

# Programación Estática y Laboratorio Web



**Formularios HTML**



**Universidad Nacional del Comahue**  
**Facultad de Informática**

# [ Formularios HTML ]

- Posibilidad de interactuar con los usuarios
- Realizar preguntas y recibir respuestas es una de las formas fundamentales de conseguir esta interacción y el lenguaje HTML proporciona la habilidad de crear formularios e insertar cajas de texto, botones de opción y otros controles que nos permitirán aprovechar esa posibilidad de interactividad en nuestras páginas Web.
- No es necesario navegar mucho tiempo por Internet para encontrar páginas Web en las que se piden datos a los navegantes.

# [ Formularios HTML ]

Name	Value
Name	<input type="text"/>
Sex	<input type="radio"/> Male <input checked="" type="radio"/> Female
Eye color	<input type="text" value="green"/>
Check all that apply	<input type="checkbox"/> Over 6 feet tall <input type="checkbox"/> Over 200 pounds
Describe your athletic ability: <input type="text"/>	
<input type="button" value="Enter my information"/>	

Name:

Gender:

Email:

Password:

Address:

Your Expertise:

- ☒ PHP
- ☒ CSS
- ☒ HTML
- ☒ Bootstrap
- ☒ jQuery

## ■ Algunos formularios

# [ Formularios HTML ]

- El lenguaje HTML consta de una serie de etiquetas que permitirán crear de forma rápida y sencilla numerosos elementos de entrada de datos. Estos elementos, que reciben el nombre de controles, serán gráficos en la mayoría de navegadores
- Un formulario no es más que un conjunto de estos controles cuya información será enviada conjuntamente cuando el usuario pulse sobre el botón de envío.
- Para crear un formulario el lenguaje HTML proporciona la etiqueta *FORM*.
- Esta etiqueta inicio <FORM>, y una etiqueta de fin, </FORM>, entre las cuales podremos insertar todos los controles que deseemos.

# [ Formularios HTML ]

- El lenguaje HTML define una etiqueta, **INPUT**, que permite insertar varios tipos de controles.
- El atributo type define que clase de control va a renderizar el navegador.
- Ejemplos
  - Cuadros de texto
  - Checkbox (“tildes”)
  - Radio Buttons.
  - Listas desplegables
- Ejemplo para texto: `<INPUT TYPE="text" />`

# [ Formularios HTML ]

- Para diferenciar los distintos elementos que podemos obtener a partir de la etiqueta *INPUT* utilizaremos el atributo *TYPE* que puede tomar los siguientes valores:
  - text,
  - password,
  - radio,
  - checkbox,
  - submit,
  - image,
  - reset,
  - file
  - hidden.

# [ Formularios HTML ]

- Identificación.
- ¿Como distinguiremos más adelante entre distintos controles?
- Existen los atributos “name” e “id” que permiten que desde javascript u otro lenguaje como php sepamos que dato representa cada control
- `<input type=“text” name=“nombre” id=“nombre”>`
- `<input type=“text” name=“apellido” id=“apellido”>`

# Formularios HTML: Campo de texto

- **<input type="text">.**
- **Genera un campo de texto.**
- Para distinguir entre las distintas caja de texto utilizaremos el atributo NAME

```
<FORM>
```

```
  <INPUT TYPE="text" NAME="mitexto_1" />
```

```
  <INPUT TYPE="text" NAME="mitexto_2" />
```

```
  <INPUT TYPE="text" NAME="mitexto_3" />
```

```
</FORM>
```



# Formularios HTML: Campo de texto

- **SIZE:** Determina la anchura del campo de texto. El valor por defecto de este atributo es 20 caracteres.
- En ocasiones convendrá decrementar o incrementar este valor por defecto según el dato esperado y el espacio asignado para el formulario en el dispositivo.

# Formularios HTML: Campo de texto

- **MAXLENGTH**: Con este atributo limitamos el número máximo de caracteres que pueden ser escritos
- El valor de este atributo puede ser mayor o menor que el especificado en SIZE, y que es totalmente independiente. Si es mayor cuando lleguemos al final del campo de texto los nuevos caracteres que insertemos irán desplazando hacia la izquierda a los primeros, que dejarán de estar a la vista
- Por último queda decir que si no usamos el atributo MAXLENGTH el número de caracteres que pueden introducirse en la caja de texto no tendrá límite.

# Formularios HTML: Campo de texto

- **VALUE:** Sirve para especificar un texto que debe aparecer por defecto en el campo de texto, antes de que el usuario escriba.
- **PLACEHOLDER:** *Texto sombreado que aparece como sugerencia del valor a escribir.*

Ingresá tu CUIT:

# Formularios HTML: Passwords

- Campos de texto para claves.
- En ocasiones puede ser conveniente pedir al usuario algún tipo de información confidencial como puede ser una clave de acceso o *password*.
- El lenguaje HTML permite utilizar, campos de texto en el que, al escribir, se mostrarán únicamente asteriscos. Para ello debemos usar ***TYPE="password"*** en la etiqueta INPUT.
- Atributos SIZE, MAXLENGTH, VALUE y ALIGN que vimos en los campos de texto.  

```
<input type="password" name="clave" size="tamaño"  
maxlength="longitud_máxima" value="clave por defecto"  
align="alineamiento">
```

# [ Formularios HTML: radio buttons ]

- `<input type="radio">`
- **Botones de elección.**
- Para insertar un botón de elección usaremos de nuevo la etiqueta *INPUT*, pero esta vez el valor del atributo **TYPE** será *radio*

```
<FORM>  
  <INPUT TYPE="radio" />  
</FORM>
```

# [ Formularios HTML: radio buttons ]

- Los **radio buttons** tiene dos estados o posiciones: seleccionado (ON) o no seleccionado (OFF)
- Inicialmente todos se encuentran en estado OFF.
- Para que inicialmente un radio buttons se encuentre seleccionado se utiliza el atributo CHECKED en la etiqueta INPUT  
`<input type="radio" checked>`

# [ Formularios HTML ]

- Los botones de elección suelen ser insertados en los formularios en grupos, dando al usuario la posibilidad de elegir entre una serie de opciones.
- Cuando una de las opciones es seleccionada el estado del radio buttons es ON y permanecerá en el hasta que pulsemos en otra opción del mismo grupo.
- Sólo uno de ellos puede estar seleccionado, por tanto cuando seleccionamos uno, aquel que estuviese seleccionado previamente dejará de estarlo.
- Para indicar que una serie de radio buttons pertenecen a un mismo grupo debemos incluir el mismo valor en el atributo NAME en cada uno ellos.
- Además debemos usar el atributo VALUE para dar un nombre distinto a cada uno de los botones.

# [ Formularios HTML: checkbox ]

- `<input type="checkbox">`
- **Checkbox** (cajas de selección)
- Al igual que los radio buttons tienen 2 estados seleccionados o no seleccionados,
- El estado por defecto es no seleccionado, a no ser que hayamos usado el atributo **CHECKED**.
- Cada caja de selección es independiente del resto, y por tanto el valor del atributo **NAME** debe ser diferente en cada una.
- Pulsando sobre cada caja de selección se cambia de seleccionado a no seleccionado, independientemente del estado del resto.



# [ Formularios HTML: checkbox ]

- Con las cajas de selección también podemos usar el atributo VALUE, en este caso no será imprescindible, ya que el nombre dado con NAME es diferente en cada caso, pero a veces puede ser conveniente.

# Formularios HTML: submit

- `<input type="submit">`
- Botón de envío del formulario.
- El botón de envío indica que ya terminamos de completar los datos y que queremos enviar la información.
- ¿En el ejemplo cual es?



Acceder

[¿Ha extraviado la contraseña?](#)

[? Aviso de Cookies](#)

# [ Formularios HTML: envio image ]

- **Botón de envio con imagen.**
- El lenguaje HTML permite una manera alternativa para insertar un botón de envío personalizado, al poder usar en lugar del botón normal que hemos visto anteriormente, una imagen.
- Para insertar este tipo de objeto debemos usar ***TYPE="image"***, mientras que la dirección de la imagen se indica con el atributo ***SRC*** al igual que hacíamos con la etiqueta ***IMG***.

# [ Formularios HTML: reset ]

- `<input type="reset" />`
- Restablece el formulario borrando lo que se haya ingresado y/o retornando a los valores por defecto.
- Este control recibe el nombre de botón de borrado o botón de reseteado (reset button).

# [ Formularios HTML: button ]

- `<input type="button" value="Cliqueame" />`
- **Botón genérico**
- Para completar la colección de botones de un formulario, nos encontramos con la posibilidad de insertar un botón genérico, es decir, que no tiene ninguna función definida, sino la que nosotros queramos darle.

# Formularios HTML: hidden

- `<input type="hidden">`
- **Campos ocultos**
- Sirven para incluir información adicional entre los datos del formulario. Este tipo de control es invisible, y por tanto el navegador no muestra ningún tipo de gráfico para advertir de su existencia.
- En general, los campos ocultos se usan para enviar información específica al programa del servidor Web que procesará los datos enviados, por ejemplo si este programa es capaz de realizar varias funciones distintas con estos datos, podemos usar un campo oculto para pasarle la información de que debe hacer con los datos de este formulario concreto

# [ Formularios HTML: textarea ]

- **TEXTAREA**
- **Este no es un input type**
- Los campos de texto multilínea o áreas de texto son unos campos que funcionan de manera similar a un editor de texto muy sencillo en el que el usuario puede escribir.
- La etiqueta usada para insertar este nuevo control es **TEXTAREA**, que consta de cuatro atributos.

# [ Formularios HTML: textarea ]

- **NAME:** El nombre del textarea.
- **ROWS:** El número de líneas del area de texto.
- **COLS:** El número de caracteres visibles de cada línea. Este atributo es similar al atributo SIZE que vimos en los campo de texto.



# [ Formularios HTML: textarea ]

- La etiqueta TEXTAREA está compuesta por una etiqueta de inicio y una etiqueta de fin.
- Entre ellas únicamente podrá insertarse texto plano (sin ninguna etiqueta HTML) y éste será mostrado como contenido inicial del área de texto.

```
<textarea name="nombre" rows="filas" cols="columnas">  
    Texto inicial  
</textarea>
```

# [ Formularios HTML: select ]

- Cuadros de selección.
- Tampoco es un type
- Su función será dar a elegir entre una serie de opciones de manera que el usuario pueda elegir una o varias de ellas.
- Existen dos formas de mostrar estas opciones, como una *lista con desplazamiento*, o mediante una *lista desplegable*.
- Para insertar un cuadro de selección usaremos la etiqueta **SELECT**, que consta de una etiqueta de inicio y de una etiqueta de fin, entre las cuales introduciremos las diferentes opciones para el usuario.
- Para insertar estas opciones usaremos una nueva etiqueta, **OPTION**

# [ Formularios HTML: select ]

```
<form>
```

```
Elija un color:<BR>
```

```
  <select name="color">
```

```
    <option>Verde
```

```
    <option>Negro
```

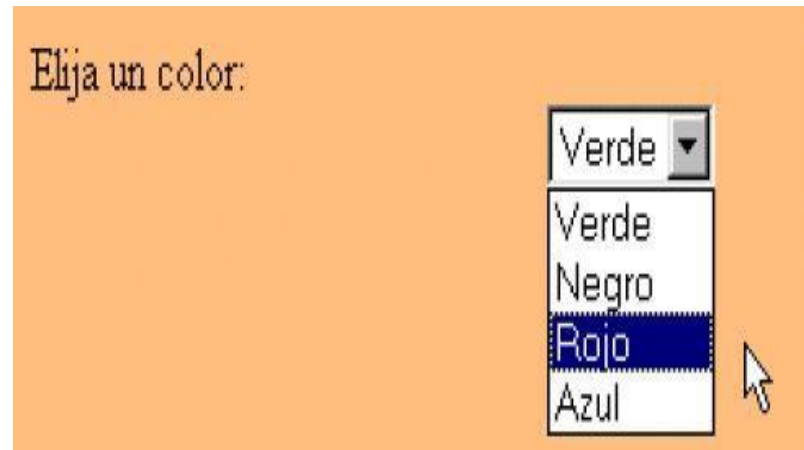
```
    <option>Rojo
```

```
    <option>Azul
```

```
  </ select >
```

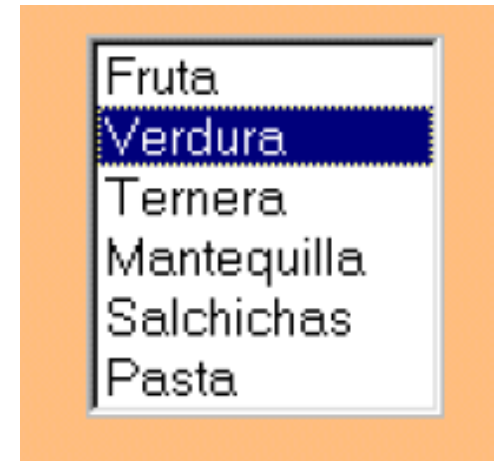
```
<INPUT TYPE="submit" VALUE="Enviar elección">
```

```
</ form >
```



# Formularios HTML: select

- El atributo, **SELECTED**, nos permitirá elegir la opción que debe estar seleccionada por defecto.
- Puede ser conveniente que todas, o al menos algunas de las posibilidades sean mostradas directamente, sin tener que pulsar un botón. Para conseguir esto el lenguaje HTML ofrece el atributo **SIZE**, que determina el número de opciones que pueden ser vistas simultáneamente.
- Si este valor es 1 (o no se usa el atributo SIZE), se mostrará una persiana desplegable, como hemos visto hasta ahora. Si es mayor, se mostrará una lista, que podrá tener barras de desplazamiento si no caben todas las opciones disponibles.



# [ Formularios HTML ]

- **COMO USAR LOS DATOS DE UN FORMULARIO**
- Estos programas deben encontrarse en algún lugar del servidorWeb. Al enviar un formulario a uno, éste procesará la información y contestará con la respuesta oportuna. Esta respuesta será en general o bien la dirección de una página o bien una nueva página.
- La etiqueta FORM consta de un atributo llamado **ACTION** donde debemos especificar la dirección URL del programa donde deben enviarse los datos. Además debe usarse otro atributo, **METHOD**, para indicarle cómo se mandarán estos datos. Este atributo puede tomar dos valores *GET* y *POST*, omitiendo todos los detalles, podemos decir que el primero se usa con formularios pequeños, mientras que el segundo será el escogido para formularios donde la cantidad de información es grande.

# [ Formularios - Input HTML5 ]

- **Input Type: color**

- `<input type="color" name="favcolor" />`

- **Input Type: date**

- `<input type="date" name="bday" />`

- **Input Type: datetime**

- `<input type="datetime" name="bdaytime" />`

- **Input Type: email**

- `<input type="email" name="email" />`

- **Input Type: month**

- `<input type="month" name="bdaymonth" />`

# [ Formularios - Input HTML5 ]

## ■ Input Type: number

- `<input type="number" name="quantity" min="1" max="5" />`

## ■ Input Type: range

- `<input type="range" name="points" min="1" max="10" />`

## ■ Input Type: search

- `<input type="search" name="googlesearch" />`

## ■ Input Type: time

- `<input type="time" name="usr_time" />`

## ■ Input Type: url

- `<input type="url" name="homepage" />`

## ■ Input Type: week

- `<input type="week" name="week_year" />`

# Referencias

---

- <https://www.w3.org/>
- <https://uniwebsidad.com/>
- <http://www.w3schools.com/html/>