



CICLO DE FORMACIÓN PROFESIONAL GRADO SUPERIOR DISEÑO DE APLICACIONES WEB

UT8 JS

Módulo: Desarrollo Web Entorno Cliente

Alumno: Daniel Rodríguez Suárez-Bustillo

Madrid enero de 2023


```

1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5 <meta charset="utf-8" />
6 <title>Form validation: Task 1</title>
7 <style>
8 body {
9 background-color: #fff;
10 color: #333;
11 font: 1em / 1.4 Helvetica Neue, Helvetica, Arial, sans-serif;
12 padding: 10px;
13 margin: 0;
14 }
15
16 * {
17 box-sizing: border-box;
18 }
19
20 span {
21 color: red;
22 }
23
24 input:invalid {
25 border: 1px dashed red;
26 }
27
28 input:valid {
29 border: 1px dashed green;
30 }
31
32 #imgPreview {
33 width: 100px;
34 height: 100px;
35 }
36 </style>
37 </head>
38
39 <body>
40 <!-- Primera versión del ejercicio 2.
41 1. Cambia los siguientes campos para que todos sean de tipo text; quita los atributos (maxlength, min, etc.)
42 que existe y añade lo correspondiente para que solo tengan patrones:
43 a. Nombre: Introducir entre 3 y 20 letras.
44 b. Apellido: Introducir entre 4 y 40 letras.
45 c. Usuario: 5 letras y a continuación 2 dos números. Ejemplo: mari01
46 d. Correo: exactamente dos 7 caracteres de usuario seguidos de @email.com (solo válidos para usuarios de email).
47 Ejemplo válido: mari01@email.com, este campo será autocompletado a partir de la información de usuario. Aunque
48 el usuario podrá modificarlo.
49 e. Número de teléfono: valor numérico de 9 cifras obligatoriamente debe comenzar por 6, 7, 8 o 9.
50 -->
51
52 <div>Ejercicio 2</div>
53 <p>Crea cuenta en red social</p>
54 <form novalidate>
55 <label for="nombre">Nombre</label>
56 <input name="nombre" id="nombre" type="text" pattern="[a-zA-Z]{3,20}" required>
57
58 <br>
59 <label for="apellido">Apellido</label>
60 <input name="apellido" id="apellido" type="text" minlength="4" maxlength="40" pattern="[a-zA-Z]{4,40}" required>
61
62 <br>
63 <label for="usuario">Usuario</label>
64 <input name="usuario" id="usuario" type="text" minlength="7" pattern="[a-zA-Z0-9]{7,40}" required>
65
66 <br>
67 <label for="correo">Correo</label>
68 <input name="correo" id="correo" type="email" required>
69
70 <br>
71 <label for="telefono">Telefono</label>
72 <input name="telefono" id="telefono" type="text" pattern="[6-9][1][0-9]{8}" required>
73
74 <br>
75 <label for="fecha">Fecha de nacimiento</label>
76 <input name="fecha" id="fecha" type="date" required>
77
78 <div class="fechaMayor"></div>
79
80 <label for="checkbox">Marca tus intereses</label></div>
81 <input type="checkbox" name="checkbox">
82 <label for="checkbox">Ciencia</label>
83 <input type="checkbox" name="checkbox">
84 <label for="checkbox">Tecnología</label>
85 <input type="checkbox" name="checkbox">
86 <label for="checkbox">Matemáticas</label>
87 <input type="checkbox" name="checkbox">
88 <label for="checkbox">Deportes</label>
89 <input type="checkbox" name="checkbox">
90 <label for="checkbox">Otro</label>
91
92 <br>
93 <label for="foto">Foto de perfil</label>
94 <input type="file" name="foto" accept="image/*" onchange="previewImage(event, 'imgPreview')">
95 <br>
96 <span class="error"></span>
97 <button id="btnValidar" type="button">Validar</button>
98 <br>
99 <br>
100 <img id="imgPreview">
101
102 </form>
103
104 <script>
105 "use strict";
106 //VALIDAMOS AL USUARIO PARA USARLO EN EL CORREO Y OBTENER EL PATRÓN CORRESPONDIENTE
107 document.querySelector('input[name="usuario"]').addEventListener('blur', () => {
108 let usuarioPersona = document.querySelector('input[name="usuario"]').value;
109 let usuarioCorreo = usuarioPersona + '@gmail.com';
110 //Creamos el correo por defecto
111 let campoCorreo = document.querySelector('input[name="correo"]').value = usuarioCorreo;
112 //introducimos el pattern por js para que se adapte al correo generado para poder controlarlo por validity
113 let prueba = document.getElementById('correo');
114 prueba.setAttribute('pattern', usuarioCorreo);
115 //NOTA*: para comprobar que los patrones se introducen correctamente, comprobarlo por consola
116 //en el cliente
117 })
118
119
120
121 //PREVUALIZACIÓN DE LA IMAGEN
122 //SI ELIMINAMOS EL ING DEL STYLE LA IMAGEN SE VERÁ COMO SU ORIGINAL
123 function previewImage(event, imgPreview) {
124 //Obtenemos el input que desencadenó la acción
125 const input = event.target;
126 //Obtenemos la etiqueta img donde cargaremos la imagen
127 var imgPreview = document.querySelector(imgPreview);
128 //Verificamos si existe una imagen seleccionada
129 if (input.files.length) return
130 //Obtenemos el archivo subido
131 var file = input.files[0];
132 //Creamos la url
133 var objectURL = URL.createObjectURL(file);
134 //Como creamos el atributo src de la etiqueta img
135 imgPreview.src = objectURL;
136 }
137
138 //
139
140 //CONTROL DE FECHA EVITANDO QUE USEN FECHAS POSTERIORES A PERSONAS QUE TIENEN MENOS DE 18 AÑOS Y MÁS DE 99
141 //Creamos la fecha del día de hoy
142 var ago = new Date()
143 //restamos 18 años para saber quien es a día de hoy ya mayor de edad
144 ago.setFullYear(ago.getFullYear()-18);
145 //Convertimos la fecha en un String y eliminamos el texto que va detrás de la t y ya que no lo vamos a usar
146 var fechaFinal = ago.toISOString().split('T')[0];
147 //restamos 99 años para saber quien es a día de hoy ya mayor de 99 años
148 ago.setFullYear(ago.getFullYear()-99);
149 //Convertimos la fecha en un String y eliminamos el texto que va detrás de la t y ya que no lo vamos a usar
150 var fechaInicial = ago.toISOString().split('T')[0];
151 //Asignamos como valores máximos y mínimos del input para poder controlarlos con validity
152 document.getElementById("fecha").setAttribute('max', fechaFinal);
153 document.getElementById("fecha").setAttribute('min', fechaInicial);
154
155 //-----

```

```

154 //-----
155
156 //-----
157
158 //VALIDADOR DEL FORMULARIO
159 //Declaración de constantes
160 const form = document.getElementById('form')[0];
161 const boton = document.querySelector('button[id="btnValidar"]');
162 const frmNombre = document.querySelector('span.error');
163
164 const nombre = document.querySelector('input[name="nombre"]');
165
166 const apellido = document.querySelector('input[name="apellido"]');
167
168 const usuario = document.querySelector('input[name="usuario"]');
169
170 const correo = document.querySelector('input[name="correo"]');
171
172 const telefono = document.querySelector('input[name="telefono"]');
173
174 const nacimiento = document.querySelector('input[name="fecha"]');
175
176
177 //Listener que observa el formulario para saber cuando esta bien el campo
178 nombre.addEventListener('input', (event) => {
179   if (nombre.validity.valid) {
180     frmNombre.innerHTML = "";
181     frmNombre.className = "error";
182   }
183 })
184
185 apellido.addEventListener('input', (event) => {
186   if (apellido.validity.valid) {
187     frmApellido.innerHTML = "";
188     frmApellido.className = "error";
189   }
190 })
191
192 usuario.addEventListener('input', (event) => {
193   if (usuario.validity.valid) {
194     frmUsuario.innerHTML = "";
195     frmUsuario.className = "error";
196   }
197 })
198
199 correo.addEventListener('input', (event) => {
200   if (correo.validity.valid) {
201     frmCorreo.innerHTML = "";
202     frmCorreo.className = "error";
203   }
204 })
205
206 telefono.addEventListener('input', (event) => {
207   if (telefono.validity.valid) {
208     frmTelefono.innerHTML = "";
209     frmTelefono.className = "error";
210   }
211 })
212
213 nacimiento.addEventListener('input', (event) => {
214   if (nacimiento.validity.valid) {
215     frmFecha.innerHTML = "";
216     frmFecha.className = "error";
217   }
218 })
219 //-----
220
221 //Listener que comprueba en orden todos los campos del formulario
222 boton.addEventListener('click', (event) => {
223   if (!nombre.checkValidity()) {
224     showErrorMessage();
225     event.preventDefault();
226   } else if (!apellido.checkValidity()) {
227     showErrorMessage();
228     event.preventDefault();
229   } else if (!usuario.checkValidity()) {
230     showErrorMessage();
231     event.preventDefault();
232   } else if (!correo.checkValidity()) {
233     showErrorMessage();
234     event.preventDefault();
235   } else if (!telefono.checkValidity()) {
236     showErrorMessage();
237     event.preventDefault();
238   } else if (!nacimiento.checkValidity()) {
239     showErrorMessage();
240     event.preventDefault();
241   }
242 });
243 //-----
244
245 //Funciones para mostrar los mensajes de error en el sitio (indicado con la etiqueta SPAN guardada en
246 //la variable "ErrorGenerico"
247 //todo ok
248 function showErrorMessage() {
249   if (nombre.validity.valueMissing) {
250     frmNombre.innerHTML = "<b>Nombre/</b> campo obligatorio";
251     frmNombre.className = "Error on el campo NAME valueMissing";
252   }
253   } else if (nombre.validity.patternMismatch) {
254     if (nombre.value.length < 3) {
255       frmNombre.innerHTML = "<b>Nombre/</b> debe tener entre 3 y 20 caracteres";
256       frmNombre.className = "Error en el campo NAME rage";
257     } else {
258       frmNombre.innerHTML = "<b>Nombre/</b> no puede contener caracteres especiales ni numeros";
259       frmNombre.className = "Error en el campo NAME mismatch";
260     }
261   }
262 }
263
264 //todo ok
265 function showErrorMessage() {
266   if (apellido.validity.valueMissing) {
267     frmApellido.innerHTML = "<b>Apellido/</b> campo obligatorio";
268   }
269   } else if (apellido.validity.patternMismatch) {
270     if (apellido.value.length < 4 || apellido.value.length > 20) {
271       frmApellido.innerHTML = "<b>Apellido/</b> debe tener entre 4 y 40 caracteres";
272     } else {
273       frmApellido.innerHTML = "<b>Apellido/</b> no puede contener caracteres especiales ni numeros";
274     }
275     frmApellido.className = "Error en el campo APELLIDO";
276   }
277 }
278
279 //todo ok
280 function showErrorMessage() {
281   if (usuario.validity.valueMissing) {
282     frmUsuario.innerHTML = "<b>Usuario/</b> campo obligatorio";
283     frmUsuario.className = "Error en el campo USERNAME missing";
284   } else if (usuario.value.length < 7) {
285     frmUsuario.innerHTML = "<b>Usuario/</b> debe tener al menos 7 caracteres";
286     frmUsuario.className = "Error en el campo USERNAME short";
287   }
288 }
289
290 //todo ok
291 function showErrorMessage() {
292   if (correo.validity.valueMissing) {
293     frmCorreo.innerHTML = "<b>Correo/</b> campo obligatorio";
294     frmCorreo.className = "Error en el campo CORREO missing";
295   }
296   } else if (correo.validity.typeMismatch) {
297     frmCorreo.innerHTML = "<b>Correo/</b> introducido no es de tipo email";
298     frmCorreo.className = "Error en el campo CORREO MISMATCH";
299   } else if (correo.value != (usuario.value + "@mail.com")) {
300     frmCorreo.innerHTML = "<b>Correo/</b> debe ser: {<b>usuario.value + '@mail.com'</b>}";
301     frmCorreo.className = "Error en el campo CORREO MISMATCH";
302   }
303 }
304
305 //todo ok
306 function showErrorMessage() {
307   if (telefono.validity.valueMissing) {
308     frmTelefono.innerHTML = "<b>Telefono/</b> campo obligatorio";
309     frmTelefono.className = "Error en el campo TELEFONO missing";
310   } else if (telefono.validity.patternMismatch) {
311     frmTelefono.innerHTML = "<b>Telefono/</b> debe comenzar por 6, 7, 8, 9 y tener una extensión fija de 9 cifras <b>Has introducido: " + telefono.value.length + " cifras en total";
312     frmTelefono.className = "Error en el campo Telefono short";
313   }
314 }
315
316 //todo ok
317 function showErrorMessage() {
318   if (nacimiento.validity.valueMissing) {
319     frmFecha.innerHTML = "<b>Fecha de Nacimiento/</b> campo obligatorio";
320     frmFecha.className = "Error en el campo FECHA missing";
321   } else if (nacimiento.validity.rangeOverflow || nacimiento.validity.rangeUnderflow) {
322     frmFecha.innerHTML = "<b>Fecha de Nacimiento/</b> la edad para menores de 18 años o mayores de 99 años";
323     frmFecha.className = "Error en el campo FECHA short";
324   }
325 }
326
327 </script>
328
329 </html>

```