

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

La empresa de invernaderos **"Atemperar"** desea contratarnos para crear una aplicación que permita convertir la temperatura de grados Fahrenheit a Celsius para determinar si la temperatura del invernadero es óptima, el sensor de temperatura del termostato que se encarga de medir dicha temperatura, solo entrega datos en grados Fahrenheit.

Etapas 1: Requerimientos

Quién Contrata: Empresa de Invernaderos Atemperar.	Qué necesita: Software de conversión de temperatura Fahrenheit - Celsius
Para qué lo necesita: Para convertir la temperatura ambiente en grados Fahrenheit a grados Celsius y así determinar qué acción realizar.	Dispositivos de Despliegue: la aplicación deberá correr y verse correctamente en diferentes navegadores (Chrome, Edge, Mozilla) y adaptarse a pantallas de computadores de escritorio, portátiles y celulares.
Capital necesario: como se trabajará con herramientas ya adquiridas el despliegue se realizará localmente hasta obtener la aprobación final del cliente, no se requiere de capital alguno.	Número de Personas: Por ser un proyecto pequeño solo se requiere de un desarrollador.
Tiempo de Entrega: El tiempo estimado inicialmente para desarrollar el trabajo y entregar el M.V.P. al cliente es de 3 Clases (2 clases x 1,5)	

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

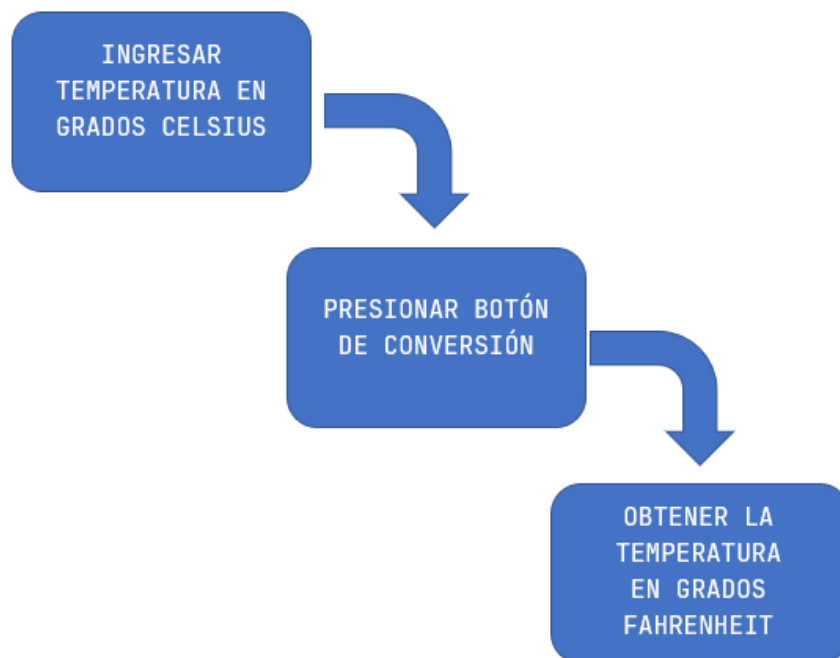
Etapa 2: Stack Tecnológico y Herramientas de desarrollo

Al tratarse de una aplicación web, se utilizarán las siguientes tecnologías de desarrollo:

HTML para crear la estructura
CSS para aplicar estilos gráficos
JAVASCRIPT para generar la interactividad
VISUAL STUDIO CODE: Editor de código
BROWSER O NAVEGADOR: para desplegar la app y visualizar cambios
DISPOSITIVOS FISICOS VARIOS: Para verificar el funcionamiento, la usabilidad y la adaptación a diferentes tamaños de pantallas.

Etapa 3: UX (Experiencia de Usuario)

Se muestra el flujo de pasos que realizará normalmente el usuario final.



PROYECTO 2: CONVERSIÓN DE TEMPERATURA

Etapa 4: UI (Interfaz de Usuario)

Se muestra un modelo gráfico aproximado del diseño de las pantallas del software.

The image displays two mockups of the user interface for a temperature conversion application. Both mockups feature the title 'Fahrenheit-Celsius' in red text with a flame icon. The top mockup shows an input field containing the number '100' and a 'Convertir' button. The bottom mockup shows an empty input field, the 'Convertir' button, and a green box at the bottom displaying the result '37.8 ° Celsius'.

Etapa 5: Tablero Kanban (Asignación de tareas)

Creamos en Trello (aplicación) un Tablero para registrar las tareas a realizar durante la ejecución del proyecto.

El tablero va a constar de las siguientes listas (columnas):

TAREAS:

Registramos todas las tareas en las que dividimos el proyecto y que están en espera de ser ejecutadas.

EN PROCESO:

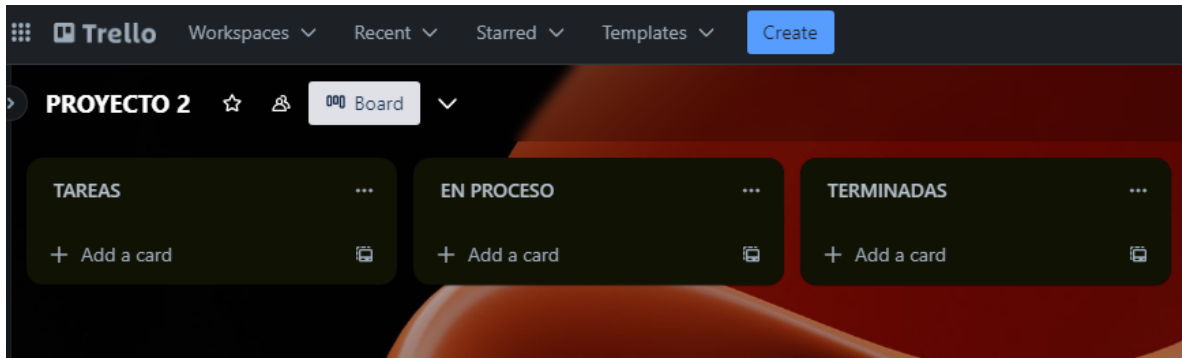
Trasladamos a esta columna las tareas que estemos ejecutando y que aún no hayan sido terminadas

TERMINADAS:

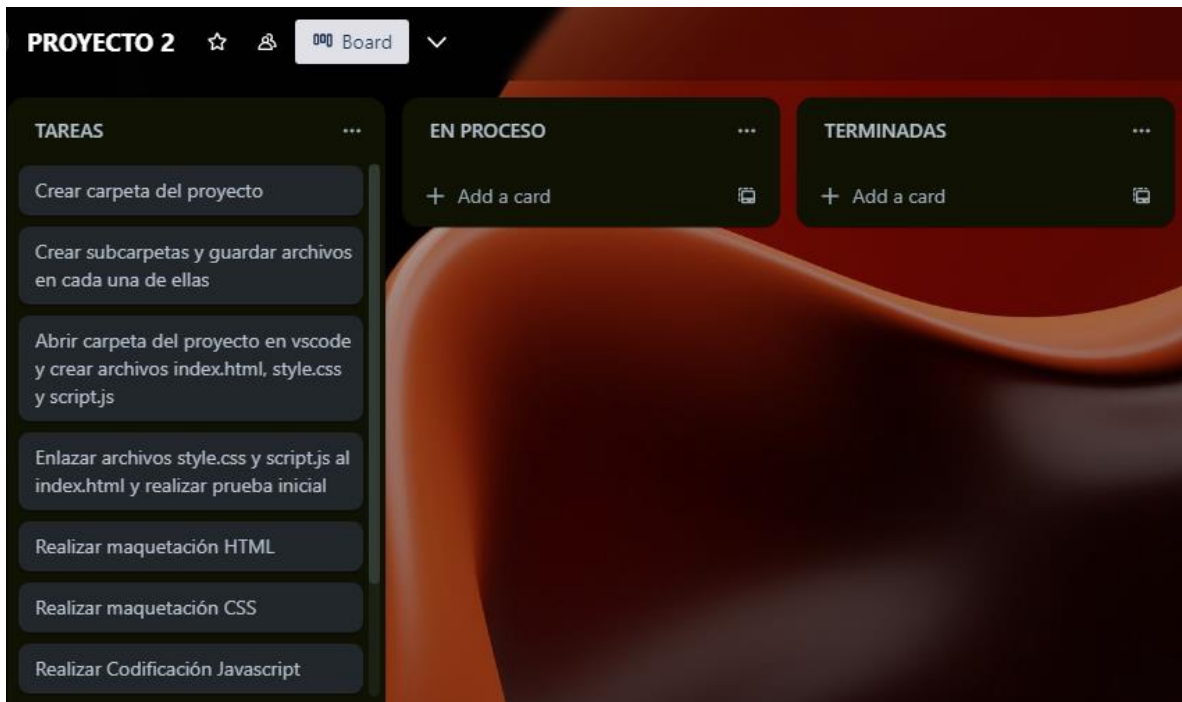
Aquí reunimos las tareas que se han terminado completamente y que liberan espacio para empezar una nueva.

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

nuestro tablero Kanban recién creado debe lucir como se muestra en la siguiente imagen.



agregaremos las tareas que vamos a ejecutar en la lista de tareas.



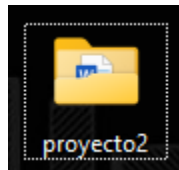
PROYECTO 2: CONVERSIÓN DE TEMPERATURA

LISTADO DE TAREAS

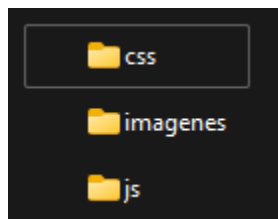
1. Crear carpeta del proyecto
2. Crear subcarpetas y guardar archivos en cada una de ellas
3. Abrir carpeta del proyecto en vscode y crear archivos index.html, style.css y script.js
4. Crear estructura HTML básica
5. Enlazar archivos style.css y script.js al index.html y realizar prueba inicial
6. Agregar el favicon
7. Realizar maquetación HTML
8. Realizar maquetación CSS
9. Realizar codificación JavaScript
10. Realizar pruebas de funcionamiento
11. Presentar el M.V.P.

Etapas 6 - Ejecución de Tareas:

1. Crear carpeta del proyecto.

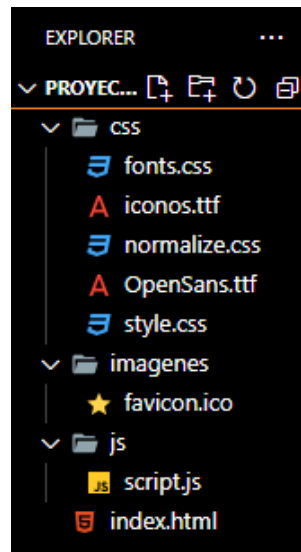


2. Crear subcarpetas



PROYECTO 2: CONVERSIÓN DE TEMPERATURA

3. Crear archivos y guardarlos en sus respectivas carpetas



4. Crear estructura HTML básica

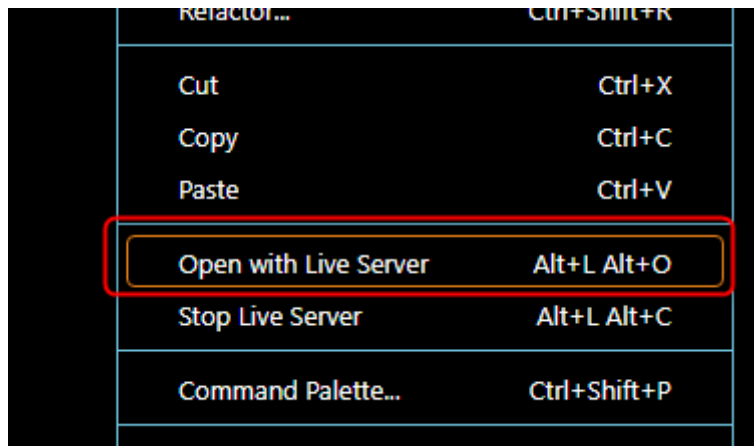
```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Conversión de Temperatura</title>
7 </head>
8 <body>
9
10 </body>
11 </html>
```

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

5. Enlazar archivos .css y script.js al index.html y realizar pruebas.

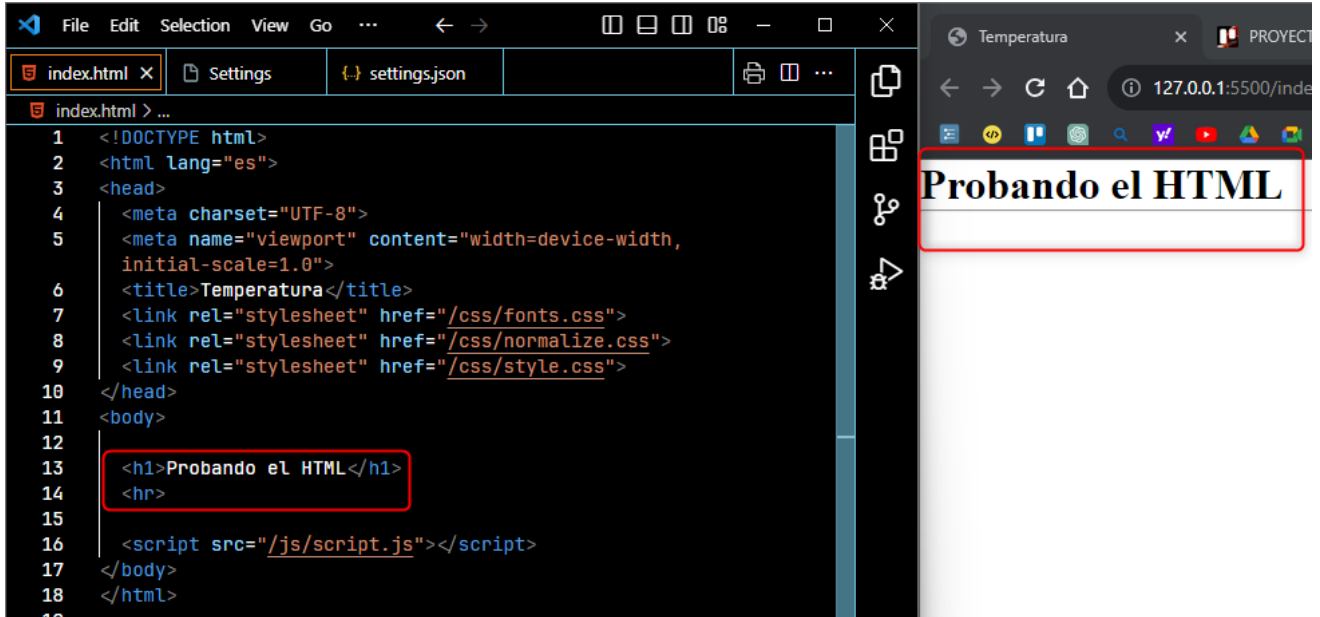
```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Conversión de Temperatura</title>
7   <link rel="stylesheet" href="/css/fonts.css">
8   <link rel="stylesheet" href="/css/normalize.css">
9   <link rel="stylesheet" href="/css/style.css">
10 </head>
11 <body>
12
13   <script src="/js/script.js"></script>
14 </body>
15 </html>
16
17
```

Ejecutamos el live server haciendo click derecho dentro de archivo index.html en cualquier espacio vacío.

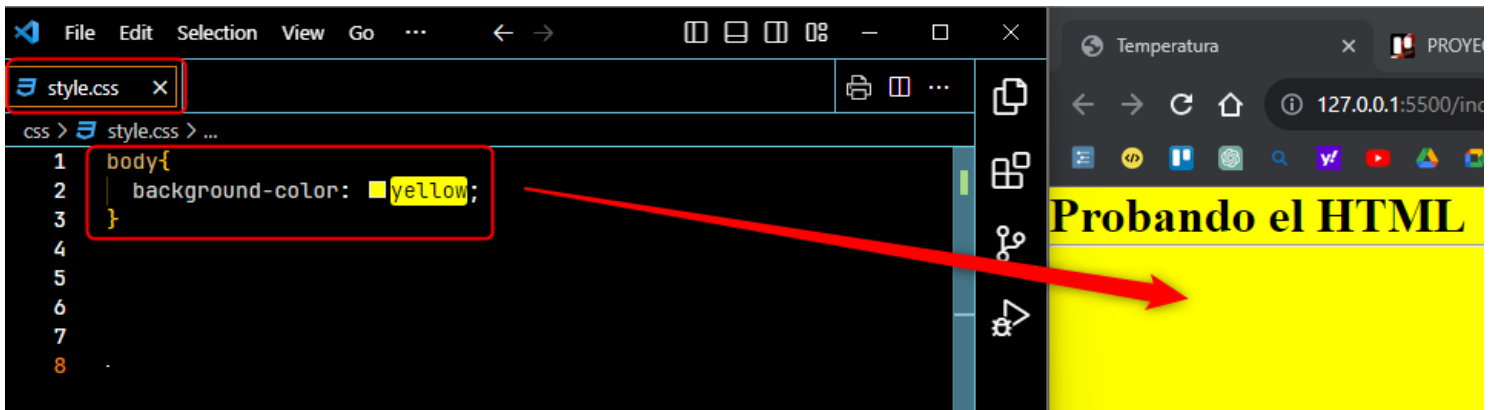


PROYECTO 2: CONVERSIÓN DE TEMPERATURA

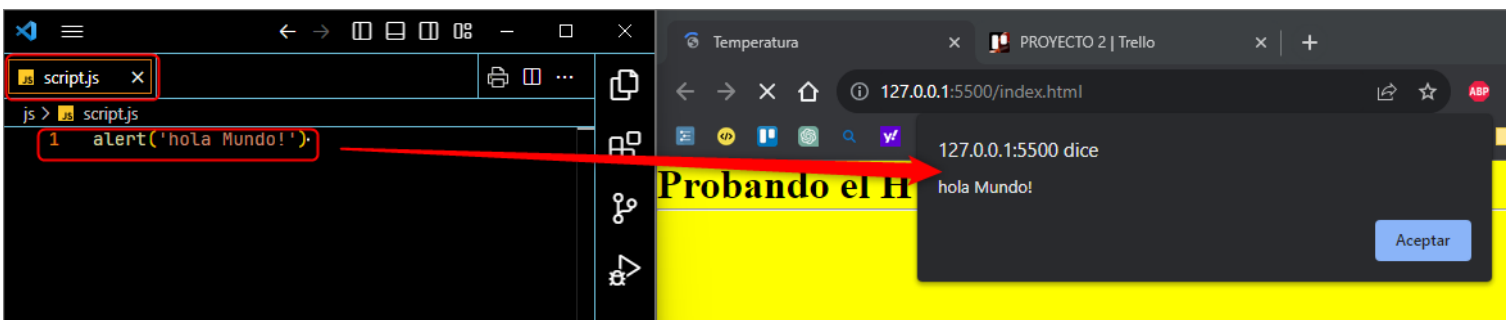
Dividimos la pantalla para colocar de un lado vscode del otro el navegador abierto por live server.



Ahora probaremos el enlace con `style.css` agregando un fondo amarillo.



Finalmente probamos el enlace con el `script.js`



PROYECTO 2: CONVERSIÓN DE TEMPERATURA

6. Agregar Favicon

En el archivo *index.html* en la sección del head agregamos el link para mostrar un icono en la pestaña del navegador

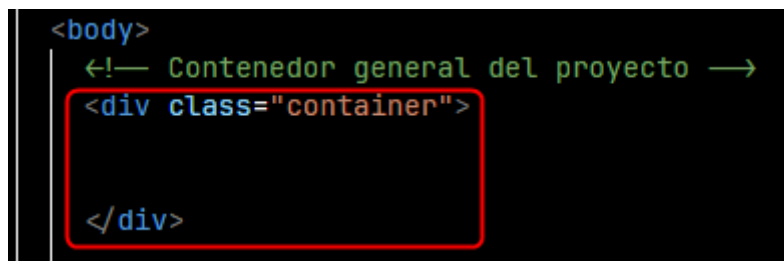


7. Maquetación HTML

Vamos a crear la estructura de nuestra interfaz de usuario.

Primero que todo eliminamos el código que habíamos digitado para pruebas de los 3 archivos para empezar desde cero.

Creamos el contenedor general de la app (container) y le agregamos la clase (class) container.

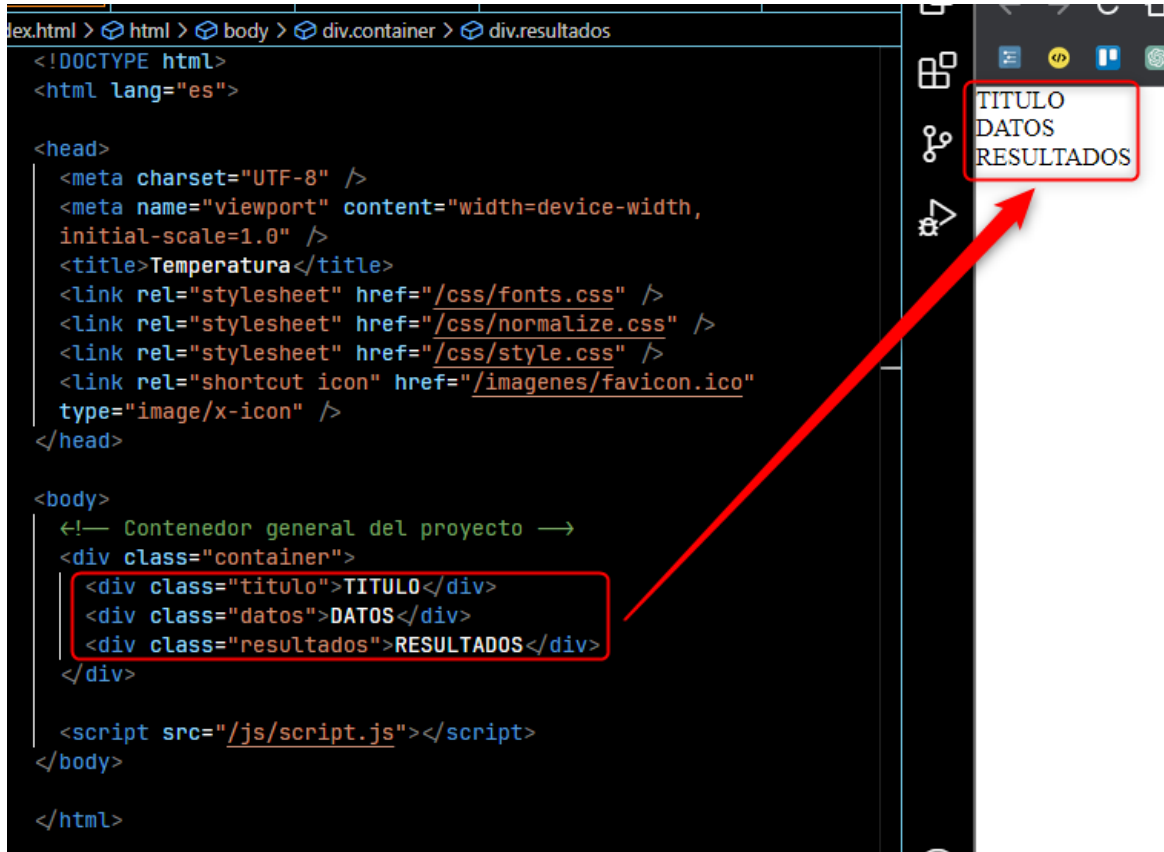


Dentro del contenedor general agregaremos contenedores para

- Título
- Datos
- Resultados

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

A su vez agregamos un texto descriptivo para ir ubicándolos en el navegador



```
<!DOCTYPE html>
<html lang="es">

<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width,
  initial-scale=1.0" />
  <title>Temperatura</title>
  <link rel="stylesheet" href="/css/fonts.css" />
  <link rel="stylesheet" href="/css/normalize.css" />
  <link rel="stylesheet" href="/css/style.css" />
  <link rel="shortcut icon" href="/imagenes/favicon.ico"
  type="image/x-icon" />
</head>

<body>
  <!-- Contenedor general del proyecto -->
  <div class="container">
    <div class="titulo">TITULO</div>
    <div class="datos">DATOS</div>
    <div class="resultados">RESULTADOS</div>
  </div>

  <script src="/js/script.js"></script>
</body>

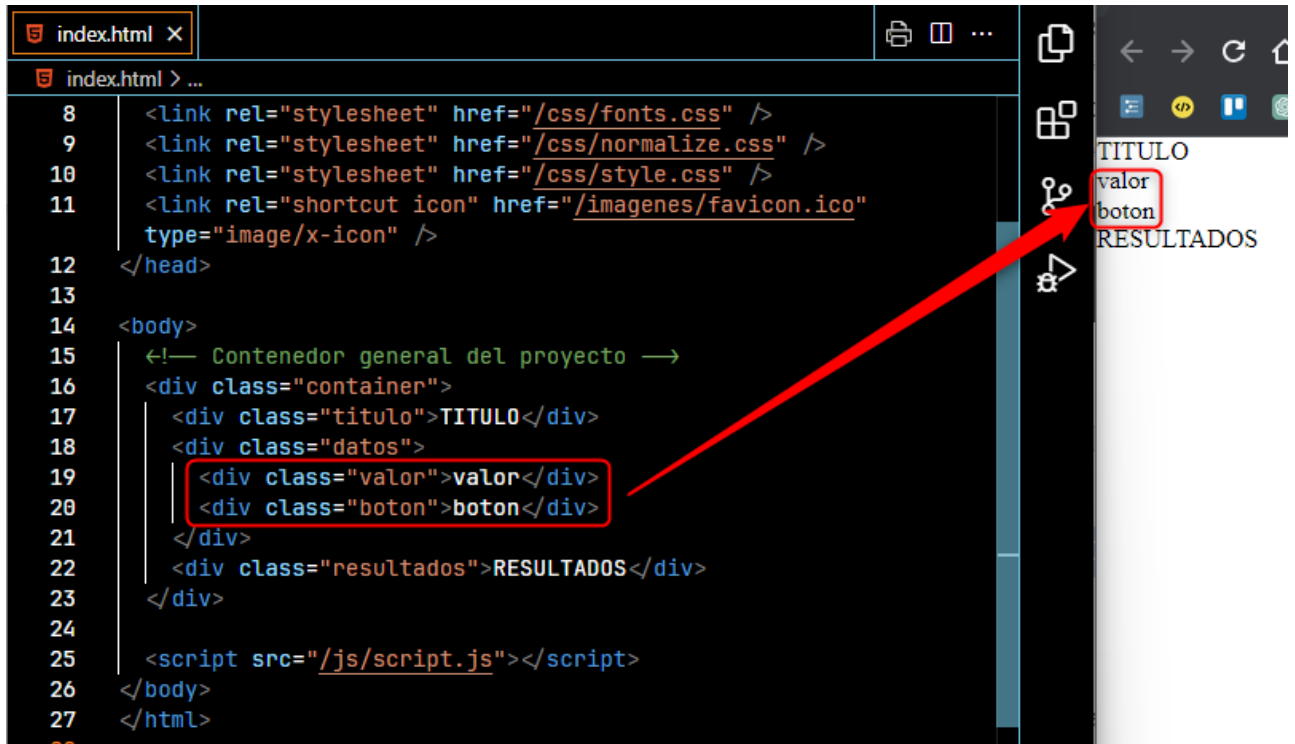
</html>
```

Dentro del contenedor de datos deben ir dos contenedores más:

- Valor
- boton

Eliminamos el texto inicial (DATOS) y colocamos textos para los nuevos contenedores (valor y boton) en minúsculas para diferenciarlos de los primeros, agregamos sus respectivas clases.

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

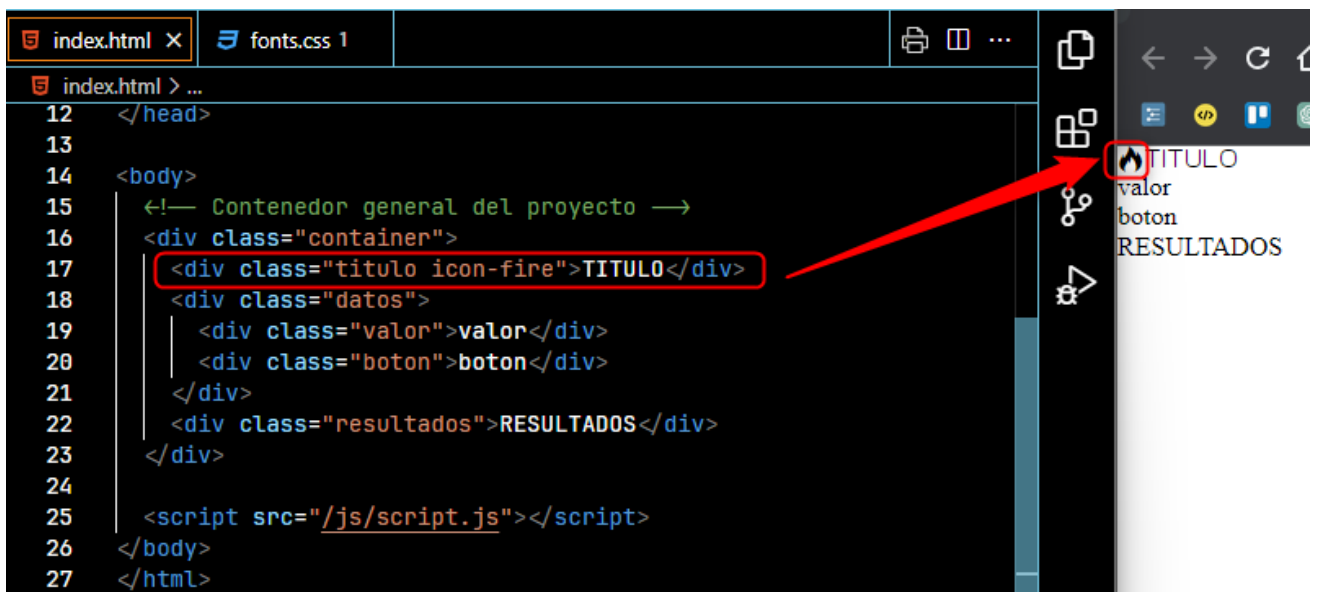


```
index.html X
index.html > ...
8 <link rel="stylesheet" href="/css/fonts.css" />
9 <link rel="stylesheet" href="/css/normalize.css" />
10 <link rel="stylesheet" href="/css/style.css" />
11 <link rel="shortcut icon" href="/imagenes/favicon.ico"
    type="image/x-icon" />
12 </head>
13
14 <body>
15   <!-- Contenedor general del proyecto -->
16   <div class="container">
17     <div class="titulo">TITULO</div>
18     <div class="datos">
19       <div class="valor">valor</div>
20       <div class="boton">boton</div>
21     </div>
22     <div class="resultados">RESULTADOS</div>
23   </div>
24
25   <script src="/js/script.js"></script>
26 </body>
27 </html>
```

DOM Tree:

- TITULO
- valor
- boton
- RESULTADOS

Ahora vamos a agregar el icono de la llama al lado del título, para ello colocaremos la clase **icon-fire** que se encargará de mostrar una llama (icono) gracias a los archivos **fonts.css** e **iconos.ttf**



```
index.html X  fonts.css 1
index.html > ...
12 </head>
13
14 <body>
15   <!-- Contenedor general del proyecto -->
16   <div class="container">
17     <div class="titulo icon-fire">TITULO</div>
18     <div class="datos">
19       <div class="valor">valor</div>
20       <div class="boton">boton</div>
21     </div>
22     <div class="resultados">RESULTADOS</div>
23   </div>
24
25   <script src="/js/script.js"></script>
26 </body>
27 </html>
```

DOM Tree:

- TITULO (with flame icon)
- valor
- boton
- RESULTADOS

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

Ahora vamos a eliminar la palabra valor de su respectivo contenedor y agregaremos un cuadro de texto tipo numérico, el cual contendrá la temperatura ingresada por el usuario.

Le agregaremos la clase y el id **valorTemp**.



```
12 </head>
13
14 <body>
15   <!-- Contenedor general del proyecto -->
16   <div class="container">
17     <div class="titulo icon-fire">TITULO</div>
18     <div class="datos">
19       <div class="valor">
20         <input type="number" class="valorTemp" id="valorTemp">
21       </div>
22       <div class="boton">boton</div>
23     </div>
24     <div class="resultados">RESULTADOS</div>
25   </div>
26
27   <script src="/js/script.js"></script>
28 </body>
29 </html>
```

De igual forma vamos a eliminar la palabra boton y en su lugar agregaremos un boton con la clase y el id **btnConvertir**



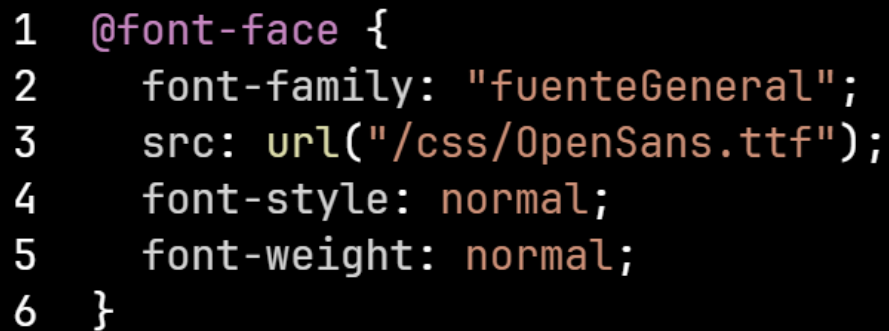
```
<!-- Contenedor general del proyecto -->
<div class="container">
  <div class="titulo icon-fire">TITULO</div>
  <div class="datos">
    <div class="valor">
      <input type="number" class="valorTemp" id="valorTemp">
    </div>
    <div class="boton">
      <input type="button" value="Convertir" class="btnConvertir" id="btnConvertir">
    </div>
  </div>
  <div class="resultados">RESULTADOS</div>
</div>

<script src="/js/script.js"></script>
</body>
</html>
```

Con esto hemos terminado la maquetación HTML.

8. Maquetación CSS

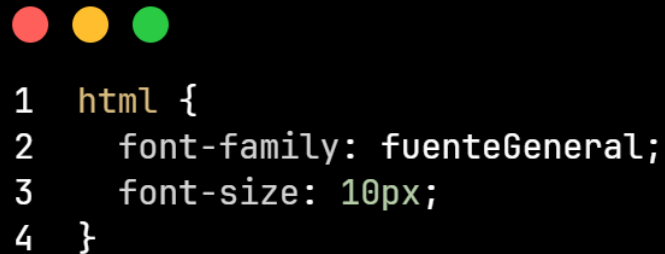
Fuente General



```
1  @font-face {  
2    font-family: "fuenteGeneral";  
3    src: url("/css/OpenSans.ttf");  
4    font-style: normal;  
5    font-weight: normal;  
6  }
```

- Asigno nombre a la fuente
- Indico ruta donde se encuentra la fuente
- Estilo normal (sin negrita, itálica, etc.)
- Peso de la fuente, grosor, intensidad

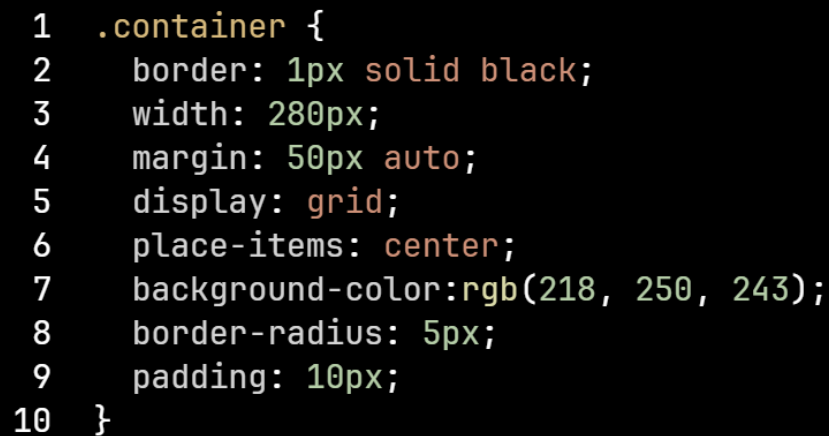
Estilos Generales de la página



```
1  html {  
2    font-family: fuenteGeneral;  
3    font-size: 10px;  
4  }
```

- Agrego la fuente a todo el documento
- Indico el tamaño base para todas las fuentes

Contenedor General (.container)

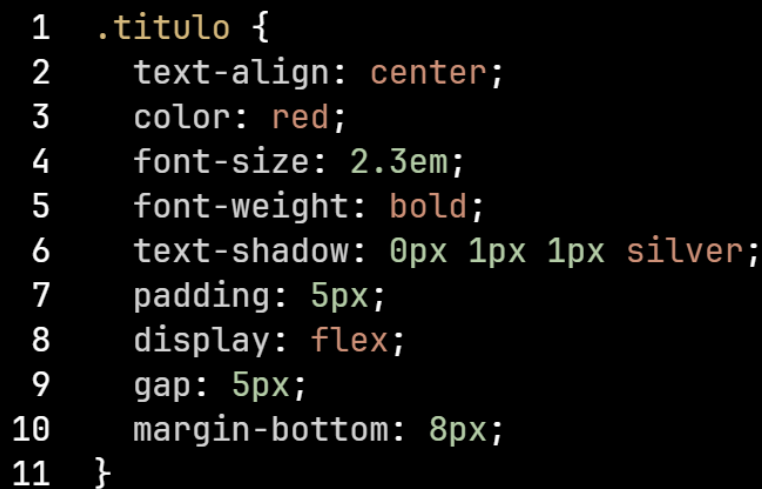


```
1  .container {  
2    border: 1px solid black;  
3    width: 280px;  
4    margin: 50px auto;  
5    display: grid;  
6    place-items: center;  
7    background-color:rgb(218, 250, 243);  
8    border-radius: 5px;  
9    padding: 10px;  
10 }
```

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

- Agregamos borde para visualizar el área del contenedor
- Se establece el ancho máximo del contenedor
- Se ubica 50px hacia abajo y se centra horizontalmente
- Se centran los elementos internos del contenedor (titulo, datos y resultados)
- Se coloca color de fondo
- Se redondean las esquinas
- Se agrega margen interno

Título (.titulo)



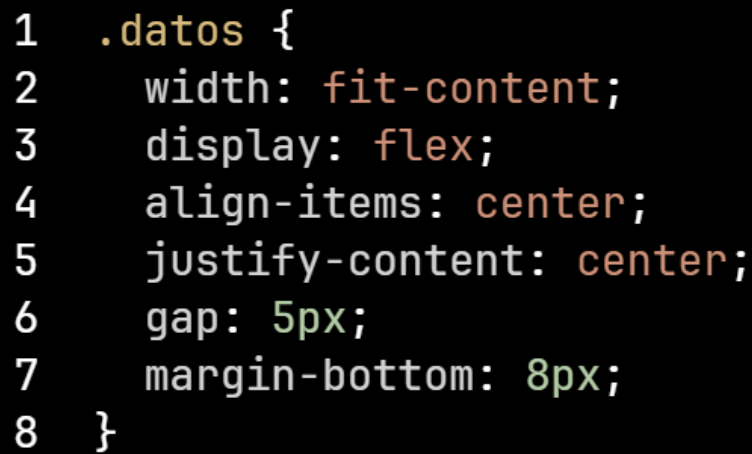
```
1  .titulo {  
2    text-align: center;  
3    color: red;  
4    font-size: 2.3em;  
5    font-weight: bold;  
6    text-shadow: 0px 1px 1px silver;  
7    padding: 5px;  
8    display: flex;  
9    gap: 5px;  
10   margin-bottom: 8px;  
11 }
```

- Se alinea el texto y el icono al centro
- Se les coloca de color rojo
- Se agranda el tamaño del texto y del icono que también es texto
- Se pone efecto de negrita
- Se pone efecto tridimensional al texto y al icono
- Se coloca borde interno

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

- Se coloca un espaciado entre el icono y el título
- Se coloca un espacio debajo del título para separarlo de los datos

Datos



```
1  .datos {  
2    width: fit-content;  
3    display: flex;  
4    align-items: center;  
5    justify-content: center;  
6    gap: 5px;  
7    margin-bottom: 8px;  
8  }
```

- Ajusto el ancho del contenedor al contenido
- Centro todos los elementos internos
- Separo los elementos internos
- Coloco espacio debajo del contenedor para separarlo del siguiente contenedor.

Cuadro de Ingreso de valor de temperatura

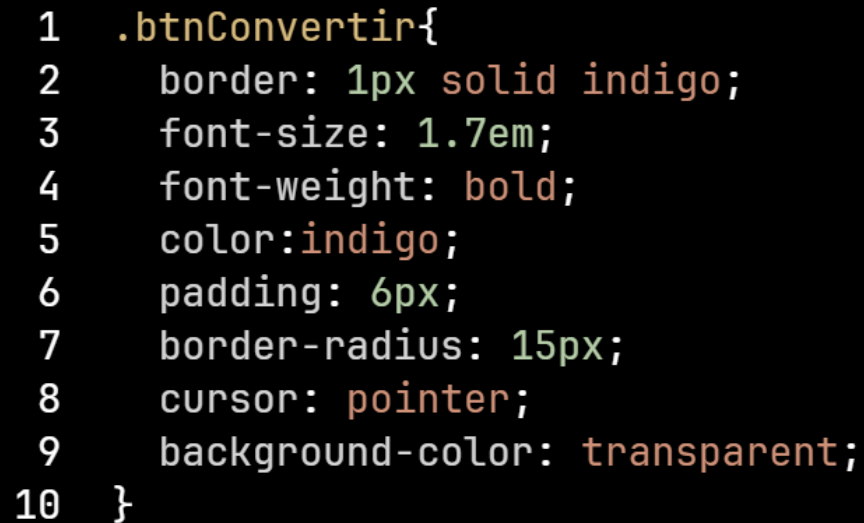
```
1  .valorTemp {
2    border: 1px solid indigo;
3    outline: none;
4    width: 7em;
5    font-size: 1.7em;
6    font-weight: 600;
7    padding: 6px;
8    text-align: center;
9    border-radius: 5px;
10   background: transparent;
11 }
```

- Coloco borde
- Elimino borde resaltado al ser seleccionado
- Asigno un ancho adecuado para escribir datos
- Coloco tamaño del texto
- Aplico espaciado interno
- Alineo el texto al centro
- Redondeo las esquinas del cuadro de texto
- Hago transparente el fondo del cuadro

Eliminar flechas de aumento y decremento

```
1  .valorTemp::-webkit-inner-spin-button,
2  .valorTemp::-webkit-outer-spin-button {
3    -webkit-appearance: none;
4  }
```

Boton de Conversión



```
1  .btnConvertir{
2    border: 1px solid indigo;
3    font-size: 1.7em;
4    font-weight: bold;
5    color:indigo;
6    padding: 6px;
7    border-radius: 15px;
8    cursor: pointer;
9    background-color: transparent;
10 }
```

- Coloco borde
- Coloco tamaño de texto
- Asigno peso o intensidad del texto
- Color de texto
- Coloco espacio interno para agrandar botón
- Redondeo esquinas del botón
- Pongo cursor de manito al pasar el mouse por encima
- Quito fondo para que quede transparente

Resultados



```
1  .resultados{
2    width: 82%;
3    padding: 10px;
4    text-align: center;
5    background: green;
6    font-size: 1.8em;
7    font-weight: bold;
8    border-radius: 5px;
9  }
```

- Aplico ancho del contenedor
- Coloco espacio interior para separar de los bordes
- Centro el texto que se vaya a mostrar
- Aplico color de fondo
- Tamaño de texto
- Intensidad del texto
- Redondeo bordes

9. Codificación JavaScript



```
1 // Creo referencias a los objetos del DOM
2 const txtTemperatura = document.getElementById("valorTemp");
3 const btnConvertir = document.getElementById("btnConvertir");
4 const cuadroResultado = document.getElementById("resultados");
```



```
1 // Hago pruebas de cada elemento
2 txtTemperatura.value=100;
3
4 btnConvertir.addEventListener('click',()=>{
5     alert('me has presionado');
6 });
7
8 cuadroResultado.innerHTML='Este es el resultado';
9
```



```
1 // oculto cuadro de respuestas
2 cuadroResultado.style.display = 'none';
```

PROYECTO 2: CONVERSIÓN DE TEMPERATURA



```
1 // asigno enter al boton convertir
2 document.addEventListener('keypress', (event) => {
3   if (event.key === 'Enter') {
4     btnConvertir.click();
5   }
6 });
```



```
1 // Ejecuto instrucciones al hacer click en el boton convertir
2 btnConvertir.addEventListener('click', () => {
3   const gradosFahrenheit = txtTemperatura.value;
4   if (gradosFahrenheit !== '') {
5     const gradosCelsius = ((gradosFahrenheit - 32) * (5 / 9)).toFixed(2);
6     cuadroResultado.innerHTML = `${gradosCelsius} ${String.fromCharCode(176)}C`;
7     // cuadroResultado.style.display = 'block';
8     txtTemperatura.value = ' ';
9   } else {
10    alert('Ingresa un valor válido!');
11  }
12 });
```

10. Pruebas de Funcionamiento

DATO INGRESADO	RESULTADO ESPERADO
VACIO	MENSAJE DE ERROR
0	-17.78 °C
20	-6.67 °C
50	10.00 °C
100	37.78 °C
250	121.11 °C
360	182.22 °C

11. Refactorización Responsiva (Desktop first)

Se realizan los siguientes cambios:

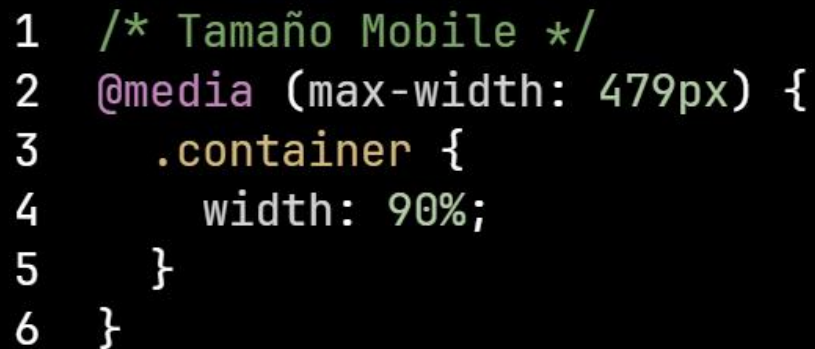
- Se cambian rutas absolutas de los links de HTML a rutas relativas para evitar errores de visualización en local.
- Modificamos el tamaño del contenedor general y agregamos fondo amarillo para pruebas

```

1  /* Tamaño Mobile */
2  @media (max-width: 479px) {
3      body{
4          background-color: yellow;
5      }
6
7      .container {
8          width: 90%;
9      }
10
11 }
```

PROYECTO 2: CONVERSIÓN DE TEMPERATURA

Una vez comprobados los cambios, eliminamos el fondo amarillo y solo dejamos el cambio del tamaño el contenedor.



```
1  /* Tamaño Mobile */
2  @media (max-width: 479px) {
3      .container {
4          width: 90%;
5      }
6  }
```

Con esto hemos terminado nuestro MVP y ya se puede visualizar mejor tanto en computadoras de escritorio como en dispositivos móviles.