

A1	A2	A3
B1		
B2		

(proviene de la página anterior)

```
<table border="1">
  <tbody>
    <tr>
      <td>A1</td>
      <td>A2</td>
      <td>A3</td>
    </tr>
  </tbody>
</table>
</td>
</tr>
<tr>
  <td>
    <table border="1">
      <tbody>
        <tr>
          <td>B1</td>
        </tr>
        <tr>
          <td>B2</td>
        </tr>
      </tbody>
    </table>
  </td>
</tr>
</table>
```

2.10 Formularios

Un formulario permite que el usuario interactúe con la página por medio de una serie de controles

2.10.1 Campo de texto

Permite crear una zona donde el usuario puede escribir y se muestra un ejemplo a continuación. Tiene algunos atributos que se usan muy a menudo:

- `type` indica el tipo de control
- `name` será el nombre de la variable en JS (no lo usaremos por ahora)
- `id` permitirá procesar el control de JS (no se usará por ahora)
- `value` permite indicar un valor por defecto
- `size` indica la anchura por defecto

```
<input type="text" name="nombre_usuario"
      id="id_nombre" value="Escriba su nombre aqui">
```

Un campo de texto puede llevar asociada una etiqueta `label` que indique al navegador que texto va con ese campo. Esto es de utilidad para programas lectores de páginas y en general para gente con discapacidad.

```
<label for="id_nombre">Nombre de usuario</label>
<input type="text" name="nombre_usuario"
      id="id_nombre" value="Escriba su nombre aqui">
```

Si el `type` de este elemento se sustituye por `password` se obtiene un control igual, pero que reemplaza el texto por símbolos que ocultan el texto.

Es posible ofrecer un conjunto de posibles opciones para un campo de texto usando un control llamado `<datalist>`. Así, supongamos que pedimos el idioma al usuario pero deseamos ofrecer la posibilidad de autocompletar mostrando algunos idiomas comunes. Se puede usar este código:

```
<br/>
Introduzca su idioma:<input type="text" id="idiomascomunes">
<datalist id="idiomas">
  <option value="Inglés"></option>
  <option value="Alemán"></option>
  <option value="Español"></option>
  <option value="Francés"></option>
</datalist>
```

2.10.2 Campo email

Si sabemos con seguridad que en un campo se va a introducir el email se puede usar este código:

```
<br/>
Introduzca su email:<input type="email" name="sexo">
```

El navegador hará **automáticamente** la comprobación de que lo que introduce el usuario es realmente un email.

2.10.3 Selector único (radio-button)

Permite elegir una sola opción de entre muchas, se necesita usar un `input` de tipo `radio`.

```
<br/>
<input type="radio" name="sexo">Masculino
<br/>
<input type="radio" name="sexo">Femenino
```

Si se desea que un control de tipo checkbox o radio aparezca marcado por defecto se debe añadir el atributo `multiple="multiple"`.

2.10.4 Selector múltiple (checkbox)

Permite elegir múltiples combinaciones de opciones. El nombre del control utilizará los corchetes para crear un vector que se procesará desde Javascript.

```
<br/>
<input type="checkbox" name="medios[]">
Coche
<br/>
<input type="checkbox" name="medios[]">
Moto
<br/>
<input type="checkbox" name="medios[]">
Bici
```

2.10.5 Lista desplegable

Permite elegir valores de una lista.

```
<select name="provincia">
  <option value="AB">Albacete</option>
  <option value="CR">Ciudad R.</option>
  <option value="CU">Cuenca</option>
</select>
```

Se debe recordar que el texto que ven los usuarios es lo que va entre las etiquetas option. El valor que comprobarán los programadores es lo que va en value En una lista desplegable se pueden elegir muchos valores usando el atributo `multiple`.

Si se desea que una opción (o varias si usamos el selector múltiple) aparezca marcada se debe usar `selected="selected"`.

2.10.6 Textareas

Permiten introducir textos muy largos:

```
<textarea rows="10" cols="15">
  Valor por defecto
</textarea>
```

2.11 Otros elementos HTML: contenido embebido y multimedia

En HTML existen otras etiquetas que permiten insertar contenido dentro del HTML que no tiene por qué ser HTML