

FORMULARIOS HTML

LIMA UD1 – Tema 2

IES Plurilingüe Antón Losada Diéguez



Tabla de contenido

1. Introducción.....	2
2. Formulario <form>	2
2.1 Atributos	2
2.1.1 Action.....	2
2.1.2 Method	2
3. Elementos del formulario.....	3
3.1 Input <input>	3
3.1.1 Text	3
3.1.2 Password.....	4
3.1.3 Radio	4
3.1.4 Checkbox.....	4
3.1.5 Submit.....	5
3.1.6 Button	5
3.1.7 Number	5
3.1.8 Time	5
3.1.9 Date	6
3.1.10 Email	6
3.1.11 Color.....	7
3.1.12 File	7
3.1.13 Range	7
3.1.14 Tel	8
3.1.15 Url	8
3.1.16 Hidden.....	8
3.2 Etiqueta <label>	8
3.3 Textarea	8
3.4 Button	9
3.5 Select	9
3.6 Datalist.....	10
3.7 Agrupación de elementos <fieldset> y <legend>.....	10

Formularios HTML

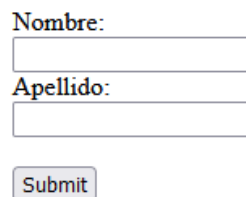
1. Introducción

Las páginas no son todas estáticas que simplemente muestren información, la mayoría tienen algún tipo de interacción con el usuario y para ello requieren en mayor o menor medida del uso de formularios para la recogida de los datos y botones para interaccionar con ellos.

2. Formulario <form>

El elemento form es el contenedor del formulario, donde se encontrarán todos y cada uno de los campos que tendrá que rellenar el usuario, los botones con los que interaccionar y el botón de envío.

```
<form action="/action_page.php" method="post">
  <label for="fname">Nombre:</label><br>
  <input type="text" id="fname" name="fname"><br>
  <label for="lname">Apellido:</label><br>
  <input type="text" id="lname" name="lname"><br><br>
  <input type="submit" value="Submit">
</form>
```



2.1 Atributos

Los formularios tienen dos atributos fundamentales, action y method.

```
<form action="/action_page.php" method="get">
```

2.1.1 Action

Este atributo estipula la página que gestionará el formulario, donde se enviará la información recogida. Si no se especifica, se envía a la página actual.

Esto lo realiza el servidor, por lo que no concierne al html y no se verá en este módulo.

2.1.2 Method

Este atributo define la forma en la que se envían los datos. Hay dos formas, dos valores posibles, get y post.

El primero, GET, envía los datos mediante la URL, visibles para el usuario y con un límite y una serie de restricciones. Si no se estipula nada, este es el método por defecto.

El segundo método, POST, envía los datos de forma oculta y sin límite.

Esto lo realiza el servidor, por lo que no concierne al html y no se verá en este módulo.

3. Elementos del formulario

Para que un formulario haga algo, es necesario añadir campos al mismo, que recojan la información necesaria.

Existen múltiples campos, desde elementos de inserción, hasta de selección, todos con sus cualidades y usos.

Prácticamente todos los elementos de un formulario tienen estos atributos:

id: El identificador es un atributo global (cualquier elemento HTML puede tenerlo) tiene que ser único, ningún otro elemento de la página puede tener un identificador igual. Se utiliza para referenciarlo con CSS y para etiquetarlo.

name: Es el nombre del elemento, también único y se utiliza para referenciarlo desde el servidor. Es habitual hacerlo coincidir con el de id.

value: El valor que tiene el campo, lo que tiene introducido. Si se estipula uno, será el valor por defecto.

placeholder: Es un texto temporal, habitualmente descriptivo o ilustrativo de lo que el usuario debería tener. Aparece en un tono claro cuando el campo está completamente vacío.

required: Es un atributo modificador, sin valor, que le indica al navegador que no envíe el formulario a no ser que el campo esté relleno. No funciona en todos los navegadores y siempre se verifica en lado del servidor, por lo que su uso no es habitual.

disabled: Es un atributo modificador, sin valor, que bloquea el campo, impidiendo que el usuario interactúe con él.

3.1 Input <input>

El elemento input representa los campos de inserción y es el más común y el más versátil. Incluye desde campos de texto hasta selectores de color o fechas.

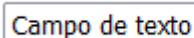
```
<input type="text" id="fname" name="fname">
```

La estructura es siempre la misma, el elemento con el tipo, su id y su nombre. El primer atributo representa el tipo de campo, cambiando completamente su funcionamiento y aspecto.

3.1.1 Text

El tipo text es de los más comunes y también el input por defecto. Se utiliza para recoger texto corto de cualquier tipo, desde un nombre hasta una frase.

```
<input type="text" id="nombre" name="nombre">
```



Campo de texto

3.1.2 Password

El tipo password se utiliza para introducir las contraseñas. A diferencia del de texto, su contenido se oculta automáticamente con puntos o asteriscos.

Los navegadores, al detectarlo, generan el mensaje de si quieres recordar la contraseña.

```
<input type="password" id="pwd" name="pwd">
```

3.1.3 Radio

Este tipo de input es especial, con una apariencia distinta y se usa para que el usuario seleccione una y solo una de entre varias opciones.

```
<input type="radio" id="html" name="fav_language" value="HTML">  
HTML<br>  
<input type="radio" id="css" name="fav_language" value="CSS" checked>  
CSS<br>  
<input type="radio" id="javascript" name="fav_language" value="JavaScript">  
JavaScript<br><br>
```

- ☐ HTML
- ☒ CSS
- ☐ JavaScript

A diferencia de los otros input, el atributo **name** tiene que coincidir entre todos los input que compongan el grupo.

Este tipo de input tiene el atributo modificador **checked** que estipula el elemento seleccionado por defecto. Solo uno puede estar seleccionado.

3.1.4 Checkbox

Este tipo es similar al radio, con la diferencia de que el usuario puede seleccionar de cero a todas las opciones.

```
<input type="checkbox" id="vehicle1" name="vehicle1" value="Bike" checked>  
I have a bike<br>  
<input type="checkbox" id="vehicle2" name="vehicle2" value="Car">  
I have a car<br>  
<input type="checkbox" id="vehicle3" name="vehicle3" value="Boat" checked>  
I have a boat<br><br>
```

- ☒ I have a bike
- ☐ I have a car
- ☒ I have a boat

A diferencia del radio, dado que cada uno puede estar seleccionado o no, los atributos **name** no pueden coincidir.

Este tipo de input tiene el atributo modificador **checked** que estipula que ese checkbox está seleccionado por defecto.

3.1.5 Submit

Este tipo representa el botón de envío del formulario. Cuando el usuario hace clic sobre él, se envían los datos.

```
<input type="submit" value="Enviar">
```



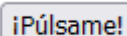
A diferencia de los demás, el atributo **value** estipula el texto del botón.

3.1.6 Button

Este es un tipo de elemento que crea un botón sin funcionalidad, es decir, que no hace nada de por sí.

Se utiliza junto con JavaScript para darle una utilidad propia.

```
<input type="button" onclick="alert('¡Hola!')" value="¡Púlsame!">
```



3.1.7 Number

Este tipo de input permite al usuario añadir un número dentro de un rango. La interacción puede ser insertando el número, con la rueda del ratón o con las flechas del campo.

```
<input type="number" id="quantity" name="quantity" min="0" max="100" step="10" value="30">
```



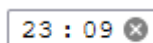
Los atributos **min** y **max** estipulan el valor mínimo y el máximo permitidos.

El atributo **step** estipula el paso, lo que aumenta o disminuye cada vez.

3.1.8 Time

Este tipo facilita el introducir una hora en formato 24h, ya que separa el campo en horas y en minutos verificando los valores. A mayores, incluye un icono para borrarlo en Firefox, mientras que en el resto de navegadores incluye un botón que abre un desplegable.

```
<input type="time" id="appt" name="appt">
```



19	38
20	39
21	40
22	41
23	42
00	43
01	44

3.1.9 Date

Este tipo de input permite introducir una fecha en modo dd/mm/aaaa con la peculiaridad de que si el usuario hace clic en el campo, aparece un calendario en el que podrá escogerla de una forma más cómoda.

Para establecer un valor por defecto, al igual que un mínimo o máximo, tendremos que estipular la fecha completa en formato internacional, es decir, aaaa-mm-dd.

```
<input type="date" id="birthday" name="birthday" value="2018-07-22" min="2018-01-01" max="2018-12-31">
```

3.1.10 Email

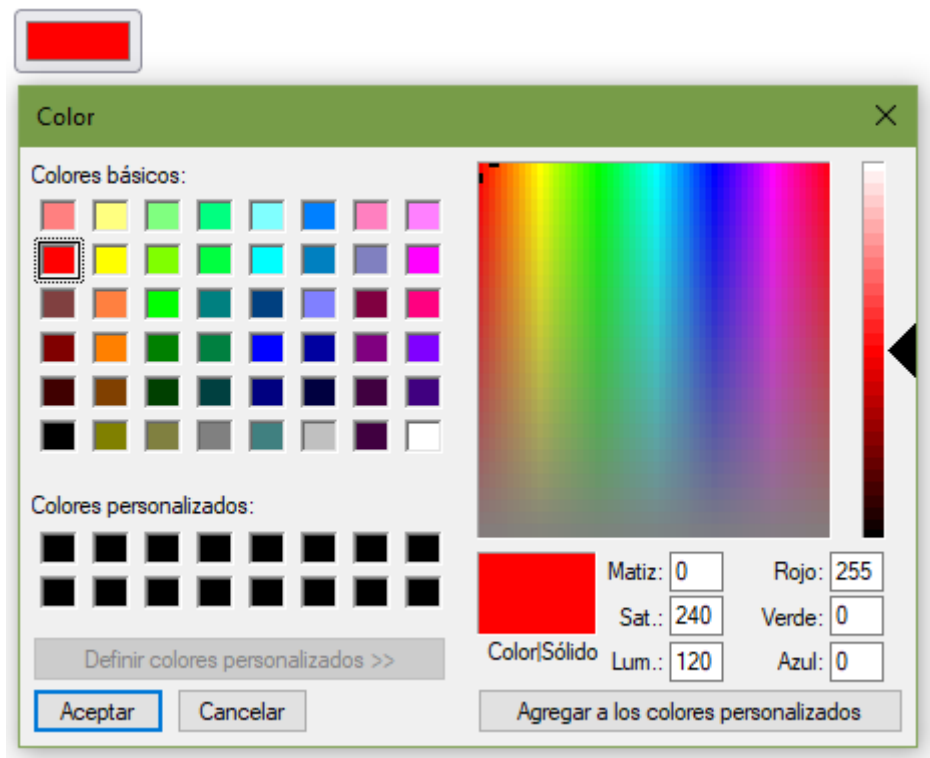
Este tipo está diseñado para introducir direcciones de correo. Aunque su apariencia es la de un campo de texto normal, el navegador no enviará el formulario a no ser que detecte que se ha introducido una dirección válida.

```
<input type="email" id="email" name="email">
```

3.1.11 Color

El input color genera un botón que permite al usuario seleccionar un color. Al hacer clic sobre él se abre un selector de colores que varía dependiendo del navegador.

```
<input type="color" id="favcolor" name="favcolor" value="#ff0000">
```



El atributo **value** estipula el color que sale por defecto en el recuadro, que varía dinámicamente cuando el usuario selecciona uno.

3.1.12 File

Este input permite al usuario añadir un fichero de su ordenador.

```
<input type="file" id="myfile" name="myfile">
```

Examinar... No se ha seleccionado ningún archivo.

Con el atributo modificador **multiple** permite añadir más de un archivo.

3.1.13 Range

El tipo range crea un selector de rango de números.

```
<input type="range" id="vol" name="vol" min="0" max="50">
```



Los atributos **min** y **max** estipulan el valor mínimo y máximo.

3.1.14 Tel

Este tipo añade un elemento similar a uno de texto, pero diseñado para números de teléfono.

```
<input type="tel" id="phone" name="phone">
```

3.1.15 Url

Este tipo está diseñado para introducir URLs. Aunque su apariencia es la de un campo de texto normal, el navegador no enviará el formulario a no ser que detecte que se ha introducido una URL válida.

```
<input type="url" id="homepage" name="homepage">
```

3.1.16 Hidden

Este tipo es especial, no se muestra al usuario y este no puede interactuar con él. Se usa para pasar información al servidor, muchas veces generada por el mismo servidor.

```
<input type="hidden" id="custId" name="custId" value="3487">
```

3.2 Etiqueta <label>

La etiqueta label permite establecer un nombre a un campo, de tal forma que aparece junto a este y, al hacer clic sobre la etiqueta, se activa el campo.

Todo campo de un formulario puede tener una etiqueta asociada.

```
<label for="fname">Nombre: </label>  
<input type="text" id="fname" name="fname"><br>
```

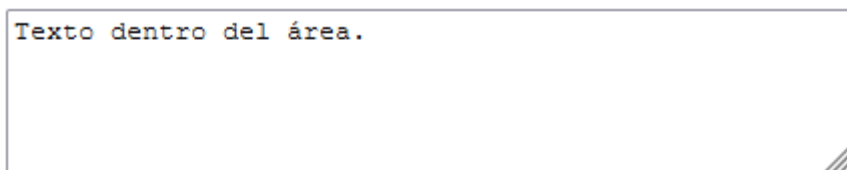
Nombre:

Para asociar la etiqueta al campo, se usa el atributo **for**, cuyo valor es el **id** del elemento a asociar.

3.3 Textarea

Este elemento es similar a un input, ya que se utiliza para obtener textos largos, de varias líneas, por parte del usuario.

```
<textarea id="w3review" name="w3review" rows="4" cols="50">Texto dentro del área.</textarea>
```



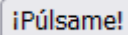
A diferencia de los input, su valor se introduce entre las etiquetas de apertura y cierre.

Los atributos **rows** u **cols** permiten establecer el tamaño inicial del área, pudiendo ser modificada por el propio usuario pinchando y arrastrando la esquina inferior derecha del recuadro.

3.4 Button

El elemento button es idéntico al input de tipo button, con la diferencia de que el valor va entre las etiquetas y no en el atributo **value**. Además, este no requiere estar dentro de un form para funcionar.

```
<button type="button" onclick="alert('¡Hola!')">¡Púlsame!</button>
```

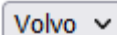


El uso del input button o el elemento button va a consideración del programador o de las necesidades de CSS. La principal ventaja de este frente al input es que no requiere de un elemento form, ya que funciona de forma independiente.

3.5 Select

Este elemento crea un desplegable para seleccionar una opción.

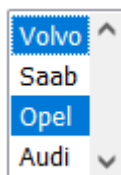
```
<select name="cars" id="cars">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi">Audi</option>
</select>
```



El elemento tiene dos niveles, la etiqueta select especifica el **name** y el **id**, mientras que las etiquetas option especifican el valor y el texto de cada opción.

La etiqueta select dispone del atributo modificador **multiple** que permite al usuario seleccionar varias opciones a la vez.

```
<select name="cars" id="cars" multiple>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="opel">Opel</option>
  <option value="audi">Audi</option>
</select>
```



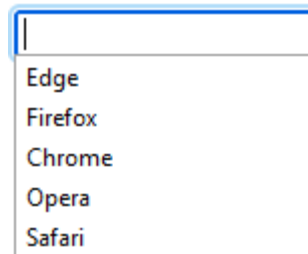
A mayores, en cualquiera de las dos opciones, la etiqueta option tiene el atributo **selected** para estipular el valor o valores seleccionados por defecto.

3.6 Datalist

Este elemento permite crear un input en el que da una serie de opciones a introducir, incluyendo el autocompletado o la búsqueda en la lista a medida que el usuario escribe.

El campo no está limitado a esas opciones, el usuario puede introducir cualquiera, simplemente sirve a modo de guía.

```
<input list="browsers" name="browser" id="browser">
<datalist id="browsers">
  <option value="Edge">
  <option value="Firefox">
  <option value="Chrome">
  <option value="Opera">
  <option value="Safari">
</datalist>
```



Para su creación requiere de un elemento input sin tipo, pero con el atributo list que relaciona ese elemento con el id del datalist.

3.7 Agrupación de elementos <fieldset> y <legend>

Para organizar mejor el formulario, existe un elemento de agrupación, el fieldset que rodea un conjunto de elementos con una caja.

La etiqueta legend, dentro del fieldset permite estipular un nombre al bloque.

```
<fieldset>
  <legend>Datos personales:</legend>
  <label for="fname">Nombre:</label>
  <input type="text" id="fname" name="fname"><br><br>
  <label for="lname">Apellido:</label>
  <input type="text" id="lname" name="lname"><br><br>
  <label for="email">Email:</label>
  <input type="email" id="email" name="email"><br><br>
  <label for="birthday">Decha de nacimiento:</label>
  <input type="date" id="birthday" name="birthday"><br><br>
  <input type="submit" value="Submit">
</fieldset>
```

Datos personales:

Nombre:

Apellido:

Email:

Decha de nacimiento:

Combinando múltiples fieldset, podemos separar un mismo formulario en varios bloques, como si fuesen secciones separadas.