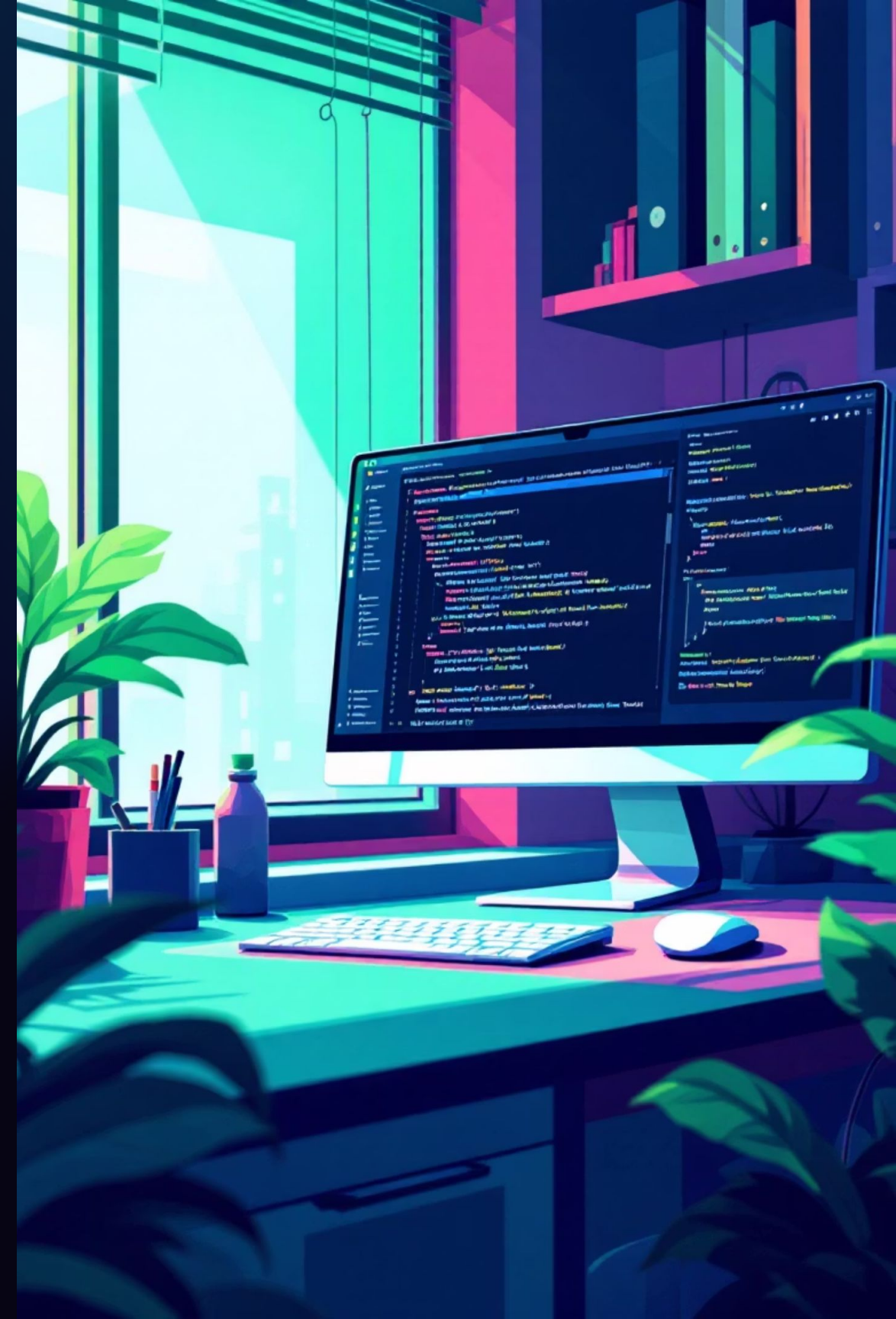


# Conceptos sobre programación web en PHP

Variables superglobales, validaciones, cabeceras de respuesta, cookies y sesiones para gestionar el estado web.

Adrián Antibón Mainé

Carlos Madroñal Sánchez



# Variables de servidor en PHP

Las variables superglobales permiten acceder a información del entorno, cliente y solicitud HTTP.

## \$\_GET

Parámetros enviados en la URL mediante HTTP GET.

1

```
<form action="Archivo.php" method="get">
  <label for="name">Nombre:</label>
  <input type="text" id="name" name="name">
  <button type="submit">Enviar</button>
</form>
```

```
<?php
echo 'Hello ' . htmlspecialchars($_GET["name"]) .
'!';
?>
```

## \$\_POST

Datos enviados mediante formularios con method="post". La diferencia con el método get es que los datos no se muestran en el url, por lo que es más seguro.

2

```
<form action="Archivo.php" method="get">
  <label for="name">Nombre:</label>
  <input type="text" id="name" name="name">
  <button type="submit">Enviar</button>
</form>
```

```
<?php
echo 'Hello ' . htmlspecialchars($_POST["name"])
. '!';
?>
```

## \$\_REQUEST

Combina \$\_GET, \$\_POST y \$\_COOKIE (evitar por seguridad).

3

```
<?php

switch($_SERVER['REQUEST_METHOD'])
{
case 'GET': $the_request = &$_GET; break;
case 'POST': $the_request = &$_POST; break;
.default:
}
?>
```

# Variables de servidor en PHP

Las variables superglobales permiten acceder a información del entorno, cliente y solicitud HTTP.

## **\$\_FILES**

Información sobre archivos subidos con enctype="multipart/form-data".

4

```
<!-- formulario.html -->
<form method="post" enctype="multipart/form-data">
  <input type="file" name="archivo">
  <button>Enviar</button>
</form>
```

```
<?php
if ($_POST && isset($_FILES['archivo'])) {
    echo "Archivo recibido: " .
    htmlspecialchars($_FILES['archivo']['name']);
}
?>
```

## **\$\_SERVER**

Información del servidor, solicitud HTTP, rutas y cabeceras.

5

```
<?php
echo "Método: " . $_SERVER['REQUEST_METHOD'] . "<br>";
echo "IP: " . $_SERVER['REMOTE_ADDR'];
?>
```

Método: GET  
IP: 83.45.233.120

## **\$\_ENV**

Variables de entorno del sistema operativo. contraseñas, rutas, tokens, usuario, idioma o entorno de ejecución

6

```
<?php
echo 'My username is ' . $_ENV["USER"] . '!!';
?>
```





# Validación de datos: isset()

La función `isset()` verifica si una variable está definida y no es null, evitando errores al acceder a datos no existentes.

```
if (isset($_GET['nombre'])) {  
    $nombre = $_GET['nombre'] ?? 'Invitado';  
} else {  
    $nombre = 'Invitado';  
}
```

# Cabeceras de respuesta en PHP

Se envían antes de cualquier salida y controlan cómo responde el servidor. Se usan con la función `header()`.

## Redirección

indica al navegador que debe ir automáticamente a otra página

```
header('Location: otra_pagina.php');
```

## Contenido JSON

indica al navegador que el contenido de la respuesta no es HTML, sino JSON

```
header('Content-Type: application/json');
```

## Caché

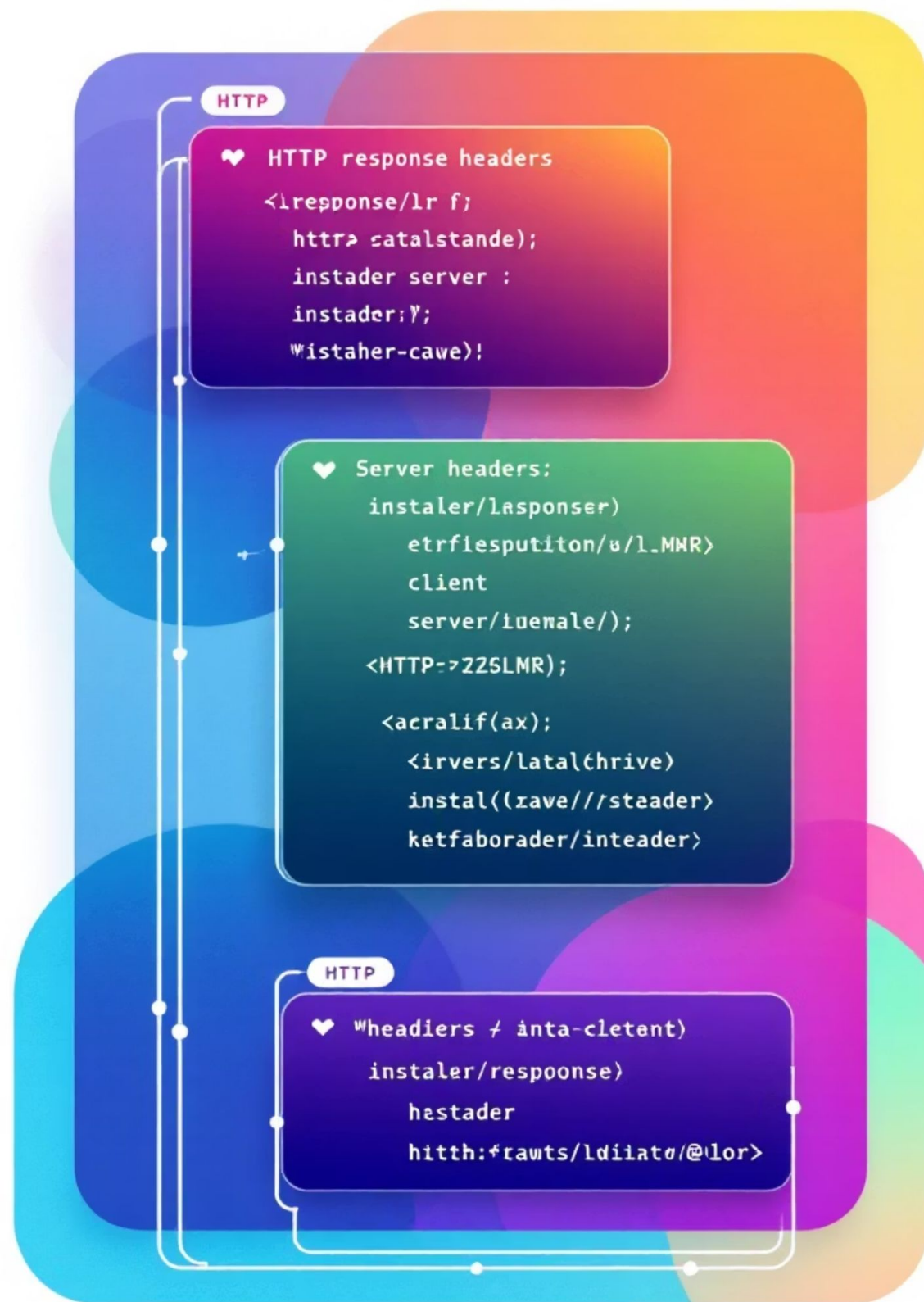
Controla si el navegador puede almacenar la respuesta en caché

```
header('Cache-Control: no-cache, no-store');
```

## Estado HTTP

Permite indicar el estado de la respuesta para saber si la operación fue correcta.

```
http_response_code(404);
```







# Gestión del estado: Cookies y Sesiones

## Cookies

Datos almacenados en el navegador del cliente. Se crean con `setcookie()` y duran según su configuración.

```
setcookie('usuario',  
'Ana', time() + 3600);
```

## Sesiones

Datos almacenados en el servidor, asociados a un ID de sesión. Se inician con `session_start()`.

```
session_start();$_SESSION[  
'logueado'] =  
true;
```

## **\$\_COOKIE**

Cookies enviadas por el cliente.

```
<?php
// Crear una cookie que dura 1 hora
setcookie("usuario", "Juan", time() + 3600);
?>
```

```
<?php
if (isset($_COOKIE["usuario"])) {
    echo "Hola " . htmlspecialchars($_COOKIE["usuario"]);
} else {
    echo "No hay cookie definida.";
}
?>
```

## **\$\_SESSION**

Variables de sesión del usuario actual, requiere session\_start().

```
<?php
session_start();
$_SESSION["name"] = "Juan";
?>
```

```
<?php
session_start();
echo "Hola " . htmlspecialchars($_SESSION["name"]) . "!";
?>
```

```
<form action="sesion.php" method="post">
    <label for="name">Nombre:</label>
    <input type="text" id="name" name="name">
    <button type="submit">Guardar en sesión</button>
</form>
```

```
<?php
session_start();
$_SESSION["name"] = $_POST["name"];
echo "Guardado en sesión.";
?>
```



# Configuración de sesiones

Parámetros clave que controlan el comportamiento de las sesiones en el servidor.

## **session.cookie\_lifetime**

Establece el tiempo de vida de la cookie de sesión en segundos.

"0": Dura mientras el navegador esté abierto.

## **session.gc\_maxlifetime**

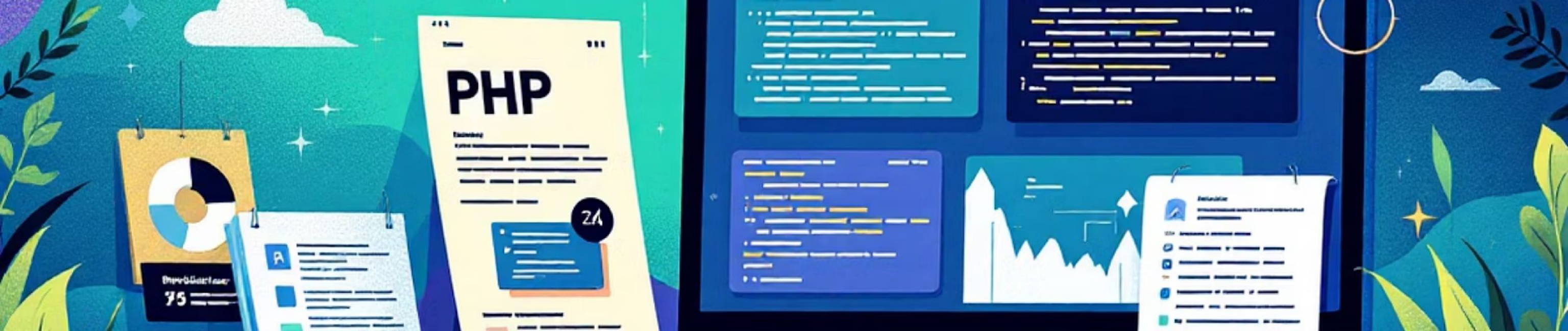
Establece el tiempo máximo que el servidor conserva datos de sesión sin actividad, independientemente de la cookie.

## **session.cookie\_secure**

Mejoran seguridad: fuerzan el envío de cookie solo por HTTPS y no accesible desde JavaScript.







# Bibliografía

Referencias oficiales de PHP para profundizar en variables superglobales, cabeceras y gestión de sesiones.

- <https://www.php.net/manual/en/reserved.variables.environment.php>
- <https://www.php.net/manual/en/reserved.variables.server.php>
- <https://www.php.net/manual/es/function.header.php>
- <https://www.php.net/manual/es/reserved.variables.session.php>

# Fin

Adrián Antibón Mainé

Carlos Madroñal Sánchez

