

UNIVERSIDAD TECNOLÓGICA DE TIJUANA



**TSU EN TECNOLOGÍAS DE LA INFORMACIÓN ÁREA
DESARROLLO DE SOFTWARE MULTIPLATAFORMA**

MATERIA:

Aplicaciones Web

ACTIVIDAD:

Super Globales \$GET y \$POST en PHP

ESTUDIANTES:

- RAMIREZ NAVARRO MARCOS DAVID

PROFESOR:

Dr. Ray Brunett Parra Galaviz

GRUPO:

3E BIS

FECHA:

17/10/2024

INTRODUCCIÓN

En el siguiente documento se hablará sobre la definición, características y se dará un ejemplo para comprender la función de las super globales `$_GET` y `$_POST`, su comportamiento en los lenguajes HTML y PHP.

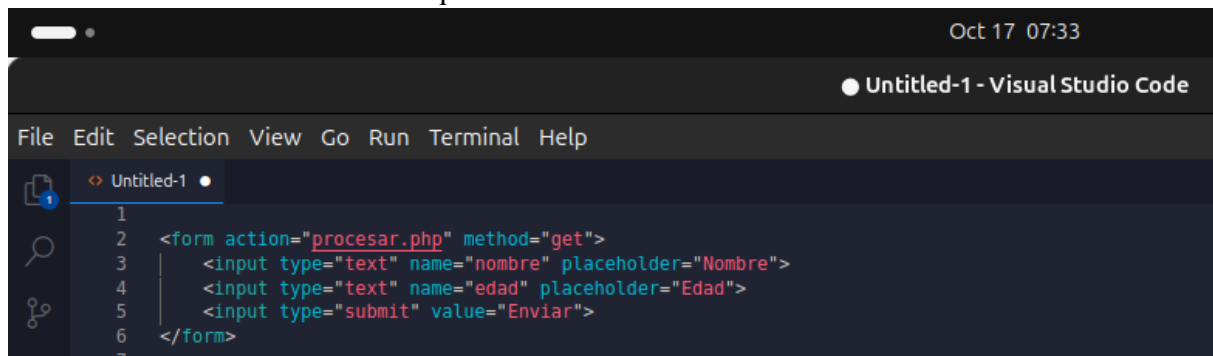
DESARROLLO

Super global \$_GET:

La superglobal GET en PHP es un arreglo asociativo que se utiliza para acceder a los datos enviados a través de la URL, específicamente mediante el método HTTP GET. Cuando un formulario se envía utilizando este método, los datos se adjuntan a la URL en forma de pares clave-valor.

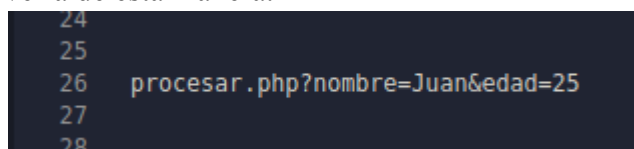
Ejemplo de su utilización:

Utilizando un formulario HTML que envía datos



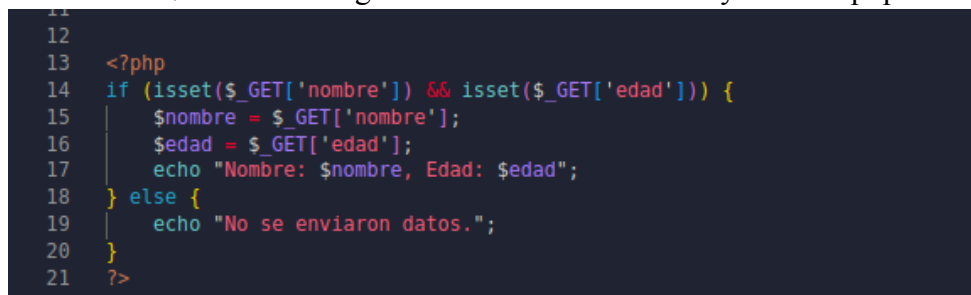
```
Oct 17 07:33
Untitled-1 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
<?php
1
2 <form action="procesar.php" method="get">
3     <input type="text" name="nombre" placeholder="Nombre">
4     <input type="text" name="edad" placeholder="Edad">
5     <input type="submit" value="Enviar">
6 </form>
7
```

Si un usuario manda valores a través del formulario para las variables “nombre” y “edad” se vería de esta manera.



```
24
25
26 procesar.php?nombre=Juan&edad=25
27
28
```

Utilizamos un condicional para validar que se mandaron datos y las variables que contienen el \$_GET se le asignan a las variables nombre y edad de php



```
11
12
13 <?php
14 if (isset($_GET['nombre']) && isset($_GET['edad'])) {
15     $nombre = $_GET['nombre'];
16     $edad = $_GET['edad'];
17     echo "Nombre: $nombre, Edad: $edad";
18 } else {
19     echo "No se enviaron datos.";
20 }
21 ?>
22
```

Características de \$_GET

Datos accesibles: Los datos enviados mediante \$_GET son accesibles a través de sus claves. Por ejemplo, \$_GET ['nombre'] devuelve el valor asociado a la clave nombre.

Limitaciones: Hay un límite en la longitud de la URL (que puede variar según el servidor y el navegador), por lo que no es adecuado para enviar grandes cantidades de datos o datos sensibles.

Visibilidad: Los datos enviados con GET son visibles en la URL, lo que puede no ser seguro para información confidencial, como contraseñas.

Uso común: Es comúnmente utilizado para búsquedas, filtros y navegación, donde los parámetros de la URL pueden ser fácilmente compartidos.

Super global \$_POST:

La superglobal \$_POST en PHP es un arreglo asociativo que se utiliza para acceder a los datos enviados a través del método HTTP POST. Este método se usa comúnmente para enviar datos de formularios, especialmente cuando se requiere enviar grandes cantidades de información o datos sensibles.

Ejemplo de su utilización:

Un formulario que utiliza el método POST está estructurado de la siguiente manera:

```
Untitled-1
1
2 <form action="procesar.php" method="post">
3   <input type="text" name="nombre" placeholder="Nombre" required>
4   <input type="password" name="contrasena" placeholder="Contraseña" required>
5   <input type="submit" value="Enviar">
6 </form>
7
```

Cuando un usuario completa el formulario y lo envía, los datos se envían al archivo procesar.php de forma segura y no aparecen en la URL.

Para acceder a los datos en PHP en el archivo procesar.php, puedes acceder a los datos enviados utilizando \$_POST

```
14 <?php
15 if (isset($_POST['nombre']) && isset($_POST['contrasena'])) {
16     $nombre = htmlspecialchars($_POST['nombre']);
17     $contrasena = htmlspecialchars($_POST['contrasena']);
18     echo "Nombre: $nombre, Contraseña: $contrasena";
19 } else {
20     echo "No se enviaron datos.";
21 }
22 ?>
23
```

Características de \$_POST

Datos accesibles: Los datos enviados mediante \$_POST son accesibles a través de sus claves. Por ejemplo, \$_POST['nombre'] devuelve el valor asociado a la clave nombre.

Sin límite en la longitud: A diferencia de \$_GET, no hay un límite estricto en la cantidad de datos que puedes enviar, lo que lo hace ideal para formularios más grandes.

Más seguro para datos sensibles: Dado que los datos no se envían en la URL, es más seguro para enviar información sensible, como contraseñas.

Uso común: Se utiliza para iniciar sesión, registrarse, enviar datos de formularios y cualquier otra situación donde se requiera enviar información a un servidor.

CONCLUSIÓN

La utilización de las super globales `$_GET` y `$_POST` es una función muy útil para enviar, recibir y validar información enviada a través de un formulario para tu código en PHP. Se comprendió la diferencia entre estas 2 super globales, una de ellas es que `$_GET` envía los datos a través de la URL, mientras que `$_POST` además de admitir mayor cantidad de texto sin limitaciones, envía la información de forma más segura debido que no manda la información a través del URL.

