

UT 5

PRÁCTICA 4

Desarrollo Web Entorno Cliente
22/12/2023



ANA BATRES



ÍNDICE

I. EJERCICIO	2-4
II. HTML	4
III RESULTADO	5
IV BIBLIOGRAFÍA	6

1. Sobre el ejercicio UT5_Práctica_3, realiza todas las validaciones utilizando los métodos y propiedades vistos en el vídeo de guía (validación a través de API en Javascript):

```
var mensaje = "";

window.onload = iniciar;

function iniciar() {
    document.getElementById("Contrasena").addEventListener('input', validarContrasena);
    document.getElementById("Telefono").addEventListener('input', validarTelefono);
    document.getElementById("Url").addEventListener('input', validarUrl);
    document.getElementById("boton").addEventListener('click', recogerFormulario);
}

function validarContrasena() {
    var contrasena = document.getElementById("Contrasena");
    if (!contrasena.checkValidity()) {
        mensaje = "Contraseña incorrecta";
        mostrarError();
        return false;
    } else {
        borrarError();
        return true;
    }
}

function validarTelefono() {
    var telefono = document.getElementById("Telefono");
    if (!telefono.checkValidity()) {
        mensaje = "Teléfono incorrecto";
        mostrarError();
        return false;
    } else {
        borrarError();
        return true;
    }
}

function validarUrl() {
    var url = document.getElementById("Url");
    if (!url.checkValidity()) {
        mensaje = "Url incorrecta";
        mostrarError();
        return false;
    } else {
        borrarError();
        return true;
    }
}
```

Declaramos una variable llamada **mensaje** con una cadena vacía para almacenar los mensajes de error.

Con el evento **window.onload** cuando la página se carga completamente, se llama a la función **iniciar()**, la cual establecerá los eventos de escucha para los diferentes campos del formulario. El cual hace que cada vez que se interactue con alguno, te vaya escribiendo si es correcto o no el contenido.

ValidarContrasena(), validarTelefono(), y validarUrl() cada una de estas funciones valida un campo específico del formulario con **checkValidity()**.

Si es valido elimina el mensaje de error si lo hubiera y devuelve true, sino envia un mensaje de error y lo muestra por pantalla.

```
function validar(e) {  
    if (!validarContrasena() || !validarTelefono() || !validarUrl()) {  
        e.preventDefault();  
        return false;  
    }  
    return true;  
}  
  
function mostrarError() {  
    document.getElementById("error").innerHTML = mensaje;  
}  
  
function borrarError() {  
    document.getElementById("error").innerHTML = "";  
    mensaje = "";  
}
```

Con la función **validar()** la cual se ejecuta cuando se hace clic en el botón para enviar el formulario, se asegura de que todos los campos del formulario sean válidos llamando a las funciones de validación anteriores. Si alguno de los campos no es válido, se evita que el formulario se envíe con **preventDefault()**;

Con las funciones **mostrarError()** y **borrarError()**: la primera muestra el mensaje de error en cada caso y la otra borra los mensajes de error para que no se repitan.

```
function recogerFormulario() {  
    var formulario = document.querySelector(".formulario");  
    var zonadatos = document.getElementById("error");  
    var inputs = formulario.elements;  
    var datos = "";  
  
    for (let i = 0; i < inputs.length; i++) {  
        if (inputs[i].id !== "boton" && validar()) {  
            datos += inputs[i].name + ": " + inputs[i].value + "<br>";  
            zonadatos.innerHTML = datos;  
        }  
    }  
}
```

Con `recogerFormulario()`, función la cual se ejecuta al hacer click en el botón. Recorre todos los elementos del formulario y asigna los datos recopilados a la zona donde anteriormente aparecía el error en caso de que este todo correcto.


HTML

```
<form class="formulario">  
  
    <label for="rango">Rango:</label>  
    <input type="range" id="Rango" name="Rango" min="0" max="100" required><br><br>  
  
    <label for="contrasena">Contraseña:</label>  
    <input type="password" id="Contrasena" name="Contrasena" maxlength="10"  
    pattern="[A-Za-z]{2,10}" required><br><br>  
  
    <label for="telefono">Teléfono:</label>  
    <input type="tel" id="Telefono" name="Telefono" pattern="[0-9]{9}"  
    maxlength="9" required><br><br>  
  
    <label for="web">URL:</label>  
    <input type="url" id="Url" name="Url" placeholder="https://example.com"  
    pattern="https?:/.+" required><br><br>  
  
    <input type="button" id="boton" value="Enviar">  
  
    <div class="error" id="error"></div>  
  
</form>
```

Si escribo solo una letra en la contraseña me sale un mensaje de error ya que la contraseña debe ser al menos 2, pero en cuando ya está en el rango permitido, el mensaje desaparece. Lo mismo con los demás campos, siguiendo los patterns de cada uno establecidos en el html.

Rango: 	Rango: 
Contraseña: <input type="password" value="•"/>	Contraseña: <input type="password" value="••"/>
Teléfono: <input type="text"/>	Teléfono: <input type="text"/>
URL: <input type="text" value="https://example.com"/>	URL: <input type="text" value="https://example.com"/>
<input type="button" value="Enviar"/>	<input type="button" value="Enviar"/>
Contraseña incorrecta	

Cuando completamos todo de manera correcta y pinchamos en enviar, se imprimen los datos. Si intentamos enviarlo con algun dato erroneo esto no funcionará.

Rango: 

Contraseña:

Teléfono:

URL:

Rango: 42
Contrasena: aa
Telefono: 111111111
Url: https://aaa.es

BIBLIOGRAFÍA

- EJERCICIO 1:

- <https://www.youtube.com/watch?v=a5QH6PSm-ew&t=479s>