



APRENDERAPROGRAMAR.COM

FORMAS DE ENVÍO DE LOS
DATOS DE UN
FORMULARIO HTML.
MÉTODOS GET Y POST Y
SUS DIFERENCIAS.
EJEMPLOS. (CU00721B)

Sección: Cursos

Categoría: Tutorial básico del programador web: HTML desde cero

Fecha revisión: 2029

Resumen: Entrega nº21 del Tutorial básico "HTML desde cero".

Autor: Enrique González Gutiérrez

FORMAS DE ENVÍO DE LOS DATOS DE UN FORMULARIO HTML. MÉTODOS GET Y POST. EJEMPLOS

Cuando un usuario rellena un formulario en una página web los datos hay que enviarlos de alguna manera. Vamos a considerar las dos formas de envío de datos posibles: usando el método POST o usando el método GET.

Por ejemplo: `<form action="http://www.aprenderaprogramar.com/prog/newuser" method="get">`



En el ejemplo anterior la acción que se ejecutará cuando el usuario pulse el botón “Enviar” (submit) será el envío de los datos a la url especificada usando el método get.

Veamos el aspecto de un formulario cualquiera para hacernos una idea general.

Formulario de comunicación

Puedes contactar con nosotros a través de este formulario o usando el correo electrónico contacto@aprenderaprogramar.com

Obligatorio*

Nombre: * ⓘ

Apellidos: ⓘ

Correo electrónico: ⓘ

País: * ⓘ

Mensaje: * ⓘ

Este formulario consta de varios campos que al usuario se le solicitan como Nombre, Apellidos, Correo electrónico, País y Mensaje. Posiblemente los nombres de los campos en el código HTML sean del tipo nombre_user, apellidos_user, email_user, pais_user y msg.

La diferencia entre los métodos get y post radica en la forma de enviar los datos a la página cuando se pulsa el botón “Enviar”. Mientras que el método GET envía los datos usando la URL, el método POST los envía de forma que no podemos verlos (en un segundo plano u “ocultos” al usuario).

Un resultado usando el método GET, a modo de ejemplo, podría ser el siguiente:

<http://www.aprenderaprogramar.com/newuser.php?nombre=Pepe&apellido=Flores&email=h52turam%40uco.es&sexo=Mujer>

En esta URL podemos distinguir varias partes:

<http://www.aprenderaprogramar.com/newuser.php> es la dirección web en sí.

El símbolo ? indica dónde empiezan los parámetros que se reciben desde el formulario que ha enviado los datos a la página.

Después del símbolo ? aparecen parejas de datos con su nombre y valor separadas por el símbolo &. Las parejas dato1=valor1, dato2=valor2, dato3=valor3... reflejan el nombre y el valor de los campos enviados por el formulario.

Por ejemplo: nombre=Pepe, apellidos=Flores, etc. nos dice que el campo del formulario que se denomina nombre llega con valor “Pepe” mientras que el campo del formulario que se denomina apellidos llega con valor “Flores”. Estos valores son recibidos en la página web de destino del formulario.

Tener en cuenta que para separar la primera pareja de la dirección web en sí se usa el símbolo ‘?’ y para separar las restantes parejas entre sí se usa el símbolo ‘&’.

Otro aspecto a tener en cuenta es que determinados caracteres no son recibidos en la URL de la misma forma exactamente en que fueron escritos en el formulario. Por ejemplo, el valor del campo email que se recibe en la URL es h52turam%40uco.es, mientras que el usuario en el formulario habrá introducido con toda seguridad h52turam@uco.es. Como vemos, el carácter @ ha sido sustituido por los caracteres %40. Estas equivalencias se introducen automáticamente en la transmisión de datos debido a que las URLs no admiten determinados caracteres como letras con tildes, arrobas y otros. No debes preocuparte por esta codificación, ya que si posteriormente rescatamos los valores mediante otros mecanismos volveremos a obtener el texto original. Simplemente, conviene conocer esta circunstancia para no pensar que están ocurriendo cosas extrañas o errores.

Hemos visto el resultado de un envío por el método GET. En el caso de un envío de datos usando el método POST, aunque estos datos también serán enviados (de una forma que podemos denominar “oculta”), no los podremos ver en la URL. Para poder recuperar los valores de los campos en el caso de un envío con el método POST necesitaríamos otras herramientas (por ejemplo valernos del lenguaje PHP para recuperar el valor de esos campos).

El resultado final con ambos métodos podemos decir que es el mismo: la información se transmite de un lado a otro. La diferencia radica en que con el método GET podemos ver directamente los parámetros pasados ya que están dentro de la URL mientras que con el método POST los parámetros quedan ocultos y para rescatarlos hay que usar otras herramientas.

Un ejemplo de uso del método post sería este:

```
<form action="http://www.aprenderaprogramar.com/prog/newuser" method="post">
```

¿ES MEJOR USAR EL MÉTODO GET O EL MÉTODO POST?

Tanto GET como POST son métodos de envío de la información de los formularios válidos y ampliamente utilizados. Cada método tiene sus ventajas y sus inconvenientes y no se puede decir que uno sea mejor que otro. Elegir entre un método y otro depende de la aplicación concreta que se esté desarrollando y es algo que dentro de las empresas de desarrollos web suelen decidir los encargados del diseño de las aplicaciones. A nosotros en este curso básico simplemente nos interesa conocer la existencia de ambos métodos y sus características.

Para terminar, en la siguiente tabla mostramos un resumen de las diferencias entre GET y POST:

MÉTODO	CONCEPTO	OBSERVACIONES
GET	GET lleva los datos de forma "visible" al cliente (navegador web). El medio de envío es la URL. Los datos los puede ver cualquiera.	Los datos son visibles por la URL, por ejemplo: www.aprenderaprogramar.com/action.php?nombre=pedro&apellidos1=gomez
POST	POST consiste en datos "ocultos" (porque el cliente no los ve) enviados por un formulario cuyo método de envío es post. Es adecuado para formularios. Los datos no son visibles.	La ventaja de usar POST es que estos datos no son visibles al usuario de la web. En el caso de usar get, el propio usuario podría modificar la URL escribiendo diferentes parámetros a los reales en su navegador, dando lugar a que la información tratada no sea la prevista.

Próxima entrega: CU00722B

Acceso al curso completo en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente: http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=69&Itemid=192