

TEMA 3

Variables del Sistema o superglobales

php.ini

include

Variables del Sistema

Las variables del sistema son variables creadas por PHP y que nos permiten obtener información de nuestros scripts.

Son también conocidas como variables de entorno o superglobales.

Son un conjunto de variables, normalmente array, que van a estar accesibles desde cualquier parte de nuestro código.

Variables del Sistema

Las variables del sistema que más vamos a usar y que describiremos con más detalle a lo largo de este tema son

- \$GLOBALS
- \$_SERVER
- \$_GET
- \$_POST
- \$_FILES
- \$_REQUEST
- \$_COOKIE
- \$_SESSION
- ...

Podemos encontrar más información en

<http://www.php.net/manual/es/reserved.variables.php>

\$GLOBALS

<http://www.php.net/manual/es/reserved.variables.globals.php>

\$GLOBALS es un array asociativo que contiene las referencias a todas las variables que están definidas en el ámbito global del script.

Los nombres de las variables son las claves del array (sin el \$).

`$GLOBALS['nombre']`

Insistimos en que el array GLOBALS para guardar variables globales no hay que utilizarlo salvo en casos muy meditados, pues en una gran mayoría de los casos tenemos alternativas más adecuadas.

`$_SERVER`

<http://www.php.net/manual/es/reserver.variables.server.php>

Es un array que contiene información, tales como cabeceras, rutas y localizaciones del código. Las entradas de este array son creadas por el servidor web. No existe garantía de que cada servidor web proporcione todas estas entradas, ya que existen servidores que pueden omitir algunas (por ejemplo por motivos de seguridad) o proporcionar otras que no se encuentran recogidas aquí.

`$_SERVER['nombre']`

`&_GET`

<http://www.php.net/manual/es/reserved.variables.get.php>

Es un array asociativo de variables pasado al script actual vía parámetros URL.

`$_GET['nombre']`

Por ejemplo:

www.midominio.com/usuario.php?id=4&nombre=pepe

Es una manera que tenemos de pasarle parámetros de entrada a un script para que se ejecute.

Lo veremos con más detalle en el Tema 4.

`&_POST`

<http://www.php.net/manual/es/reserved.variables.post.php>

Es un array asociativo que variables que se le pasan al script actual a través del método HTTP POST. El usuario no las ve directamente, a diferencia de `$_GET`.

`$_POST['nombre']`

Al igual que la anterior `$_GET`, entraremos en detalles más adelante en el tema 4.

`$_FILES`

<http://www.php.net/manual/es/reserved.variables.files.php>

Es otra variable de tipo array asociativo que contiene elementos cargados al script actual a través del método POST.

`$_FILES['nombre']`

Se utiliza para tratar la subida de archivos al servidor desde un formulario de entrada.

Esta variable la veremos en un tema específico de subida y manejo de archivos en el servidor.

`$_COOKIE`

<http://www.php.net/manual/es/reserved.variables.cookies.php>

Es un array asociativo que contiene variables que le podemos pasar al script actual a través de cookies HTTP.

`$_COOKIES['nombre']`

Ampliaremos información más adelante.

`$_REQUEST`

<http://www.php.es/manual/es/reserved.variables.request.php>

Es un array asociativo que por defecto contiene todo el contenido de los array `$_POST`, `$_GET` y `$_COOKIE`.

`$_REQUEST['nombre']`

`$_SESSION`

<http://www.php.net/manual/es/reserved.variables.session.php>

Es un array asociativo que contiene las variables que queremos almacenar de la sesión y que queremos que estén disponibles en el script actual.

`$_SESSION['nombre']`

También ampliaremos información más adelante

phpinfo()

Función de PHP que nos permite ver toda la información sobre nuestra versión de PHP instalada.

Nos muestra visualmente el archivo de configuración de php.ini

```
<?php
```

```
    phpinfo();
```

```
?>
```


php.ini

Archivo de configuración de PHP que podemos modificar a nuestro gusto para una configuración básica o avanzada de nuestro servidor, ya sea local o contratado.

En servidores dedicados lo manejamos nosotros mismos

En hosting o servidores compartidos no podemos acceder a este archivo, pues afecta a todo lo que haya en el servidor.

Para modificarlo, lo podemos encontrar en la carpeta de PHP, que dependerá de qué despliegue hayamos hecho o tengamos en nuestro servidor.

También se puede modificar en tiempo de ejecución, ya que PHP dispone de una serie de funciones para ello, pero sólo se modifican para el script que estamos ejecutando. Si ejecutamos otro, tomará los valores por defecto de php.ini

php.ini

Directivas interesantes de php.ini:

- `max_execution_time` (Tiempo máximo de ejecución de un script).
- `display_errors` (Mostrar o no los errores en el navegador).
- `error_reporting` (Que tipo de mensajes queremos que aparezcan).
- `upload_max_filesize` (Establece un tamaño máximo para los upload).
- `error_log` (Nos dice donde se encuentra nuestro log de errores de PHP).

Include

Función PHP que nos permite incluir y evaluar archivos específicos en nuestro código.

La sintaxis a emplear es:

```
include ("NombreArchivo");
```

```
include "NombreArchivo";
```

Cualquiera de las dos es válida.

Si el archivo a incluir está en otro directorio, hemos de incluir la ruta, ya sea relativa o absoluta.

include_once

Si tenemos un archivo que por alguna circunstancia se pueda incluir dos veces y no es esa nuestra intención, podemos usar

```
include_once("NombreArchivo");
```

```
include_once "NombreArchivo";
```

Cualquiera de las dos es válida.

Su función es comprobar si el archivo que queremos incluir ya ha sido incluido o no. Si no lo ha sido, lo incluye, y si ya se incluyó anteriormente, ignora la línea y no hace nada.

requiere

Funciona igual que include, y también se puede utilizar como require_once.

```
require("NombreArchivo");
```

```
require_once("NombreArchivo");
```

```
require "NombreArchivo";
```

```
require_once "NombreArchivo";
```

include y require se diferencian en que la primera si falla por algún motivo (p.e. no encuentra el archivo) muestra un warning y la segunda genera un error fatal, no pudiendo seguir con la ejecución de nuestro script.