

Desarrollo Web con HTML

Módulo 3

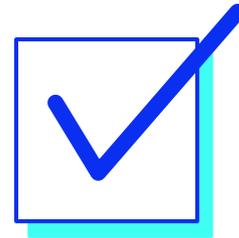
Formularios avanzados: nuevos elementos

Campos de opción: *checkbox*

Los *checkbox* o **casillas de verificación** son controles de formulario que permiten al usuario **seleccionar y deseleccionar opciones de forma individual**.

Aunque en ocasiones se muestran varios *checkbox* juntos, cada uno de ellos es **completamente independiente del resto**.

Por este motivo, se utilizan cuando el usuario puede **activar y desactivar varias opciones relacionadas pero no excluyentes**.



Para incluir casillas de verificación en nuestros formularios, debemos asignar el valor **checkbox** al atributo **type** de nuestro elemento **input**.

El texto que se encuentra al lado de cada checkbox debe **añadirse manualmente fuera del control del formulario**, como se muestra en el ejemplo del próximo *slide*. Si no se añade un texto al lado de la etiqueta **<input>** del checkbox, el usuario solo ve un pequeño cuadrado sin ninguna información relativa a la finalidad de esa casilla.

El valor del atributo **value**, junto con el valor del atributo **name**, es la **información que llega al servidor cuando el usuario envía el formulario**.

Puestos de trabajo buscados

- Dirección
- Técnico
- Empleado



Si queremos chequear el elemento de manera **predeterminada**, debemos aplicarle el **atributo checked**. No hacerlo será suficiente para que **no esté chequeado**.

En el ejemplo, el último elemento aparecerá **chequeado en el navegador**. El usuario puede **deschequearlo si desea**.

- Me voy en bici
- Me voy en auto
- Me voy en barco

```
<form action="form.phpV">
  <input type="checkbox" name="bici" value="bicicleta">Me voy en bici<br>
  <input type="checkbox" name="auto" value="automovil">Me voy en auto<br>
  <input type="checkbox" name="barco" value="barquito" checked>Me voy en barco<br>
</form>
```

Campos de opción: *radio button*

Los controles de tipo *radio button* son similares a los controles de tipo *checkbox*, pero presentan una diferencia muy importante: **son mutuamente excluyentes**.

Los *radio button* se utilizan cuando el usuario **solamente puede escoger una opción entre las distintas opciones** que se le presentan. Cuando se selecciona una, **automáticamente se deselecciona la otra opción**.

- 
- A diagram showing two radio buttons. The first radio button is selected, indicated by a solid black dot in the center of the circle. The second radio button is unselected, shown as an empty circle. The text 'Ejemplo 1' and 'Ejemplo 2' are positioned to the right of their respective radio buttons. The entire diagram is enclosed in a light blue rectangular border with a cyan shadow on the right and bottom sides.
- Ejemplo 1
 - Ejemplo 2

Para poder trabajar con estos elementos, lo haremos como se muestra más abajo.

También podemos trabajar con el atributo **checked** con el mismo propósito que en otros elementos.

¿Eres mayor de edad?

Si

No

```
¿Eres mayor de edad?<br>  
<input type="radio" name="edad" value="si" checked="checked">Si  
<input type="radio" name="edad" value="no">No
```

Botones en formularios

Botón de envío de formulario

La mayoría de **formularios** dispone de un botón para **enviar al servidor los datos introducidos por el usuario**, por ejemplo:

```
<input type="submit">
```

```
<input type="submit" value="Enviar">
```



Enviar

- El **valor del atributo type** para este control de formulario es **submit**. El navegador se encarga de enviar automáticamente los datos cuando el usuario pincha sobre este tipo de botón.
- El valor del atributo **value** es el **texto que muestra el botón**. Si no establecemos el atributo **value**, el navegador muestra el texto predefinido *“Enviar”*.

Button en lugar de *input type submit*

Existe otro botón de envío, llamado `<button>` `</button>`. Este cumple el mismo propósito que el `<input type="submit">` y se trabaja como se muestra en el ejemplo debajo.

```
<button> Enviar </button>
```

Reset de formulario

Aunque su uso fue muy popular hace unos años, la mayoría de formularios ya no utilizan este tipo de botón. Se trata de un botón especial que **borra todos los datos introducidos por el usuario y devuelve el formulario a su estado original**.

Es importante que **todos los botones antes mencionados se encuentren dentro del form**, de forma tal que se vinculen con él.

¿Qué sucede si los botones se encuentran por fuera del formulario?

Si los botones de reset, submit o el elemento button se encontrasen por fuera del formulario, **se deben vincular de alguna otra manera, dado que sino no funcionará.** Es decir, si se hace clic en submit este no enviará el *form* y el reset no lo reseteará.

¿Cómo vincularlos si los botones no están anidados?

Una forma de vincularlos si no están anidados es a través del **atributo form**, cuyo valor será el **id del formulario en cuestión.**

```
<form action="codigo.php" method="post" id="formulario">
</form>

<input type="reset" form="formulario">
<input type="submit" value="enviar" form="formulario">
```

Botones independientes

Los botones de envío, tanto el **button** como el **submit**, pueden tener **valores distintos al formulario de origen**.

Ejemplo: tenemos dos botones, uno enviará la información a un archivo PHP y el otro a otro, quizás para conseguir diferentes fines.

```
<form action="codigo.php" method="post" id="formulario">
</form>

<input type="reset" formaction="codigo1.php" form="formulario">
<input type="submit" value="enviar" formaction="codigo2.php" form="formulario">
```

Para saber más sugerimos revisar sitio oficial de [Mozilla](#).

Formularios avanzados: integración con IA

Uso de IA para formularios

Si bien ya se viene utilizando IA para HTML desde el módulo 1, cuando tratamos de que *ChatGPT* nos ayude a distinguir correctamente con ejemplos y explicaciones más específicas, la diferencia entre elementos semánticos y cómo utilizarlos, es importante entender que la IA tienen muchas otras **ventajas para los desarrolladores o maquetadores**.

Primero, entender que **la IA no va a reemplazar a las personas**, en este caso los maquetadores o programadores, sino que da una gran mano en la **mejora y rapidez** con la cual se trabaja.

La IA ayuda a potenciarse y es un mundo al cual debemos adaptarnos para aprovechar sus ventajas, pero también conocer sus desventajas, para evitar confianza absoluta y errores innecesarios. Recordemos que muchas extensiones y programas basados en IA están en desarrollo y aún mantienen *prompts* complejos y con errores, por lo cual nunca debemos confiar en ellos un 100% para tomar decisiones.

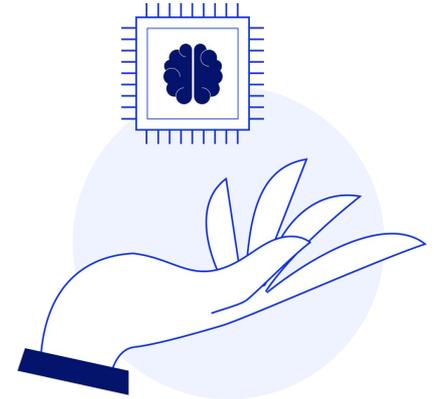
La IA es el futuro en varios aspectos, no sólo del desarrollo y la programación, y debemos **aprender a utilizarla como complemento de nuestro saber previo**.

ChatGPT for Google

El vasto mundo de los **formularios en HTML** poseen una infinidad de elementos, de los cuales hemos aprendido a trabajar con los más comunes o necesarios.

Sin embargo, muchas veces debemos trabajar con otros elementos más complejos, o que tienen particularidades que son interesantes de comprender en profundidad.

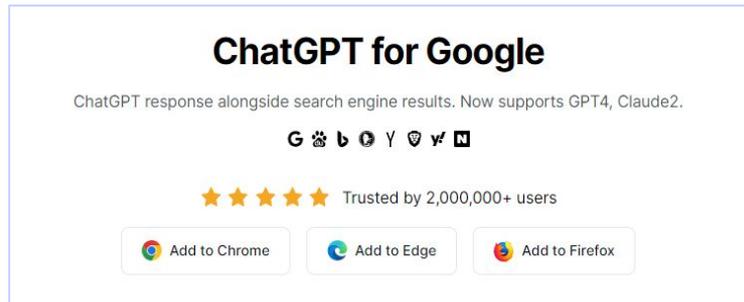
Para eso, utilizaremos la extensión ***ChatGPT for Google***.



Instalación de *ChatGPT for Google*

1. Ir a la [página oficial](#).
2. Loguearse con el usuario.
3. Instalar la extensión.

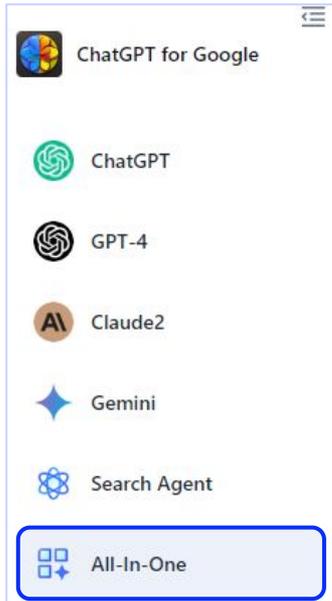
Disponible para instalar en Google, IE o Firefox.



4. Buscar la extensión desde la **barra de extensiones** del navegador.



5. Se abrirá la siguiente pantalla, donde se puede trabajar con cualquiera de las opciones: en este caso la opción **All-in-One**.



6. El **prompt** a utilizar será el siguiente. Cópialo y pégalo:

“Desarrollar un código de HTML que incluya un formulario que contenga un data picker de mes y otro de hora, que no contenga nada de validación generada por JavaScript o FlatPickr”.

Fue necesario ayudar a la extensión, ya que de lo contrario habría incluido una extensión de JavaScript innecesaria para este proceso.

7. El **resultado** es el siguiente:



```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Formulario con Date y Time Picker</title>
</head>
<body>

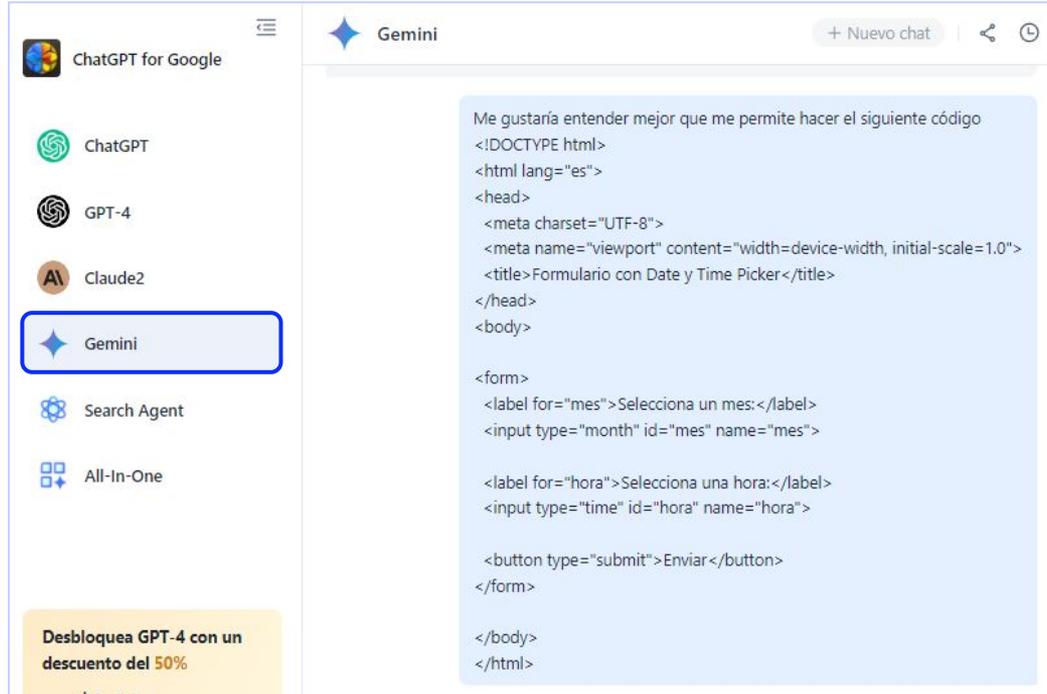
<form>
  <label for="mes">Selecciona un mes:</label>
  <input type="month" id="mes" name="mes">

  <label for="hora">Selecciona una hora:</label>
  <input type="time" id="hora" name="hora">

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

</body>
</html>
```

8. Copiar el **código** y seleccionar la opción **> Gemini**, que es ideal para realizar el siguiente proceso: se quiere entender mejor qué sucedió con el *prompt* y qué función realiza cada elemento generado. Para eso realizar un *prompt* similar al siguiente:



The screenshot shows the Gemini chat interface. On the left, a sidebar lists AI models: ChatGPT for Google, ChatGPT, GPT-4, Claude2, Gemini (highlighted with a blue border), Search Agent, and All-In-One. At the bottom of the sidebar, there is a yellow banner that reads "Desbloquea GPT-4 con un descuento del 50%". The main chat area is titled "Gemini" and contains a message from the user: "Me gustaría entender mejor que me permite hacer el siguiente código". Below the message is a code block containing HTML code for a form with a date and time picker.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Formulario con Date y Time Picker</title>
</head>
<body>

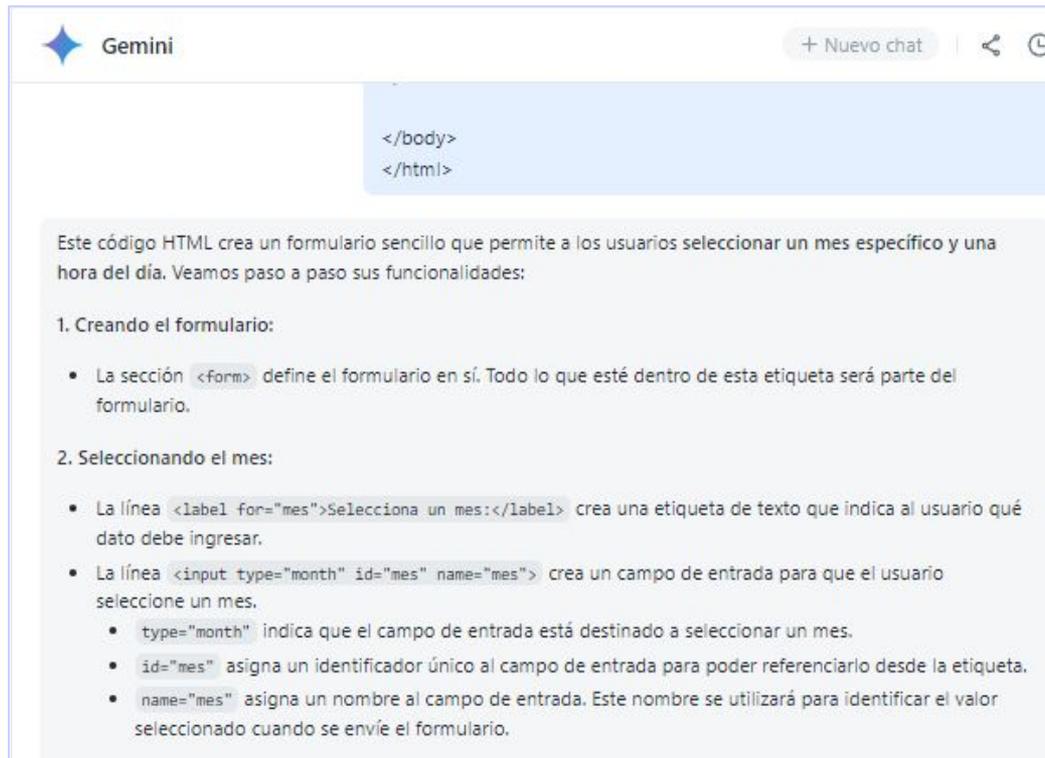
<form>
  <label for="mes">Selecciona un mes:</label>
  <input type="month" id="mes" name="mes">

  <label for="hora">Selecciona una hora:</label>
  <input type="time" id="hora" name="hora">

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

</body>
</html>
```

La **respuesta** será más que interesante, brindará detalles específicos incluso sobre otros elementos que ya conocemos aparte de los *data pickers* o elementos de calendario.



Gemini + Nuevo chat |  

```
</body>
</html>
```

Este código HTML crea un formulario sencillo que permite a los usuarios seleccionar un mes específico y una hora del día. Veamos paso a paso sus funcionalidades:

1. Creando el formulario:
 - La sección `<form>` define el formulario en sí. Todo lo que esté dentro de esta etiqueta será parte del formulario.
2. Seleccionando el mes:
 - La línea `<label for="mes">Selecciona un mes:</label>` crea una etiqueta de texto que indica al usuario qué dato debe ingresar.
 - La línea `<input type="month" id="mes" name="mes">` crea un campo de entrada para que el usuario seleccione un mes.
 - `type="month"` indica que el campo de entrada está destinado a seleccionar un mes.
 - `id="mes"` asigna un identificador único al campo de entrada para poder referenciarlo desde la etiqueta.
 - `name="mes"` asigna un nombre al campo de entrada. Este nombre se utilizará para identificar el valor seleccionado cuando se envíe el formulario.

Revisión

- Repasar **los elementos de formulario** vistos.
- Incorporar **variantes y diferentes atributos**.
- Guardar el archivo en una carpeta llamada **“Web”** o **“Sitio web”**.
- Realizar las **preguntas necesarias** a tu docente antes de continuar.



**¡Sigamos
trabajando!**