

PROGRAMA DE ESTUDIOS

DESARROLLO DE SISTEMAS DE INFORMACIÓN FRONTEND ORIENTADO A SERVICIOS

Tema

FORMULARIO EN HTML PARTE I

FORMULARIO EN HTML PARTE I

Los formularios en HTML permiten a los usuarios ingresar y enviar datos al servidor, que luego se procesan para diversos fines, como registrar información, realizar búsquedas, enviar mensajes, entre otros. A continuación, analizaremos los atributos esenciales que controlan el comportamiento de un formulario.

1. Atributo name

- **Concepto:**

El atributo name proporciona un identificador único para los campos dentro del formulario. Este nombre se usa en el envío de datos al servidor para asociar un valor específico a cada campo.

- **Características:**

- Ayuda a identificar los datos enviados.
- Cada campo de un formulario (como un campo de texto, checkbox, etc.) debe tener un name único para facilitar el procesamiento de datos.

Ejemplo:

```
html
```

```
<form>
    <label for="nombre">Nombre:</label>
    <input type="text" id="nombre" name="nombre">
</form>
```

En este caso, el campo de entrada de texto tiene el atributo name="nombre", que se enviará con el valor ingresado en el formulario.

2. Atributo action

- **Concepto:**

El atributo action define la URL del servidor o el recurso al que se enviarán los datos del formulario cuando se haga clic en el botón de envío.

- **Características:**

- Permite especificar un archivo o script (PHP, JavaScript, etc.) que procesará los datos del formulario.
- Si el action está vacío (action=""), los datos se enviarán a la misma página que contiene el formulario.

Ejemplo:

```
<form action="/procesar_datos.php">
    <label for="email">Correo Electrónico:</label>
    <input type="email" id="email" name="email">
</form>
```

Aquí, el formulario enviará los datos al archivo procesar_datos.php.

3. Atributo method

- **Concepto:**

El atributo method define cómo se envían los datos del formulario. Los valores más comunes son post y get.

- **Características:**

- **method="post":**
 - Envía los datos en el cuerpo de la solicitud HTTP.
 - Es más seguro para enviar información sensible, ya que los datos no aparecen en la URL.
 - Comúnmente usado para formularios de registro, inicio de sesión, y donde se necesita manipular datos en el servidor.
- **method="get":**
 - Envía los datos a través de la URL, lo que permite verlos directamente en la barra de direcciones.
 - Es más adecuado para formularios de búsqueda o donde no se necesita privacidad de datos.
 - Los datos enviados tienen un límite de caracteres, por lo que no es adecuado para enviar grandes volúmenes de información.

Ejemplo con method="post":

```
html
```

```
<form action="/registrar_usuario.php" method="post">
    <label for="usuario">Usuario:</label>
    <input type="text" id="usuario" name="usuario">
</form>
```

Aquí, los datos se enviarán usando el método POST, ideal para formularios de registro.

Ejemplo con method="get":

html

```
<form action="/buscar.php" method="get">
    <label for="query">Buscar:</label>
    <input type="text" id="query" name="query">
</form>
```

En este caso, el formulario utiliza el método GET, adecuado para enviar datos de búsqueda.

4. Atributo target

- **Concepto:**

El atributo target especifica dónde se mostrará la respuesta después de enviar el formulario. Puede abrir los resultados en la misma pestaña, en una nueva ventana, en un marco específico o en una ventana emergente.

- **Características:**

- **Valores comunes de target:**

- _self: Abre el resultado en la misma ventana o pestaña (valor predeterminado).
 - _blank: Abre el resultado en una nueva pestaña o ventana.
 - _parent: Abre el resultado en el marco o ventana superior.

- `_top`: Abre el resultado en la ventana completa, eliminando cualquier marco.

Ejemplo:

html

```
<form action="/registro_completo.php" target="_blank">
    <label for="username">Usuario:</label>
    <input type="text" id="username" name="username">
</form>
```

En este ejemplo, el formulario abrirá la respuesta en una nueva ventana o pestaña gracias a `target="_blank"`.

Ejercicios Prácticos

Ejercicio 1: Formulario de Registro de Usuario

- **Objetivo:** Crear un formulario de registro que envíe los datos a un archivo llamado `registro.php` usando el método `post`.
- **Instrucciones:**

1. Crea un formulario con campos para nombre, correo electrónico y contraseña.
2. Usa `method="post"` para enviar la información de manera segura.
3. Incluye un botón de envío para enviar los datos.

Ejercicio 2: Formulario de Búsqueda en el Sitio Web

- **Objetivo:** Crear un formulario de búsqueda que envíe datos a busqueda.php usando el método get.

- **Instrucciones:**

1. Crea un formulario con un campo para ingresar un término de búsqueda.
2. Usa method="get" para que los datos aparezcan en la URL.
3. Configura el atributo target para que el resultado se abra en una nueva pestaña.



