

## 02-09c formularios HTML5 . FieldSet. Datalist, progress y Meter

Nombre		Curso	
Apellidos		Fecha	

### Sumario

02-09 formularios HTML5.....	1
→ Agrupación de controles, fieldset.....	2
→ Etiquetado de controles.....	3
→ Datalist.....	4
→ Controles avanzados.....	5
→ Cuadro de tipo keygen.....	5
→ Control de tipo output.....	6
→ Barras de resultados.....	7
barras de medida, elemento meter.....	7
barra de progreso, elemento progress.....	8

## → Agrupación de controles, fieldset

La etiqueta **fieldset** permite agrupar controles para que visualmente sea más cómodo el relleno de los controles. Visualmente los controles aparecerán recuadrados y se suele utilizar una etiqueta inmediatamente interior a **fieldset** que es **legend** que contiene el texto que encabezará al grupo de controles.

Ejemplo:

```
<form action="control_form.php" method="get">
<fieldset >
  <legend>Datos personales</legend>

  <input type="radio" name="sexo" value="hombre" id="hombre">
  <label for="hombre"> Hombre</label><br>
  <input type="radio" name="sexo" value="mujer" id="mujer" checked="checked">
  <label for="mujer"> Mujer</label><br>

  <label for="estadocivil"> Estado civil: Casado</label>
  <input type="checkbox" name="estadocivil"
    id="estadocivil" value="casado"
    checked="checked">

  <br>
</fieldset>
<input type="submit" value="Enviar" >
</form>
```

El resultado:



Ilustración 105. Controles agrupados

## → Etiquetado de controles

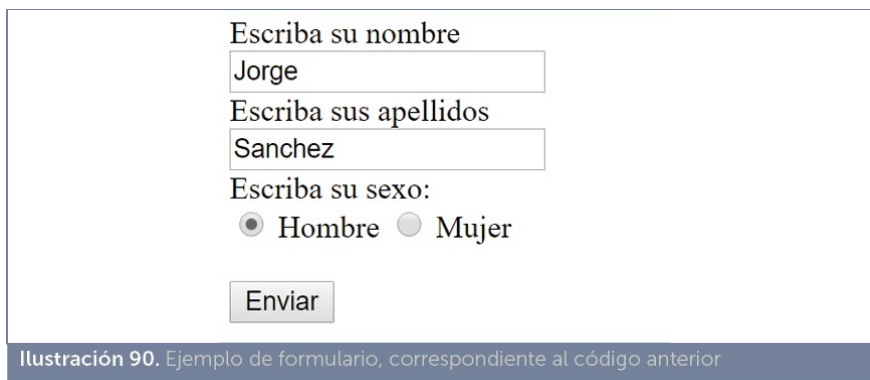
Su labor es colocar una etiqueta explicativa a cada control.

**label** permite una mejor usabilidad (los navegadores reconocen **label** y, por ejemplo, al hacer clic en el texto de la etiqueta, el control asociado recibe el foco) y una mejor semántica del código.

Los elementos **label** utilizan un atributo **for** que sirve para asociar la etiqueta (que es el texto que aparece en pantalla acompañando al control) al control concreto. El valor de **for** se corresponde con el valor indicado en el atributo **id** del control.

Ejemplo para etiquetar un control de tipo **textarea**:

```
<form action="control_form.php" method="get">  
  
  <label for="textoid">Escribe un texto descriptivo:</label>  
  
  <textarea rows="10" cols="40" name="texto" id="Textoid" ></textarea><br>  
  <input type="submit" value="enviar"><br>  
</form>
```



Formulario de ejemplo:

Escriba su nombre

Escriba sus apellidos

Escriba su sexo:  
☒ Hombre ☐ Mujer

Ilustración 90. Ejemplo de formulario, correspondiente al código anterior

## → Datalist ( Text + Combobox)

Se trata de un elemento HTML 5 muy potente. Permite añadir entradas a un control de cuadro de texto (y también a cuadros especializados como los de email, url,...). La forma de utilizarlo consiste en usar el atributo HTML5 **list** existente en la etiqueta **input**. Ese atributo asociará el cuadro de texto a la lista de valores a través del identificador del **datalist**. Luego dentro de **datalist** se colocan etiquetas **option** para cada opción en la lista (al estilo de los cuadros de tipo select).

Ejemplo:

```
<form action="recogida.php" method="get">
  <label for="gustos">
    Escribe en qué te gusta pasar tu tiempo de ocio
  </label>
  <input type="text" id="gustos" name="gustos" list="listaGustos" >

  <datalist id="listaGustos">

    <option label="deportes" value="Deportes" >
    <option label="teatro" value="Teatro">
    <option label="cine" value="Cine">
    <option label="leer" value="Leer">
  </datalist>

  <input type="submit" value="enviar"><br>
</form>
```

No aparece un cuadro combinado, será un cuadro de texto en el que al arrimar el ratón aparecen las entradas indicadas (como sugerencias de escritura), pero podremos escribir lo que queramos (sin elegir ninguna opción de la lista).

El código anterior permitiría esta situación:

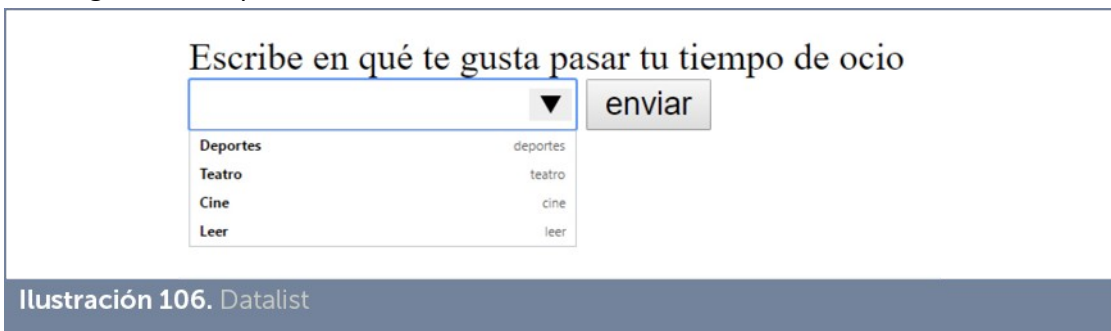


Ilustración 106. Datalist

El usuario puede elegir cualquiera de las opciones de la lista de sugerencias, pero también puede escribir lo que quiera en el cuadro.

Sin embargo esta forma no funciona en algunos navegadores (solo en los modernos y Safari no ha incorporado esta etiqueta), por ello se suele usar un truco que consiste en meter un elemento **select** dentro del elemento **datalist**. Los navegadores modernos ignorarán la etiqueta **select** y los viejos el **datalist**.

```
<form action="recogida.php" method="get">
  <label for="gustos">
    Escribe en qué te gusta pasar tu tiempo de ocio
  </label>
  <datalist id="listaGustos">

    <select name="listaGustos">

      <option value="deportes">Deportes</option>
      <option value="teatro">Teatro</option>
      <option value="cine">Cine</option>
      <option value="leer">Leer</option>

    </select>

  </datalist>

  <input id="gustos" name="gustos" list="listaGustos">
  <input type="submit" value="enviar"><br>
</form>
```

## → Cuadro de tipo keygen

Es una nueva etiqueta de formulario nacida en HTML 5 que permite enviar claves de cifrado de datos. El elemento genera una lista en la que podemos elegir el tipo de cifrado. Al enviar los datos se generan las dos claves: la pública y la privada, el ordenador local guarda la clave privada y el remoto la pública.

Es una opción para poder cifrar los datos del formulario a fin de autentificar la autenticidad del cliente. Ni Internet Explorer ni Safari son compatibles con este elemento, lo que limita su uso. Hoy en día es más fiable enviar los datos de los formularios usando el protocolo **https**.

La lista de atributos de keygen es:

atributo	uso
----------	-----

**autofocus** El control obtendrá el foco cuando el formulario se cargue

**challenge** Es una cadena de desafío que se presentará junto con la clave pública

**disable** Deshabilita el control

atributo    uso

**form**        Permite especificar el formulario al que se asociará este control (de uso obligado en caso de estar fuera de una etiqueta **form**)

**keytype**    Indica el tipo de cifrado. Puede ser: **rsa**, **dsa** o **ec**. Por defecto cifra mediante RSA.

**name**        Nombre del control

```
<html>
<form method="post" action="gestionarclave.php">
  <keygen name="RSA public key" challenge="123456789" KEYTYPE="RSA">
  <input type="submit" name="createcert" value="Pulsa aquí para generar tu clave
Publica">
</form>
</html>
```

## → Control de tipo output

La etiqueta output (no disponible en Internet Explorer ni en Safari) se utiliza para mostrar resultados procedentes de cálculos sobre otros controles del formulario.

El valor y el nombre de los elementos *output* no se envían como los demás datos del formulario, solo sirve para ver el resultado en la página. **No es obligatorio su uso, pero da un sentido claro a la función del contenido.**

El ejemplo siguiente consigue mostrar la suma de los dos números que el usuario indique en sendos cuadros numéricos, esta suma se muestra rápidamente gracias al lenguaje JavaScript:

```
<form oninput="res.value=parseInt(n1.value)+parseInt(n1.value)">

<input type="number" id="n1" name="n1" value="0">

+<input type="number" id="n2" name="n2" value="0">

=<output name="total" for="n1 n2" id="total"></output>

</form>
```

Resultado del código:

9 + 18 = 27

**Ilustración 107.** Resultado de la suma a través de un elemento output

El atributo **for**, especifica el nombre de los controles que afectan al resultado de **output**; en ese atributo cada identificador se separa con espacios.

## → Barras de resultados

### ***barras de medida, elemento meter***

Se trata de un tipo de barra que sirven para indicar de forma gráfica un determinado valor fracción o parte de algo. El elemento **meter**, que no funciona en Internet Explorer, es el que permite este uso.

Atributos:

atributo    significado

**max**        Máximo valor del rango, ese será el valor relacionado con el final de la barra

**min**        Mínimo valor del rango, es el valor relacionado con el valor de la barra.

**high**        Valor más alto a alcanzar (meta). Por debajo de este valor, el color de la barra se muestra de color amarillo (aún no hemos alcanzado el valor a alcanzar)

**low**        Valor más bajo en el rango de la barra de progreso. Por debajo de este valor, la barra se muestra roja (puesto que no hemos alcanzado el mínimo)

**value**        Valor actual de la barra. La barra se colorea desde el inicio hasta llegar a la posición que le correspondería a este valor en la barra.

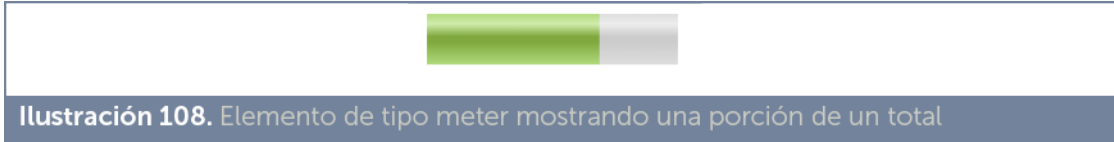
**optimum**    Valor marcado como óptimo en la barra de progreso. No provoca ningún efecto visual en la actualidad.

**form**        Identificador del formulario al que pertenece este control (salvo que el control esté dentro de una etiqueta **form**)

Ejemplo:

```
<meter value="138" min="1" max="200" high="150" low="60" >138</meter>
```

El resultado es:



Solo mediante JavaScript se obtiene realmente partido de este control (al poder variar y capturar sus valores de forma dinámica). En todo caso, es una forma muy gráfica de presentar ciertos datos.

### ***barra de progreso, elemento progress***

La idea es la misma que en el caso anterior, pero ahora se usa para mostrar información sobre el grado de cumplimiento de una determinada tarea. Esta barra es más sencilla de configurar. Usa estos atributos:

atributo significado

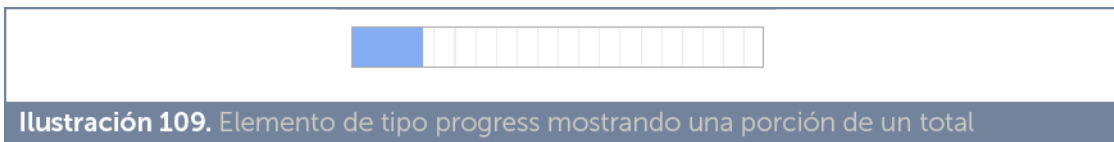
**max** Máximo valor. Se toma este valor como la meta a alcanzar

**value** Valor actual. Desde el inicio hasta el valor se pintará en la barra de progreso

Ejemplo de barra tipo progress:

```
<progress max="100" value="35">35%</progress>
```

El resultado es:

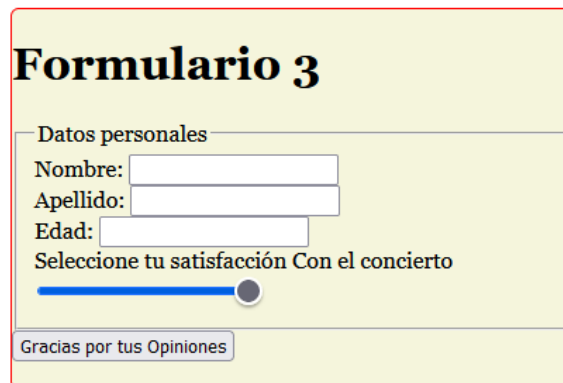




## Práctica 10 a Tablas Básicas

### Ejercicio 1

Crea un formulario como este usando fieldtext



**Formulario 3**

Datos personales

Nombre:

Apellido:

Edad:

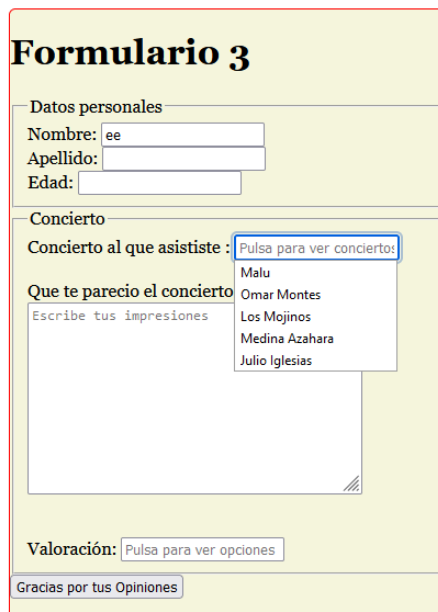
Selecciona tu satisfacción Con el concierto

Gracias por tus Opiniones

\* Estilo css usado : `style="width: 400px; margin-left:auto ;margin-right:auto;background-color: beige;border-color:brown;border: 1px solid red ;border-radius:6px ;"`

### Ejercicio 2

Modifica el anterior para Crear un formulario como este usando datalist y textarea



**Formulario 3**

Datos personales

Nombre:

Apellido:

Edad:

Concierto

Concierto al que asististe :

Que te pareció el concierto

Escribe tus impresiones

Valoración:

Gracias por tus Opiniones

## Ejercicio 3

Crear un formulario como este usando datalist y la etiqueta progress. Añade un javascript para que cuando pulse genera aleatorio ponga datos aleatorios en los valores del progress

```
<script>
function cambio() {
    Velocidad.value=Math.random()*200;
    Fuerza.value=Math.random()*200;
    Persuasion.value=Math.random()*200;
    Espionaje.value=Math.random()*200;
}
</script>
<button onclick=" cambio()"> Genera tu personaje </button>
```

## Ejercicio 4

Modifica el anterior para cambiar los progress por meter

Modifica tambien para dependiendo de la valoracion sea un personaje en el input text debes poner onchange()

### Formulario 3

Genera tu personaje para el juego

Nombre:

Valoración:

Escoge la edad

Datos del personaje

Fuerza

Velocidad

Persuasión

Espionaje

### Formulario Meter

Genera tu personaje para el juego

Nombre:

Valoración:

Escoge la edad

Datos del personaje

Fuerza

Velocidad

Persuasión

Espionaje