

Formularios

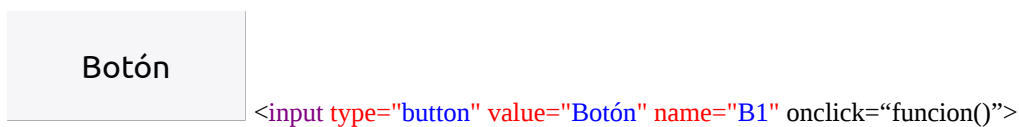
Los formularios son los elementos que permiten la entrada de datos. Llevan cuadros de texto, botones, etc. Podemos acceder a estos elementos como a cualquier otro de la página, por sus atributos, su identificador, su clase, etc. Pero además podemos usarlos para interactuar con el usuario: pedirle datos o selecciones y actuar en consecuencia.

La definición de elementos de formulario en una página se ha visto en la parte de HTML. Dirígete allí si no lo recuerdas, aquí vamos a ceñirnos a su uso con Javascript normal o con jquery

Botón

Existen tres clases de botones: los de envío y reset, que son específicos de formularios y los generales, que pueden aparecer en cualquier parte.

El botón genérico no tiene una función definida, es el utilizado para activar funciones de Javascript.



Estos botones no tienen por qué ir en formularios, pueden ponerse en cualquier punto de la página, con la etiqueta BUTTON

Ejemplo:

```
<button id='boton2' onclick='adelantar()'><b>Adelante</b></button>.
```

Cuadros de texto

Es un control de tipo TEXT (permite al operador cargar caracteres por teclado).

Un uso avanzado es el de mostrar información variable o recibir datos para cálculos, todo esto con Javascript.

Con un ejemplo veremos estos controles: Formulario que permita meter el nombre y edad de una persona:

```
<!doctype html>
<html>
<head><SCRIPT src="jquery.js"></script>
</head>
<body>

<script language="JavaScript">
function mostrar()
{
  var nom=$("#nom").val();
  var ed=$("#edad").val();
  alert('Nombre:' + nom);
  alert('Y edad:' + ed);
}
</script>

<form >
  Nombre:
  <input type="text" id="nom"><br>
  Edad:
  <input type="text" id="edad"><br>
  <input type="button" value="Confirmar" onClick="mostrar()">
</form>
```

```
</body>
</html>
```

En este problema tenemos un botón y dos cuadros de texto. La función que se activa al presionar el botón se llama mostrar.

La función 'mostrar' accede a los contenidos de los dos controles de tipo TEXT:

```
var nom=$("#nom").val();
var ed=$("#edad").val();
```

Para acceder a los textos cargados debemos indicar val(). Para hacer más clara la función, guardamos en dos variables auxiliares los contenidos de los controles de tipo TEXT.

Veamos otro ejemplo: una función que toma los valores de un formulario, y devuelve en un cuadro de texto el valor de la primera casilla elevado a la potencia de la segunda:

```
<script>
function potencia() {
var i=0;
var resul=1;
var exponente = $("#elevado").val()
var num = $("#base").val()
resul=Math.pow(num,exponente);
$("#resultado").val(resul);
}
</script>
```

Veamos la parte HTML de este documento:

```
<form>
Número: <input type="text" id="base">
Potencia: <input type="text" id="elevado">
Resultado: <input type="text" id="resultado">
<input type="button" name="poten" Value="Calcular potencia"
onClick="potencia()">
</form>
```

Nota: Esto sirve también para elementos `textarea`, `password`, `hidden`

Listas desplegadas

Uno de los elementos más utilizados de un formulario es el menú de opciones: una lista desplegable.

Cuando tenemos un menú de opciones y seleccionamos una opción, es posible sacar por medio de Javascript cuál fue la opción seleccionada.

Ejemplo para referirnos al atributo `value` de la opción seleccionada en un `select` cuyo `id` sea `menu`:

```
$("#menu").val()
```

Si observas, indicamos primero el nombre del `select`, y con `val()` sacamos el `value`

Es muy común poner juntos un menú de opciones y un botón y al seleccionar algún elemento y pulsar en el botón nuestro navegador hará lo seleccionado; para lograr esto tenemos que obtener el valor del elemento seleccionado del menú. Veamos un ejemplo para construir nuestro propio menú de direcciones:

Inicio	Click para ir
--------	---------------

El código para el menú y el botón es:

```
<form>
<select id="menu">
<option value="http://web.unican.es">
Inicio
</option>
<option value="https://www.qwant.com">
Buscador
</option>
</select>
<input type="button" value="Pulsa para ir"
onclick="vamos( )" ></p>
</form>
```

Observa que la función **vamos()** se activará con el evento onclick sobre el botón. El código para la función es:

```
<script language="javascript"><!--
function vamos( ){
window.location = $("#menu").val()
}
-->
</script>
```

De la forma antedicha obtenemos el valor (value) de la opción. Cuando escribas el código para el menú, recuerda que el valor (value) de cada opción debe ser la dirección de la página a la cuál será llevado el usuario. y entre `<option>` `</option>` va la etiqueta (o sea el texto que se mostrará en el menú).

window.location se utiliza para decir cual será la **ubicación (location)** que tendrá la **ventana (window)**, es decir, la nueva dirección.

También podemos hacer que se ejecute una función Javascript cada vez que el usuario hace una selección de la lista. Esto lo hacemos con OnChange.

Para sacar el texto (no el value) de la etiqueta seleccionada en un SELECT que se llame "select1" sería por ejemplo:

```
$("#select1 option:selected").text()
```

Otro ejemplo completo que muestra el empleo de un control SELECT es:

```
<!doctype html>
<html>
<head>
<SCRIPT src="jquery.js"></script>
</head>
<body>
<script language="JavaScript">
function cambiarColor()
{
textoselec=$("#select1 option:selected").text();
valueselec=$("#select1").val();
$("#text2").val(textoselec);
$("#text3").val(valueselec);
}
</script>
<form name="form1">
<select size="1" id="select1" ONCHANGE="cambiarColor()">
<option value="0xff0000">Rojo</option>
<option value="0x00ff00">Verde</option>
<option value="0x0000ff">Azul</option>
</select></p>
<br>
```

```
Texto seleccionado:<input type="text" id="text2"><br>
Valor asociado:<input type="text" id="text3"><br>
</form>
</body>
</html>
```

Fíjate primero en el objeto SELECT (es HTML), y cómo acceder luego a sus valores desde Javascript.

La forma de indicar la función a llamar cuando ocurra un cambio es: onChange="cambiarColor()".

Por cada opción del objeto SELECT tenemos una línea OPTION, por ejemplo:

```
<option value="0xff0000">Rojo</option>
```

Donde Rojo es el texto que se visualiza en el navegador para esa opción y `value` es el valor asociado a dicha opción.

Analizando la función `cambiarColor()` podemos ver cómo obtenemos los resultados seleccionados del objeto SELECT.

Cuadros de marca

El cuadro de marca o “checkbox” es la pequeña casilla que se muestra en algunos formularios.

El atributo `checked` indica si esta casilla está o no activada. La forma de activarla o desactivarla con Javascript es:

```
$("#C1")[0].checked = true // Para Activar
$("#C1")[0].checked = false // Para desactivar
```

Ejemplo: activar y desactivarlo con botones. Éste es el código:

```
<form>
<p><input type="button" value="Pulsame para activar checkbox" name="B1"
onclick="$('#C1')[0].checked=true"> <br>
<input type="checkbox" id="C1" value="ON"><br>
<input type="button" value="Pulsame para desactivar" name="B2" onclick="$('#C1')
[0].checked=false"> </p>
</form>
```

Otro ejemplo: Una página que muestre 4 lenguajes de programación que el usuario puede seleccionar si los conoce. Luego mostrar un mensaje indicando la cantidad de lenguajes que ha seleccionado.

```
<!doctype html>
<html>
<head>
<SCRIPT src="jquery.js"></script>
</head>
<body>
<script language="JavaScript">
function contarSeleccionados()
{
var cant=0;
if ($("#lenguaje1")[0].checked)
cant++;
if ($("#lenguaje2")[0].checked)
cant++;
if ($("#lenguaje3")[0].checked)
cant++;
if ($("#lenguaje4")[0].checked)
cant++;
alert('Conoce ' + cant + ' lenguajes');
}
```

```

</script>
<form>
<input type="checkbox" id="lenguaje1">JavaScript
<br>
<input type="checkbox" id="lenguaje2">PHP
<br>
<input type="checkbox" id="lenguaje3">JSP
<br>
<input type="checkbox" id="lenguaje4">VB.Net
<br>
<input type="button" value="Mostrar" onClick="contarSeleccionados()">
</form>
</body>
</html>

```

Cuando se presiona el botón se llama a la función Javascript `contarSeleccionados()`. En ella verificamos uno a uno cada control `checkbox` accediendo a la propiedad `checked` que almacena true o false según esté o no seleccionado el control.

Ponemos un `if` para cada `checkbox`:

```

if ($("#lenguaje1")[0].checked)
    cant++;

```

Al contador `cant` lo definimos e inicializamos en cero antes de los `if`. Mostramos finalmente el resultado final.

Selección uno de grupo (botones “radio”)

Los botones de radio sirven para elegir una de varias opciones. Ejemplo: Mostrar cuatro objetos de tipo RADIO que permitan seleccionar los estudios que tiene un usuario:

```

<!doctype html>
<html>
<head><TITLE>Prueba</TITLE>
<SCRIPT src="jquery.js">
</script>
</head>
<body>
<script language="JavaScript">
function mostrarSeleccionado()
{
    if ($("#estud0")[0].checked)
        alert('no tienes estudios');
    if ($("#estud1")[0].checked)
        alert('tienes estudios primarios');
    if ($("#estud2")[0].checked)
        alert('tienes estudios secundarios');
    if ($("#estud3")[0].checked)
        alert('tienes estudios universitarios');
}
</script>
<form name="form1">
<input type="radio" name="estudios" id="estud0">Sin
estudios
<br>
<input type="radio" name="estudios" id="estud1">Primarios
<br>
<input type="radio" name="estudios" id="estud2">Secundarios
<br>
<input type="radio" name="estudios" id="estud3">Universitarios

```

```
<br>
<input type="button" value="Mostrar" onClick="mostrarSeleccionado()">
</form>
</body>
</html>
```

Ya sabes que todos los objetos de tipo RADIO que van enlazados tienen el mismo `name`.

Igual que el `checkbox`, la propiedad `checked` da verdadero o falso, según esté o no seleccionado el control radio.

Si hemos puesto atributos `value` que nos interesen, para obtener el `value` del seleccionado sería:

```
$("#input[name=nombre]:checked").val()
```

Para marcar uno, por ejemplo el 0 (que es el primero) sería:

```
$('#input[name=nombre]')[0].checked = true;
```