

Programación Web

Temas de la clase 8:

- Elementos de un formulario - parte 2.
- Atributos adicionales y validación.
- Formularios y Javascript.

Elementos de formulario - Menú desplegable

Otro elemento que puede usarse en formularios es el menú desplegable que se crea con las etiquetas **Select** y **Option**.

```
<select name="opciones" múltiple size="2">
```

```
  <option value="op1" selected>Opción 1</option>
```

```
  <option value="op2">Opción 2</option>
```

```
</select>
```

Elementos de formulario - Menú desplegable

El atributo **name** sirve para agrupar las opciones dentro de un mismo nombre, al igual que en el elemento **input:radio**.

El atributo **múltiple** del ejemplo anterior permite seleccionar varios elementos del menú desplegable al mismo tiempo.

El atributo **size** indica cuántas opciones de la lista serán visibles sin que el usuario deba dar un clic en la lista.

Elementos de formulario - Textarea

El elemento **textarea** permite agregar un cuadro de texto editable, similar a los **input:text** pero este tiene la opción de ser redimensionado por el usuario y puede tener varias líneas de alto. Estos elementos se suelen usar en los campos de los formularios que permiten al usuario escribir historias o comentarios.

```
<textarea cols="30" rows="5"></textarea>
```

Los atributos **cols** y **rows** indican el tamaño en caracteres horizontal y vertical respectivamente.

Atributos adicionales - Múltiple en email

El **input:email** admite el uso del atributo **multiple** al igual que en el menú desplegable, permitiendo agregar varias direcciones de correo electrónico separados por comas.

```
<input type="email" multiple placeholder="correo@gmail.com">
```

Atributos adicionales - Múltiple en file

El **input:file** admite el uso del atributo **multiple** al igual que en el menú desplegable y en email, permitiendo seleccionar varios archivos.

<input type="file" multiple>

Atributos adicionales - Pattern

El atributo **pattern** nos permite añadir una primera capa de validación para nuestros **input** en los formularios, usando este atributo podemos indicar que caracteres son considerados válidos para nuestros **input**.

Estas secuencias de caracteres para los pattern se llaman Expresiones Regulares, y se abrevian “RegExp”. A continuación un ejemplo de uso:

```
<input type="password" pattern="[a-zA-Z0-9]{8,16}">
```

Con el código anterior estamos indicando que los valores válidos para nuestro input son, letras de la “a” a la “z” en minúscula y mayúscula, además de números del 0 al 9, y un mínimo de 8 hasta 16 caracteres.

Atributos adicionales - Pattern

Para crear patrones, pueden hacerse manualmente o utilizar la página www.html5pattern.com la cual ofrece opciones de patrones para copiar y pegar en nuestro código **html**, según se quiera usar para nombres, tarjetas, fechas, passwords, etc. también nos permite crear un patrón por nosotros mismos y probarlo en la página para comprobar su funcionamiento usando el enlace “**Make your own!**”.

Las expresiones regulares usadas en **pattern** pueden ser desde muy simples a muy complejas, es necesario que en **html** se acompañen con la etiqueta **title** del **input** para indicar las reglas del pattern.

Atributos adicionales - Step - Number

En los **input:number** podemos utilizar el atributo **step** para indicar cual es el incremento para el valor del número en nuestro **input** que se considera válido.

Por ejemplo: **<input type="number" min="0" max="10" step="2">**

Con lo escrito anteriormente tenemos valores aceptados como válidos entre el 0 y el 10, pero con intervalos de 2, entonces, cada vez que incrementamos el valor será de a 2, y del mismo modo, solo estos números serán válidos para nuestro **input**: 0,2,4,6,8,10.

Formularios y JavaScript en HTML

Podemos agregar funciones a los botones de nuestro formulario, o incluso a botones que no pertenezcan a un formulario mediante el uso de código **javascript** de las siguientes dos formas:

- 1) **<button onclick="alert('Javascript en html')">Ver Mensaje</button>**
- 2) **<button onclick="verMensaje()">Ver Mensaje</button>**

<script>

function verMensaje() { alert("Javascript en html") }

</script>

Formularios y JavaScript en HTML

En la diapositiva anterior, tenemos la primera opción que utiliza javascript en línea y la segunda que utiliza javascript interno, del mismo que se puede usar CSS. También, al igual que en CSS podemos incluir un archivo javascript en nuestra página de la siguiente forma:

```
<script type="text/javascript" src="dirección del archivo.js" defer></script>
```

El atributo **defer** es opcional, e indica que este archivo **javascript** debería cargarse solamente cuando el resto de la página se haya cargado.

Formularios y Javascript - Selección de elementos

Podemos obtener los datos del formulario en JavaScript de la siguiente forma:

```
let nombre = document.getElementById("id de mi input");
```

let se utiliza para asignar variables, en este caso, ahora nombre es una variable que contiene una referencia a un elemento input con el id correspondiente.

Formularios y Javascript - Obtener el valor

- El valor que contiene el **input** puede obtenerse de la siguiente forma:

nombre.**value**;

- En caso de elementos **select option**:

elementoSelect.**options**[elementoSelect.**selectedIndex**].**value**;

- Si tratamos con **radio**:

let selectedRadio =

document.querySelector("input[name='radioName']:checked");

selectedRadio.**value**;

Formularios y PHP - Introducción

Si en el valor **action** del formulario escribimos la dirección de un archivo o página **php**, éste será quien procese la información de nuestro formulario.

Vamos a ver un pequeño ejemplo con un formulario y el método **POST**.

Nuestro formulario deberá permitir el ingreso de un correo, un asunto y un mensaje.

Luego vamos a armar un pequeño código en lenguaje **php** que tomará los datos introducidos en el formulario para luego enviar un correo electrónico.

Nuestros archivos deberán ser ubicados en la carpeta

“C:/Windows/xampp/htdocs” o donde tengamos instalado el programa **XAMPP** el cual nos permite tener un servidor local de **php**.

Formularios y PHP - Introducción

Código en archivo PHP:

```
<?php
```

```
    $para = $_POST["email_para"];
```

```
    $asunto = $_POST["email_asunto"];
```

```
    $mensaje = $_POST["email_mensaje"];
```

```
    mail($para, $asunto, $mensaje);
```

```
?>
```