

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

49

- Al enviar un formulario podemos procesar el contenido con **JavaScript** para asegurarnos de que todos los datos sean correctos:
  - ▣ Campos obligatorios,
  - ▣ Emails válidos,
  - ▣ Fortaleza de una contraseña,
  - ▣ Habilitar o deshabilitar campos según lo seleccionado,
  - ▣ Etc
- Siempre debemos validar en servidor, porque se puede desactivar JavaScript y esta validación no se realizaría.
- También podemos utilizar la validación de **HTML5**.

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.1. Validación básica con HTML5

50

- **HTML5 ya incorpora una validación básica sin tener que utilizar JavaScript.**
- Podemos utilizar atributos como max, pattern, minlength, etc.

[https://www.w3schools.com/tags/tag\\_input.asp](https://www.w3schools.com/tags/tag_input.asp)

- También existen nuevos inputs como color, date, number, etc y algunos no están soportados por todos los navegadores...¡Cuidado con eso!

[https://www.w3schools.com/html/html\\_form\\_input\\_types.asp](https://www.w3schools.com/html/html_form_input_types.asp)

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.1. Validación básica con HTML5

51

```
<form action="procesar.php" method="post" id="miFormulario">
  <table>
    <tr>
      <td><label for="nombre">Nombre*: </label></td>
      <td>
        <input type="text" name="nombre" id="nombre" maxlength="15" pattern="[A-Za-z]{2,15}" title="Introduce entre 2 y 15 letras" required="required"/>
      </td>
    </tr>
    <tr>
      <td><label for="edad">Edad*: </label></td>
      <td>
        <input type="number" name="edad" id="edad" min="18" max="100" required="required"/>
      </td>
    </tr>
    <tr>
      <td><label for="telefono">Telefono*: </label></td>
      <td>
        <input type="text" name="telefono" id="telefono" pattern="[0-9]{9}" title="Numero de 9 cifras" required="required"/>
      </td>
    </tr>
  </table>
  <input type="submit" value="Enviar" id="enviar" />
  <input type="reset" value="Borrar" id="borrar" />
</form>
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.2. Validación básica con JavaScript

52

### □ 2 opciones:

- ▣ En el formulario `<form>` añadir el atributo `onsubmit` con el resultado de la validación

`onsubmit="return validar()";`

### ▣ **Añadir el evento al botón submit:**

`document.getElementById("enviar").addEventListener('click',validar,false);`

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.2. Validación básica con JavaScript

53

**1ª opción: <form ..... onsubmit="return validar()">**

```
<script>

function validar()    {
    if( comprobación--)
    {
        //mensaje de error
        return false;
    }
    //...
    return true;
}

</script>
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.2. Validación básica con JavaScript

54

### 2ª opción: añadir evento al botón enviar

1. Definir una función “iniciar” que se va a cargar sólo cuando el documento se haya cargado completamente:

```
window.addEventListener("load", iniciar);  
  
function iniciar() {  
    document.getElementById("enviar").addEventListener('click',  
        validar, false);  
}
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.2. Validación básica con JavaScript

55

### 2ª opción: añadir evento al botón enviar

2. La función validar recibe el evento por defecto asociado al botón enviar. Validamos cada apartado llamando a las funciones correspondientes:

```
function validar(e) {  
    if (validaNombre() && validaTelefono() && validaFecha() && validaCheck() &&  
        confirm("Pulsa aceptar si deseas enviar el formulario")) {  
        return true;  
    } else {  
        e.preventDefault(); //Cancela el evento de envío por defecto asignado al botón  
        return false;  
    }  
}
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.2. Validación básica con JavaScript

56

### EJEMPLO:

#### Formulario

Nombre:

Telefono:

Fecha de nacimiento:

Sexo : ☒ Hombre ☐ Mujer

Mayor de 18: ☐

### Validación de un campo de texto obligatorio:

```
function validaNombre() {  
    var elemento = document.getElementById("nombre");  
    limpiarError(elemento);  
    if (elemento.value == "") {  
        alert("El campo no puede estar vacío");  
        error(elemento);  
        return false;  
    }  
    return true;  
}
```



# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.2. Validación básica con JavaScript

57

**Validación de todos los campos de texto obligatorios:**

```
var formulario =  
document.getElementById("miFormulario")  
;  
for (var i=0;  
i<formulario.elements.length; i++){  
    if  
    (formulario.elements[i].type=="text"  
    " &&  
    formulario.elements[i].value==""){  
        alert("El campo  
        "+formulario.elements[i].name+"  
        no puede estar en blanco");  
  
        formulario.elements[i].className  
        ="error";  
        formulario.elements[i].focus();  
        return false;  
    }  
}  
return true;
```

**Validación de un campo de texto como numérico:**

```
function validaTelefono() {  
    var elemento =  
    document.getElementById("telefono");  
    if (isNaN(elemento.value)) {  
        alert("El campo teléfono tiene  
        que ser numérico");  
        return false;  
    }  
    return true;  
}
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.2. Validación básica con JavaScript

58

### Validación de un campo de fecha:

```
function validaFecha() {  
    var dia =  
    document.getElementById("dia").value;  
    var mes =  
    document.getElementById("mes").value;  
    var ano =  
    document.getElementById("ano").value;  
  
    var fecha = new Date(ano, mes, dia);  
    if (isNaN(fecha)) {  
        alert("Los campos de la fecha son  
        incorrectos");  
        return false;  
    }  
    return true;  
}
```

### Validación de un checkbox

```
function validaCheck() {  
    var campoCheck =  
    document.getElementById("mayor"  
    );  
    if (!campoCheck.checked) {  
        alert("Debes ser mayor de  
        edad");  
        return false  
    }  
    return true;  
}
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.2. Validación básica con JavaScript

59

- **Modificación de la apariencia de un campo erróneo:**
  - ▣ Creamos una clase en css y cuando el campo sea erróneo le aplicamos esa clase:

```
.error {  
    border: solid 2px #FF0000;  
}
```

```
function error(elemento) {  
    elemento.className = "error";  
    elemento.focus();  
}  
  
function limpiarError(elemento) {  
    elemento.className = "";  
}
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.3. Validación avanzada con HTML5 y JavaScript

60

- Podemos utilizar la validación de HTML5 y modificar los mensajes de error o de validación con JavaScript.

[https://www.w3schools.com/js/js\\_validation.asp](https://www.w3schools.com/js/js_validation.asp)

[https://www.w3schools.com/js/js\\_validation\\_api.asp](https://www.w3schools.com/js/js_validation_api.asp)

- Partimos del ejemplo con el formulario:

### Formulario

Nombre\*:

Edad\*:

Telefono\*:

Añadimos al HTML:

```
<style>
    .error {
        border: solid 2px #FF0000;
    }
</style>
```

```
</table>
<p id="mensajeError"></p>
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.3. Validación avanzada con HTML5 y JavaScript

61

```
window.addEventListener("click",iniciar);

function iniciar() {
    document.getElementById("enviar").addEventListener('click', validar, false);
}

function validaNombre() {
    var elemento = document.getElementById("nombre");
    //Comprobamos que la validación HTML5 es correcta
    if (!elemento.checkValidity()) {
        //Mostramos un mensaje diferente al mensaje por defecto
        //Si el elemento no tiene un valor
        if (elemento.validity.valueMissing) {
            error2(elemento, "Debe introducir un nombre")
        }
        //Si el elemento no coincide con el patrón
        if (elemento.validity.patternMismatch) {
            error2(elemento, "El nombre debe tener entre 2 y 15 caracteres");
        }
        //error(elemento);
        return false;
    }
    return true;
}
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.3. Validación avanzada con HTML5 y JavaScript

62

```
function validaEdad() {  
    var elemento = document.getElementById("edad");  
    if (!elemento.checkValidity()) {  
        if (elemento.validity.valueMissing) {  
            error2(elemento, "Debe introducir una edad")  
        }  
        //Si se sobrepasa el rango  
        if (elemento.validity.rangeOverflow) {  
            error2(elemento, "El valor debe ser menor de 100")  
        }  
        //Si el rango es menor al establecido  
        if (elemento.validity.rangeUnderflow) {  
            error2(elemento, "El valor debe ser mayor o igual que 18");  
        }  
        //error(elemento);  
        return false;  
    }  
    return true;  
}
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.3. Validación avanzada con HTML5 y JavaScript

63

```
function validaTelefono() {
    var elemento = document.getElementById("telefono");
    if (!elemento.checkValidity()) {
        if (elemento.validity.valueMissing) {
            error2(elemento, "Debe introducir un teléfono")
        }
        //Si el patrón no coincide
        if (elemento.validity.patternMismatch) {
            error2(elemento, "El telefono debe tener 9 numeros");
        } //error(elemento);
        return false;
    }
    return true;
}

function validar(e) {
    borrarError();//borra las marcas de errores previos
    if (validaNombre() && validaEdad() && validaTelefono() &&
        confirm("Pulsa aceptar si deseas enviar el formulario")) {
        return true
    } else {
        e.preventDefault(); //No manda el formulario
        return false;
    }
}
```

# 4. ENVÍO Y VALIDACIÓN DE FORMULARIOS

## 4.3. Validación avanzada con HTML5 y JavaScript

64

```
function error(elemento) {
    //validationMesagge viene predefinido y muestra el mensaje de error
    por defecto
    document.getElementById("mensajeError").innerHTML =
    elemento.validationMessage;
    elemento.className = "error";
    elemento.focus();
}

function error2(elemento, mensaje) {
    document.getElementById("mensajeError").innerHTML = mensaje;
    elemento.className = "error";
    elemento.focus();
}

function borrarError() {
    var formulario = document.forms[0]; //cogemos el primer form
    //elements devuelve todos los inputs
    for (var i = 0; i < formulario.elements.length; i++) {
        formulario.elements[i].className = "";
    }
}
```