
  <label for="other">Anything else you want to add?</label>
  <textarea id="other" name="other"></textarea><br />
  <input type="submit" value="Report Abduction" name="submit" />
</form>
                                                                 O botão "submeter"
O formulário fica contido
                             Nenhuma surpresa aqui - o
                                                                 diz ao formulário
dentro de tags <form> de
                            formulário é 100% puro
                                                                 para executar a ação
abertura e de fechamento.
                             código HTML!
                                                                 exposta ali.
```

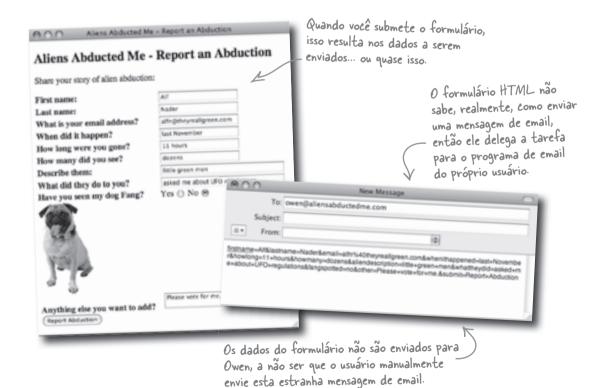


Teste o formulário "Relatar uma Abdução".

Baixe o código da página "Relatar uma Abdução" no site da Alta Books, em www.altabooks.com.br. O código estará no cadastro deste Livro. O folder ch1 contém o formulário de Owen em report.html, bem como uma folha de estilo (style.css) e uma imagem de Fang (fang.jpg).

Abra a página report.html em um editor de texto e troque o endereço de email de Owen pelo seu. Em seguida, abra a página em um navegador web, digite algumas informações no formulário e clique no botão "Relatar Abdução" (Report Abduction).





Então, o que você achou? Você recebeu os dados do formulário como uma mensagem de email na sua caixa de entrada?

O formulário HTML apresenta problemas

O formulário "Relatar uma Abdução" está no ar, operante, mas ele não consegue obter muitas informações dos usuários. Será que a abdução de Fang corresponde a um incidente isolado... ou tem algo de errado com o formulário. Vejamos o que os usuários têm a dizer.

Eu observei algo parecido com o seguinte no campo Assunto:

?When=&Where=. Estou confuso. s Abducted Me - Report an Abd

Quando clico no botão, ele abre o meu programa de email, o Outlook, e nada do que digitei, durante 15 minutos, ficou registrado no formulário!

Apareceu um email em branco. Todas as respostas que eu tive o trabalho de digitar foram ignoradas. Deveriam abduzir este formulário idiotal



0

Não aconteceu nada, porque o meu navegador não tem um cliente de email padrão... seja lá o que isso signifique.

0

Por algum motivo, o formulário de Owen está extraindo mais frustração do que informação dos visitantes do seu site.



O que está acontecendo aqui? Você tem ideias sobre como consertar o formulário?

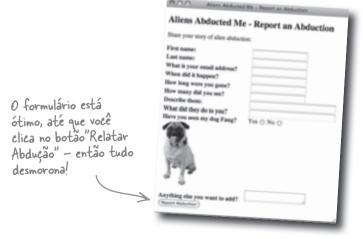
Parece estar tudo OK com o formulário. Será que o problema tem algo a ver com aquela parte do "mailto"?



Sim. O código HTML do formulário está correto, mas "mailto" não é uma boa maneira de enviar dados

O formulário de Owen funciona bem até o usuário clicar no botão "Relatar Abdução". Nesse ponto, você passa a depender do recurso de mailto para empacotar os dados do formulário em uma mensagem de email. Mas este email não é enviado automaticamente – ele é criado no programa de mail padrão do computador do usuário. E o pior: o usuário precisa **enviar ele mesmo o email**, para que os dados sejam realmente enviados para você! Você não tem nenhum controle sobre a entrega do email, significando que ele poderá – ou não – fazer a viagem do seu formulário web, passando pelo navegador do usuário e pelo seu cliente de email, até chegar a você como uma mensagem de email. Nada bom.

Você precisa de uma forma para ter o controle sobre a entrega do formulário web. Mais especificamente, você precisa que o PHP empacote os dados em uma mensagem de email, e depois se certifique de que essa mensagem será enviada. Isto envolve desviar a sua atenção do **cliente** (HTML, mailto, etc.) para o **servidor** (PHP).



HTML roda no cliente

O formulário de Owen foi escrito em HTML puro, com uma ação mailto que tenta enviar os dados via email. Embora a página report. html venha de um servidor web, ela é preenchida e processada inteiramente no navegador web do usuário. Agui está. Eu gostaria O navegador requisita a de receber a página página web de Owen, a 0 "Relatar uma Abducão". qual inclui o formulário 0 por favor. 0 0 O software navegador do seu computador roda O servidor retorna o agui, também O software servidor web código HTML para a conhecido como de Owen roda aqui, também página web. CLIENTE. conhecido como SERVIDOR. O usuário preenche o Agora, eu gostaria de Ahm, não formulário e o submete. submeter o formulário tem nenhum trabalho com os dados que o usuário para mim aqui. <form action digitou, por favor. "mailto: ... 0 0 A tag de ação do formulário diz ao navegador para Owen Poderá pedir ao programa de email receber ou do usuário que crie uma não o email. mensagem de email. O programa de email do usuário cria uma mensagem com os O servidor jamais dados do formulário – mas é o entra em contato usuário que precisa enviá-los, com os dados manualmente, a Owen. digitados em O papel do servidor, aqui, limita-se a apenas entregar a página web ao formulários que navegador. Quando o usuário submete o formulário, o navegador (cliente!) usem mailto. precisa resolver sozinho a questão de como fazer os dados serem enviados

via email. O cliente não está equipado para entregar dados do formulário -

esse é um trabalho para o servidor.

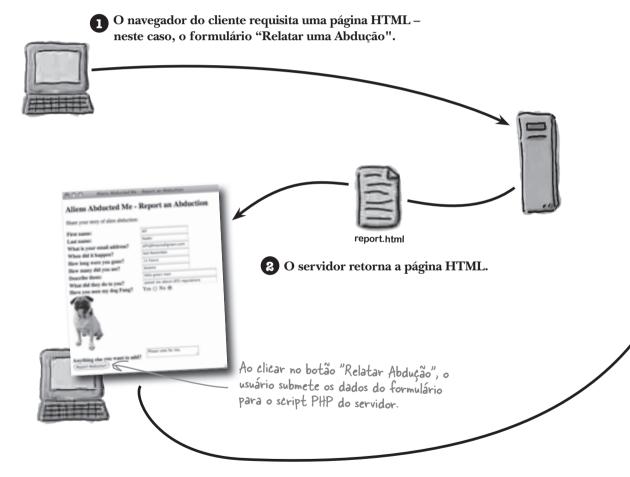
PHP roda no servidor

O PHP lhe permite controlar os dados que o usuário digita no formulário, enviando-os para você de forma transparente. O usuário digita o seu caso de abdução no formulário, clica no botão "Relatar Abdução" e pronto! O código PHP cria a mensagem de email, a envia para você e então gera uma página de confirmação para o usuário. Agui está. Eu gostaria de 🚹 O navegador requisita a receber a página "Relatar página web de Owen. 0 uma Abdução", por favor. 0 ত O servidor responde com o código HTML para a página web. O usuário preenche o formulário e o submete, Eu processo enviando os dados a um script PHP presente as informações do no servidor. formulário e envio o email. Agora, eu gostaria de submeter o formulário, 0 <form action = por favor. "report.php" 0 0 O script PHP gera uma página de confirmação em HTML e envia os dados para Owen, por email. O servidor envia uma confirmação em HTML ao navegador. Owen recebe, com certeza, um email agradavelmente formatado. Marque os lugares onde você acha que os scripts PHP devem residir: no cliente no servidor em ambos em nenhum dos dois

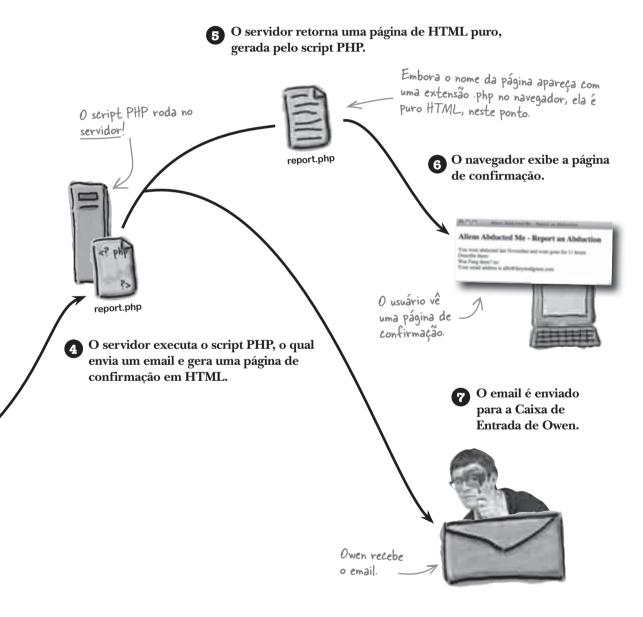
Os scripts PHP rodam no servidor

O código PHP roda no servidor e é armazenado em **scripts** PHP que geralmente tem a extensão .php. Os scripts PHP frequentemente se parecem bastante com páginas HTML normais, porque contêm tanto código HTML quanto CSS. De fato, quando o servidor executa um script PHP, o resultado final é sempre HTML e CSS puros. Assim, todo script PHP acaba se transformando em HTML e CSS, uma vez terminada a sua execução no servidor.

Vamos dar uma olhada mais detalhada no modo como o script PHP modifica o fluxo do formulário web de Owen.



O usuário preenche o formulário e o submete, fazendo o navegador repassar os dados para um script PHP no servidor. O PHP é uma linguagem de programação do lado do servidor (serve-side) -pois ele roda em um servidor web.



OK. Mas o que, realmente, faz o script PHP ser executado no servidor?



0

O atributo **action** dos elementos do formulário é o que conecta este último ao script PHP, fazendo o script ser executado quando o formulário é submetido.

Os formulários são criados usando-se a tag HTML <form> e cada tag <form> possui um atributo action. O arquivo cujo nome você inclui com o atributo action é usado pelo servidor web para processar o formulário quando este é submetido. Assim, se o script PHP de Owen se chamar report.php, então a tag <form> para conectá-lo ao formulário se parecerá com o seguinte:

<form action = "report.php" method = "post">

Este é o nome do seu script PHP. - Quando o usuário clica no botão "Relatar Abdução", a ação do formulário faz o script report. php ser executado **no servidor**, para processamento dos dados.



O atributo action
da tag <form>
é o que faz o
script PHP ser
executado no
servidor, quando
o formulário é
submetido.



report.php

não existem Perguntas Idiotas

P: O que significa PHP?

PHP é um acrônimo que originalmente significava Personal Home Pages (Páginas Pessoais). De alguma forma, o acrônimo teve o seu significado modificado para PHP: Hypertext Processor (Processador de Hipertexto PHP). Este último é considerado um acrônimo recursivo, porque ele referencia a si mesmo – um acrônimo (PHP) dentro do acrônimo. Inteligente? Confuso? Você decide!

P: Mesmo que o meu navegador esteja mostrando que o nome da página termina com .php, ela ainda é puro HTML? Como é possível?

R: Isso é possível porque a página começa a sua vida como código PHP no servidor, mas é transformada em código HTML antes de ser enviada para o navegador. Assim, o servidor executa o código PHP e o converte em HTML antes de enviá-lo para ser visualizado no navegador. Isso significa que, mesmo que o arquivo .php contenha código PHP, o navegador nunca vê esse código – ele vê apenas o código HTML, resultado da execução do código PHP no servidor.

P: Mas não é verdade que toda página web se origina no servidor, inclusive as páginas HTML puras, em arquivos .html?

R: Sim. Todos os arquivos que compõem um site ficam armazenados no servidor - . html , .css , .php, etc. Mas nem todos eles são processados pelo servidor. Os arquivos HTML e CSS, bem como arquivos gráficos, são enviados diretamente para o navegador do cliente, sem preocupações sobre o conteúdo deles. Os arquivos PHP são diferentes porque contêm códigos que são processados e executados no servidor web. Não é o código PHP que é enviado ao navegador, e sim os resultados obtidos ao se executar o código PHP, - esses resultados são HTML e CSS puros.

Use PHP para acessar os dados do formulário

Então, Owen precisa de um script PHP capaz de fazer com que as informações do formulário cheguem a ele de forma mais confiável do que através do e-mail. Vamos criar esse script.

Não se preocupe em entender tudo agora – nós chegaremos lá:

```
Os scripts PHP
                                       Ah, é aqui que
                 frequentemente
                                                              E perfeitamente
                                       as coisas ficam
                 começam parecendo-
                                                             normal um script PHP
                                       interessantes -
                 se bastante com uma
                                                             incluir tags e atributos
                                       este é o começo
                  página HTML normal.
<html>
                                                             HTML normais.
                                       do código PHP
                                       propriamente dito.
<head>
  <title>Aliens Abducted Me - Report an Abduction</title>
                                                                  O bloco de código
</head>
                                                                   inteiro é PHP... o
<body>
                                                                   restante do script é
  <h2>Aliens Abducted Me - Report an Abduction</h2>
                                                                   HTML normal.
<?php
                                                                       Este pedaço de
  $when_it_happened = $_POST['whenithappened'];
                                                                       código PHP obtém
  $how_long = $_POST['howlong'];
                                                                       os dados do
                                                                       formulário, para
  $alien_description = $_POST['description'];
                                                                      que eles possam
  $fang_spotted = $_POST['fangspotted'];
                                                                       ser exibidos
                                                                      como parte de
  $email = $_POST['email'];
                                                                      uma página de
                                                                      confirmação.
  echo 'Thanks for submitting the form. <br />';
  echo 'You were abducted ' . $when_it_happened;
                                                                      Aqui, nós usamos
  echo ' and were gone for ' . $how_long . '<br />';
                                                                      PHP para gerar
  echo 'Describe them: ' . $alien_description . '<br />';
                                                                      código HTML a
                                                                      partir dos dados
  echo 'Was Fang there? ' . $fang_spotted . '<br />';
                                                                      do formulário.
  echo 'Your email address is ' . $email;
?>
                     Assim como uma página web normal,
</body>
                      este script PHP termina fechando
</html>
                      quaisquer tags HTML abertas.
```



Modifique o formulário de Owen de forma que ele use um script PHP para processar os dados do formulário.

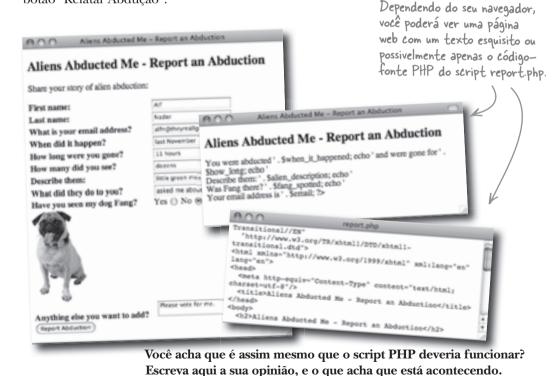
Crie um novo arquivo de texto chamado report. php e digite nele todo o código da página anterior. Esse é o script que irá processar o formulário web de Owen.

O script PHP ainda não está conectado ao formulário; abra a página report.html em um editor de textos e modifique a ação do formulário para report.php em vez de mailto.



<form action = "report.php" method = "post">

Abra a página report.html em um navegador web, digite algumas informações do seu caso de abdução no formulário e clique no botão "Relatar Abdução".



você está aqui ▶

Os scripts PHP precisam ficar em um servidor!

A não ser que você, por acaso, tenha um servidor web rodando no seu computador local, o script report.php não poderá ser executado quando você submeter o formulário "Relatar uma Abdução". Lembre-se, o PHP é uma linguagem de programação e precisa de um ambiente onde possa rodar. Esse ambiente é um servidor web com suporte a PHP. Scripts PHP e páginas web que dependem deles **precisam ser colocadas em um servidor web real**, não basta apenas abrir um script diretamente a partir de um sistema de arquivos local.

E se você tiver um servidor web instalado localmente, e ele tiver suporte a PHP, então poderá testar os scripts PHP diretamente no seu computador.

Os navegadores web não sabem nada sobre PHP e, portanto, não são capazes de rodar scripts PHP.



Ao contrário de páginas HTML, que podem ser abertas localmente em um navegador web, os scripts PHP precisam sempre ser "abertos" através de uma URL, a partir de um servidor web.

Este script PHP representa apenas um monte de código sem sentido para o navegador web.

O servidor web entende este código PHP e executa o script!



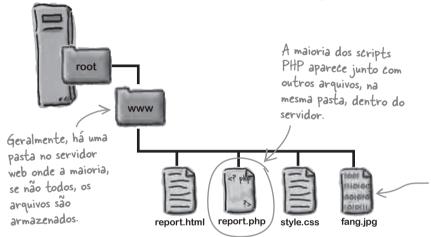
Os servidores web com suporte a PHP são equipados para rodar scripts PHP e transformálos em páginas HTML que os navegadores possam entender.

Uma forma rápida de saber se uma Página web está sendo entregue por um servidor web é olhar se a URL começa com "http:". Páginas web abertas como arquivos locais sempre começam com "file:"

Os scripts PHP devem ser executados em um servidor web, ou então não funcionarão.

Coloque os seus scripts PHP no servidor

É perfeitamente válido criar e editar scripts PHP no seu computador local. Porém, você precisa colocar os arquivos em um servidor web antes de executá-los. Os arquivos PHP, frequentemente, são colocados junto com arquivos HTML em um servidor. Não há nada fora do comum em colocar scripts PHP no servidor web – basta colocá-los em um lugar onde as suas páginas web possam acessá-los. Para fazer o upload (envio) dos arquivos para o servidor, você precisará da ajuda de um utilitário, como por exemplo, um programa de FTP (File Transfer Protocol ou Protocolo de Transferência de Arquivos).



As imagens são, às vezes, armazenadas em uma pasta própria, para melhor organização no servidor... mas não neste caso.

Fazer o upload dos seus scripts PHP para um servidor web não é o suficiente – esse servidor também precisa ter o PHP instalado nele. Alguns servidores incluem o PHP por padrão, alguns não.

não existem

Perguntas Idiotas

 $\mathop{\text{\rm P:}}\nolimits$ Como eu posso saber se o meu servidor web tem o PHP instalado?

R: Você pode perguntar ao seu administrador, ou à sua empresa de hospedagem de sites, ou então pode realizar um pequeno teste, você mesmo. Crie um arquivo de texto chamado teste.php e coloque o sequinte código nele:



Agora, envie (upload) o teste.php para o seu servidor, e então digite a sua URL em um navegador web. Se o PHP estiver instalado no seu servidor, aparecerão várias informações detalhadas sobre o PHP, incluindo a sua versão. Bingo!



Se você não tiver o PHP instalado no seu servidor web, consulte

o Apêndice ii.

Nele, encontrará instruções para ter o PHP instalado e operante no seu servidor.

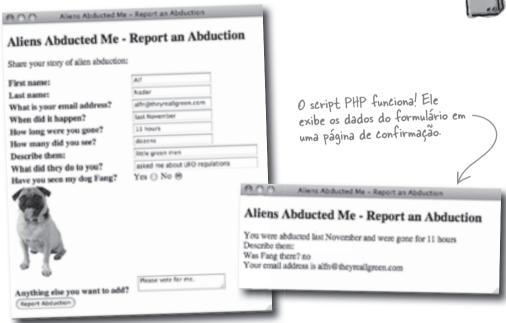
Lembre-se de deletar o script phpinfo quando estiver terminado, desta forma, ninguém mais consegue ver isto



Faça o upload dos arquivos da página "Relatar uma Abdução" e teste o formulário... novamente.

Envie os arquivos report.html, report.php, style. css e fang.jpg para um servidor web que tenha o PHP instalado. Digite a URL da página report.html no seu navegador, preencha o formulário com informações do seu caso de abdução, e clique no botão "Relatar Abdução".





Legal. Agora só é preciso adicionar um pouco de código PHP para enviar os dados do formulário por email.



Isso mesmo. O script report.php ainda precisa de algum código para enviar os dados para Owen por email.

Mas isso não é problema, porque o PHP oferece uma função, que é um pedaço de código pronto reutilizável, que você pode usar para enviar mensagens por email. Você só precisa determinar o que a mensagem deve informar, e então pode usar o PHP para criá-la e enviá-la.

Um momentinho! Nós nem sabemos como o script report.php original funciona, e agora ainda temos de torná-lo capaz de enviar emails. Isso, é tipo, muito difícil... o que vocês estão pensando!?

É verdade. Para fazer mais coisas com o PHP, é preciso saber mais sobre o PHP

Assim, para adicionar a funcionalidade de email ao script **report.php** de Owen, você terá de se aprofundar um pouco mais no PHP e precisará ter um entendimento sólido de como o script funciona – pelo menos até aqui.

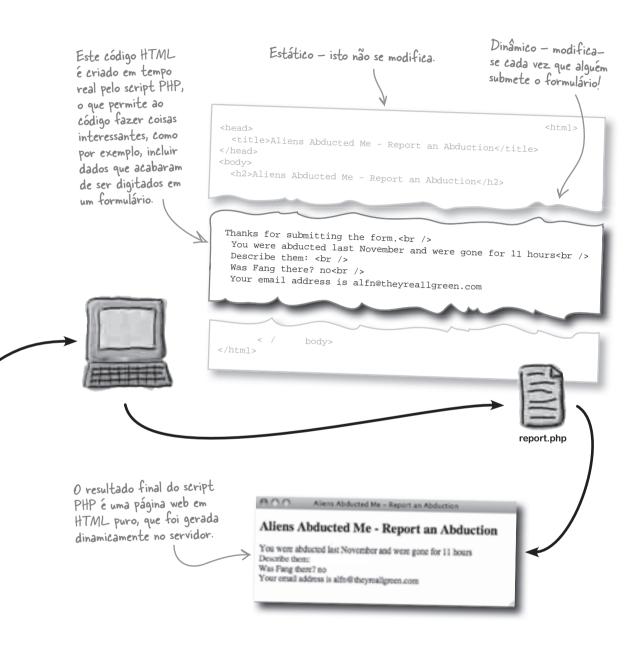


O servidor transforma PHP em HTML

Para que se entenda como um script PHP funciona, o principal é entender o que acontece com o script quando ele é executado no servidor. A maioria dos scripts PHP contém tanto código PHP quanto HTML, e o PHP é executado e transformado em HTML antes de o servidor enviar tudo – na forma de HTML – para o navegador web do cliente. No script report.php de Owen, o código PHP gera a maior parte do conteúdo HTML no corpo da página de confirmação. O código HTML em torno do PHP é enviado sem modificações.

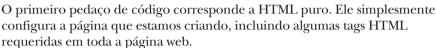
Este código HTML é enviado sem modificações para o navegador.

Este código PHP é <title>Aliens Abducted Me - Report an Abduction</title> executado pelo servidor e gera o código HTML <h2>Aliens Abducted Me - Report an Abduction</h2> contendo os dados que foram digitados no formulário. <?php \$when_it_happened = \$_POST['whenithappened']; \$how_long = \$_POST['howlong']; \$alien_description = \$_POST['aliendescription']; \$fang_spotted = \$_POST['fangspotted']; \$email = \$_POST['email']; echo 'Thanks for submitting the form.
'; echo 'You were abducted ' . \$when_it_happened; echo ' and were gone for ' . \$how_long . '
'; echo 'Describe them: ' . \$alien_description . '
'; echo 'Was Fang there?' . \$fang_spotted . '
'; echo 'Your email address is ' . \$email; 2> report.php Mais código HTML estático, que é transmitido pelo servidor ao navegador sem modificações.



Pesconstruindo o script PHP de Owen

O script report.php é acionado pelo formulário "Relatar uma Abdução" e o seu trabalho (no momento) é acessar os dados digitados e gerar uma página de confirmação. Vejamos como.



```
<html>
<head>
<title>Aliens Abducted Me - Report an Abduction</title>
</head>
<body>
<h2>Aliens Abducted Me - Report an Abduction</h2>
```

Aqui as coisas começam a ficar interessantes. Estamos prontos para sair do código HTML e entrar no PHP. A tag <?php abre uma seção de código PHP – tudo o que vier depois desta tag é puro PHP.

```
<?php
```

<? php ?>

report.php

Sim, este código

HTML é mínimo —
normalmente você teria
um DOCTYPE, tags
<meta>, etc., mas
estamos mantendo as
coisas simples, aqui.

Daqui em diante, estaremos lidando com código PHP... pelo menos até chegarmos à tag de fechamento ?>.

Este código reúne os dados do formulário e os armazena em variáveis individuais, para que possamos acessá-los posteriormente. As **variáveis** do PHP permitem que você armazene valores, sejam eles números, textos ou outros tipos de dados.

```
$when_it_happened = $_POST['whenithappened'];
$how_long = $_POST['howlong'];
$alien_description = $_POST['description'];
$fang_spotted = $_POST['fangspotted']
$email = $_POST['email'];
Cada linha do código PHP

reúne os dados do field
do formulário para uma
nova variável
```

Agora sim! Aqui, as variáveis que acabamos de criar são colocadas em ação, sendo inseridas no código HTML, gerado dinamicamente. O comando echo ocasiona o output (saída) de código HTML, que é retornado diretamente para o navegador web.

```
echo 'Thanks for submitting the form.<br/>
echo 'You were abducted ' . $when_it_happened;<br/>
echo ' and were gone for ' . $how_long . '<br />';<br/>
echo 'Describe them: ' . $alien_description . '<br />';<br/>
echo 'Was Fang there? ' . $fang_spotted . '<br />';<br/>
echo 'Your email address is ' . $email;
```

A tag ?> correspondente a <? encerra a seção de código PHP. Daqui em diante, estamos de volta ao código HTML normal.

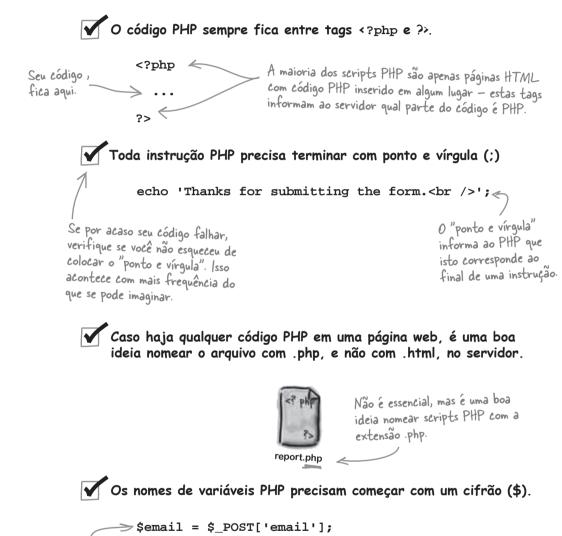
```
__ lsto fecha o
__ código php
```

Isto finaliza o código PHP – depois desta tag, estamos de volta ao HTML normal.

```
</bd>
Agora, termine a página fechando as tags
HTML que havíamos aberto.
```

⁰ seu código Algumas regras PHP para a sua vida

O script report.php de Owen revela algumas regras fundamentais da linguagem PHP, que se aplicam a todo script PHP. Vamos dar uma olhada nelas.



O cifrão identifica claramente a variável PHP, responsável por armazenar informações dentro de um script PHP.

Observando as variáveis usadas no script report.php, você consegue ver quaisquer outras regras do PHP referentes a variáveis? Escreva-as aqui!

Encontrando o nome perfeito para a sua variável

Além de começar com \$, os nomes de variáveis PHP são também "case- um contaîner sensitive". Mas isso não é tudo – existem outras regras importantes para os nomes que você pode dar às suas variáveis. Algumas delas se referem onde você pode à sintaxe, significando que o seu código falhará se você as ignorar, enquanto outras outras obras correspondem a apenas boas ideias transmitidas de uma geração de programadores PHP para outra.

Vamos começar com as regras oficiais que irão definitivamente causar problemas se você ignorá-las ao nomear suas variáveis. Siga estas regras para criar nomes de variáveis válidas. Entendido!

A variável corresponde a armazenar dados, e cada variável tem um nome único.

✓ O primeiro caracter tem de ser um cifrão (\$).

O nome da variável precisa ter no mínimo um caracter.

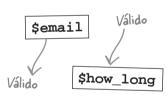
Sem contar o \$, que é requerido no nome de toda variável

O primeiro caracter depois do cifrão pode ser uma letra ou um "underscore" (), e os caracteres seguintes podem ser uma letra, um underscore ou um número.

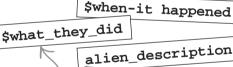
Espaços e caracteres especiais, fora _ e \$, não são permitidos em nenhuma parte do nome de uma variável

variáveis PHP.

Inválido Nomes de variáveis PHP não podem conter hifens nem espaços.



\$fang_spotted Inválido Hífens não são Permitidos em nomes de



Válido

Inválido Nomes de variáveis PHP Precisam começar com um cifrão (\$).

Se você não seguir estas regras, o seu código não funcionará. Existem ainda, outras regras que podem ser seguidas - por uma questão de "convenções" de programação. Estas são regras que ajudam a tomar o código PHP um pouco mais consistente e fácil de ser lido.

Use minúsculas nos nomes das variáveis.



Em nomes de variáveis com mais de uma palavra, devem começar separe-as com "underscores".

Se você ignorar estas duas últimas regras, não serão geradas falhas no seu código. Certamente, você encontrará códigos PHP que não seguem estas regras e, mesmo assim, funcionam perfeitamente. Isso ocorre porque estas regras - sobre as quais estamos nos referindo tratam-se de uma convenção estilística. Elas tornam-se úteis quando começam a criar e nomear suas próprias variáveis.

Os nomes das Variáveis PHP com um cifrão(\$). e não podem conter espaços.

não existem

Perguntas İdiotas

P: Faz diferença escrever os comandos PHP em maiúsculas ou minúsculas?

R: Sim e não. Em geral, o PHP não é "case-sensitive", portanto, você pode misturar maiúsculas e minúsculas na maioria dos comandos. Isso significa que você pode usar echo, ECHO ou EchO. Entretanto, por uma questão de convenção, é uma ótima ideia ser consistente nos seus scripts. A maioria dos programadores PHP prefere usar minúsculas na maior parte dos seus códigos, e é por isso que verá a grafia echo sendo usada nos exemplos de códigos deste livro.

P: Então, mesmo sendo uma má prática de programação, posso misturar maiúsculas e minúsculas no meu código PHP?

R: Não completamente. A exceção corresponde aos nomes das variáveis, o que também se aplica às localidades de armazenamento de dados que você cria. Vejamos a variável \$email usada no script "Relatar uma Abdução", como exemplo. O nome desta variável é "case-sensitive", portanto, você não pode se referir a ela como \$EMAIL ou \$eMail. Todos os nomes de variáveis. em PHP, são "case-sensitive", por isso, é importante nomear as variáveis com cuidado e referenciá-las de forma consistente, em todo o código. Falaremos mais sobre nomes de variáveis em outro momento.

Realmente não tem problema colocar código PHP e HTML no mesmo arquivo?

R: Nenhum problema. Na verdade, em muitos casos, é absolutamente necessário fazer isso.

P: E por que colocar PHP e HTML no mesmo arquivo?

R: Porque o verdadeiro propósito um servidor web é servir páginas web HTML

para os navegadores. O PHP não muda esse fato. O que o PHP lhe permite fazer é modificar o conteúdo HTML em tempo real, inserindo dados como a data de hoje, informações retiradas de um banco de dados, ou até mesmo valores calculados, como o valor total do pedido, em um site de compras. Assim, o PHP permite que você manipule o HTML que compõe as páginas web, em vez de tê-las criadas estaticamente, apenas uma vez. É bastante comum ter uma página HTML com código PHP aqui e ali para inserir dados importantes, ou alterar programaticamente o HTML de alguma forma.

P: O código PHP inserido em um arquivo HTML precisa estar na sua própria linha ou posso colocá-lo em uma linha HTML, como parte do atributo de uma tag HTML, por exemplo?

R: Fora a necessidade de colocar o seu código PHP dentro das tags <?php e ?>, não há restrições sobre como inserir esse código no HTML. Na verdade, é necessário colocar, frequentemente, um bloco de código PHP no meio do código HTML, por exemplo, quando está sendo configurado o atributo de uma tag HTML. Esse é um uso perfeitamente legítimo do PHP.

P: Já vi códigos PHP iniciados por <? como tag de abertura, em vez de <?php. Isso está certo?

R: Na verdade, não. Tecnicamente isso é permitido, mas não recomendado. É preciso que uma configuração do servidor esteja habilitada para que a tag abreviada (<?) funcione. A tag <? php usual sempre funciona, então é melhor simplesmente usá-la, sabendo que o seu código vai funcionar.

P: Se o servidor web sempre retorna HTML puro para o navegador do cliente, por que as URLs mostram o nome do script PHP, como, por exemplo, paginaweb.php?

R: Lembre-se de que toda página web corresponde ao resultado de uma comunicação de duas vias, envolvendo uma requisição feita pelo navegador cliente e uma resposta do servidor web. A URL é a base da requisição, enquanto conteúdo retornado pelo servidor é a resposta. Os scripts PHP são requisitados da mesma forma que páginas HTML normais, através de URLs digitadas no navegador ou linkadas a partir de outras páginas, ou como ações de formulários. Isso explica porque a URL de uma "página" PHP mostra o nome do script.

A outra metade da equação é a resposta do servidor, que é o código resultante gerado pelo script PHP. Uma vez que a maioria dos scripts PHP gera HTML, faz sentido que esse código seja HTML e não PHP. Assim, não é por acidente que a URL referencie um arquivo .php no servidor, o que faz o código PHP ser executado no servidor, resultando em um conteúdo puramente HTML, que é retornado ao navegador.

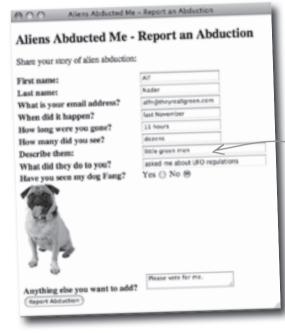
P: As variáveis PHP podem armazenar outros tipos de dados?

R: Certamente. Você pode usá-las para armazenar dados Booleanos (verdadeiro/falso). E dados numéricos podem ser tanto números inteiros quanto de ponto flutuante (decimais). Há também os "arrays", os quais armazenam um conjunto de dados, bem como objetos, os quais associam um conjunto de dados ao código usado para manipulá-los. Os "arrays" serão abordados um pouco mais adiante ainda neste capítulo, enquanto os "objetos" são assunto do Capítulo 12. Existe também um tipo de dado especial chamado NULL, que representa "nenhum valor". Por exemplo, uma variável que não tenha nenhum valor atribuído a ela é considerada NULL.

0

Ou a memória do PHP não é tão boa assim ou há algo de errado com o script... há alguns dados que não aparecem.





Claramente, foi digitada uma descrição (describe) dos aliens no formulário...

...mas a descrição não aparece na página de confirmação.

Aliens Abduct d Me - Report an Abduction

You were abduct ast November and were gone for 11 hours
Describe them:
Was Fang them? no
Your email address is alfn@theyreallgreen.com

Aponte seu lápis

Há um problema com os dados referentes à descrição dos aliens, no script report.php de Owen. Circule as linhas do código que você acha que tem a ver com o problema e escreva o que elas estão causando. Alguma ideia do que há de errado?

```
<html>
 <head>
   <title>Aliens Abducted Me - Report an Abduction</title>
 </head>
 <body>
   <h2>Aliens Abducted Me - Report an Abduction</h2>
 <?php
   $when_it_happened = $_POST['whenithappened'];
  $how_long = $_POST['howlong'];
  $alien_description = $_POST['description'];
  $fang_spotted = $_POST['fangspotted']
  $email = $_POST['email'];
  echo 'Thanks for submitting the form.<br />';
  echo 'You were abducted ' . $when_it_happened;
  echo ' and were gone for ' . $how_long . '<br />';
  echo 'Describe them: ' . $alien_description . '<br />';
  echo 'Was Fang there? ' . $fang_spotted . '<br />';
  echo 'Your email address is ' . $email;
?>
</body>
</html>
```

29

report.php

Aponte seu lápis Solução

Há um problema com os dados referentes à descrição dos aliens, no script report.php de Owen. Circule as linhas do código que você acha que tem a ver com o problema e escreva o que elas estão causando. Alguma ideia do que há de errado?

Esta linha do código obtém a descrição dos aliens do respectivo campo no formulário HTML, e os dados armazena em uma variável PHP chamada falien_description.

Este código combina a descrição dos aliens com alguns outros textos e códigos HTML e envia tudo isso para o navegador.

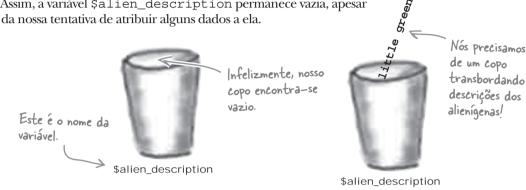
```
<html>
 <head>
   <title>Aliens Abducted Me - Report an Abduction</title>
 </head>
 <body>
   <h2>Aliens Abducted Me - Report an Abduction</h2>
 <?php
   $when_it_happened = $_POST['whenithappened'];
   $how_long = $_POST['howlong'];
  $alien_description = $_POST['description'];
   $fang_spotted = $_POST['fangspotted']
  $email = $_POST['email'];
  echo 'Thanks for submitting the form.<br />';
  echo 'You were abducted ' . $when_it_happened;
  echo ' and were gone for ' . $how_long . '<br />';
  echo 'Describe them: ' . $alien_description
  echo 'Was Fang there? ' . $fang_spotted
  echo 'Your email address is ' . $email;
?>
</body>
</html>
```

Por algum motivo, a variável falien_ description parece estar vazia... nada bom.

report.php

Variáveis servem para armazenar dados de scripts

As variáveis PHP são como containers que armazenam informações, da mesma forma que um copo armazena uma bebida. Uma vez que a variável \$alien_description está vazia, nós sabemos que os dados correspondentes do formulário não estão chegando até ela. Assim, a variável \$alien_description permanece vazia, apesar da nossa tentativa de atribuir alguns dados a ela.



Uma forma de consertar o script seria atribuir a string exata que estamos esperando para a variável \$alien_description, desta forma:



Este código armazena, definitivamente, o texto 'homenzinhos verdes' na variável \$alien_description. Mas nós resolvemos um problema criando outro – este código faz a descrição ser sempre a mesma, independentemente do que o usuário digitar no formulário.



Por algum motivo, a atribuição dos dados do formulário referentes à descrição dos alienígenas à variável \$alien_description não está funcionando.

\$alien_description = \$_POST['description'];

O que você acha que está errado com este código?

O problema obviamente tem algo a ver com aquele negócio de \$_POST. Mas não tenho ideia do que possa ser.

O problema, de fato, tem algo a ver com \$_POST, que é um mecanismo usado para que os dados do formulário sejam transferidos para um script.

O cifrão no início de \$_POST é uma dica... \$_POST é um container! Mais especificamente, \$_POST é um conjunto de locais de armazenamento usados para que os dados de um formulário web fiquem armazenados. No caso de Owen, todos os dados enviados para script report.php são armazenados quando alguém preenche o formulário e clica no botão de "Relatar Abdução". Assim, para acessar os dados do formulário e fazer qualquer coisa com eles, nós temos que usar \$_POST . Lembra-se deste código?

```
$when_it_happened = $_POST['whenithappened'];

$how_long = $_POST['howlong'];

$alien_description = $_POST['description'];

$fang_spotted = $_POST['fangspotted'];

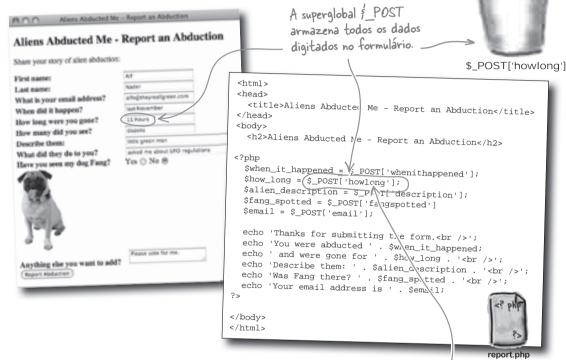
$email = $_POST['email'];

O mesmo acontece aqui, exceto pelo fato de que os dados referentes à duração da abdução são atribuídos à variável femail.
```

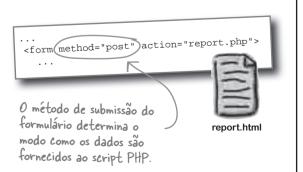
Assim, os dados de cada campo do formulário "Relatar uma Abdução" são acessados usando-se post. Mas o que exatamente é post. uma variável?

[†]_POST é uma variável especial que armazena dados do formulário

\$_POST é uma variável especial, conhecida também como **superglobal**, porque ela é interna ao PHP e fica disponível para o script inteiro. \$_POST já existe quando o seu script é executado – você não a cria, como as outras variáveis no PHP.



A superglobal \$_POST é vinculada diretamente ao método de submissão usado pelo formulário HTML. Se o método tiver sido definido como post, então todos os dados são empacotados dentro da superglobal \$_POST, onde cada informação pode ser retirada e usada conforme necessário.



O nome "howlong" ("por quanto tempo") vem do atributo name usado na tag <input> correspondente a este campo do formulário.



Como você acha que a superglobal \$_POST funciona? Como ela é capaz de armazenar múltiplos valores originários de todas aquelas caixas de texto do formulário de Owen?

⁹_POST transporta os dados do formulário para o seu script

\$_POST é um tipo especial de container PHP, conhecido como **array**, que armazena um conjunto de variáveis sob um único nome. Quando o usuário submete o formulário de Owen, os dados que ele digitou nos campos são armazenados no array \$ POST, cujo trabalho é

Aliens Abducted Me - Report an Abduction

Yes O No 6

Share your story of allen abduction

on my dog Fang?

What is your email addr

How many did you see?

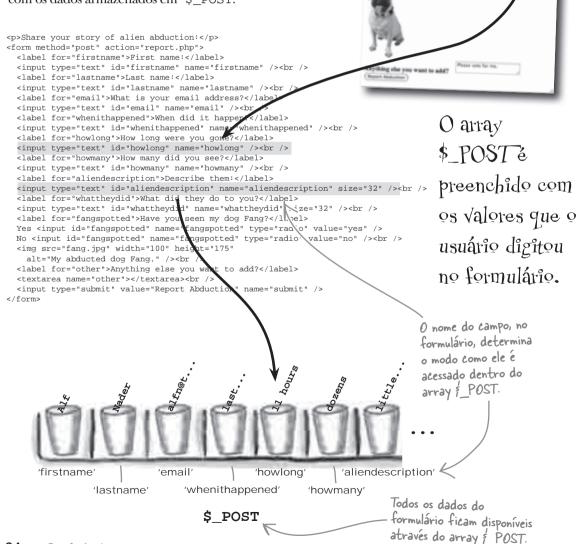
Describe them: What did they do to you?

When did it happen? How long were you gone?

repassar esses dados para o script.

Cada elemento do array \$_POST corresponde a uma determinada informação, digitada em um determinado campo do formulário. Para acessar os dados referentes a um campo específico, você usa o nome desse campo com \$_POST.

Por exemplo, a duração da abdução é armazenada em \$_POST['howlong']. O código HTML do formulário de Owen revela o modo como os nomes do formulário se relacionam com os dados armazenados em \$_POST.



Aponte seu lápis

Examine o código de report.php que está fazendo a descrição dos aliens ficar em branco e depois escreva a solução para o problema. Dica: use o código HTML do formulário exposto na página anterior para ajudar a isolar o problema.

```
<ht.ml>
                        <head>
                          <title>Aliens Abducted Me - Report an Abduction</title>
                        </head>
                        <body>
                          <h2>Aliens Abducted Me - Report an Abduction</h2>
                        <?php
                         $when_it_happened = $_POST['whenithappened'];
                          $how_long = $_POST['howlong'];
                         $alien_description = $_POST['description'];
                         $fang_spotted = $_POST['fangspotted']
Lembre-se:
                         $email = $_POST['email'];
anteriormente, nós
isolamos o problema
para essas duas linhas
                         echo 'Thanks for submitting the form.<br />';
de código.
                         echo 'You were abducted ' . $when_it_happened;
                         echo ' and were gone for ' . $how_long . '<br />';
                         echo 'Describe them: ' . $alien_description . '<br />';
                         echo 'Was Fang there? ' . $fang_spotted . '<br />';
                         echo 'Your email address is ' . $email;
                      ?>
                      </body>
                      </html>
                                                                          report.php
```

Aponte seu lápis Solução

Examine o código de report.php que está fazendo a descrição dos aliens ficar em branco e depois escreva a solução para o problema. Dica: use o código HTML do formulário exposto na página anterior para ajudar a isolar o problema.

```
<input type="text" id="aliendescription" name=("aliendescription")</pre>
                                                                  an Abduct
                           </head>
                           <body>
 O nome do campo
                             Aliens Abducted Me - Report an Abduction</h2>
 em report.html
 é "aliendescription",
                           <?php
 diferente do nome
                             $when_it_happened = $\_OST['whenithappened'];
"description" usado com
                             $how_long = $_POST['howlong
 & POST.
                                                            'aliendescription'
                             $alien_description = $_POST['
                             $fang_spotted = $_POST['fangspotted']
                            $email = $_POST['email'];
 Nós precisamos
modificar & POST
                            echo 'Thanks for submitting the form.<br />';
Para Corrigir o
                            echo 'You were abducted ' . $when_it_happened;
nome do campo:
"aliendescription".
                            echo ' and were gone for ' . $how_long . '<br />';
                            echo 'Describe them: ' . $alien_description . '<br />';
                            echo 'Was Fang there? ' . $fang_spotted . '<br />';
                            echo 'Your email address is ' . $email;
                          ?>
                          </body>
                          </html>
                                                                              report.php
```



Conserte e teste o script.

Modifique a linha de código problemática em report.php e depois a envie para o seu servidor web. Abra a página report. html no seu navegador, preencha o formulário com as informações do seu caso de abdução e clique no botão "Relatar Abdução" para submeter o formulário agora corrigido script.



Excelente. Mas acontece que ainda estão faltando *alguns* dados do formulário...



Aponte seu lápis

Há alguns dados digitados no formulário "Relatar uma Abdução" que ainda não estamos usando. Lembre-se, esses dados incluem informações vitais sobre um caso de abdução, as quais poderiam ajudar Owen a encontrar seu cão Fang. Portanto, precisamos acessar ou filtrar todos os dados do caso e armazená-los em variáveis PHP.

> Atualmente o script report.php está ignorando cinco campos do formulário. Inacreditável

```
How many did you see?
                                                                  Describe them:
                                                                                                   e green mer
                                                                  What did they do to you?
                                                                                                 asked me about UFO regulations
                                                                  Have you seen my dog Fang?
                                                                                                Yes () No (6)
 <form method="post" action="report.php">
    <label for="firstname">First name:</label>
    <input type="text" id="firstname" name="firstname"</pre>
    <label for="lastname">Last name:</label>
    <input type="text" id="lastname" name="lastname" /><</pre>
    <label for="email">What is your email address?</labe</pre>
    <input type="text" id="email" name="email" /><br />
    <label for="whenithappened">When did it happen?</lak</pre>
    <input type="text" id="whenithappened" name="whenith")</pre>
    <label for="howlong">How long were you gone?</label;</pre>
    <input type="text" id="howlong" name="howlong" /><br/>b;
    <label for="howmany">How many did you see?</label>
    <input type="text" id="howmany" name="howmany" /><b=</pre>
    <label for="aliendescription">Describe them:</label>
    <input type="text" id="aliendescription" name="aliendescription" size="32"</pre>
    <label for="whattheydid">What did they do to you?</label>
    <input type="text" id="whattheydid" name="whattheydid" size="32" /><br />
    <label for="fangspotted">Have you seen my dog Fang?</label>
    Yes <input id="fangspotted" name="fangspotted" type="radio" value="yes" />
No <input id="fangspotted" name="fangspotted" type="radio" value="no" /><br/>
/> />
    <img src="fang.jpg" width="100" height="175"</pre>
      alt="My abducted dog Fang." /><br />
    <label for="other">Anything else you want to add?</label>
    <tactarea id="other" name="other"></textarea></r/>
<input type="submit" value="Report Abduction" name="submit" />
  </form>
</body>
</html>
```



Escreva o código PHP a fim de criar quatro novas variáveis que armazenem os dados do formulário que estão faltando: \$name, \$how_many, \$what_they_dide \$other. Dica: Crie a variável \$name de forma que ela armazene o **nome completo** do usuário.

Aliens Abducted Me - Report an Abduction

Aliens Abducted Me - Report an Abduction

A tag <input> de cada campo

do formulário é

a partir do PHP.

a chave para acesso aos dados,

Share your story of alien abduction:

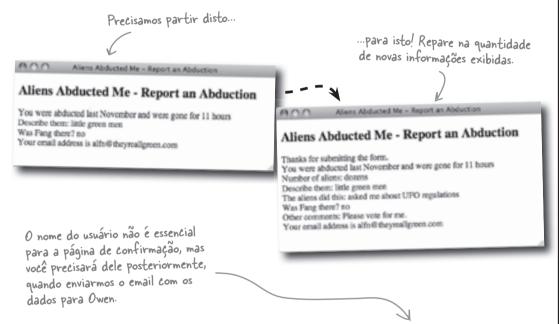
What is your email address?

When did it happen?

How long were you gone?

First name: Last name:

Seu trabalho ainda não terminou. A página de confirmação gerada pelo script PHP precisa usar essas novas variáveis para exibir mais informações sobre a abdução.



Usando todas as variáveis que acabou de criar, exceto \$name, complete o código abaixo, o qual irá gerar uma página de confirmação mais informativa.

```
echo 'Thanks for submitting the form.<br/>
'You were abducted ' . $when_it_happened;
echo ' and were gone for ' . $how_long . '<br />';
echo 'Describe them: ' . $alien_description . '<br />';
echo 'Was Fang there? ' . $fang_spotted . '<br />';
echo 'Your email address is ' . $email;
```

Aponte seu lápis Solucão

Há alguns dados digitados no formulário "Relatar uma Abdução" que ainda não estamos usando. Lembre-se: esses dados incluem informações vitais sobre um caso de abdução, as quais poderiam ajudar Owen a encontrar seu cão Fang. Portanto, precisamos acessar ou filtrar todos os dados do caso e armazená-los em variáveis PHP.

> Atualmente o script report.php está ignorando cinco campos do formulário. Inacreditável!

What did they do to you? asked me about UFO regulations flave you seen my dog Fang? Yes () No (0) <form method="post" action="report.php"> <label for="firstname">First name:</label> <input type="text" id="firstname" name="firstname" /</pre> <label for="lastname">Last name:</label> <input type="text" id="lastname" name="lastname" /><</pre> <label for="email">What is your email address?</labe</pre> <input type="text" id="email" name="email" />
 <label for="whenithappened">When did it happen?</lak</pre> <input type="text" id="whenithappened" name="whenith"</pre> <label for="howlong">How long were you gone?</label> <input type="text" id="howlong" name="howlong" />

 <label for="howmany">How many did you see?</label> <input type="text" id="howmany" name="howmany" /><b=</pre> <label for="aliendescription">Describe them:</label> <input type="text" id="aliendescription" name="aliendescription" size="32"</pre> <label for="whattheydid">What did they do to you?</label> <input type="text" id="whattheydid" name="whattheydid" size="32" />
 <label for="fangspotted">Have you seen my dog Fang?</label> Yes <input id="fangspotted" name="fangspotted" type="radio" value="yes" />
No <input id="fangspotted" name="fangspotted" type="radio" value="no" />
/> /> <imq src="fang.jpg" width="100" height="175"</pre> alt="My abducted dog Fang." />
 <label for="other">Anything else you want to add?</label> <textarea id="other" name="other"></textarea>
 <input type="submit" value="Report Abduction" name="submit" /> </form> </body>

A tag <input> de cada campo do formulário é a chave para acesso aos dados, a partir do PHP

Aliens Abducted Me - Report an Abd

Aliens Abducted Me - Report an Abduction

Share your story of alien abduction:

What is your email address?

When did it happen?

How long were you gone? How many did you see?

First name: Last name:

Describe them:

report.html

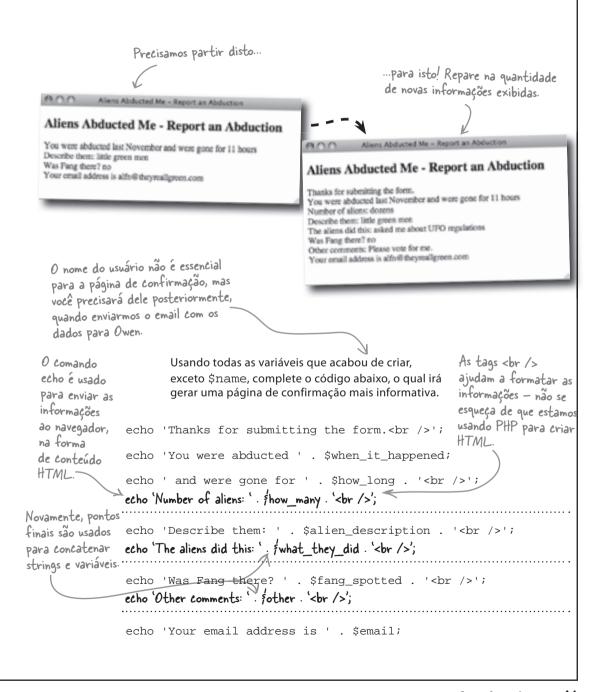
O ponto final the permite juntar várias strings de texto em uma só – um processo conhecido como concatenação.

Escreva o código PHP a fim de criar quatro novas variáveis que armazenem os dados do formulário que estão faltando: \$name, \$how_many, \$what_they_dide \$other. Dica: Crie a variável \$name de forma que ela armazene o **nome completo** do usuário.

fname = f_POST['firstname']. ' . f_POST['lastname']; show many = & POST['howmany']; fwhat_they_did = f_POST['whattheydid']; fother = f_POST['other'];

Este espaço separa o Primeiro nome do sobrenome.

Seu trabalho ainda não terminou. A página de confirmação gerada pelo script PHP precisa usar essas novas variáveis para exibir mais informações sobre a abdução.





Ajuste o script de Owen e teste as modificações.

Adicione o código das novas variáveis a report.php, bem como o código que envia as variáveis para o navegador, com o HTML formatado. Em seguida, envie o script para o seu servidor web, abra a página report.html no seu navegador e preencha o formulário com as informações do seu caso de abdução. Finalmente, clique no botão "Relatar Abdução" para submeter o formulário e ver os resultados.

não existem

Perguntas İdiotas

P: O que realmente acontece quando eu faço a concatenação de várias strings, usando pontos finais?

R: Concatenar, neste caso, significa juntar diversas strings para formar outra, totalmente diferente. O resultado final é sempre uma única string, não importa quantas tenham existido originalmente. Assim, quando você concatena strings como parte de um comando echo, o PHP, primeiramente, as combina em uma só string e depois envia esta última para o navegador.

P: Quando eu concateno uma variável com uma string, a variável precisa conter texto?

R: Não. Embora a concatenação sempre resulte em uma string, as variáveis não precisam conter strings para que você possa usá-las em um processo de concatenação. Por exemplo, digamos que uma variável contenha um número; o PHP primeiramente converte o número em uma string e depois a inclui na concatenação.

 $\operatorname{\mathcal{P}}$: O que acontece com o código PHP no navegador?

R: Nada. E isso porque o código PHP nunca é visto pelo navegador. O código PHP é executado no servidor e se transforma em código HTML, este sim sendo enviado para o navegador. Assim, o navegador nem sequer fica sabendo da existência do PHP – as páginas web chegam como puros HTML e CSS.

P: OK, então como, exatamente, o servidor transforma o código PHP em códigos HTML e CSS?

R: Em primeiro lugar, lembre-se de que, por padrão, assume-se que o código em um script PHP é código HTML. Você pode identificar o código PHP dentro de um script colocando-o entre tags <?php e?>. O servidor vê essas tags e entende que deve rodar o código dentro delas como PHP e todo o código fora dessas tags é informado ao navegador como sendo um HMTL.

P: Certo. Mas isso ainda não explica como o código PHP é transformado em HTML/CSS. E então?

R: Ah, é aí que o comando echo entra em cena. Você pode pensar no echo como um comando para enviar as informações que estejam fora do domínio das tags <?php e?>. Assim, o comando echo representa a chave para a capacidade do PHP de gerar código HTML/CSS, dinamicamente. Concatenando strings de texto com variáveis PHP, você pode construir código HTML em tempo real e depois usar echo para enviá-lo ao navegador como parte da página web resultante. Um bom exemplo disso, no script report.php, é quando a tag
br /> é anexada ao final de um pedaço de texto para gerar uma quebra de linha no HTML resultante.

A página de confirmação é útil para o usuário, mas não para mim.
Ainda preciso que os dados do formulário sejam enviados para mim através de um email.

C p D fd H oil un m

O script PHP ainda precisa enviar os dados do formulário para Owen.

Da forma como está, o script report.php está usando os dados do formulário "Relatar uma Abdução" para gerar uma página de confirmação HTML para o usuário. Mas isso ainda não está resolvendo o problema original, que baseia-se no envio de uma mensagem por email para Owen, uma vez submetido o formulário. Ele quer apenas receber uma simples mensagem de email em modo texto, mais ou menos como a seguinte:

De forma semelhante à página web de confirmação, esta mensagem de email consiste em texto estático, combinado com dados do formulário.

Alf Nader foi abduzido em novembro passado e esteve ausente por 11 horas.

Número de aliens: dúzias

Descrição dos aliens: homenzinhos verdes

O que eles fizeram: me perguntaram sobre leis para as OVNIs

Fang foi visto: não

Outros comentários: Por favor vote em mim.

Esta mensagem de email pode ser gerada a partir de código PHP, criando-se uma string que combine textos estáticos como "Outros comentários:" com dados do formulário que estejam armazenados em variáveis.

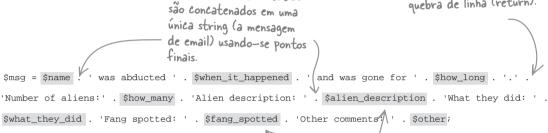
Escreva aqui como você criaria uma mensagem de email a partir de texto estático e variáveis PHP.

Criando o corpo da mensagem de email com PHP

Variáveis e texto estático

Você já viu como um ponto final pode ser usado em códigos PHP para concatenar várias strings de texto em uma só. Agora precisa usar novamente a concatenação para criar uma string que será a mensagem de email, com variáveis sendo usadas juntamente com texto estático.

A maioria dos editores de texto, automaticamente, quebra o texto para a linha seguinte, mesmo que você não inclua no código a sua própria quebra de linha (return).



Um problema ao se criar uma string desse tamanho é que ela demanda uma grande linha de código PHP, difícil de ler e entender. Você pode dividir o código PHP em várias linhas, para deixá-lo mais fácil de ser estudado. Certifique-se, apenas, de separar o código em pontos onde o espaçamento não faça diferença, como por exemplo, entre duas strings concatenadas e não no meio de uma string. Em seguida, coloque um ponto e vírgula no final da última linha de código, para finalizar a instrução PHP.

Lembre-se: cada variável armazena uma string de texto retirada do formulário "Relatar uma Abdução".

```
| sto na verdade é apenas uma grande linha de código dividida em várias linhas.

Smsg = $name . ' was abducted ' . $when_it_happened . ' and was gone for ' . $how_long . '.' .

'Number of aliens: ' . $how_many .

'Alien description: ' . $alien_description .

'What they did: ' . $what_they_did .

'Fang spotted: ' . $fang_spotted .

'Other comments: ' . $other:

Você ainda precisa finalizar a instrução, com um ponto e virgula.
```

Quando uma linha de código PHP é: deliberadamente, dividida em várias linhas, costuma-se indentar as linhas após a primeira, para que se perceba, com mais facilidade, quais linhas compõem uma mesma instrução no seu código.

Uma longa linha de código PHP pode ser dividida em várias linhas, desde que você tenha cuidado na hora de separar o código.

Esse código PHP certamente ficou bonito. Mas sem formatação, a mensagem de email não vai ficar toda bagunçada?

0

Sim. Só porque o código PHP está bem organizado, não quer dizer que o seu output, (saída) automaticamente, também ficará bonito.

Organizar o código PHP para que você possa entendê-lo melhor é completamente diferente de formatar o output que os usuários irão ver. Você normalmente usará tags HTML para formatar o output do código PHP, uma vez que, na maioria dos casos, o PHP é usado para se gerar, dinamicamente, uma página web. Porém não neste caso.

Aqui, estamos gerando uma mensagem de email, que é puro texto, e não HTML. Precisamos lidar com o fato de que a mensagem, atualmente, se parece com o resultado seguinte:

Alf Nader foi abduzido em novembro passado e esteve ausente por 11 horas. Número de aliens: dúzias Descrição dos aliens: homenzinhos verdes O que eles fizeram: me perguntaram sobre leis para as OVNIs Fang foi visto: não Outros comentários: Por favor vote em mim.

Hm... isto NÃO é o que Owen tinha em mente para a sua mensagem de e-mail.

não existem

Perguntas İdiotas

Existe alguma maneira de se usar formatação HTML em emails que você envia a partir de um script PHP?

R: Existe. Mas isso requer um passo adicional, que envolve definir o cabeçalho do tipo de conteúdo para a mensagem. Cabeçalhos e tipos de conteúdo vão um pouco além do escopo desta discussão, por isso, estamos nos limitando a mensagens de email em formato de texto simples. Você aprenderá mais sobre os cabeçalhos no Capítulo 6, portanto, não se preocupe – você verá como enviar emails em HTML mais adiante.



Como você reformataria a mensagem de email para torná-la mais fácil de ler?

Até mesmo texto simples pode ser formatado... um pouco

Uma vez que Owen está enviando mensagens de email como texto simples, sem nenhuma formatação especial HTML, ele não pode colocar tags
 /> para adicionar quebras de linha para separar o conteúdo. Mas ele pode usar caracteres newline (nova linha), que são escapados na forma \n. Assim, em qualquer lugar do email onde apareça \n, será inserida uma quebra de linha, fazendo o conteúdo que vier a seguir se iniciar na linha seguinte. Eis o novo código para a mensagem de email, com os newlines adicionados:

Caracteres de escape em PHP começam com uma barra invertida ().

```
In é usado para
                                     que sejam inseridos
                                     caracteres newline na
                                     mensagem de email.
$msg = $name . ' was abducted ' . \$when_it_happened . ' and was gone for ' . $how_long . '.\n'
  'Number of aliens: ' . $how_many . '\n' .
  'Alien description: ' . $alien_description . '\n' .
  'What they did: ' . $what_they_did . '\n' .
  'Fang spotted: ' . $fang_spotted . '\n' .
  'Other comments: ' . $other;
```

Alf Nader foi abduzido em novembro passado e esteve ausente por 11 horas n Número de aliens: dúzias n Descrição dos aliens: homenzinhos verdes no que eles fizeram: me perguntaram sobre leis para as OVNIs n Fang foi visto: nãq n Outros comentários: Por favor vote em mim.

Os newlines me parecem uma boa ideia... pena que o código não parece funcionar.

0

 ${\rm P:}\;$ O que é, exatamente, um caracter de escape?

R: Um caracter de escape é um caracter que: ou é difícil de digitar ou que, por algum motivo, poderia causar confusão no código PHP. Você pode estar familiarizado com os caracteres de escape do HTML, programados de uma forma um pouco diferente, como por exemplo, © ou © para o símbolo de copyright. O PHP tem um conjunto bastante pequeno de caracteres de escape que são úteis para se escrever coisas que poderiam se confundir com a própria linguagem, como por exemplo, aspas simples (\'), aspas duplas (\") e, é claro, newlines (\n).

O In está aparecendo como texto normal, em vez de um caracter newline ... nada bom.



Newlines precisam de aspas duplas

O problema com o código de Owen é que o PHP trata as strings de forma diferente, conforme elas se apresentem: dentro de aspas simples ou duplas. Mais especificamente, os caracteres newline (\n) só podem ser escapados dentro de aspas duplas. Assim, a mensagem de email com os dados do formulário precisa ser construída usando-se strings dentro de aspas duplas, para as newlines funcionarem.

Mas a história das aspas simples x duplas não termina aí. As strings dentro de aspas simples são consideradas como texto puro, enquanto que o PHP processa as strings dentro de aspas duplas em busca de variáveis. Quando uma variável é encontrada dentro de uma string em aspas duplas, o PHP insere o seu valor dentro da string, como se tivesse ocorrido uma "concatenação". Assim, a string dentro de aspas duplas não só é necessária para o funcionamento dos newlines na mensagem de email, como também nos permite simplificar o código, colocando as variáveis diretamente na string.

concatenação não é mais necessária, uma vez que as variáveis podem ser referenciadas diretamente dentro da string entre aspas duplas.

```
$msg = "$name was abducted $when_it_happened and was gone for $how_long.\n"
```

"Number of aliens: $how_many\n$ " .

"Alien description: \$alien_description\n"

"What they did: $\$ \$what_they_did\n"_ $\$.

"Fang spotted: \$fang_spotted\n"

"Other comments: \$other";

Não é preciso de uma nova linha quando for a linha final. Os caracteres newline agora são interpretados corretamente, graças à string apresentada entre aspas duplas.

Mas ainda precisamos dividir a mensagem em várias strings concatenadas para tornar o código mais legível.

não existem

Perguntas İdiotas

P: Se as aspas duplas são tão legais, por que é que por que fizemos uso, praticamente, aspas simples, até agora?

R: Bem, tenha em mente que as aspas simples não são processadas pelo PHP de nenhuma maneira, o que as torna ideias para strings que são puro texto, sem nenhuma variável inserida. Assim, continuaremos a usar aspas simples ao longo do livro, a não ser que haja um bom motivo para usar as duplas. A coisa mais importante nesta questão de usar aspas simples ou duplas para as strings, é tentar manter-se o mais consistente possível.

O que acontece se eu tiver de usar um apóstrofo dentro de uma string que esteja dentro de aspas simples, como por exemplo, em 'He's lost!' ("ele está perdido!")?

R: É aí que os caracteres de escape são úteis. Para

usar um apóstrofo em uma string dentro de aspas simples, basta fazer escape dele como \', desta forma: '
He\'s lost! '. O mesmo se aplica se você quiser usar uma só aspa dupla dentro de uma string delimitada por aspas duplas – use \". Não é preciso fazer escape das aspas quando não houver conflito, como por exemplo, ao usar um apóstrofo em uma string contida entre aspas duplas: "He's lost!".

P: Então, com as aspas simples, pode-se usar \'mas não \n. Como saber quais caracteres de escape eu posso usar dentro de aspas simples?

R: As strings dentro de aspas simples só permitem os caracteres de escape \('\end{e}\) \(\). Todos os outros caracteres de escape só podem ser usados em strings dentro de aspas duplas.

Crie uma mensagem de email para Owen

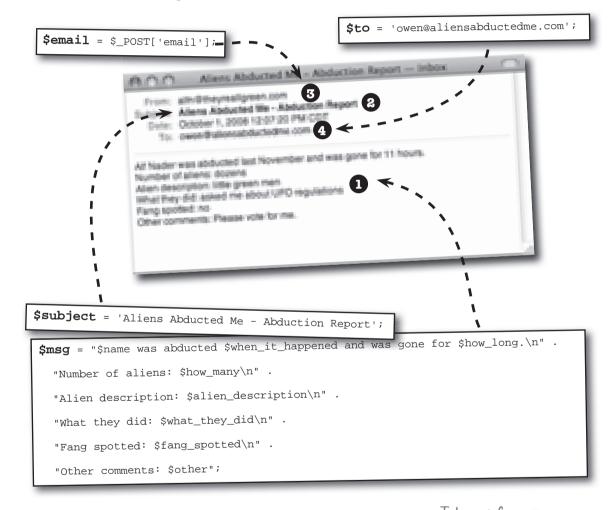
Com o corpo da mensagem gerado como uma string, você pode prosseguir e montar o restante do email de Owen. Uma mensagem de email é mais do que o corpo – existem várias partes diferentes. Embora algumas sejam opcionais, as seguintes informações são usadas, praticamente, em todo o email:



Este exemplo de mensagem de email revela que a maior parte do conteúdo fica no corpo da mensagem, o que já está pronto. Só é preciso bolar algo para o campo assunto, preencher os endereços do remetente e do destinatário... e, é claro, usar o PHP de alguma forma para, realmente, enviar a mensagem!

As variáveis armazenam as partes do email

Nós já temos o corpo da mensagem armazenado em \$msg, mais ainda faltam o assunto e os endereços do remetente ("from") e do destinatário ("to"). O assunto e o endereço "to" podem ser definidos como texto estático em novas variáveis, enquanto que o endereço "from" já está armazenado na variável \$email, graças ao código processador do formulário que escrevemos anteriormente, neste capítulo.



Corpo da mensagem

Todas as informações estão nos seus lugares!

Assunto da mensagem



- 3 O endereço de email do remetente (campo FROM da mensagem).
- 4 O endereço de email do destinatário (campo TO da mensagem).

Enviando uma mensagem de email com PHP A função

Então, você está pronto para escrever o código PHP que enviará a mensagem mail(), do PHP, a Owen por email. Isto requer a função interna mail () do PHP, a qual envia uma mensagem com base em informações que você fornece.

1) assunto da O endereço de mail(\$to, \$subject, \$msg);

en√ia uma mensagem de email a partir de um script.

Essas três informações são requeridas pela função mail (), portanto, você sempre tem de fornecê-las. O endereço de email "from" não é exigido, mas ainda é uma boa ideia incluí-lo. Para especificar o campo "from" ao chamar a função mail (), é preciso usar um outro argumento da função, junto a alguma concatenação de strings.

1) texto 'From:' precisa ser colocado antes do endereço, ao se especificar o endereço de email do remetente.



O ponto final é útil mais uma vez, para concatenar 'From:' com o endereço de email de Owen.

não existem

Perguntas İdiotas

· Há mais alguma coisa que possa ser especificada como parte de uma mensagem de email, além do endereço "from"?

R: Sim. Você pode especificar também recipientes "copy" ("cópia") e "blind copy" ("cópia oculta") da mesma forma que o "from" - basta usar 'Cc: ' ou 'Bcc: ' em vez de

'From: '. Se quiser especificar tanto um "from" quanto um "copy", você terá de separá-los com uma combinação de caracteres return e newline (\r) , desta forma:

É isso mesmo, dois caracteres de "From: " . \$from . "\r\nCc: " . \$cc

Precisamos usar aspas duplas aqui, uma vez que estamos usando os caracteres de escape \r e \n.

Mas então, como é que nós **usamos**, realmente, a função mail()?

0



Basta adicionar ao seu script o código para chamar mail(). A linha de código que chama a função mail() é tudo o que você precisa para

enviar a mensagem de email. Certifique-se de que este código apareça no script após o código que cria as variáveis para o email, e pronto. Eis aqui o código completo para o script report. php de Owen, incluindo a chamada para a função mail().

```
<html>
   <head>
    <title>Aliens Abducted Me - Report an Abduction</title>
  </head>
                                                                    Peque todos os dados
    <h2>Aliens Abducted Me - Report an Abduction</h2>
                                                                    do formulário no array
                                                                    & POST e coloque-os
  <?php
    $name = $_POST['firstname'] . ' ' . $_POST['lastname'];
                                                                    em variáveis individuais.
    $when_it_happened = $_POST['whenithappened'];
    $how_long = $_POST['howlong'];
    $how_many = $_POST['howmany'];
    $alien_description = $_POST['aliendescription'];
                                                                  Certifique-se de
   $what_they_did = $_POST['whattheydid'];
                                                                  trocar este endereço
   $fang_spotted = $_POST['fangspotted'];
   $email = $_POST['email'];
                                                                  de email pelo seu
   $other = $_POST['other'];
                                                                  Próprio, para testar o
   $to = 'owen@aliensabductedme.com';
                                                                  script.
   $subject = 'Aliens Abducted Me - Abduction Report';
   msg = "name was abducted $when_it_happened and was gone for $how_long.\n" .
     "Number of aliens: $how_many\n"
     "Alien description: \alpha = \beta 
     "What they did: \ what_they_did\n" .
     "Fang spotted: $fang_spotted\n" .
                                                               Junte os diferentes
     "Other comments: $other";
  mail($to, $subject, $msg, 'From:' . $email);
                                                                pedaços da
  echo 'Thanks for submitting the form.<br />';
                                                               mensagem a ser
  echo 'You were abducted ' . $when_it_happened;
echo ' and were gone for ' . $how_long . '<br />';
echo 'Number of aliens: ' . $how_many . '<br />';
                                                               enviada para Owen.
echo 'Describe them: ' . $alien_description . '<br />';
  echo 'The aliens did this: ' . $what_they_did . '<br />';
  echo 'Was Fang there? ' . $fang_spotted . '<br />'; echo 'Other comments: ' . $other . '<br />';
  echo 'Your email address is ' . $email;
</body>
</html>
```

Gere uma página HTML, em tempo real, para confirmar que o formulário foi submetido

com sucesso.

Envie a

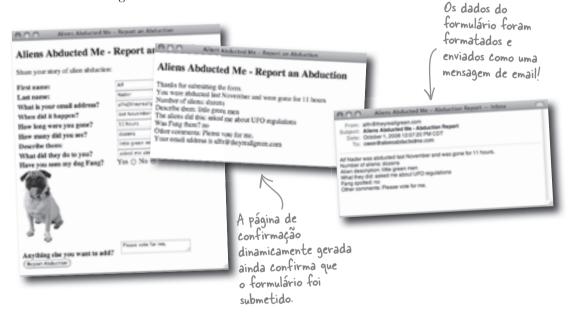
mensagem.

report.php



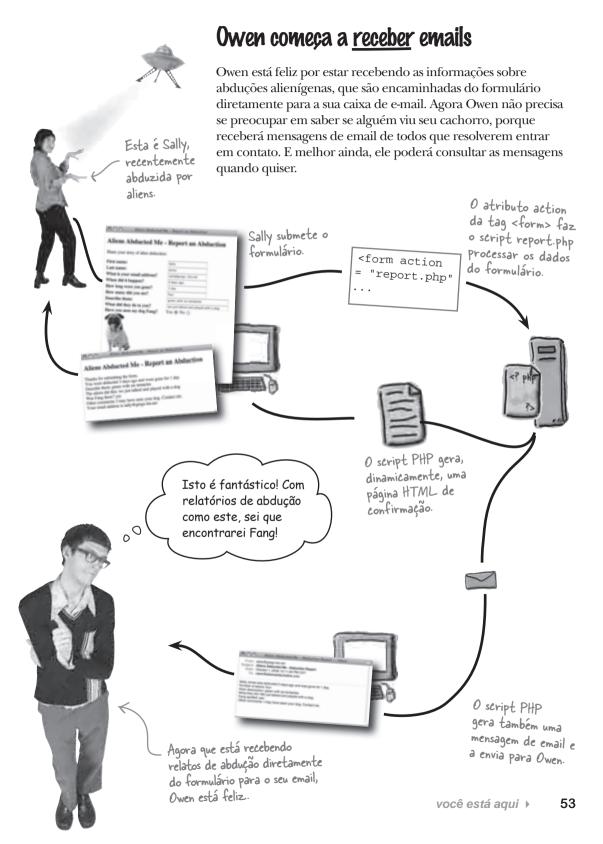
Finalize o script de Owen e depois teste-o.

Adicione as três novas variáveis para o email (\$to, \$ subject e \$msg) ao script report.php, bem como a chamada à função mail(). Certifique-se de configurar a variável \$to com o o seu próprio endereço de email e não o de Owen! Envie (upload) o script para o seu servidor web, abra-o no seu navegador e preencha o formulário com informações do seu caso de abdução. Clique no botão "Relatar Abdução" para submeter o formulário. Espere alguns segundos e depois verifique se a mensagem está na sua Caixa de Entrada de emails.



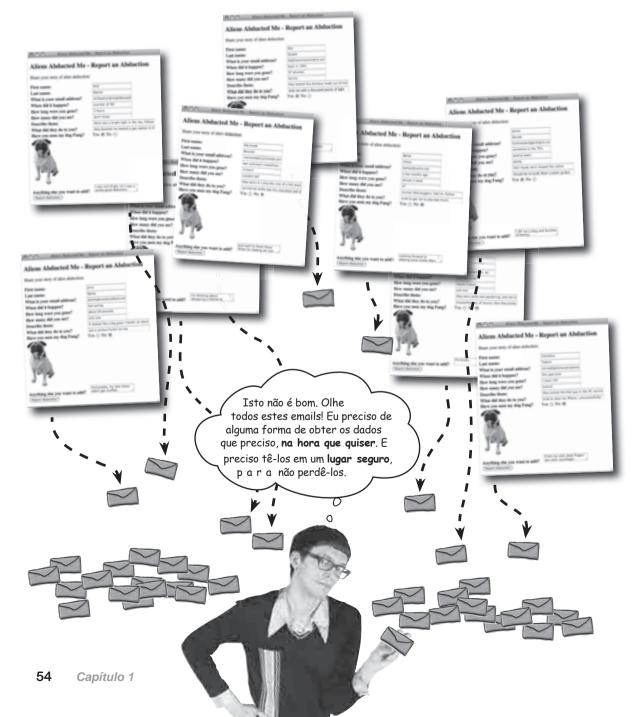
Você poderá ter de configurar o PHP no seu servidor web, para que ele saiba como enviar emails.

Se a função mail() não estiver funcionando para você, o problema poderá poderá estar no suporte a email, que não está corretamente configurado na sua instalação do PHP. Consulte www.php.net/mail para mais detalhes sobre como configurar os recursos de email no seu servidor web.



Owen começa a perder emails

A boa notícia é que Owen está recebendo emails. A má notícia é que ele está recebendo muitos, muitos emails. Tantos que está tendo dificuldade em acompanhá-los. A sua Caixa de Entrada está lotada, e ele já apagou alguns acidentalmente... Owen precisa de uma forma melhor de armazenar os dados sobre as abduções.



QUEM FAZ Q QUÊ?

Os aliens mexeram com o seu cérebro? Coloque as coisas no lugar ligando cada componente HTML ou PHP à descrição do que eles fazem.

HTML Um software que serve para visualizar e interagir

com páginas web. Age como o lado do cliente em

comunicações na web.

PHP Um comando PHP usado para fazer o output de algum

conteúdo, como, por exemplo, texto simples ou código HTML.

formulário Web Estas tags são usadas para delimitar o código PHP a fim de

que o servidor web saiba que deve processá-lo e executá-lo.

navegador Um array interno do PHP que armazena os dados

submetidos usando o método "post".

<?php ?> Uma linguagem de programação usada para a criação

de scripts, que são executados em um servidor web.

Variável Toda string precisa estar dentro de uma variável.

aspas Um software para enviar páginas web, o qual age

como um servidor em comunicações na web.

echo Uma linguagem marcação usada para descrever a estrutura

do conteúdo das páginas web, que são visualizadas em um

navegador web.

PHP que ficam acessíveis a todos os scripts.

servîdor web

Uma série de campos de input, em uma página web,

usados para se obter informações dos usuários.

uma função interna do PHP que envia mensagens de email.

superglobal Um local de armazenamento, em um script PHP, que

tem o seu próprio nome e tipo único de dados.

maîl() Um tipo de armazenamento de dados em PHP que

permite que várias informações sejam armazenadas em

um mesmo local.



Os aliens mexeram com o seu cérebro? Coloque as coisas no lugar ligando cada componente HTML ou PHP à descrição do que eles fazem.

HTML
PHP
formulário Web
navegador

Um software que serve para visualizar e interagir com páginas web. Age como o lado do cliente em comunicações na web.

Um comando PHP usado para fazer o output de algum 'conteúdo, como, por exemplo, texto simples ou código HTML.

Estas tags são usadas para delimitar o código PHP a fim de que o servidor web saiba que deve processá-lo e executá-lo.

Um array interno do PHP que armazena os dados submetidos usando o método "post".

·Uma linguagem de programação usada para a criação de scripts, que são executados em um servidor web.

-Toda string precisa estar dentro de uma variável.

Um software para enviar páginas web, o qual age como um servidor em comunicações na web.

Uma linguagem marcação usada para descrever a estrutura do conteúdo das páginas web, que são visualizadas em um navegador web.

Um nome usado para descrever as variáveis internas do PHP que ficam acessíveis a todos os scripts.

Uma série de campos de input, em uma página web, usados para se obter informações dos usuários.

.Uma função interna do PHP que envia mensagens de email.

Um local de armazenamento, em um script PHP, que tem o seu próprio nome e tipo único de dados.

Um tipo de armazenamento de dados em PHP que permite que várias informações sejam armazenadas em um mesmo local.

<?php ?>

variável

aspas

echo

\$_POST

servidor web

array

superglobal

mail()

56

Sua Caixa de Ferramentas do PHP e MySQL

No Capítulo 1, você aprendeu como utilizar o PHP para dar vida nova ao formulário web de Owen. Veja quanta coisa já foi aprendida...

Variável

Um container para armazenamento de algum dado. No PHP, as variáveis devem começar com um cifrão, desta forma: fnome_da_variavel.

\$_POST

Uma variável especial, que armazena os dados de formulários.

PHP

Uma linguagem de scripts, que roda no lado do servidor, que lhe permite manipular o conteúdo de páginas web no servidor antes da página ser entregue ao navegador cliente.

Script PHP

Um arquivo de texto que contém código PHP para executar tarefas em um servidor web.

MySQL

Um aplicativo que lhe permite armazenar dados em bancos de dados e tabelas, inserir e recuperar informações usando a linguagem SQL.

SQL

Uma linguagem de consulta usada para a interação com aplicativos de bancos de dados, como o MySQL. <?php ?p>

Estas tags devem "circundar" todo o código PHP nos seus scripts PHP. mail()

A função do PHP para enviar emails. Ela trata o assunto do email, o corpo do texto e o endereço do destinatário como parâmetros (opcionalmente, você pode especificar o endereço do remetente, também).

echo

O comando PHP usado para enviar output para a janela do navegador. A sua sintaxe é:

echo 'Hello World';

Lado do cliente

É interpretado unicamente pelo navegador web do cliente:

Lado do servidor

Interpretado por um servidor web, e não pelo cliente.

array

Uma estrutura de dados que armazena um conjunto de valores. Cada valor possui um índice que você pode usar para acessar o array.

Caracter de escape

Usado para representar caracteres no código PHP que sejam difíceis de digitar ou que poderiam entrar em conflito com outras partes do código, como por exemplo \n (newlines).