



Campos de texto

Introducción

La etiqueta `<input>` puede tener diferentes `types`, entre los que se encuentran los de tipo texto, que serán:

```
<input type="email"> el teclado mostrará botones especiales  
como .com y @  
<input type="hidden"> info que queremos enviar con el  
formulario pero que no se vea en la pantalla  
<input type="password"> no aparecerán los caracteres visibles  
<input type="search"> para búsquedas  
<input type="tel"> tipo teléfono, desplegará teclado numérico  
<input type="text"> (por defecto)  
<input type="url">  
<textarea>
```

A estos inputs les pueden acompañar los atributos `autocomplete`, que indica si queremos que aparezca información anteriormente escrita en este campo y `placeholder`, que será un texto a modo de sugerencia/ayuda para el usuario.

Existe además uno complementario que es [spellcheck](#) para que el navegador compruebe la ortografía cuando escribimos.

Input de tipo text

El campo de texto, como bien indica su nombre, es el que nos permite insertar texto en un formulario. El usuario podrá insertar texto visible sobre el campo.

```
<input type="text" id="identificador" size="tamaño"
      name="nombre" value="texto por defecto"/>
```

De esta forma, si queremos crear un campo de texto para poder insertar 70 caracteres que contenga un email, lo definiremos de la siguiente forma.

```
<input type="text" id="email" name="email"
      size="70" value="usuario@dominio.com"/>
```

Input de tipo password

Cuando un usuario inserte una contraseña dentro de un formulario necesitaremos, casi seguro, que el valor de la contraseña no aparezca y que por el contrario aparezcan caracteres como asteriscos.

```
<input type="password" id="identificador" size="tamaño"
      name="nombre"/>
```

Si queremos definir un campo que acepte contraseñas de 20 caracteres lo codificaremos de la siguiente forma:

```
<input type="password" id="clave" name="clave" size="20"/>
```

Etiqueta textarea

Además de los inputs tenemos la etiqueta `<textarea>` que es muy similar al input con el `type="text"` con la diferencia que no se autocierra y nos permite recoger textos con mayor volumen, como son los comentarios, descripciones, observaciones... Esta etiqueta también puede utilizar el atributo `id` para enlazarse con un `<label>`.

La etiqueta `<textarea>` define un control de entrada de texto de varias líneas. Además, se usa a menudo en un formulario, para recopilar las entradas de los usuarios, como comentarios o reseñas.

Como curiosidad, comentar que un `<textarea>` puede contener una cantidad ilimitada de caracteres y el texto se muestra por defecto en una fuente de ancho fijo (generalmente Courier).

El atributo `name` es necesario para hacer referencia a los datos del formulario después de enviar el formulario. Asimismo, el atributo `id` es necesario para asociar el área de texto con una `<label>`.

Como siempre os dejamos un ejemplo de código para mayor facilidad:

```
<label for="comments">Comentarios:</label>
<textarea id="comments">Este será el contenido
de los comentarios</textarea>
```

Algunos de los atributos más usados son

`minlength` y `maxlength` en los input de tipo `text` son usados para indicar un tamaño mínimo o máximo de caracteres en una entrada de datos. Así podremos restringir un input para que al menos tenga `n` caracteres, o que no se exceda de `n` caracteres.

En el ejemplo estamos diciendo que el input de tipo texto referente al "nick" debe ser mínimo de 2 caracteres de largo y máximo 8.

```
<form method="post" action="/user">
  <label for="nick">nickName:</label>
  <input type="text" name="nick" placeholder="Su nick"
    minlength="2" maxlength="8">
  <button type="submit">Enviar</button>
</form>
```

Qué hemos aprendido

1. Los tipos de inputs de texto que existen.
2. Los atributos que los acompañan.
3. Cómo usar textareas.