

# Formularios HTML

Programación IV

Prof. Karlinna Chaves



# Formularios HTML <Form>

- Hasta ahora hemos utilizado el HTML como un lenguaje para la presentación de contenidos (imágenes, textos, listas, tablas, etc) pudiendo el usuario interactuar con ese contenido pero sin participar.
- Sin embargo con los formularios podemos recibir datos del usuario: nombre, apellidos, números, valores [true|false], email, ficheros, etc.



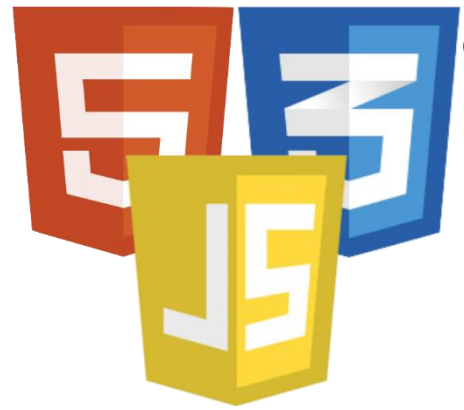
# Formularios HTML <Form>

- Al llenar un formulario, y darle “Enviar”, la información se envía al servidor y esta es utilizada con el fin que se solicitó.
- ¿Qué utilidad se le puede dar a esta información? En general, **un formulario html nos sirve para acceder a otra URL** (dirección web) que recibe la información.
- Cuando la nueva URL recibe la información pueden ocurrir varias cosas:
  - que se muestre en pantalla,
  - que se escriba en una base de datos,
  - que nos muestre una imagen distinta según la información que hayamos enviado...



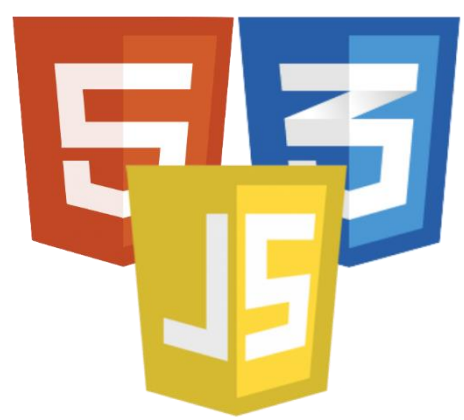
# Formularios HTML <Form>

- Algunas de las cosas que hemos citado no se pueden hacer usando HTML.
- Por ejemplo, no podemos escribir en una base de datos usando HTML.
- Sin embargo, sí podemos recoger la información en un formulario o página web HTML y luego acceder a una dirección web de tipo php, asp, jsp... que son tipos de páginas web que permiten realizar otro tipo de funciones más avanzadas relacionadas con la programación, como es la escritura en bases de datos para almacenar la información.



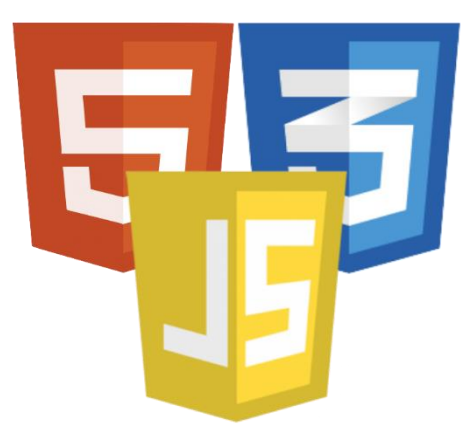
# Formularios HTML <Form>

- Posibilidad de tener varios formularios en una web:
- Cuando creamos un formulario en HTML, todo el contenido que lleva lo incluimos entre las etiquetas de apertura <form> y de cierre </form>. Estas dos etiquetas delimitan el formulario.
- Dentro de una página web puede haber varios formularios, por ejemplo podríamos tener dos formularios en una página web para que los usuarios de un tipo rellenen un formulario y los de otro tipo otro formulario



# Formularios HTML

- Etiqueta `<form></ form >`
  - Todo formulario se encuentra contenido dentro de una etiqueta `<form>`, la cual funciona como un contenedor de los distintos campos que conforman el formulario.
  - Atributos:
    - `action = "URL"` indica hacia donde enviaremos los datos recogidos por el formulario. Podemos también enviar a un email con `action="mailto:alguien@mail.com"` (desaconsejado)
    - `method = "POST|GET"` indicamos el modo en el que enviaremos los datos.
      - Aunque no existe un modo 100% seguro, se considera POST más seguro dado que envía los datos dentro de la cabecera HTTP; mientras que los datos enviados por GET aparecen en la URL y son más fácilmente visibles.



# Formularios HTML

- Etiqueta `<form></form>`
  - Atributos:
    - `method = "POST|GET"`
      - Diferencias entre enviar por GET y POST:
        - POST permite enviar más información que GET (el límite de la URL).
        - POST permite enviar ficheros adjuntos. GET no.
        - Los datos por POST se almacenan en la cabecera HTTP. Los de GET en la URL.
    - ¿Cuándo utilizar GET o POST?
      - Como norma general se utiliza GET para los formularios de búsqueda (como en Google, donde vemos los términos de la búsqueda en la URL).
      - Cuando queramos realizar una modificación contra una BD de datos por medio de un formulario utilizaremos POST por ser más seguro, admitir un mayor número de información y soportar el envío de ficheros.



# Formularios HTML

- Etiqueta `<form></form>`

- Atributos:

- `enctype` = "application/x-www-form-urlencoded | multipart/form-data | text/plain" Tipo de codificación en la que se envía el formulario. Generalmente no se indica salvo en el caso que enviemos ficheros adjuntos.
    - `accept-charset` = "UTF-8, ISO-8859-1, etc" Establecemos la codificación de los datos contenidos, que por defecto coincide con el del documento HTML:
    - `name` o `id`. Es una buena práctica asignar siempre un identificador o un `name` a un formulario. Para así identificarlo y poder realizar validaciones previas en Java Script.
    - `autocomplete` = "on" Permite autocompletar datos del formulario en función de datos previamente introducidos por el usuario. Incluso es un atributo que se puede aplicar a otros `input` de modo que puede estar activado en el formulario y desactivado para un campo (o viceversa).
    - `novalidate` Permite desactivar la validación de formularios en HTML5.





# Elementos de Formularios HTML

- **Etiqueta < input />**
  - La etiqueta <input> permite incluir los distintos controles del formulario o campos donde poder introducir los datos. Es una etiqueta simple por lo que en XHTML se escribe <input/>
  - Existen 10 tipos de campos distintos en XHTML y aún más con HTML5:
    - text | password | checkbox | radio | submit | reset | file | hidden | image | button



# Atributos básicos de un control

- **1.- name:** identifica un control dentro de un formulario. Este atributo nos va a resultar necesario para poder rescatar la información.
- **2.-value:** cada control tiene un valor inicial y un valor actual. Normalmente, el valor inicial de un control puede especificarse con el atributo value. El valor actual del control se hace en primer lugar igual al valor inicial. A partir de ese momento, el valor actual del control puede ser modificado mediante la acción del usuario. El valor inicial de un control no cambia. Así, cuando se carga el formulario, el valor actual de cada control se restablece a su valor inicial. Si el control no tiene un valor inicial, el efecto de una reinicialización o carga del formulario sobre ese control es indefinido. Indefinido significa que no sabemos exactamente lo que va a pasar. Para que no ocurra esto, es preferible establecer siempre un valor inicial.



# Atributos básicos de un control

- **3.-id:** el valor de este atributo permite relacionar un control con una etiqueta.
- Por ejemplo, si un control tiene por id el valor `id="email"`, esto significa que ese control está relacionado con la etiqueta (label) cuyo atributo `for` es `for="email"`.



# Elementos de Formularios HTML

- Etiqueta `< input type="text" />`
  - Es el tipo de campo más común. Se muestra un texto vacío en donde usuario puede escribir texto.

Nombre

```
<label>Nombre
```

```
    <input type="text" id="nombre" name="nombre" />
```

```
</label>
```

```
.
```



# Elementos de Formularios HTML

- Etiqueta `< input type="password" />`
  - Como su propio nombre indica se utiliza para escribir contraseñas o contenido que no queramos que se vea en pantalla.

Contraseña

`<label>Contraseña`

`<input type="password" id="contrasena" name="contrasena"/>`

`</label>`



# Elementos de Formularios HTML

- Etiqueta `< input type="checkbox" />`
  - Son casillas de verificación que permiten al usuario seleccionar una o varias opciones. Son campos independientes unos del otro.
  - Para los controles checkbox y radiobutton permite indicar qué opción aparece preseleccionada

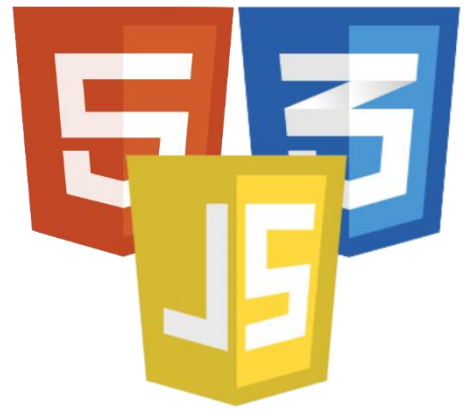
☒ New York ☒ Boston ☐ San Francisco ☐ Wasington DC

```
<label><input type="checkbox" name="ny" checked> New York</label>
```

```
<label><input type="checkbox" name="bos" checked> Boston</label>
```

```
<label><input type="checkbox" name="sfo"> San Francisco</label>
```

```
<label><input type="checkbox" name="dc"> Wasington DC</label>
```



# Elementos de Formularios HTML

- Etiqueta `<input type="radio" />`
  - Los elementos de este tipo son similares a los checkbox pero con la distinción de que son excluyentes. Cuando se selecciona una opción automáticamente se desmarca la otra.
  - Es importante destacar que dado que son excluyentes usan el mismo atributo name

Sexo

☒ Hombre    ☐ Mujer

```
<label><input type="radio" name="sexo" value="hombre" checked="checked" /> Hombre</label>
```

```
<label><input type="radio" name="sexo" value="mujer" /> Mujer</label>
```



# Elementos de Formularios HTML

- Etiqueta `< input type="button" />`
  - En ocasiones nos podemos encontrar con formularios especialmente complejos que necesitan más botones que un reset o un enviar. (Por ejemplo una calculadora).
  - Estos botones en sí no realizan acciones en concreto sino que sirven como disparadores de funciones JavaScript.

```
<input type="button" name="sumar" value="Sumar" />
```

```
<input type="button" name="restar" value="Restar" />
```

```
<input type="button" name="dividir" value="Dividir" />
```

```
<input type="button" name="multiplicar" value="Multiplicar" />
```





# Elementos de Formularios HTML

- Etiqueta `< input type="submit" />`
  - Genera un botón para enviar los datos del formulario.



```
<input type="submit" name="buscar" value="Buscar" />
```

- Etiqueta `< input type="reset" />`
  - Permite borrar los campos y resetear el formulario.



```
<input type="reset" name="limpiar" value="Borrar datos del formulario" />
```



# Elementos de Formularios HTML

- Etiqueta `<input type="file" />`
  - Por medio de este tipo de campo podemos incluir archivos para ser subidos al servidor. En HTML no existe ninguna limitación para en formatos y tamaño de los archivos a subir. Sin embargo todos los servidores incluirán un límite (generalmente entre 2 y 8 Mbytes) en el tamaño de los ficheros.

Fichero adjunto

```
<form method="post" enctype="multipart/form-data">
```

Los formularios que adjunten ficheros tendrán el atributo `enctype` o no funcionarán.



# Elementos de Formularios HTML

- Etiqueta `< input type="hidden" />`
  - En ciertos casos será necesario pasar datos en el formulario no visibles para el usuario (aunque aparecerán en código HTML visibles). Generalmente son datos necesarios para la gestión, controles de seguridad o bien que ya se le han preguntado al usuario.

```
<input type="hidden" name="id_usuario" value="109" />
```



# Elementos de Formularios HTML

- Etiqueta `< input />`

- Otros atributos:

- `name="nombre"`: Cada elemento de un formulario hay que identificarlo con un atributo name, para poder identificarlo y de este modo cuando llegue a su destino se pueda procesar.
    - `value="texto"`: Podemos establecer el valor inicial del campo.
    - `alt="descripción"`: Como en las imágenes nos permite incluir una descripción.
    - `tabindex="numero"`: Nos permite especificar el orden que seguiremos en la selección de un campo de formulario cuando pulsemos la tecla de tabulador.
    - `placeholder=""`: Nos permite incluir una descripción sobre el uso del campo en los tipo text.
    - `size="tamaño"`: Tamaño del control (para los campos de texto y de password se refiere al número de caracteres, en el resto de controles se refiere a su tamaño en píxel).
    - `maxlength="tamaño"`: Máximo número de caracteres para los controles de texto y de password.



# Elementos de Formularios HTML

- Etiqueta `< input />`
  - Otros atributos:
    - Readonly: El elemento muestra un valor pero no es posible modificarlo.
    - Disabled: El elemento input aparece desactivado y no es posible interactuar con él. Por ejemplo un botón de submit. No se envían sus datos. (Se usa también con `<select>` y `<textarea>`)
    - Required: Hasta que el campo no ha sido completado no se puede enviar el formulario.



# Elementos de Formularios HTML5

- **Etiqueta `< button>` `< /button>`**

- En HTML5 se ha introducido el tag `<button>` como un elemento independiente, pero que también se puede incluir en formularios.

```
<button type="submit">Enviar</button>
```

- **Etiqueta `<textarea>` `< /textarea>`**

- La etiqueta `<textarea>` se utiliza cuando se debe de introducir una gran cantidad de texto, dado que se puede especificar altura y anchura.
- Se utilizan los atributos `row=` y `cols=` para dotar de altura y anchura en líneas o columnas al elemento.

```
<form action="" method="post">  
  <textarea id="descripcion" name="descripcion" cols="40" rows="5"></textarea>  
</form>
```



# Elementos de Formularios HTML5

- **Etiqueta < select> < /select>**
  - Los denominados campos <select> visualmente tienen la apariencia de listas de elementos u opciones.
  - Se compone de un contenedor <select> y tantos elementos <option> como posibles valores o alternativas que queramos contener.
  - Lista con un sólo elemento visible.

```
<label for="asiento">Asiento</label> <br/>
<select id="asiento" name="asiento">
  <option value="" selected>- Elija asiento -</option>
  <option value="ventanilla">Ventanilla</option>
  <option value="pasillo">Pasillo</option>
  <option value="central">central</option>
</select>
```

**Asiento**

- Elija asiento - ▼



# Elementos de Formularios HTML5

- Etiqueta `< select >` `< /select >`
  - Lista con **varios** elementos visibles.

```
<label for="asiento">Asiento</label> <br/>  
<select id="asiento" name="asiento" size="2">  
  <option value="" selected>- Elija asiento -</option>  
  <option value="ventanilla">Ventanilla</option>  
  <option value="pasillo">Pasillo</option>  
  <option value="central">central</option>  
</select>
```



A screenshot of a web browser displaying a dropdown menu. The menu is open, showing two visible options: 'Pasillo' and 'Central'. The 'Central' option is currently selected and highlighted in blue. The menu has a standard scroll bar on the right side.





# Elementos de Formularios HTML5

- Etiqueta `< select >` `< /select >`

- Lista con **selección múltiple** de elementos.

```
<label for="so">Sistema operativo</label> <br/>
```

```
<select id="so" name="so" size="4" multiple>
```

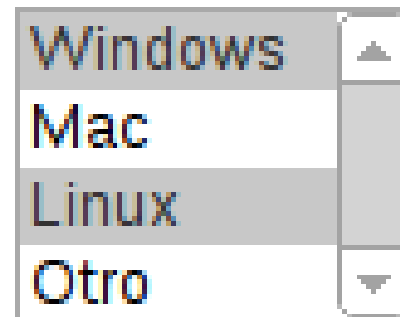
```
  <option value="windows" selected>Windows</option>
```

```
  <option value="mac">Mac</option>
```

```
  <option value="linux">Linux</option>
```

```
  <option value="otro">Otro</option>
```

```
</select>
```



# Elementos de Formularios HTML5

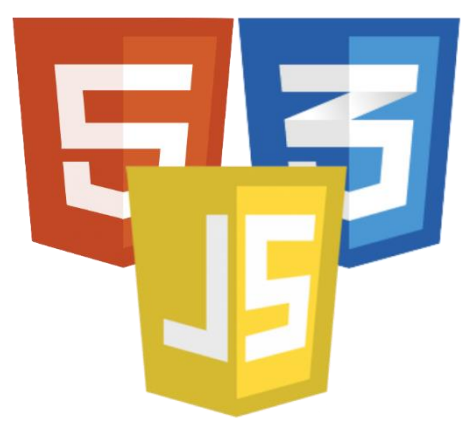
- Etiqueta `< select>` `< /select>`
  - En los campos `<select>` podemos además agrupar opciones visualmente con `<optgroup>`.

```
<label for="asiento">Asiento</label> <br/>
<select id="asiento" name="asiento">
  <optgroup label="Primera clase">
    <option value="pri-pas">Pasillo</option>
    <option value="pri-ven">Ventanilla</option>
  </optgroup>
  <optgroup label="Segunda clase">
    <option value="seg-pas">Pasillo</option>
    <option value="seg-ven">Ventanilla</option>
  </optgroup>
</select>
```



# Elementos de Formularios HTML5

- Etiqueta `<input>` `</input>`
  - Se han añadido nuevos tipos de campos `<input>` con diferente soporte en los distintos navegadores.
  - color, date, datetime, datetime-local, email, month, number, range, search, tel, time, url, week.
- Para referencia consultar:
  - [https://www.w3schools.com/html/html\\_form\\_input\\_types.asp](https://www.w3schools.com/html/html_form_input_types.asp)
  - [https://www.w3schools.com/html/html\\_form\\_attributes.asp](https://www.w3schools.com/html/html_form_attributes.asp)
  - <http://www.htmlcinco.com/nuevos-tipos-de-input-en-html5/>



# Elementos de Formularios HTML5

- **Etiqueta `< input>` `< /input>`**

- `type="number"` Podemos enviar enteros, estableciendo un mínimo y un máximo como input de texto.
- `type="range"` Podemos enviar enteros, estableciendo un mínimo y un máximo utilizando un control gráfico.
- `type="color"` Nos permite seleccionar desde un “color picker” un color y enviarlo como hexadecimal.
- `type="date"` Nos permite seleccionar una fecha. En los navegadores más avanzados aparece un calendario.
- `type="datetime"` Nos permite seleccionar fecha y hora (con zona horaria).



# Elementos de Formularios HTML5

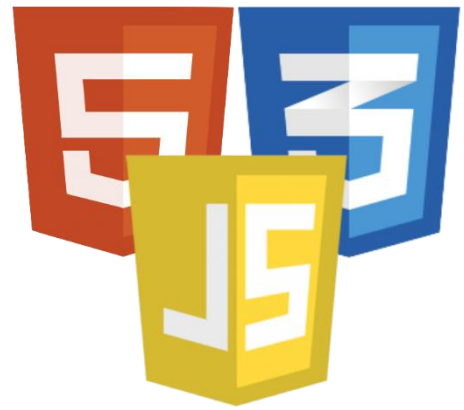
- **Etiqueta `< input>` `< /input>`**
  - `type="datetime-local"` Nos permite seleccionar fecha y hora (sin zona horaria).
  - `type="month"`, `type="week"`, `type="time"` Nos permiten seleccionar distintas unidades de tiempo
  - `type="email"` Permite enviar una dirección de email con una pobre validación.
  - `type="url"` Permite enviar una url con una pobre validación.
  - `type="tel"` Para el envío de números de teléfono.
  - `type="search"` Para definir campos de búsqueda.



# Elementos de Formularios HTML5

- Etiqueta `< datalist>` `< /datalist>`
  - Permite vincular a un input de texto una lista de posibles valores, lo que nos permite emular una funcionalidad de autocompletado.

```
<input list="browsers" name="browsers">  
<datalist id="browsers">  
  <option value="IE">  
  <option value="Firefox">  
  <option value="Chrome">  
  <option value="Opera">  
  <option value="Safari">  
</datalist>
```



# Elementos de Formularios HTML5

- Ejemplos
  - En la siguiente URL se encuentra ejemplos de las etiquetas para formularios:
    - <https://html5doctor.com/demos/forms/forms-example.html>

