



Especialidad: Técnico en Programación

Módulo 3. Submódulo2. Desarrolla aplicaciones que se ejecutan en el cliente.

Prof: Hilda Lucía Rodríguez Gómez

Competencia Profesional: Emplea JavaScript para validar formato de datos

FORMULARIOS Y API FORMS

HTML5 incorpora nuevos tipos para el elemento **<input>**, una API completa para validar y procesar formularios, y atributos para mejorar esta interface; algunos tipos tienen condiciones de validación implícitas, otros solo declaran un propósito para el campo que ayudará a los navegadores a presentar el formulario en pantalla. Así tenemos:

**<input type=" " name=" " id=" ">**

Type=	Uso
email	valida la entrada como una dirección de email.
search	da información al navegador sobre el propósito del campo (búsqueda) para ayudar a presentar el formulario en pantalla.
url	valida la entrada como una dirección web.
tel	da información al navegador sobre el propósito del campo (número telefónico) para ayudar a presentar el formulario en pantalla.
number	valida la entrada como un número. Puede ser combinado con otros atributos (como <b>min</b> , <b>max</b> y <b>step</b> ) para limitar los números permitidos.
range	genera un nuevo control en pantalla para la selección de números. La entrada es limitada por los atributos <b>min</b> , <b>max</b> y <b>step</b> . El atributo <b>value</b> establece el valor inicial para el elemento.
date	valida la entrada como una fecha en el formato <b>año-mes-día</b> .
month	valida la entrada como una fecha en el formato <b>año-mes</b> .
week	valida la entrada como una fecha en el formato <b>año-semana</b> donde el segundo valor es representado por una letra W y el número de la semana.
time	valida la entrada como una hora en el formato <b>hora:minutos:segundos</b> . También puede tomar otras sintaxis como <b>hora:minutos</b> .
datetime	valida la entrada como fecha y hora completa, incluyendo zona horaria.
datetime-local	valida la entrada como una fecha y hora completa, sin zona horaria.
color	valida la entrada como un valor de color.

Atributos para ayudar al control de validación

autocomplete	especifica si los valores insertados serán almacenados para futura referencia. Puede tomar dos valores: <b>on</b> y <b>off</b> .
autofocus	Este es un atributo booleano que enfoca el elemento en el que se encuentra cuando la página es cargada.
novalidate	es exclusivo para elementos <b>&lt;form&gt;</b> . Es un atributo booleano que establece si el formulario será validado por el navegador o no.
formnovalidate	es exclusivo para elementos de formulario individuales. Es un atributo booleano que establece si el elemento será validado por el navegador o no.
placeholder	ofrece información que orientará al usuario sobre la entrada esperada. Su valor puede ser una palabra simple o un texto corto, y será mostrado como una marca de agua dentro del campo hasta que el elemento es enfocado.
required	declara al elemento como requerido para validación. Es un atributo booleano que no dejará que el formulario sea enviado hasta que una entrada para el campo sea provista.
pattern	especifica una expresión regular contra la cual la entrada será validada.
multiple	Este es un atributo booleano que permite ingresar múltiples valores en el mismo campo (como múltiples cuentas de email, por ejemplo). Los valores deben ser separados por coma.
form	asocia el elemento al formulario. El valor provisto debe ser el valor del atributo <b>id</b> del elemento <b>&lt;form&gt;</b> .
list	asocia el elemento con un elemento <b>&lt;datalist&gt;</b> para mostrar una lista de posibles valores para el campo. El valor provisto debe ser el valor del atributo <b>id</b> del elemento <b>&lt;datalist&gt;</b> .



## Elementos

HTML5 también incluye nuevos elementos que ayudan a mejorar y expandir formularios.

<b>&lt;datalist&gt;</b>	Este elemento hace posible incluir una lista de opciones predefinidas que será mostrada en un elemento <b>&lt;input&gt;</b> como valores sugeridos. La lista es construida con el elemento <b>&lt;option&gt;</b> y cada opción es declarada con los atributos <b>value</b> y <b>label</b> . Esta lista de opciones se relaciona con un elemento <b>&lt;input&gt;</b> por medio del atributo <b>list</b> .
<b>&lt;progress&gt;</b>	Este elemento representa el estado en la evolución de una tarea (por ejemplo, una descarga).
<b>&lt;meter&gt;</b>	Este elemento representa una medida, como el tráfico de un sitio web.
<b>&lt;output&gt;</b>	Este elemento presenta un valor de salida para aplicaciones dinámicas.

## Métodos

Algunos de los métodos que incluye HTML5 son:

<b>setCustomValidity(mensaje)</b>	Este método nos permite declarar un error y proveer un mensaje de error para un proceso de validación personalizado. Para anular el error, debemos llamar al método con una cadena de texto vacía como atributo.
<b>checkValidity()</b>	Este método solicita al navegador iniciar el proceso de validación. Activa el proceso de validación provisto por el navegador sin la necesidad de enviar el formulario. Este método retorna <b>true</b> (verdadero) si el elemento es válido.

## Eventos

Los eventos incorporados para esta API son los siguientes:

<b>invalid</b>	Este evento es disparado cuando un elemento inválido es detectado durante el proceso de validación.
<b>forminput</b>	Este evento es disparado cuando un formulario recibe la entrada del usuario.
<b>formchange</b>	Este evento es disparado cuando un cambio ocurre en el formulario.

## Estado

API Forms provee un grupo de atributos para controlar estados en un proceso de validación personalizado.

<b>valid</b>	Este estado es un estado de validación general. Retorna <b>true</b> (verdadero) cuando ninguno de los estados restantes es <b>true</b> (verdadero), lo que significa que el elemento es válido.
<b>valueMissing</b>	Este estado es <b>true</b> (verdadero) cuando el atributo <b>required</b> fue incluido en el elemento y el campo está vacío.
<b>typeMismatch</b>	Este estado es <b>true</b> (verdadero) cuando la entrada no es el valor esperado de acuerdo al tipo de campo (por ejemplo, cuando se espera un email o una URL).
<b>patternMismatch</b>	Este estado es <b>true</b> (verdadero) cuando la entrada no es un valor admitido por la expresión regular especificada con el atributo <b>pattern</b> .
<b>tooLong</b>	Este estado es <b>true</b> (verdadero) cuando el largo de la entrada es mayor que el valor especificado en el atributo <b>maxlength</b> .
<b>rangeUnderflow</b>	Este estado es <b>true</b> (verdadero) cuando la entrada es menor que el valor declarado para el atributo <b>min</b> .
<b>rangeOverflow</b>	Este estado es <b>true</b> (verdadero) cuando la entrada es mayor que el valor declarado para el atributo <b>max</b> .
<b>stepMismatch</b>	Este estado es <b>true</b> (verdadero) cuando el valor declarado para el atributo <b>step</b> no corresponde con los valores en los atributos <b>min</b> , <b>max</b> y <b>value</b> .
<b>customError</b>	Este estado es <b>true</b> (verdadero) cuando un error personalizado fue declarado para el elemento.