

RECUPERAR DATOS DE FORMULARIOS. VARIABLES \$_REQUEST, \$_GET e \$_POST

Cando un usuario pulsa o botón enviar dun formulario, a información que contiñan os seus campos é enviada a unha dirección URL desde onde teremos que recuperala para tratala dalgunha maneira. Por exemplo, se realiza unha compra, teremos que recuperar os datos para completar o proceso de pago. A información do formulario "viaxa" almacenada en variables que poderemos recuperar e utilizar mediante PHP.

A recuperación de variables con PHP é bastante fácil, xa sexan datos dun formulario enviado polo método post (parámetros non visibles) ou polo método get (parámetros na URL si visibles).

Primeiro de nada hai que aclarar que hai varios métodos para recuperar variables con PHP. Para os formularios temos os métodos GET, POST, e REQUEST.

RECUPERAR VARIABLES CON GET

GET permítenos recuperar parámetros desde a URL ou desde formularios enviados co método GET. Escribe este código e gárdao cun nome de arquivo como exemplo1.html. A continuación, sube o ficheiro ao servidor, introduce un nome no campo do formulario e visualiza o resultado.

```
<form name="formulario" method="get" action="exemploGet.php">
  Nombre: <input type="text" name="nombre" value="">
  <input type="submit" />
</form>
```

Antes de pulsar sobre o botón enviar, debemos crear o ficheiro exemploGet.php que é o indicado no formulario para o envío dos datos.

Escribe este código e gárdao cun nome de arquivo como exemploGet.php.

```
<?php
$nombre = $_GET['nombre'];
echo $nombre;
?>
```

A continuación, sube o ficheiro ao servidor na mesma cartafol onde subiches o exemplo1.html

Se tivésemos máis dun campo no formulario, teríamos tantas liñas como as seguintes, que fisen necesarias:

```
$apellidos = $_GET['apellidos'];
$telefono = $_GET['movil'];
$edadPersona = $_GET['edad'];
```

Fíxache en que unha cousa é a variable na que almacenamos a información recuperada, e outra cousa é o nome do campo do formulario de onde provén. Por exemplo en `$apellidos = $_GET['apellidos'];` coinciden o nome da variable que utilizamos co nome do campo do formulario. Con todo, en `$telefono = $_GET['movil'];` non coinciden. Neste caso, o campo que provén do formulario chámase "movil" mentres que a información que veña nese campo almacenámola nunha variable á que chamamos `$telefono`. Finalmente, en `$edadPersona = $_GET['edad'];` estamos a almacenar nunha variable á que chamamos `$edadPersona` a información proveniente dun campo do formulario denominado "edad".

Con frecuencia os nomes das variables e dos campos do formulario fanse coincidir, pero noutras ocasións non. Isto queda a elección do programador.

RECUPERAR VARIABLES CON POST

POST permítenos recuperar datos enviados desde formularios co método POST de xeito invisible para o usuario.

```
<form name="formulario" method="post" action="exemploPost.php">  
  Nombre: <input type="text" name="nombre" value="">  
  <input type="submit" />  
</form>
```

Creamos o ficheiro php:

```
<?php  
$nombre = $_POST['nombre'];  
echo $nombre;  
?>
```

En POST ocorre o mesmo que no casi de GET, teremos tantas variables e POST como entradas teñamos no formulario:

```
$apellidos = $_POST['apellidos'];
```

```
$telefono = $_POST['movil'];
```

```
$edadPersona = $_POST['edad'];
```

RECUPERAR VARIABLES CON REQUEST

REQUEST permítenos capturar variables enviadas desde formularios cos métodos GET ou POST.

Por exemplo, podemos ter:

```
<form name="formulario" method="get" action="exemploRequest.php">  
  Nombre: <input type="text" name="nombre" value="">  
  <input type="submit" />  
</form>
```

ou

```
<form name="formulario" method="post" action=" exemploRequest.php">  
  Nombre: <input type="text" name="nombre" value="">  
  <input type="submit" />  
</form>
```

no primeiro caso enviamos o dato con GET e no segundo con POST.

Bastaría con empregar o seguinte código, para poder recoller o dato, independentemente de que se teña empregado POST ou GET:

```
<?php  
$nombre = $_REQUEST['nombre'];  
echo $nombre;  
?>
```

Exercicio resolto

Deseñar un formulario web que pida a altura e o diámetro dun cilindro en metros. Unha vez o usuario introduza os datos e pulse o botón calcular, deberá calcularse o volume do cilindro e mostrarse o resultado no navegador. O envío de datos debe facerse por POST e a recuperación con REQUEST.

Solución en pseudocódigo que habería que apuntar nun papel antes de comenzar a programar:

1. Inicio
2. Mostrar “Introduza o diámetro, en metros” : Pedir D
3. Mostrar “Introduza a altura, en metros” : Pedir H
4. $R = D/2$: $Pi = 3,141593$
5. $V = Pi * (R ^ 2) * H$
6. Mostrar “O volume do cilindro é de”, V, “metros cúbicos”
7. Fin

Escomenzaríamos por programar a parte HTML:

```

<!DOCTYPE html>
<html>
    <head>
        <title>Exemplo captura de datos</title>
        <meta charset="utf-8">
    </head>
    <body>
        <form name="formularioDatos" method="post" action="exemploRequest.php">
            <p> CÁLCULO DO VOLUME DUN CILINDRO </p>
            <br/>
            Introduzca o diámetro en metros: <input type="text" name="diam" value="">
            <br/> <br/>
            Introduzca a altura en metros: <input type="text" name="altu" value="">
            <br/> <br/>
            <input value="Calcular" type="submit" />
        </form>
    </body>
</html>

```

Creamos o programa en PHP exemploRequest.php:

```

<?php
    $diametro = $_REQUEST['diam'];
    $altura = $_REQUEST['altu'];
    $radio = $diametro/2;
    $Pi = 3.141593;
    $volumen = $Pi*$radio*$radio*$altura;
    echo "<br/> O volume do cilindro é de". $volumen. "metros cúbicos";
?>

```