



Curso de Operador de Tecnologia da Informação e Comunicações

Desenvolvimento Web

PHP - Programação



Objetivo de
Aprendizado

Entender como funcionam os formulários que utilizam PHP com HTML e quais as particularidades dos métodos GET e POST.



A linguagem PHP é um das mais utilizadas no mundo devido a sua versatilidade e também por causa da facilidade de aprendizado. Existem várias ferramentas que podem ser utilizada com o PHP relacionadas a frameworks que podem ajudar no desenvolvimento rápido e seguro.

Formulários em PHP



Uma das características mais fortes do PHP é a forma como ele trata os **formulários HTML**. Quando um formulário **html** é submetido para um script **PHP** qualquer variável de um formulário será automaticamente disponível para este script. Existem dois métodos de passagem de parâmetros: **Get** e **Post**. O tipo do método é indicado no parâmetro **method** da tag **form**. Exemplo de formulário:

```
<html>
  <body>
    <form action="recebe_dados.php" method="POST">
      Nome:<input type="text" name="nome"/>
      Idade:<input type="text" name="idade"/>

      <input type="submit" value="Enviar" name="enviar"/>
      <input type="reset" name="limpar" value="Limpar"/>
    </form>
  </body>
</html>
```

MÉTODO GET

O método **Get** é o método padrão para o envio de dados, isto significa que se nenhum método for especificado no parâmetro **method** da tag **form** então o método **Get** será utilizado. O método **Get** passa os argumentos de uma página para a próxima, como parte da URL (Uniform Resource Locator) [Converse et al., 2004]. O método **GET** anexa os nomes das variáveis e seus valores na URL designado no atributo **action** do formulário com o separador (?). O símbolo (**&**) indica o início de uma nova variável e o caracter (**=**) separa as variáveis dos seus respectivos valores.

Exemplo:

http://www.meusite.com.br/recebe_dados.php?nome=Aluno&idade=19

Por motivo de **segurança**, o método Get **não deve ser usado** para enviar senhas ou informações confidenciais. O método também não é adequado para variáveis com valores superiores a 2000 caracteres.

Exemplo de formulário usando o método Get

```
<html>
  <body>
    <form action="recebe_dados.php" method="GET">
      Nome:<input type="text" name="nome"/>
      Idade:<input type="text" name="idade"/>

      <input type="submit" value="Enviar" name="enviar">
      <input type="reset" name="limpar" value="Limpar"/>
    </form>
  </body>
</html>
```

Variável \$_GET

Ao clicar no botão submit do formulário da página HTML. Os dados do formulário são enviados para o arquivo PHP. O PHP utiliza da variável **\$_GET** para recolher os dados do formulário enviadas pelo método **GET**. Exemplo: arquivo **recebe_dados.php**

```
<html>
  <body>
    Olá
    <?php echo $_GET["nome"];?>
    <br> você tem
    <?php echo $_GET["idade"];?>
    anos.
  </body>
</html>
```

Saída em HTML será a seguinte:

Olá Aluno.

Você tem 19 anos.

Exemplo 2:

```
<html>
  <body>
    <form method="GET" action="cor.php">

      <p>Informe seu nome:</p><input type="text" name="nome"></p>
      <p>Escolha uma cor:</p>
      <input type="radio" name="cor" value="r"/>Vermelho
      <input type="radio" name="cor" value="g"/>Verde
      <input type="radio" name="cor" value="b"/>Azul</p>

      <input type="submit" value="Enviar"/>
    </form>
  </body>
</html>
```

Arquivo “cor.php”

```

<?php
switch($_GET["cor"]){
    case"r":$background="rgb(255,0,0)";
    break;

    case"g":$background="rgb(0,255,0)";
    break;

    case"b":$background="rgb(0,0,255)";
    break;

    default:$background="rgb(255,255,255)";
    break;
}
?>

<html>
<body style="background:<?php echo $background; ?>;">
    <php? echo"<h1>Olá".$_GET["nome"]."</h1>";?>
</body>
</html>

```

Método POST

Quando o método POST é utilizado no parâmetro method da tag form, os dados do formulário são encapsulados dentro do corpo da mensagem encaminhada ao servidor. Com o **POST** não haverá nenhuma mudança visível na URL.

http://www.meusite.com.br/recebe_dados.php

O método **POST** é o método **mais seguro** e pode ser usado para enviar um número maior de informações. Com esse método é possível enviar imagens ou outros tipos de arquivos.

Variável \$_POST

A variável **\$_POST** é usada para recolher os dados enviados pelo método **POST** do formulário. Exemplo de formulário usando o método **POST**

```
<html>
  <body>
<p>Escolha um curso</p>
<form name="informacoes" method="POST" action="curso.php">

  <select name="curso">
    <option value="ADS"selected>Tecnologia em Análise e Desenvolvimento
    de Sistemas
    </option>
    <option value="Automação">Tecnologia em Automação Industrial
    </option>
    <optionvalue="Matemática">Licenciatura em Matemática
    </option>
    <optionvalue="TMSI">Técnico de Manutenção e Suporte em Informática
    </option>
    <optionvalue="TAI">Técnico em Automação Industrial
    </option>

  </select></p>

  <input name="enviar"type="submit" value="Enviar">
</form>

</body>

</html>
```

Arquivo "curso.php"

```
<html>
  <body>
    <h3>Resultado</h3>

    <?php
      echo "<p>Cursoselecionado:".$_POST['curso']. "</p>\n";
    ?>
  </body>
</html>
```


Variável \$_REQUEST

A variável predefinida **\$_REQUEST** contém as variáveis **\$_GET**, **\$_POST** e **\$_COOKIE**. A variável **\$_REQUEST** pode ser usada para recolher os dados enviados pelos métodos **GET** e **POST**. Exemplo

```
<?php
if(isset($_REQUEST['enviar'])){
    echo "</p><h2>Dados do formulário</h2><hr></p>";
    echo "Nome:{"$_REQUEST['nome']}</br>";
    echo "Comentário:{"$_REQUEST['comentario']}</p><hr>";
}else{
    echo "<h2>Formulário</h2>
        <formname='informacoes' method='POST' action=#>Informe seu
        nome:
            <input name='nome' type='text'><br/><br/>
        Escreva um comentário:</p>
            <textarea name='comentario' rows='3' cols='25'>
            </textarea>
        <br/><br/>
            <input name='enviar' type='submit' value='Enviar'>
        </form>";
}
?>
```

