# Formularios

## Método POST

De todas maneras existe un problema si intentamos acceder a la vista del fichero recible.php directamente a través de la url (es decir no accedemos directamente desde el formulario).

**Notice**: Undefined variable: nombre in **C:\Users\xurxo\Desktop\CURSO\_MF0492\_3\php\PROYECTOS\FORMARIO\_1\recibe.php** on line **6**

Esto es debido a que el array $\_POST se crea tras pulsar el botón submit. Y si no se pulsa no existe directamente esta variable.

Hay varias soluciones. Una de ellas es ver si este $\_POST tiene valores. Con un simple if:

<?php

if($\_POST){

    //tratamos las variables puesto que la página se carga desde nuestro formulario

}else{

    //redirigimos

    header('Location: url\_que\_deseemos');

}

La url que solemos poner en caso de redirección suele ser la de nuestro formulario.

El código un poco más optimizado sería:

<?php

if(!$\_POST)

    //redirigimos

    header('Location: http://localhost/PROYECTOS/FORMARIO\_1/');

//todo nuestro código. No hace falta ya else

Por último el método POST nos garantiza seguridad con información sensible, por ejemplo, un formulario de acceso, o cuando hay que enviar un gran volumen de información o también imágenes.

## Método GET

Obviamente el código es muy parecido. Puesto que si existe un array $\_POST. Existe un arreglo $\_GET

Obviamente se opera de la misma manera. Este método presenta problemas puesto que nos permite inyectar código:

[http://localhost/PROYECTOS/FORMARIO\_1/recibe.php?nombre=<h1>nombre</h1](http://localhost/PROYECTOS/FORMARIO_1/recibe.php?nombre=%3ch1%3enombre%3c/h1)>

[http://localhost/PROYECTOS/FORMARIO\_1/recibe.php?nombre=<script>alert(‘Xurxo’)</](http://localhost/PROYECTOS/FORMARIO_1/recibe.php?nombre=%3cscript%3ealert(‘Xurxo’)%3c/)script>

Por tanto a lo que vamos. En el método GET todos los datos van por URL. No pasa nada si la utilizamos en datos no confidenciales. Por ejemplo para marcar las páginas de artículos de un blog.

Ejercicio

Inyecta código en recibe.php una vez recibido los datos por método GET

Existe una forma para que no nos puedan inyectar código en php y es envolver nuestras variables con la función htmlspecialchars(). Por ejemplo:

if(!$\_GET)

header('Location: http://localhost/PROYECTOS/FORMARIO\_1/');

// echo "<pre>";

// print\_r($\_GET);

// echo "</pre>";

$nombre=$\_GET["nombre"];

echo htmlspecialchars($nombre);

Ejercicio

Vuelve a inyectar código en recibe.php una vez recibido los datos por método GET y con htmlspecialchars()