



Frontend I

Clase 16

Formularios



**Certified
Developer**
The Ultimate Tech Degree

DigitalHouse >
Coding School



Temas

1

Introducción

2

Repaso

3

Cierre

4

Actividad



Profundizamos en CSS - Variables globales

- En CSS, las variables globales son una característica que permite definir valores que se pueden reutilizar en todo el documento. Estas variables se definen usando la pseudo-clase **:root**.
- La pseudo-clase **:root** se utiliza para seleccionar el **elemento raíz del documento**, que es el elemento HTML. Al definir las variables en **:root**, estamos creando variables globales que se pueden utilizar en cualquier lugar del documento.
- Para definir una variable global, se utiliza la sintaxis **--nombre-variable: valor**. Luego, para utilizar la variable global en cualquier lugar del documento, se utiliza la sintaxis **var(--nombre-variable)**.

```
:root {  
  --color-primario: red;  
}
```

```
.elemento {  
  background-color: var(--color-primario);  
}
```

- De esta manera, podemos reutilizar el valor de la variable global "color-primario" en cualquier lugar del documento donde sea necesario.



Profundizamos en CSS - Importando CSS

"**@import**" es una regla de CSS que permite importar un archivo CSS en otro archivo CSS. Esta regla puede ser útil para mantener la **organización** y **modularidad** del código CSS, ya que permite separar diferentes partes de un proyecto en archivos CSS separados y luego importarlos en un archivo principal.

```
> OPEN EDITORS
  ▾ VARIABLES CSS
    ▾ CSS
      # estilos.css
      # main.css
      # variables.css
      ↕ clase.md
      <> index.html

css > # estilos.css > ...
1  @import "variables.css";
2  @import "main.css";
3
4  h1{
5      color: var(--colorPrimario);
6  }
7
```



Formularios

*** Nombre**

Apellido
Email
***Tel**
Password
Mensaje

Seleccione su nacionalidad:

Argentina

Enviar **Resetear**

Login

Datos personales

Nombre y apellido (Obligatorio)

Tel
Email
Mensaje

Tipo de Documento

☐ DNI
☐ CUIL
☐ CUIT

Imagen Documento

Ninguno archivo selec.

Imagen Documento

Ninguno archivo selec.

Fecha de Nacimiento

Lenguajes

Seleccione los lenguajes que conoce

☐ HTML5 ☐ CSS ☐ JS

Trabajo

Seleccione modalidad trabajo:

Remoto

Seleccione su color favorito:

☐ *Acepto los términos. **Resetear** **Enviar**



Etiqueta Form

```
<form action="" method="POST"> ...  
</form>
```

La etiqueta HTML **<form>** se utiliza para crear un formulario en una página web para que los usuarios puedan enviar información. Los elementos que se incluyen dentro de la etiqueta **<form>** son los campos que se van a enviar.

La etiqueta **<form>** tiene dos atributos principales:

method: especifica el método HTTP que se utilizará para enviar los datos del formulario al servidor. Los dos métodos más comunes son **GET** y **POST**. GET se utiliza para enviar datos de formulario en la URL, mientras que POST se utiliza para enviar datos de forma oculta en el cuerpo de la solicitud.

action: especifica la **URL del servidor que procesará los datos enviados** desde el formulario. Cuando se envía el formulario, los datos se envían a la URL especificada en el atributo action.





Etiqueta input

La etiqueta **<input>** es utilizada para crear campos de entrada de datos en un formulario. Puede tener varios atributos, entre ellos:

type: especifica el tipo de entrada de datos que se debe utilizar. Algunos valores comunes para este atributo son **text**, **password**, **email**, **number**, **date**, **checkbox**, **radio**, etc.

name: especifica el nombre del campo de entrada de datos. Este nombre es utilizado para identificar el campo cuando se envía el formulario.

value: especifica el valor predeterminado del campo de entrada de datos. Este valor se mostrará en el campo cuando se cargue la página.

placeholder: especifica un texto de muestra en el campo de entrada de datos que desaparecerá cuando el usuario empiece a escribir en el campo.

required: especifica que el campo de entrada de datos es obligatorio. Si este atributo se agrega, el formulario no se puede enviar a menos que el campo tenga algún valor.

disabled: especifica que el campo de entrada de datos está desactivado. Esto significa que el usuario no puede interactuar con el campo.





Etiqueta label

El atributo id se utiliza para identificar un elemento en particular en una página web y debe ser único en toda la página. El atributo for se utiliza en la etiqueta label para asociar el campo de entrada de datos (generalmente un elemento input) con la etiqueta label. Esto mejora la accesibilidad del formulario ya que los usuarios pueden hacer clic en la etiqueta para seleccionar el campo de entrada de datos correspondiente.

```
<label for="nombre">Nombre:</label>  
<input type="text" id="nombre" name="nombre">
```





Etiqueta label

Algunos de los beneficios del uso del atributo **for** son los siguientes:

Accesibilidad: Los usuarios con discapacidad visual utilizan a menudo lectores de pantalla para navegar por las páginas web. Al asociar una etiqueta label con un campo de entrada de datos utilizando el atributo for, el **lector de pantalla puede leer el contenido de la etiqueta label** cuando se selecciona el campo de entrada de datos correspondiente. Esto mejora la accesibilidad del formulario.

Usabilidad: Al hacer clic en la etiqueta label, el campo de entrada de datos correspondiente **se selecciona automáticamente**. Esto significa que los usuarios no tienen que hacer clic en el campo de entrada de datos específico para seleccionarlo, lo que puede mejorar la usabilidad del formulario.

SEO: El uso de la etiqueta label y su atributo for también **puede ayudar con la optimización de motores de búsqueda (SEO)**. Al incluir la etiqueta label y su atributo for en el código HTML de la página, se proporciona una información más completa y estructurada para los motores de búsqueda, lo que puede mejorar la visibilidad de la página en los resultados de búsqueda.





Campos obligatorios

A login form titled "Login" with two required fields: "*Nombre" and "Apellido". The "Nombre" field contains the text "Gabriela". The "Apellido" field is empty and has a red border. A tooltip message with an exclamation mark icon says "Por favor, completar correctamente". Below the "Apellido" field are three more empty input fields. At the bottom left is a blue button labeled "Enviar".

El atributo **required** en un elemento input de HTML indica que el campo de entrada de datos es **obligatorio** y no se puede enviar el formulario sin que se haya completado ese campo. Si el campo de entrada de datos se deja en blanco, el navegador mostrará un mensaje de error predeterminado para indicar que el campo es obligatorio.

El atributo **oninvalid** se utiliza para especificar una acción que se debe realizar cuando un campo de entrada de datos que tiene el atributo **required** no se ha completado correctamente. Este atributo puede apuntar a una función de JavaScript que realiza alguna acción específica, como mostrar un mensaje de error personalizado.

```
oninvalid="setCustomValidity('Por favor, completar correctamente')"
```



Email, teléfono y password

El campo de entrada de datos **email** verifica que la entrada del usuario tenga un formato de correo electrónico válido, y puede mostrar un mensaje de error si el formato es incorrecto.

El campo de entrada de datos **teléfono** verifica que la entrada del usuario tenga un formato de número de teléfono válido, y puede mostrar un mensaje de error si el formato es incorrecto.

El campo de entrada de datos **password** oculta la entrada del usuario con asteriscos u otros caracteres para proteger la privacidad del usuario.

Además de los atributos básico, como type, name y id, estos campos también pueden tener atributos específicos como **minlength**, **maxlength**, **pattern** y **placeholder** para ayudar a limitar y validar la entrada del usuario.

A login form titled "Login" in blue text. It contains five input fields with labels in blue text to their left: "* Nombre" with the value "Gabriela", "Apellido" with the value "Perez", "Email" with the value "digital@house.com", "*Tel" with the value "42266659", and "Password" with masked characters ".....". Each input field has a thin grey border.



Textarea

Diferencias entre input y textarea

Tipo de entrada de datos: El elemento input se utiliza para recibir una entrada de datos de una sola línea, como un nombre de usuario o una dirección de correo electrónico. El elemento textarea se utiliza para recibir una entrada de datos de varias líneas, como un comentario o una descripción.

Tamaño de entrada: El tamaño de entrada del elemento input está limitado por el atributo size y el atributo maxlength, mientras que el tamaño de entrada del elemento textarea está limitado por el número de filas y columnas que se le asignan.

Disposición de la entrada: El elemento input suele tener una disposición en línea, mientras que el elemento textarea suele tener una disposición en bloque.

Interfaz de usuario: El elemento input puede tener diferentes tipos de entrada, como texto, contraseña, número, fecha, etc. El elemento textarea solo acepta entradas de texto.

Etiquetas: A diferencia del input, el elemento textarea tiene etiquetas de apertura y de cierre.





Select

El elemento **select** en HTML se utiliza para crear un menú desplegable que permite al usuario seleccionar una opción de una lista de opciones predefinidas. Cada opción en el menú desplegable se crea con la etiqueta **option** y se especifica mediante el atributo **value**.

El atributo **name** en el elemento select se utiliza para identificar el campo en el servidor y el atributo **id** se utiliza para asociar la etiqueta **label** al campo. El atributo **size** se puede usar para especificar el número de opciones que se mostrarán en el menú desplegable a la vez.

El atributo **multiple** se puede agregar al elemento select para permitir la selección de varias opciones a la vez. El atributo **disabled** se puede agregar a las opciones para deshabilitarlas, lo que significa que el usuario no podrá seleccionarlas.

```
<label for="pais">Seleccione su nacionalidad:</label>
<select name="País" id="pais">
  <option value="Argentina">Argentina</option>
  <option value="Brasil">Brasil</option>
  <option value="Colombia">Colombia</option>
  <option value="Venezuela">Venezuela</option>
</select>
<input type="submit" value="Enviar">
```

Seleccione su nacionalidad:



Input type submit y reset

Los elementos input con el atributo **type="submit"** y **type="reset"** son dos tipos comunes de botones utilizados en los formularios HTML, dentro de form.

El elemento input con **type="submit"** se utiliza para enviar un formulario al servidor. Cuando se hace clic en un botón de este tipo, los **datos del formulario se envían al servidor especificado** en el atributo action del formulario. El atributo **value** se utiliza para especificar el texto que aparecerá en el botón.

El elemento input con **type="reset"** se utiliza para **restablecer los valores del formulario a su estado original**. Cuando se hace clic en un botón de este tipo, todos los campos del formulario se restablecen a sus valores predeterminados. El atributo value se utiliza para especificar el texto que aparecerá en el botón.

```
<input type="submit" value="Enviar" />  
<input type="reset" value="Resetear" />
```





¿Input, button o enlace?

- Si deseas enviar el formulario para procesamiento, debes usar un **input type submit** o un **button type submit**. Ambos elementos enviarán el formulario cuando se hace clic en ellos. La diferencia principal entre los dos es que el **button type submit puede tener contenido dentro de él**, mientras que el input type submit no puede.
- Si deseas agregar un botón dentro del formulario para realizar una acción específica que no involucre enviar el formulario, debes usar un **button type button**. Puedes agregar un controlador de eventos a este botón para realizar una acción específica al hacer clic en él, como mostrar u ocultar elementos en la página.
- Si deseas agregar un **enlace** que **abra una página** o **realice una acción en la página** actual sin enviar el formulario, debes usar un enlace `<a>`. Los enlaces `<a>` son útiles para realizar acciones en la página actual, como mostrar u ocultar elementos, sin enviar el formulario.



Agrupamiento

- La etiqueta **fieldset** se utiliza para agrupar elementos relacionados dentro de un formulario. Se puede utilizar para crear una sección dentro del formulario y también para aplicar un borde alrededor del grupo de elementos. Además, fieldset también se puede utilizar para deshabilitar todo el grupo de elementos en un solo paso.
- Por otro lado, la etiqueta **optgroup** se utiliza dentro de un elemento select para agrupar opciones relacionadas. Permite organizar las opciones en subconjuntos y hacer que el menú desplegable sea más fácil de usar para el usuario.



Formularios avanzados

div

h3

Login

Datos personales

Nombre y apellido (Obligatorio)

Tel

Tipo de Documento

☐ DNI ☐ CUIL ☐ CUIT

Imagen Documento

No file chosen

Fecha de Nacimiento

Lenguajes

Seleccione los lenguajes que conoce

☐ HTML ☐ CSS ☐ JS

Trabajo

Seleccione modalidad trabajo:

☐ Acepto los términos.

text

tel

form

date

checkbox

select>option

radio

file

fieldset

legend

fieldset

legend

fieldset

legend

checkbox



Formularios avanzados: radio button

Los botones de radio (o "**radio buttons**" en inglés) son elementos de formulario HTML que permiten al usuario seleccionar una opción exclusiva de un conjunto de opciones. **Solo se puede seleccionar una opción a la vez.**

Los botones de radio se crean utilizando la etiqueta `<input>` con el atributo `type` establecido en "radio". Cada botón de radio debe tener un `name` común para agrupar los botones de opción y permitir que el navegador identifique qué botón de opción está seleccionado.

Además, cada botón de radio puede tener un `value` para representar la opción seleccionada. **El valor predeterminado es "on".** Si se establece un valor, se enviará ese valor al servidor cuando se envíe el formulario.





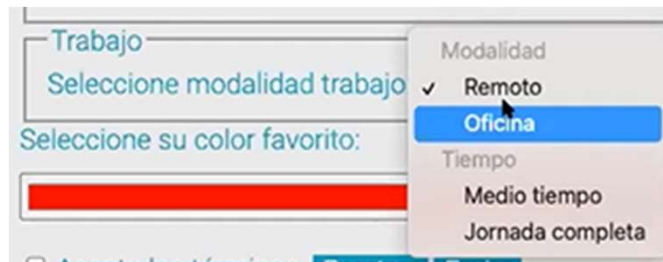
Formularios avanzados: checkbox

Este campo permite a los usuarios **seleccionar una o varias opciones** entre un conjunto de opciones disponibles.

Cuando un usuario marca una casilla de verificación, la información se envía como una cadena de texto que indica el valor de la casilla seleccionada. Si la casilla no está marcada, no se envía información para esa casilla.

El campo de entrada de casilla de verificación también admite el atributo "**checked**", que se puede utilizar para premarcar una casilla de verificación al cargar la página. Además, el atributo "name" se utiliza para identificar el campo de entrada y se utiliza cuando se envía el formulario.

Con optgroup:



The screenshot shows a web form with a dropdown menu. The dropdown is open, showing a list of options. The options are grouped into two categories: 'Modalidad' and 'Tiempo'. Under 'Modalidad', there are two options: 'Remoto' (which is checked with a small 'x' icon) and 'Oficina' (which is highlighted in blue). Under 'Tiempo', there are two options: 'Medio tiempo' and 'Jornada completa'. The form also includes a text input field labeled 'Trabajo' and a label 'Seleccione su color favorito:' with a red bar below it.



Formularios avanzados: file

<input type="file"> es un elemento de formulario en HTML que permite al usuario seleccionar uno o varios archivos de su sistema de archivos local para ser cargados al servidor. Cuando se utiliza la etiqueta **<input type="file">**, se crea un botón que abre un cuadro de diálogo de selección de archivos en el sistema del usuario. Una vez que el usuario ha seleccionado el archivo o los archivos que desea cargar, se puede enviar el formulario al servidor y los archivos se pueden procesar y almacenar.

Algunos de los **atributos básicos** que puede tener son:

accept: Este atributo especifica los tipos de archivo que se pueden seleccionar en el cuadro de diálogo de selección de archivos.

multiple: Este atributo permite que se seleccione más de un archivo para cargar. Usar ctrl +.

name: Este atributo especifica el nombre del campo de archivo que se enviará al servidor. Este nombre se utilizará para identificar el archivo en el lado del servidor.

required: Este atributo indica que el campo de archivo es obligatorio y debe completarse antes de enviar el formulario.

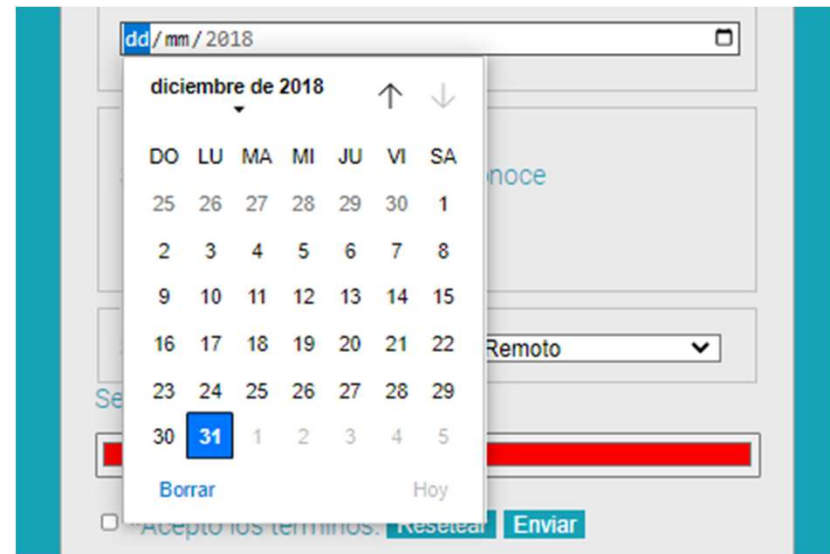




Formularios avanzados: date

Este campo permite a los usuarios seleccionar una fecha utilizando un calendario o una interfaz similar. Cuando un usuario selecciona una fecha, la información se envía en el formato **"AAAA-MM-DD"** (año-mes-día) como una cadena de texto.

El campo de entrada de fecha también admite los atributos **"min"** y **"max"**, que se pueden utilizar para definir un rango de fechas permitidas. Además, el atributo **"step"** se puede utilizar para definir el intervalo entre las fechas disponibles, por ejemplo, **"step=1"** permitiría seleccionar fechas diarias, mientras que **"step=7"** permitiría seleccionar fechas semanales.



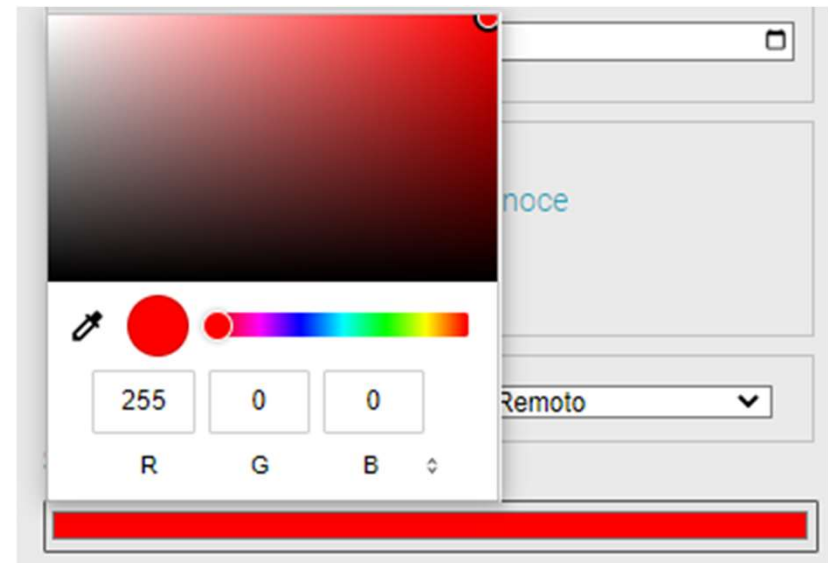


Formularios avanzados: color

Este campo permite a los usuarios seleccionar un color utilizando una **interfaz de selección de color**.

Cuando un usuario selecciona un color, la información se envía en **formato hexadecimal** de 6 dígitos (por ejemplo, #FF0000 para rojo) como una cadena de texto.

El campo de entrada de color también admite los atributos **"min"** y **"max"**, que se pueden utilizar para definir un rango de colores permitidos. Además, el atributo **"step"** se puede utilizar para definir el intervalo entre los colores disponibles.





Formularios avanzados: tabindex

El atributo "**tabindex**" se utiliza en los formularios HTML para establecer el orden en el que se enfocan los elementos de entrada cuando el usuario presiona la tecla "Tab".

Cada elemento de entrada en un formulario tiene un orden de enfoque predeterminado basado en su posición en el código HTML. Sin embargo, el atributo "tabindex" permite modificar este orden de enfoque predeterminado.

El valor del atributo "tabindex" indica el orden en el que se enfocarán los elementos de entrada. Los elementos de entrada con un valor de "tabindex" menor se enfocarán primero, seguidos de los elementos de entrada con un valor de "tabindex" mayor.

Es recomendable utilizar el atributo "tabindex" con precaución y solo cuando sea necesario para mejorar la usabilidad y accesibilidad del formulario. También se debe evitar asignar valores de "tabindex" arbitrarios o no secuenciales, ya que esto puede confundir a los usuarios.





Actividad



<https://www.figma.com/file/9XZ12ORXQuvMNg3ydKQtFo/Formulario-II?node-id=842-2&t=2FkQdqLw3obDPsw-0>





Conclusiones

Hoy pudimos conocer y profundizar sobre formularios, elementos fundamentales en HTML que nos permiten recolectar información de los usuarios. Aprendimos que es importante utilizar las etiquetas `<form>` y sus atributos, y que dentro de ellos podemos agregar diversos tipos de elementos, como campos de texto, botones, selectores, etc.

En resumen, los formularios son una herramienta esencial para recolectar información de los usuarios en la web, y es importante utilizarlos correctamente para mejorar la experiencia del usuario y garantizar la calidad de la información recibida.



DigitalHouse>
Coding School