



Estructura



Diseño visual

# HTML5 y CSS3 parte 3

## Trabajando con formularios y tablas

### Comenzando la página de contacto

Ocuparemos algunas cosas del proyecto anterior ([productos.html](#)). Ocuparemos todo lo que contiene la parte del: (<**head**></**headheader**></**headerfooter**></**footer**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <link rel="stylesheet" href="reset.css">
  <link rel="stylesheet" href="productos.css">
  <title>Contacto</title>
</head>
<body>
  <header>
    <div class="caja">
      <h1></h1>
      <nav>
        <ul>
          <li><a href="index.html">HOME</a></li>
          <li><a href="productos.html">PRODUCTOS</a></li>
          <li><a href="contacto.html">CONTACTO</a> </li>
        </ul>
      </nav>
    </div>
  </header>

  <main>
    TEST
  </main>

  <footer>
    
    <p class="copyright">&copy; Copyright - Barberia Alura - 2022</p>
  </footer>
</body>
</html>
```





Estructura



Diseño visual

## Funcionamiento de Formularios

**<form>**

Para decidir cuándo debemos utilizar un formulario en nuestras páginas, se pueden analizar algunos motivos:

- 1) Cuando queremos enviar datos a un servidor
- 2) Cuando queremos capturar información que el usuario escribe
- 3) Cuando queremos enviar los datos a otra página

¿Cuáles realmente se pueden tener en cuenta?

Todos los motivos pueden ser tomados en cuenta

Un formulario puede servir, entre otras cosas, para enviar datos a otra página, capturar información que el usuario escribe y enviar datos a un servidor.

## Estructura básica

¿Cuál es la *etiqueta* que usamos para emparejar un `input`, para tener un título para la entrada de datos? ➡ `<label>`

La *etiqueta* `<label>` es una etiqueta para la entrada de datos para el `<input>`.



Estructura



Diseño visual

## for

Hace referencia al (**id= ""**) "input" donde se ingresa la información.

```

<form>

  <label for="nombreapellido">Nombre y Apellido</label>
  <input type="text" id="nombreapellido">

  <label for="correoelectronico">Correo Electrónico</label>
  <input type="text" id="correoelectronico">

  <label for="telefono">Teléfono</label>
  <input type="text" id="telefono">

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

</form>

```

## Estilos para formulario

### ➤ Posicionando formulario en el centro.

- ✓ Las pantallas por lo general tienen 1270px, una buena práctica es poner **940px**.
- ✓ En **margin** ponemos: **0** (tanto superior como inferior) **auto** (automático los costados).

```

main {
  width: 940px;
  margin: 0 auto;
}

```



Estructura



Diseño visual

- Posicionando formulario de forma Vertical.

```
form label {
  display: block;
}

form input {
  display: block;
}
```

- Aplicando “espaciado externo” al formulario.  
(40px “superior, inferior” 0 “los costados”)

```
form {
  margin: 40px 0;
}
```

- Mejorando aspecto visual: “labels” - “inputs”.  
(0 “superior” 0 “derecha” 10px “abajo”)

```
form label {
  display: block;
  font-size: 20px;
  margin: 0 0 10px;
}
```

```
form input {
  display: block;
  margin: 0 0 20px;
}
```

- Agrandando la caja del input:  
(10px “superior, inferior” 25px “los costados”)

```
padding: 10px 25px;
```

- Que la caja del input ocupe: 50% del main (width: 940px)

```
width: 50%;
```

```
form input {
  display: block;
  margin: 0 0 20px;
  padding: 10px 25px;
  width: 50%;
}
```



Nombre y Apellido

Correo Electrónico

Teléfono

Enviar Formulario



EXITOS



Estructura



Diseño visual

## Input con espacio entre el borde y el contenido

¿Qué propiedad debo cambiar para que mi `<input>` tenga un espacio entre el borde y el contenido? ➡ `padding`

La propiedad `padding` sirve para cambiar el espaciado interno, entre el contenido y el borde.

## Lo que aprendimos

- A crear un formulario HTML.
- La *etiqueta* que lo representa es `<form>`.
- La *etiqueta* `<input>`, para la entrada de datos del usuario.
- A crear una *etiqueta* para el input con la *etiqueta* `<label>`.
- A conectar un `input` con su `label`.
- Colocamos un `id` al `input` y asociamos ese `id` al atributo `for` del `label`.
- Algunos tipos de `input`, como `text` y `submit`.
- Qué `label` e `input` por padrón posee el `display inline`.
- A estilizar nuestro formulario.

## Formulario más completo

### <textarea>

Representa un control para la edición multilínea de texto sin formato.

```
<label for="mensaje">Mensaje</label>
<textarea id="mensaje" cols="68" rows="10"></textarea>
```

### <input type="radio">

Colecciones de botones de radio que describen un conjunto de opciones relacionadas.

Solo se puede seleccionar un botón de radio en un grupo dado al mismo tiempo.

```
<div>
  <p>Cómo le gustaría que lo contactemos ?</p>

  <label for="radio-email">Email</label>
  <input type="radio" name="contacto" value="email" id="radio-email">

  <label for="radio-telefono">Teléfono</label>
  <input type="radio" name="contacto" value="telefono" id="radio-telefono">

  <label for="radio-whatsapp">WhatsApp</label>
  <input type="radio" name="contacto" value="whatsapp" id="radio-whatsapp">
</div>
```

- Con `name=""` le indicamos que pertenecen al mismo grupo de `input` (Sino se seleccionan los 3 inputs a la vez)

¿Cuál es la propiedad de un `input` del tipo `radio` que necesito crear para que todos los elementos formen parte del mismo grupo?

name

La propiedad `name` solo se puede “completar” una vez, es por eso que, cuando seleccionamos uno de los ítems, se anula el otro, manteniendo solamente uno seleccionado.

### <input type="checkbox">

Se representan como casillas que se marcan cuando se activan. Se comporta como una casilla de verificación, le permite seleccionar valores únicos para enviar en un formulario (o no).

```
<label><input type="checkbox">Le gustaría recibir novedades de la Barbería Alura ?</label>
```

- Como buena práctica le colocamos dentro del `<label>` al (`<input type="checkbox">`)



Estructura



Diseño visual

Mensaje

Cómo le gustaría que lo contactemos ?

Email

☐

Teléfono

☐

WhatsApp

☐☐

Le gustaría recibir novedades de la Barbería Alura ?

Enviar Formulario

## CSS para inputs más complejos

- Replicar estilos en Css para varios selectores.  
Al separar por la (,) puedo aplicar las mismas propiedades igual que al resto de elementos (**input**).

```
form input, form textarea {
  display: block;
  margin: 0 0 20px;
  padding: 10px 25px;
  width: 50%;
}
```

- Creando la clase (**class="input-padron"**) a los **inputs** correspondientes:

```
<label for="nombreapellido">Nombre y Apellido</label>
<input type="text" id="nombreapellido" class="input-padron">

<label for="correoelectronico">Correo Electrónico</label>
<input type="text" id="correoelectronico" class="input-padron">

<label for="telefono">Teléfono</label>
<input type="text" id="telefono" class="input-padron">

<label for="mensaje">Mensaje</label>
<textarea id="mensaje" cols="68" rows="10" class="input-padron"></textarea>
```

- De esta manera podremos aplicar estilos de manera específica a los **inputs** (**class="input-padron"**).
- ✓ Lo cual nos permitirá aplicar estilos Css a los (**<input type="radio">**) que creamos.

```
form input, form textarea {
  display: block;
  margin: 0 0 20px;
  padding: 10px 25px;
  width: 50%;
}
```

Antes



```
.input-padron{
  display: block;
  margin: 0 0 20px;
  padding: 10px 25px;
  width: 50%;
}
```

Actualmente

EXITOS





Estructura



Diseño visual

Cómo le gustaría que lo contactemos ?

Email

☐

Teléfono

☐

Anteriormente

WhatsApp

☐☐

Le gustaría recibir novedades de la Barbería Alura ?

Enviar Formulario

➤ Redefiniendo (`<input type="radio">`)

```

<div>
  <p>Cómo le gustaría que lo contactemos ?</p>

  <label for="radio-email"><input type="radio" name="contacto" value="email" id="radio-email">Email</label>

  <label for="radio-telefono"><input type="radio" name="contacto" value="telefono" id="radio-telefono">Teléfono</label>

  <label for="radio-whatsapp"><input type="radio" name="contacto" value="whatsapp" id="radio-whatsapp">WhatsApp</label>

</div>

<label><input type="checkbox">Le gustaría recibir novedades de la Barbería Alura ?</label>

```

- Después de haber Redefinido los (`<input type="radio">`):

Cómo le gustaría que lo contactemos ?

☐ Email

☐ Teléfono

☐ WhatsApp

☐ Le gustaría recibir novedades de la Barbería Alura ?

Enviar Formulario

- Replicando estilo Css (`<p>`) al texto "Cómo le gustaría que lo contactemos".

```

form label, form p {
  display: block;
  font-size: 20px;
  margin: 0 0 10px;
}

```

EXITOS





Estructura



Diseño visual

¿Cuál es el mejor enfoque cuando tenemos una configuración de CSS aplicada a un ítem y queremos replicarla en un ítem similar?

- ✓ Crear un clase

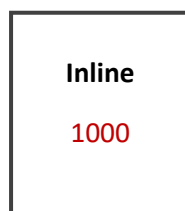
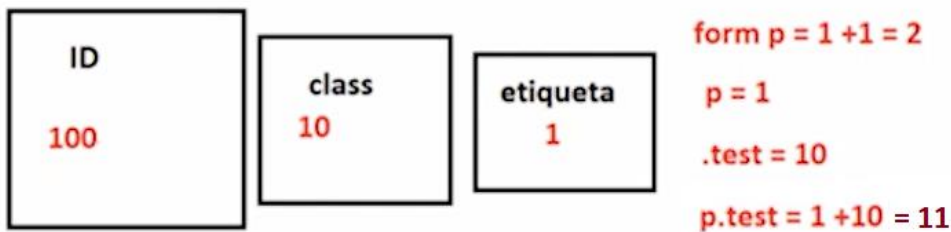
¡Alternativa correcta! Cuando tenemos un estilo que puede ser repetido, una buena estrategia es extraer eso a una clase y usar la clase para aplicar esos estilos en nuestros elementos.

- ✓ Usar selectores de ítems separados por coma

¡Alternativa correcta! Esta es una forma sencilla de hacer que la misma configuración de CSS funcione para los dos ítems, ya sean selectores de elementos, de ids o de clases.

## 👉 Jerarquía en el CSS

- Al aplicar una propiedad en Css:
  - El peso que tiene más valor aplica el estilo Css. "por jerarquía"



- También existe las configuraciones inline "aplica - Html". Que tiene mayor peso

```
<p style="color: green;">Cómo le gustaría que lo contactemos ?</p>
```

```
form p {
  color: blue;
}

p {
  color: red;
}

.test {
  color: yellow;
}

p.test {
  color: orange;
}

#test {
  color: pink;
}
```

## 👉 Seleccionando opciones

<select>

Representa un control que muestra un menú de opciones.

Las opciones contenidas en el menú son representadas por elementos <option>.

```
<div>
  <p>En cuál horario prefiere ser atendido ?</p>

  <select>
    <option>Mañana</option>
    <option>Tarde</option>
    <option>Noche</option>
  </select>
</div>
```



Estructura



Diseño visual

## Selects y options

la *etiqueta* `<option>` . ¿Cuál es su utilidad?

- ✓ Para crear los ítems seleccionables de un `<select>`

¡Alternativa correcta! La estructura de la *etiqueta* `<select>` está compuesta de uno o más `<option>` .

## Lo que aprendimos

- El `textarea` , para entradas de texto de más de una fila.
- El `input` de tipo `radio` .
- Cómo agrupar varios `input` de tipo `radio` , impidiendo que más de un `input` sea seleccionado.
- El `input` del tipo `checkbox` .
- Que podemos crear un `input` dentro de un `label` , asociándolos.
- Más estilos para nuestra página.
- Cómo funciona la jerarquía en CSS .
- El `select` , que es selector, un campo de selección de un ítem, y el `option` , que representa cada opción del selector.



Estructura



Diseño visual

## Inputs para celulares

```
<input type="text">
```

```
<input type="email">
```

```
<input type="tel">
```

```
<input type="number">
```

```
<input type="password">
```

```
<input type="date">
```

```
<input type="datetime">
```

```
<input type="search">
```

En la página: <http://mobileinputtypes.com/>

Nos muestra cómo poner los **inputs** en sus diferentes tipos de formato que tiene.

```

<label for="nombreapellido">Nombre y Apellido</label>
<input type="text" id="nombreapellido" class="input-padron">

<label for="correoelectronico">Correo Electrónico</label>
<input type="email" id="correoelectronico" class="input-padron">

<label for="telefono">Teléfono</label>
<input type="tel" id="telefono" class="input-padron">

```

¿Por qué motivos es importante usar los tipos correctos de `<input>` para los usuarios que acceden a la página vía celular?

- ✓ Para tener datos más correctos en el formulario

¡Alternativa correcta! Es mucho mejor tener datos correctos cuando tenemos nuestro formulario completo.

- ✓ Para facilitar la vida del usuario

¡Alternativa correcta! Es extremadamente importante facilitar la vida de nuestros usuarios siempre.



Estructura



Diseño visual

## Datos importantes en los inputs

### Required (requerido)

Atributo HTML: Indica que el usuario debe especificar un valor para la entrada antes de que se pueda enviar el formulario.

### Placeholder

Atributo HTML: Especifica una breve sugerencia que describe el valor esperado de un campo de entrada.

```

<label for="nombreapellido">Nombre y Apellido</label>
<input type="text" id="nombreapellido" class="input-padron" required>

<label for="correoelectronico">Correo Electrónico</label>
<input type="email" id="correoelectronico" class="input-padron" required placeholder="email@dominio.com">

<label for="telefono">Teléfono</label>
<input type="tel" id="telefono" class="input-padron" required placeholder="(xx) xxxx xxxx">

<label for="mensaje">Mensaje</label>
<textarea id="mensaje" cols="68" rows="10" class="input-padron" required></textarea>

```

### Checked

Atributo HTML: puede ser utilizado con:

- ☞ `<input type = "checkbox">` para indicar que la casilla está marcada, o con
- ☞ `<input type = "radio">` para indicar que el botón está activado.

```

<input type="radio" name="contacto" value="whatsapp" id="radio-whatsapp" checked> WhatsApp

```

```

<input type="checkbox" class="checkbox" checked> ¿Le gustaría recibir novedades de la Barbería Alura?

```

## ¿Cómo convertimos un campo común en un campo obligatorio?

- ✓ Agregando la propiedad `required`

¡Alternativa correcta! Cuando agregamos la propiedad `required`, el navegador nos ayuda a validar si ese campo está rellenado.



Estructura



Diseño visual

## 👉 Mejorando la semántica del formulario

**<fieldset>** (grupo de campos)

Permite organizar en grupos los campos de un formulario. (los grupos hacen referencia al mismo grupo)

Nota: no usa la etiqueta **<p>** Si no la etiqueta (**<legend>**)

```

<fieldset>
  <p>¿En cuál horario prefiere ser atendido?</p>
  <select>
    <option>Mañana</option>
    <option>Tarde</option>
    <option>Noche</option>
  </select>
</fieldset>

```

```

<fieldset>
  <legend>¿En cuál horario prefiere ser atendido?</legend>
  <select>
    <option>Mañana</option>
    <option>Tarde</option>
    <option>Noche</option>
  </select>
</fieldset>

```

¿Qué *etiqueta* usamos para marcar un título de un grupo de campos en el formulario?

✓ **<legend>**

¡Alternativa correcta! Es la *etiqueta* **<legend>** que usamos para un título de un grupo de campos en cualquier formulario.

- En una imagen  el atributo Html.

### Alt

Especifica un texto alternativo para una imagen, si la imagen **no se puede mostrar**.

**alt="Logo de la Barbería Alura"**

## Lo que aprendimos

- Algunos tipos de *inputs* para el celular: `email`, `tel`, `number`, `password`, `date`, `datetime`, `month` y `search`.
- Cómo configurar campos obligatorios, o sea, exigir al usuario para que algunos campos sean completados, a través del atributo `required`.
- Cómo mostrar una sugerencia para completar los campos, a través del atributo `placeholder`.
- Cómo dejar una opción marcada por padrón en nuestros `input` `radio` y `checkbox` a través del atributo `checked`.
- Cómo estructurar mejor nuestro código con `fieldset` y `legend`.
- Cómo agregar una alternativa a la imagen, describiéndola, con el atributo `alt`.



## 👉 Qué son las transiciones

```
<input type="submit" value="Enviar Formulario" class="enviar">
```

➤ Al **input** "submit" creamos la **clase** (enviar) para aplicar las propiedades Css,

- ❖ **Padding.** – espaciamiento interno.
- ❖ **Font-size.** – tamaño de texto.
- ❖ **Font-wieght: bold.** – poner en negrita "texto".
- ❖ **Color.** – aplica al texto.
- ❖ **Background.** – color fondo al elemento "submit"
- ❖ **Border: none.** – quitar el color "contorno-bordes".
- ❖ **Border-radius.** – poner bordes redondeados.
- ❖ **Transition: 1s.** – proporciona una transición visual de un elemento.  
(trabaja con **Background-color:**)

```
.enviar {
  width: 40%;
  padding: 15px 0;
  font-size: 18px;
  font-weight: bold;
  color: white;
  background: orange;
  border: none;
  border-radius: 5px;
  transition: 1s background;
  cursor: pointer;
}
```

- ❖ **Cursor.** – cambia el cursor del mouse.  
- ❖ **Background-color:** - cambia el color del boton "submit"

```
.enviar:hover {
  background-color: darkorange;
}
```



## 👉 Entendiendo Transformaciones

❖ Cambiamos: `transition: 1s background;`

Por: `transition: 1s all;`

la transición aplicara todos los elementos

- ❖ **transform: scale(1.20);**  
Aumenta un elemento proporcionalmente. (submit)  
Analizando No es necesario:  
**rotate(35deg)** Gira (35 grados) el submit.

```
.enviar:hover {
  background-color: darkorange;
  transform: scale(1.20) rotate(35deg);
}
```



Estructura



Diseño visual

## Lo que aprendimos

- Cómo colocar estilos al botón de envío de formulario.
- A realizar transiciones en nuestros elementos, con la propiedad CSS `transition`.
- A modificar el estilo del puntero del mouse, al pasar sobre un determinado elemento, a través de la propiedad CSS `cursor`.
- A realizar transformaciones en nuestros elementos, como aumentar proporcionalmente la escala de determinado elemento o rotarlo, a través de la propiedad CSS `transform`.

### 👉 Tabla básica

#### `<table>`

Utilizamos la *etiqueta* `<table>` para crear una tabla.

Consta de celdas de tabla dentro de filas y columnas.



Utilizamos la *etiqueta* `<tr>` para marcar la fila de una tabla.



Utilizamos la *etiqueta* `<td>` para marcar una columna de una tabla.

```

<table>
  <tr>
    <td>Día</td>
    <td>Horario</td>
  </tr>
  <tr>
    <td>Lunes</td>
    <td>08h ~ 20h</td>
  </tr>
  <tr>
    <td>Miércoles</td>
    <td>08h ~ 20h</td>
  </tr>
  <tr>
    <td>Viernes</td>
    <td>08h ~ 20h</td>
  </tr>

```

Día	Horario
Lunes	08h ~ 20h
Miércoles	08h ~ 20h
Viernes	08h ~ 20h





Estructura



Diseño visual

## 🔗 Etiquetas semánticas para tablas

- ❖ Para estructurar mejor nuestra tabla “de una forma más semántica”
  - Utilizamos (<thead>) para organizar “mejor” el encabezado de la tabla.
  - Además, cambiamos (<td>) por (<th>) -> Observamos (Centra “Día y Horario”)

```
<thead>
  <tr>
    <th>Día</th>
    <th>Horario</th>
  </tr>
</thead>
```

Día	Horario
Lunes	08h ~ 20h
Miércoles	08h ~ 20h
Viernes	08h ~ 20h

- Utilizamos (<tbody>) para organizar “mejor” el cuerpo de la tabla.

```
<tbody>
  <tr>
    <td>Lunes</td>
    <td>08h ~ 20h</td>
  </tr>
  <tr>
    <td>Miércoles</td>
    <td>08h ~ 20h</td>
  </tr>
  <tr>
    <td>Viernes</td>
    <td>08h ~ 20h</td>
  </tr>
</tbody>
```

Día	Horario
Lunes	08h ~ 20h
Miércoles	08h ~ 20h
Viernes	08h ~ 20h

## 🔗 Estilos en las tablas

- Aplicando estilos

```
table {
  margin: 40px 40px;
}

thead {
  background: #555555;
  color: white;
  font-weight: bold;
}
```

Día	Horario
Lunes	08h ~ 20h
Miércoles	08h ~ 20h
Viernes	08h ~ 20h

- Dibujar “líneas” de la tabla.

```
td, th {
  border: 1px solid #000000;
}
```

Día	Horario
Lunes	08h ~ 20h
Miércoles	08h ~ 20h
Viernes	08h ~ 20h

- Padding “espaciamiento interno” de la tabla.

```
td, th {
  border: 1px solid #000000;
  padding: 8px 15px;
}
```

Día	Horario
Lunes	08h ~ 20h
Miércoles	08h ~ 20h
Viernes	08h ~ 20h



Estructura



Diseño visual

## Lo que aprendimos

- A crear una tabla HTML.
  - La *etiqueta* `table` , representa la tabla.
  - La *etiqueta* `tr` , representa la fila de la tabla.
  - La *etiqueta* `td` , representa la celda de la tabla.
  - La *etiqueta* `thead` , representa el encabezado de la tabla.
  - La *etiqueta* `tbody` , representa el cuerpo de la tabla.
  - La *etiqueta* `th` , representa la celda del encabezado de la tabla.
  - La *etiqueta* `tfoot` , representa el pie de página de la tabla.
- A colocar los estilos respectivos a nuestra tabla.