

[< VOLTAR](#)

Resgatando informações de Formulários HTML

Neste tópico apresentaremos como resgatar informações advindas de um formulário.

NESTE TÓPICO

- > Introdução
- > GET x POST
- > Quando usar GET?
- > Quando usar POST? [Marcar tópico](#)
- > Referências



Introdução

Uma das características marcantes do PHP é como ele trata formulários HTML. É importante entender que qualquer controle (campos) em um formulário irá automaticamente ficar disponível para usarmos em nossos scripts PHP.

Vamos ver um exemplo simples!

```
1. <form action="action.php" method="post">
2.     <p>Nome: <input type="text" name="txtnome" /></p>
3.     <p>Idade: <input type="text" name="txtidade" /></p>
4.     <p><input type="submit" value="Enviar" /></p>
5. </form>
```

O código acima, produzirá a tela abaixo:

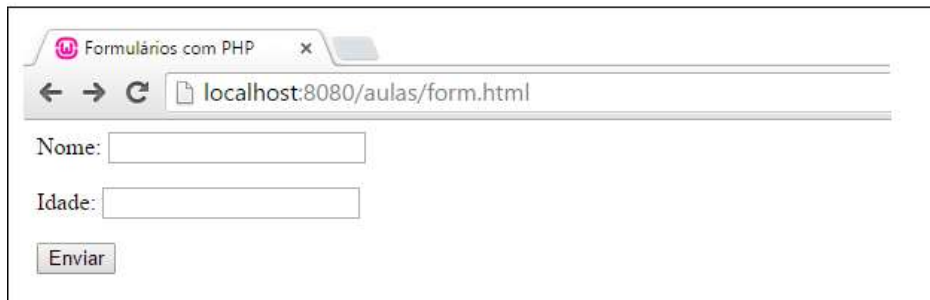


Figura 1: Manipulando formulários com PHP.

Fonte: Fonte: Prof. Adriano Milanez.

Relembrando o **Tópico VIII – Variáveis Pré-definidas**, podemos receber informações de um formulário utilizando os métodos do PHP, `$_GET[]` e `$_POST[]`

Como percebemos o exemplo acima, é de um formulário simples, sem nada especial. Quando o usuário preencher este formulário e clicar no botão enviar, a página `action.php` é chamada. Neste arquivo nós teremos algo como este:

1. Olá, <?php echo \$_POST['txtnome']; ?>.
2. Você tem <?php echo \$_POST['txtidade']; ?> anos.

O código acima, nos dará a tela abaixo como resultado:



Figura 2: Exemplo simples de como receber dados utilizando o método `$_POST[]` do PHP.

Fonte: Fonte: Prof. Adriano Milanez.

Perceberam, que o método utilizado foi o POST, mas e se estivéssemos utilizando o método GET? Vejamos o exemplo abaixo:

1. <form action="action.php" method="get">
2. <p>Nome: <input type="text" name="txtnome" /></p>
3. <p>Idade: <input type="text" name="txtidade" /></p>
4. <p><input type="submit" value="Enviar" /></p>
5. </form>

E quando o usuário preencher este formulário e clicar no botão enviar, a página `action.php` é chamada novamente. Neste arquivo nós teremos algo como este:

1. Olá, <?php echo \$_GET['txtnome']; ?>.
2. Você tem <?php echo \$_GET['txtidade']; ?> anos.

E as informações enviadas pelo formulário, iriam abertas na *url*, ou seja, segurança dos dados nula, como podemos notar na imagem abaixo:



Figura 3: Exemplo simples de como receber dados utilizando método `$_GET[]` do PHP

Fonte: Fonte: Prof. Adriano Milanez.

GET x POST

Vale lembrar que podemos utilizar tanto GET e POST num formulário e para recebermos as informações utilizaremos `$_GET[]` e `$_POST[]`, respectivamente. Como vimos no **Tópico VIII – Variáveis Pré-definidas**, `$_GET[]` e `$_POST[]`, são variáveis superglobais, o que significa que elas estão sempre acessíveis, independentemente do âmbito de aplicação - e você pode acessá-los de qualquer função, classe ou arquivo sem ter que fazer nada especial.

- `$_GET[]` é um *array* de variáveis passadas para o script atual através dos parâmetros de *url*.
- `$_POST[]` é um conjunto de variáveis passadas para o script atual através do método HTTP POST.

Quando usar GET?

Como já vimos as informações enviadas utilizando o método GET, deixa todos os nomes e valores de variáveis exibidos na *url*. Em navegadores mais antigos existe limitação na quantidade de informações a serem enviadas, esta limitação é de cerca de 2000 caracteres; nos navegadores mais recentes, essa limitação praticamente inexistente. No entanto, porque as variáveis são exibidas no *url*, é possível marcar a página. Isto pode ser útil em alguns casos.

NUNCA utilize GET para o envio de senhas ou outras informações confidenciais, como por exemplo, número de cartão de crédito, conta corrente!

Quando usar POST?

As informações enviadas utilizando o método POST são incorporados dentro do corpo da solicitação HTTP e não existe limite sobre a quantidade de informações a serem enviadas. Além disso POST suporta funcionalidades avançadas, tais como o upload de arquivos para o servidor.

Para informações confidenciais, tipo senha, cartões de crédito, senha bancária, sugere-se fortemente a utilização de POST.

Agora que fizemos um breve *review*, vamos falar sobre manipulação de formulário utilizando PHP.

A manipulação de dados de formulários é uma das coisas mais importantes e que todo desenvolvedor deve saber, pois normalmente é por meio dela que existe a comunicação da aplicação com o banco de dados. Saber manipular formulários é tão importante quanto saber como enviar as informações a um banco de dados.

Vamos tomar o formulário HTML, abaixo como exemplo:


```

1.  <html>
2.      <head>
3.          <title> Formulários com PHP</title>
4.          <style>
5.              .error {color: #FF0000;}
6.          </style>
7.      <?php
8.          // define variables and set to empty values
9.          $generoErr = $bandaErr= $infoErr = "";
10.         $genero = $banda = $info = "";
11.         if ($_SERVER["REQUEST_METHOD"] == "POST") {
12.             if (empty($_POST["rdo_genero"])) {
13.                 $generoErr = "Selecione um gênero musical";
14.             } else {
15.                 $genero = $_POST["rdo_genero"];
16.             }
17.             if (empty($_POST["txt_banda"])) {
18.                 $bandaErr = "Digite o nome de sua banda favorita";
19.             } else {
20.                 $banda = $_POST["txt_banda"];
21.             }
22.             if (empty($_POST["info_banda"])) {
23.                 $infoErr = "Digite informações sobre sua banda favorita";
24.             } else {
25.                 $info = $_POST["info_banda"];
26.             }
27.         }
28.     ?>
29. </head>
30. <body>
31.     <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
32.         <span class="error">* campo requerido</span>
33.         <p>Selecione um gênero musical:<span class="error">* <?php echo $generoErr;?></span>
34.             <br>
35.             <input type="radio" name="rdo_genero" value="Classica" /> Clássica<br>
36.             <input type="radio" name="rdo_genero" value="MPB" /> MPB<br>
37.             <input type="radio" name="rdo_genero" value="Pagode" /> Pagode<br>
38.             <input type="radio" name="rdo_genero" value="Rock" /> Rock<br>
39.             <input type="radio" name="rdo_genero" value="Samba" /> Samba<br>
40.             <input type="radio" name="rdo_genero" value="Sertanejo" /> Sertanejo<br>
41.         </p>
42.         <p>Digite sua banda favorita: <input type="text" name="txt_banda" /><span class="error">*<?php echo $bandaErr;?></span></p>
43.         <p>Informações sobre sua banda favorita:<br><textarea cols="50" rows="10" name="info_banda" /></textarea><span class="error">*<?php echo $infoErr;?></span></p>
44.         <p><input type="submit" value="Enviar" /></p>
45.     </form>
46. <?php
47.     if (($_SERVER["REQUEST_METHOD"] == "POST") && (!empty($_POST["rdo_genero"])) && !empty($_POST["txt_banda"])) && !empty($_POST["info_banda"])){ ?>
48.         <p>Os dados enviados foram:
49.         <?php
50.             echo "<h2>Your Input:</h2>";
51.             echo $genero;
52.             echo "<br>";
53.             echo $banda;
54.             echo "<br>";
55.             echo $info;
56.         }
57.     ?>
58. </p>
59. </body>
60. </html>

```



```

1.  <html>
2.      <head>
3.          <title> Formulários com PHP</title>
4.          <style>
5.              .error {color: #FF0000;}
6.          </style>
7.      <?php
8.          // define variables and set to empty values
9.          $generoErr = $bandaErr= $infoErr = "";
10.         $genero = $banda = $info = "";
11.         if ($_SERVER["REQUEST_METHOD"] == "POST") {
12.             if (empty($_POST["rdo_genero"])) {
13.                 $generoErr = "Selecione um gênero musical";
14.             } else {
15.                 $genero = $_POST["rdo_genero"];
16.             }
17.             if (empty($_POST["txt_banda"])) {
18.                 $bandaErr = "Digite o nome de sua banda favorita";
19.             } else {
20.                 $banda = $_POST["txt_banda"];
21.             }
22.             if (empty($_POST["info_banda"])) {
23.                 $infoErr = "Digite informações sobre sua banda favorita";
24.             } else {
25.                 $info = $_POST["info_banda"];
26.             }
27.         }
28.     ?>
29. </head>
30. <body>
31.     <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
32.         <span class="error">* campo requerido</span>
33.         <p>Selecione um gênero musical:<span class="error">* <?php echo $generoErr;?></span>
34.             <br>
35.             <input type="radio" name="rdo_genero" value="Classica" /> Clássica<br>
36.             <input type="radio" name="rdo_genero" value="MPB" /> MPB<br>
37.             <input type="radio" name="rdo_genero" value="Pagode" /> Pagode<br>
38.             <input type="radio" name="rdo_genero" value="Rock" /> Rock<br>
39.             <input type="radio" name="rdo_genero" value="Samba" /> Samba<br>
40.             <input type="radio" name="rdo_genero" value="Sertanejo" /> Sertanejo<br>
41.         </p>
42.         <p>Digite sua banda favorita: <input type="text" name="txt_banda" /><span class="error">*<?php echo $bandaErr;?></span></p>
43.         <p>Informações sobre sua banda favorita:<br><textarea cols="50" rows="10" name="info_banda" /></textarea><span class="error">*<?php echo $infoErr;?></span></p>
44.         <p><input type="submit" value="Enviar" /></p>
45.     </form>
46. <?php
47.     if (($_SERVER["REQUEST_METHOD"] == "POST") && (!empty($_POST["rdo_genero"])) && !empty($_POST["txt_banda"]) && !empty($_POST["info_banda"])){ ?>
48.         <p>Os dados enviados foram:
49.             <?php
50.                 echo "<h2>Your Input:</h2>";
51.                 echo $genero;
52.                 echo "<br>";
53.                 echo $banda;
54.                 echo "<br>";
55.                 echo $info;
56.             }
57.         ?>
58.     </p>
59. </body>
60. </html>

```


O código acima produzirá a tela abaixo:



Formulários com PHP x

localhost:8080/aulas/form1.html

Selecione um gênero musical:

- ☐ Clássica
- ☐ MPB
- ☐ Pagode
- ☐ Rock
- ☐ Samba
- ☐ Sertanejo

Digite sua banda favorita:

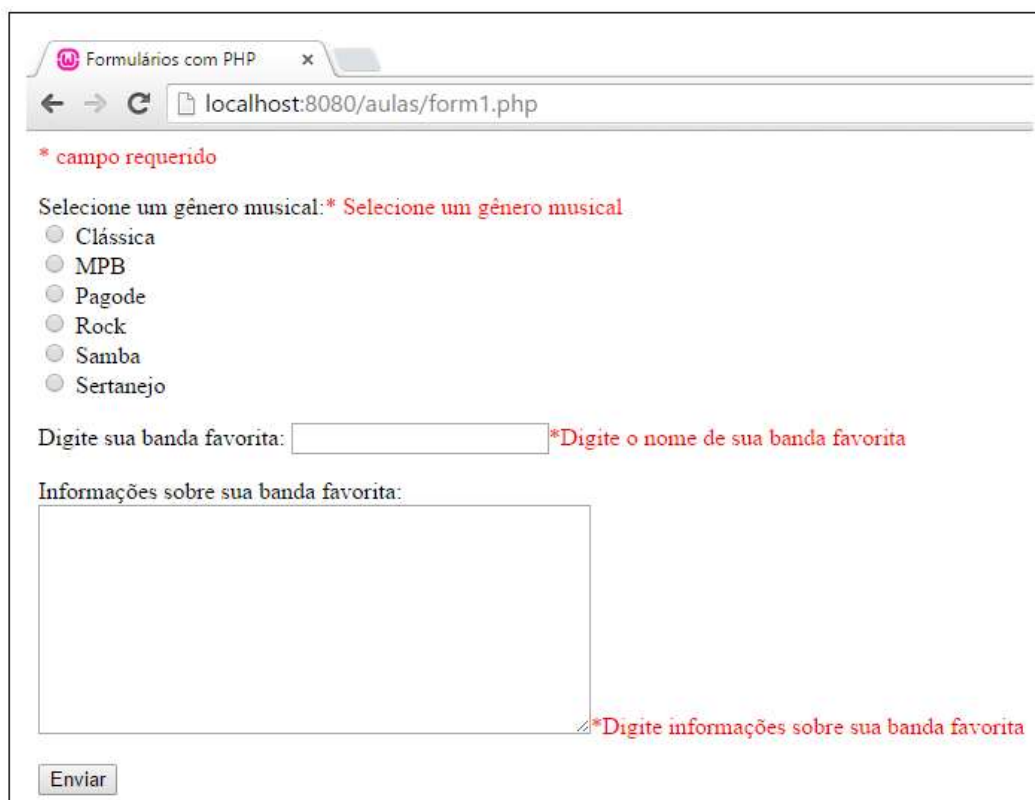
Informações sobre sua banda favorita:

Enviar

Figura 4: Resultado do processamento de um formulário.

Fonte: Fonte: Prof. Adriano Milanez.

Supondo clicássemos no botão "enviar" e que não preenchêssemos os campos obrigatório, teremos o seguinte resultado:



Formulários com PHP x

localhost:8080/aulas/form1.php

* campo requerido

Selecione um gênero musical: * Selecione um gênero musical

- ☐ Clássica
- ☐ MPB
- ☐ Pagode
- ☐ Rock
- ☐ Samba
- ☐ Sertanejo

Digite sua banda favorita: * Digite o nome de sua banda favorita

Informações sobre sua banda favorita:

* Digite informações sobre sua banda favorita

Enviar

Figura5: Mensagens de erro sendo exibidas.

Fonte: Fonte: Prof. Adriano Milanez.

Espero que o conteúdo tenha sido proveitoso. Estudem e até o próximo tópico!

Quiz

Exercício Final

Resgatando informações de Formulários HTML

INICIAR ➤

Referências

CONVERSE, T. & PARK, J. **PHP 5 - A Bíblia**. 2. Ed. Rio de Janeiro: Campus, 2005.

DALL'OGGIO, P. **Programando com orientação a objetos (Inclui *Design Patterns*)**. São Paulo: Novatec, 2009.

MARTIN, J. **Princípios de análise e projeto baseado em objetos**. Rio de Janeiro: Editora Campus, 1994.

MATHEUS, D. **Variáveis e constantes no PHP**. Disponível em <<http://www.diogomatheus.com.br/blog/php/variaveis-e-constantes-no-php/>>, acessado em 19/09/2015, às 12h15min. Publicado em 26/09/2012.

MUTO, C. A. **PHP e MySQL: Guia Introductório**. 3. Ed. Rio de Janeiro: Brasport, 2006.

NIEDERAUER, J. **Desenvolvendo Websites com PHP**. 2. Ed. São Paulo: Novatec, 2004.

PHP. **PHP: Introdução. Manual PHP**. Disponível em <http://php.net/manual/pt_BR/language.variables.basics.php>. Acessado em 04/10/2015, às 14h20min.

PHP. **PHP: Variáveis pré-definidas**. Disponível em <http://php.net/manual/pt_BR/reserved.variables.php>. Acessado em 15/10/2015, às 18h15min.

SOARES, L.; AUGUSTO, B. **Aprendendo a Linguagem PHP**. 1. Ed. Rio de Janeiro: Ciência Moderna, 2007.

SOARES, W. **PHP 5 - Conceitos, Programação e Integração com Banco de Dados**. 4. Ed. São Paulo: Érica, 2004.

TANSLEY, D. **Como criar Web Pages rápidas e eficientes usando PHP e MySQL**. 1. Ed. Rio de Janeiro: Ciência Moderna, 2002.



Avalie este tópico



ANTERIOR
Variáveis pré-definidas

Biblioteca
(<https://www.uninove.br/conhec-a-uninove/biblioteca/sobre-a-biblioteca/apresentacao/>)
Portal Uninove
(<http://www.uninove.br>)
Mapa do Site

Índice

Ajuda?
PRÓXIMO
(<https://ava.uninove.br/criacao-de-funcoes/>)

© Todos os direitos reservados

