

<b>4. Xeración dinámica de páxinas web .....</b>	<b>2</b>
4.2. Persistencia de información ente distintos documentos web .....	2
O método GET .....	2
O método POST .....	3
A variable \$_SERVER .....	4

## 4. Xeración dinámica de páxinas web

### 4.2. Persistencia de información ente distintos documentos web

A **persistencia da información entre diferentes documentos web** fai referencia ao feito de que estes **poidan intercambiar información** de maneira que ao pasar dunha páxina a outra, a información dinámica, por exemplo, **os datos que introduciu o usuario nun formulario, non se perdan**.

O Hypertext Transfer Protocol (HTTP) está deseñado para permitir comunicacións entre clientes e servidores. HTTP é un protocolo solicitude-resposta entre cliente e servidor. O navegador web sería o cliente, e a aplicación que alberga o sitio web sería o servidor. O cliente envía unha solicitude HTTP ao servidor, e o servidor devolve unha resposta ao cliente. A resposta contén información de estado e o contido solicitado polo cliente. Existen dous métodos de solicitude HTTP: **GET e POST**. Na presente actividade mostrarase cómo manexar estes métodos de comunicación mediante a linguaxe PHP.

PHP conta cunha serie de **variables predefinidas chamadas superglobais** que están dispoñibles ao longo de tódolos ámbitos nos documentos do sitios web. Xa se utilizaron algunhas nas actividades precedentes: **\$\_FILES** e **\$REQUEST**. Nesta actividade expóñanse outras necesarias para o intercambio de información entre diferentes documentos web.

Polo tanto, os conceptos e destrezas que se manexarán nesta actividade son os que seguen:

- Manexo de solicitudes polo método GET desde PHP.
- Xestión das solicitudes polo método POST mediante PHP.
- Utilización da variable superglobal **\$\_SERVER** de PHP na xestión de formularios.

#### O método GET

En PHP, a variable superglobal **\$\_GET** utilízase para almacenar os valores dun formulario no que se fai o **submit co método GET**. A **información** enviada entre formularios co método GET ten como característica principal que é visible por todo o mundo xa que é **mostrada na barra de direccións do navegador**. A **cantidade** de información que se pode enviar por este método é **limitada**.

O seguinte formulario utiliza o método GET para o envío dun nome e dunha idade.

Exemplo de formulario que usa o método GET	
<pre>&lt;form action="exemplo.php" method="get"&gt;   &lt;p&gt;Nome: &lt;input type="text" name="nome"&gt;&lt;/p&gt;   &lt;p&gt;Idade: &lt;input type="text" name="idade"&gt;&lt;/p&gt;   &lt;input type="submit"&gt; &lt;/form&gt;</pre>	<p>Nome: <input type="text" value="Brais"/></p> <p>Idade: <input type="text" value="19"/></p> <p><input type="submit" value="Enviar"/></p>

No exemplo, a URL que se envía ao servidor ao facer submit é:

`http://192.168.0.2/ud6a2/exemplo.php?nome=B&idade=19`

O formato das URL xeradas cando o método de envío é GET, consta da URL na que se atopa o ficheiro solicitado ao servidor, seguido do carácter **?**, o nome do primeiro paráme-

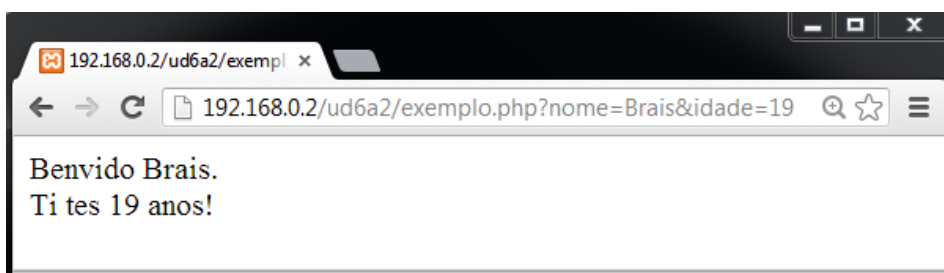
tro enviado, seguido do carácter =, e a continuación o valor do parámetro. Se houberse máis dun parámetro, coma no exemplo, engádesse un carácter &, e a continuación indicárase o nome do seguinte parámetro, =, e o seu valor.

O arquivo que recolle os parámetros enviados por GET, fai uso do array `$_GET`, no que cada elemento será accesible mediante o nome do parámetro. Por exemplo, para acceder ao valor do parámetro nome, usarase `$_GET["nome"]`. O código do ficheiro que recolle os parámetros GET sería o que segue.

#### Código do ficheiro exemplo.php

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title></title>
  </head>
  <body>
    Benvido <?php echo $_GET["nome"]; ?>. <br/>
    Ti tes <?php echo $_GET["idade"]; ?> anos!
  </body>
</html>
```

Cando o usuario pulsa no botón submit, a URL que se envía ao servidor, e paxina coa que o servidor responde amósanse na seguinte imaxe.



Como xa se mencionou, no método GET todos os nomes e valores dos parámetros son mostrados na URL, polo tanto, **este método non debería ser utilizado cando se envíen contrasinais ou información confidencial**. Un aspecto positivo é que, grazas a que a URL contén as variables, será posible gardala na caché do navegador, no historial de navegación, ou ser engadida a favoritos. Por outra banda, este método no é adecuado para enviar parámetros con valores de gran tamaño, polo xeneral, que **non excedan os 2000 caracteres**. Este método debería utilizarse para consultar información, máis que para o envío de datos.

**Unha característica a destacar no método GET é que se poden enviar parámetros escribindo directamente a URL sen que sexa necesario sempre facer o submit dun formulario. Sobre isto versará a tarefa .**

## O método POST

En PHP, a **variable superglobal** `$_POST`, que é un array que contén as variables que son enviadas nun formulario mediante o método POST. A **información enviada** por este método é **invisible a outros** e non ten **límites** en canto ao tamaño da información a enviar. Así a todo, cómpre ter en conta que na configuración do intérprete PHP, **no arquivo php.ini** existe unha directiva `post_max_size` na que se establece o tamaño máximo dos datos a enviar mediante POST a través de PHP.

No seguinte formulario envíase mediante o método POST dous parámetros: nome e idade.

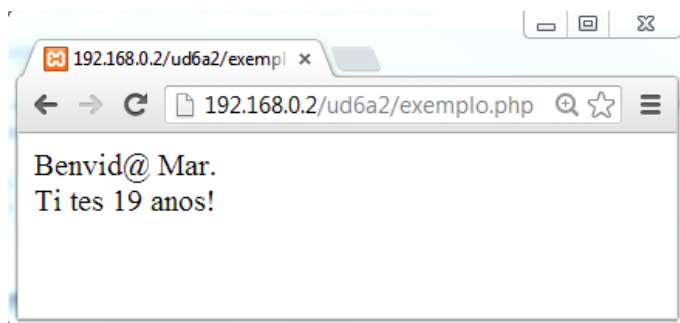
Exemplo de formulario que usa o método POST	
<pre>&lt;form action="exemplo.php" method="post"&gt;   &lt;p&gt;Nome: &lt;input type="text" name="nome"&gt;&lt;/p&gt;   &lt;p&gt;Idade: &lt;input type="text" name="idade"&gt;&lt;/p&gt;   &lt;input type="submit"&gt; &lt;/form&gt;</pre>	<p>Nome: <input type="text" value="Mar"/></p> <p>Idade: <input type="text" value="19"/></p> <p><input type="button" value="Enviar"/></p>

Neste caso a URL xerada na solicitude do ficheiro exemplo.php non contén ningún parámetro: `http://192.168.0.2/ud6a2/exemplo.php`

No código de exemplo.php, recolleranse os parámetros recibidos accedendo aos elementos do array superglobal `$_POST`, indicando como elemento o nome do parámetro ao que se desexa acceder. Por exemplo, para acceder ao nome escríbese `$_POST["nome"]`. O código do ficheiro exemplo.php se mostra a continuación.

Código do ficheiro exemplo.php
<pre>&lt;!DOCTYPE html&gt; &lt;html&gt;   &lt;head&gt;     &lt;meta http-equiv="Content-Type" content="text/html; charset=UTF-8"&gt;     &lt;title&gt;&lt;/title&gt;   &lt;/head&gt;   &lt;body&gt;     Benvid@ &lt;?php echo \$_POST["nome"]; ?&gt;.&lt;br/&gt;     Ti tes &lt;?php echo \$_POST["idade"]; ?&gt; anos!   &lt;/body&gt; &lt;/html&gt;</pre>

O resultado acceder a exemplo.php, xunto cao URL, se amosa na seguinte imaxe.



Ao non estar contidos os parámetros na URL, a páxina xerada non pode ser gardada na caché do navegador, nin no historial, nin nos favoritos. Secasí, non existen restricións no tamaño da información a enviar, soamente a directiva `post_max_size` para facelo desde PHP.

Cómpre indicar que a variable superglobal `$_REQUEST`, que foi utilizada nos exemplos da actividade anterior, é un array que contén tanto os parámetros das variables `$_GET` como os de `$_POST`.

## A variable `$_SERVER`

A variable superglobal `$_SERVER` é un array asociativo que contén información sobre cabeceiras, rutas e localizacións de scripts. Os elementos deste array son creados polo servidor, agora ben, non existe garantía de que cada servidor proporcione todos os elementos.

A continuación amósase un script PHP que informa acerca da IP do servidor, IP do cliente e a localización do código do ficheiro que se está a executar.

Exemplo de uso do array \$_SERVER[]	
<pre> &lt;!DOCTYPE html&gt; &lt;html&gt;   &lt;head&gt;     &lt;meta http-equiv="Content-Type" content="text/html; charset=UTF-8"&gt;     &lt;title&gt;&lt;/title&gt;   &lt;/head&gt;   &lt;body&gt;     Benvid@ &lt;?php echo \$_POST["nome"]; ?&gt;.&lt;br/&gt;     Ti tes &lt;?php echo \$_POST["idade"]; ?&gt; anos!     &lt;p&gt;IP do servidor: &lt;?php echo \$_SERVER["SERVER_ADDR"]; ?&gt;&lt;/p&gt;     &lt;p&gt;IP do cliente: &lt;?php echo \$_SERVER["REMOTE_ADDR"]; ?&gt;&lt;/p&gt;     &lt;p&gt;Localización do script PHP: &lt;?php echo \$_SERVER["PHP_SELF"];     ?&gt;&lt;/p&gt;   &lt;/body&gt; &lt;/html&gt; </pre>	<pre> Benvid@ Mar. Ti tes 19 anos! IP do servidor: 192.168.0.2 IP do cliente: 192.168.0.3 Localización do script PHP: /ud6a2/exemplo.php </pre>

En particular, resulta de especial interese o elemento `$_SERVER["PHP_SELF"]`, que contén a localización relativa do arquivo que contén o script PHP, é dicir, o nome dos directo- rios en nome do ficheiro, pero sen o nome do servidor. Por exemplo, para a URL `http://www.actividade.com/exercicios/exemplo.php` a variable `$_SERVER["PHP_SELF"]` conterá `exercicios/exemplo.php`.

A importancia da variable `$_SERVER["PHP_SELF"]` é que **se usará para enlazar as pá- xinas consigo mesmas**. Por exemplo, unha páxina pode conter un formulario que envíe os datos a esa mesma páxina. Para isto, **o atributo action do formulario deberá conter o nome da propia páxina**. No posto de indicar o nome do ficheiro, é máis práctico utilizar `$_SERVER["PHP_SELF"]`. Unha vantaxe é que se cambiase de nome ou localiza- ción o arquivo, o código seguiría funcionando.

A continuación estudarase o caso dun **documento web que xestiona el mesmo os datos enviados polo formulario**. Conterá un formulario que se recargará se non supera un pro- ceso de validación ou verificación, de xeito que se lle informará ao usuario sobre o erro para que introduza correctamente os datos. Os **campos que estaban ben cubertos mante- rán o valor que xa introducira o usuario**, desta maneira o usuario soamente terá que **reescribir os campos con erros**. Suporase que todos os campos son obrigatorios.

No código do script **deberase distinguir entre a primeira vez que se mostra o formula- rio, en as posteriores**. A primeira vez os controis mostraranse baleiros, mentres que nas posteriores recargas do formulario, os controis mostrarán a información que o usuario in- troducira na vez anterior. Para isto, **comprobarase se o array \$\_POST[] ten cero ele- mentos** (primeira vez que se carga o formulario) **ou non**.

Para estruturar dunha mellor maneira o código do script, definirase unha **función** para construír o formulario, **eliminado desta forma redundancia no código**. Esta función re- cibirá como parámetros cada un dos valores dos controis do formulario. Cando o formula- rio se cargue por vez primeira, pasaránselle valores baleiros, e nas recargas posteriores pa- saránselle os valores almacenados no array `$_POST`. No caso do control ficheiro, aínda que conta co parámetro value, a maioría dos navegadores ignorarán este parámetro por motivos de seguridade. Polo tanto, será o usuario quen seleccione o ficheiro sempre, e non se encherá automaticamente o control ao recargar o formulario.

Finalmente, mostraranse os erros que se puidesen producir, e en caso en de no producirse ningún, non se mostrará xa o formulario, indicarse que os datos foron enviados correc- tamente, e mostrarase o conxunto de datos recibidos.

## Código do exemplo de uso de \$\_SERVER["PHP\_SELF"]

```
<?php
function incForm($marMod, $nKm, $urlFab, $com, $eqEx){?>

<form action="<?php echo $_SERVER["PHP_SELF"];?>" method="post">
    <p>Marca e modelo: <input type="text" name="marMod" value="<?php echo $marMod;?>" /></p>
    <p>Nº km: <input type="text" name="nKm" value="<?php echo $nKm;?>" /></p>
    <p>URL do fabricante:<input type="text" name="urlFab" value="<?php echo $urlFab;?>" /></p>
    <p>Combustible: <input type="radio" name="com" value="Diesel" <?php if ($com=="Diesel") echo "checked"?> Diesel
    <input type="radio" name="com" value="Gasolina" <?php if ($com=="Gasolina") echo "checked"?> Gasolina
    <input type="radio" name="com" value="Eléctrico/Híbrido" <?php if ($com=="Eléctrico/Híbrido") echo "checked"?> Eléctrico/Híbrido
    </p>
    <p>Equipamento extra:<select name="eqEx[]" size="3" multiple="multiple">
        <option <?php if (in_array("Pintura metalizada", $eqEx)){echo "selected='selected'";}?>Pintura metalizada</option>
        <option <?php if (in_array("Sistema de navegación", $eqEx)){echo "selected='selected'";}?>Sistema de navegación</option>
        <option <?php if (in_array("Reposabrazos central dianteiro", $eqEx)){echo "selected='selected'";}?>Reposabrazos central dianteiro</option>
    </select></p>
    <p><input type="reset" value="Borrar"/> <input type="submit" value="Enviar" /></p>
</form>
<?php
}
?>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
        <title></title>
    </head>
    <body>
        <?php
        if (count($_POST)==0)
            incForm("", "", "", array());
        else{
            $cadeaErros="";
            $marMod= htmlspecialchars(trim(strip_tags($_POST["marMod"])), ENT_QUOTES, "ISO-8859-1");
            if ($marMod=="")
                $cadeaErros.= "O campo Marca e modelo non pode estar valerio. ";

            $nKm= htmlspecialchars(trim(strip_tags($_POST["nKm"])), ENT_QUOTES, "ISO-8859-1");
            if ($nKm=="")
                $cadeaErros.= "O campo Nº km non pode estar baleiro. ";
            else
                if (filter_var($nKm, FILTER_VALIDATE_INT)){
                    if ((int)$nKm<=0)
                        $cadeaErros.= "O Nº km deber ser maior que 0. ";
                } else {
                    $cadeaErros.= "O Nº km non é un número enterio. ";
                }

            $urlFab= htmlspecialchars(trim(strip_tags($_POST["urlFab"])), ENT_QUOTES, "ISO-8859-1");
            if ($urlFab == "")
                $cadeaErros.= "O campo URL do fabricante non pode estar baleiro. ";
            else{
                if (filter_var($urlFab, FILTER_VALIDATE_URL))
                    $cadeaErros.= "A URL do fabricante especificada non é válida.";
            }

            $mod=(isset($_POST["com"]))
            ? $_POST["com"]
            : "";

            $eqEx=(isset($_POST["eqEx"]))
            ? $_POST["eqEx"]
            : array();

            if ($cadeaErros != ""){
                incForm($marMod,$nKm, $urlFab, $mod, $eqEx);
                echo "<strong><em>$cadeaErros</em></strong>";
            }
        }
    }
}
```

```

    }else{
        echo "<p>O formulario encheuse correctamente. Os datos recibidos son:</p>";
        echo "<p>Marca e modelo: ".$_POST["marMod"]."</p>";
        echo "<p>Nº km: ".$_POST["nKm"]."</p>";
        echo "<p>URL do fabricante: ".$_POST["urlFab"]."</p>";
        echo "<p>Combustible: ".$_POST["com"]."</p>";
        echo "<p>Equipamento extra: ";
        for ($i=0; $i<count($_POST["eqEx"]); $i++){
            echo $_POST["eqEx"][$i];
            if ($i==(count($_POST["eqEx"])-1))
                echo ".";
            else
                echo ",";
        }
        echo "</p>";
    }
}
?>
</body>
</html>

```

### Exemplos de execución do anterior código:

Exemplos de execución do código que fai uso de \$_SERVER["PHP_SELF"]	
<p>Marca e modelo: <input type="text" value="VOLVO C70 2.4 140cv"/></p> <p>Nº km: <input type="text" value="78000"/></p> <p>URL do fabricante: <input type="text" value="http://www.volvocars.com"/></p> <p>Combustible: <input type="radio"/> Diesel <input checked="" type="radio"/> Gasolina <input type="radio"/> Eléctrico/Híbrido</p> <p>Equipamento extra: <div><div>Pintura metalizada</div><div>Sistema de navegación</div><div>Reposabrazos central dianteiro</div></div></p> <p><input type="button" value="Borrar"/> <input type="button" value="Enviar"/></p>	<p>O formulario encheuse correctamente. Os datos recibidos son:</p> <p>Marca e modelo: VOLVO C70 2.4 140cv</p> <p>Nº km: 78000</p> <p>URL do fabricante: http://www.volvocars.com</p> <p>Combustible: Gasolina</p> <p>Equipamento extra: Pintura metalizada, Sistema de navegación.</p>
<p>Marca e modelo: <input type="text" value="VOLVO C70 2.4 140cv"/></p> <p>Nº km: <input type="text"/></p> <p>URL do fabricante: <input type="text" value="TT"/></p> <p>Combustible: <input type="radio"/> Diesel <input type="radio"/> Gasolina <input type="radio"/> Eléctrico/Híbrido</p> <p>Equipamento extra: <div><div>Pintura metalizada</div><div>Sistema de navegación</div><div>Reposabrazos central dianteiro</div></div></p> <p><input type="button" value="Borrar"/> <input type="button" value="Enviar"/></p>	<p>Marca e modelo: <input type="text" value="VOLVO C70 2.4 140cv"/></p> <p>Nº km: <input type="text"/></p> <p>URL do fabricante: <input type="text" value="TT"/></p> <p>Combustible: <input type="radio"/> Diesel <input type="radio"/> Gasolina <input type="radio"/> Eléctrico/Híbrido</p> <p>Equipamento extra: <div><div>Pintura metalizada</div><div>Sistema de navegación</div><div>Reposabrazos central dianteiro</div></div></p> <p><input type="button" value="Borrar"/> <input type="button" value="Enviar"/></p> <p><i>O campo Nº km non pode estar valerio. A URL do fabricante especificada non é válida.</i></p>