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.

**Etapa 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)** | |

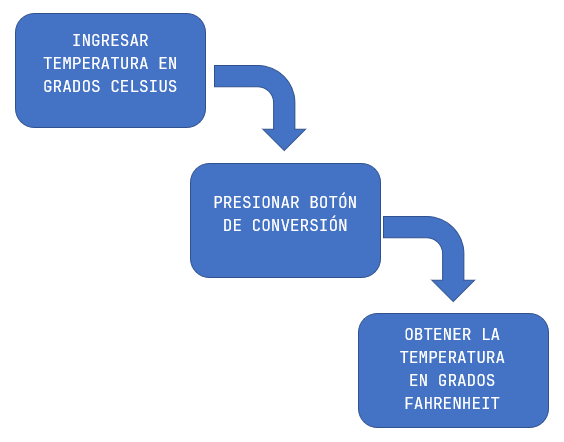
**Etapa 2: Stack Tecnológico y Herramientas de desarrollo**

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. |

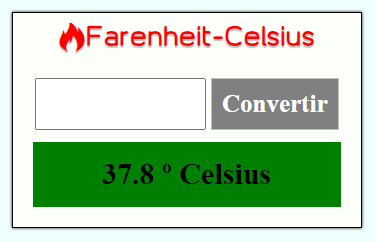
**Etapa 3: UX (Experiencia de Usuario)**

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

****

**Etapa 4: UI (Interfaz de Usuario)**

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

****

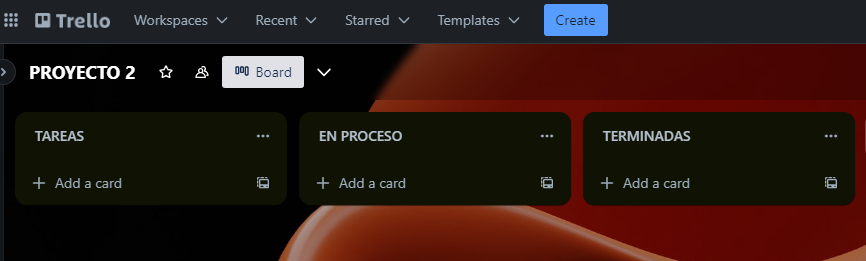
**Etapa 5: Tablero Kanban (Asignación de tareas)**

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. |

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.



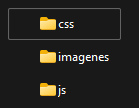
|  |
| --- |
| LISTADO DE TAREAS |
| 1. Crear carpeta del proyecto |
| 1. Crear subcarpetas y guardar archivos en cada una de ellas |
| 1. Abrir carpeta del proyecto en vscode y crear archivos index.html, style.css y script.js |
| 1. Crear estructura HTML básica |
| 1. Enlazar archivos style.css y script.js al index.html y realizar prueba inicial |
| 1. Agregar el favicon |
| 1. Realizar maquetación HTML |
| 1. Realizar maquetación CSS |
| 1. Realizar codificación JavaScript |
| 1. Realizar pruebas de funcionamiento |
| 1. Presentar el M.V.P. |

**Etapa 6 - Ejecución de Tareas:**

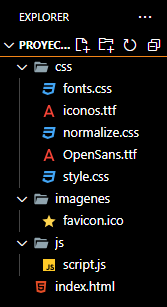
1. **Crear carpeta del proyecto.**



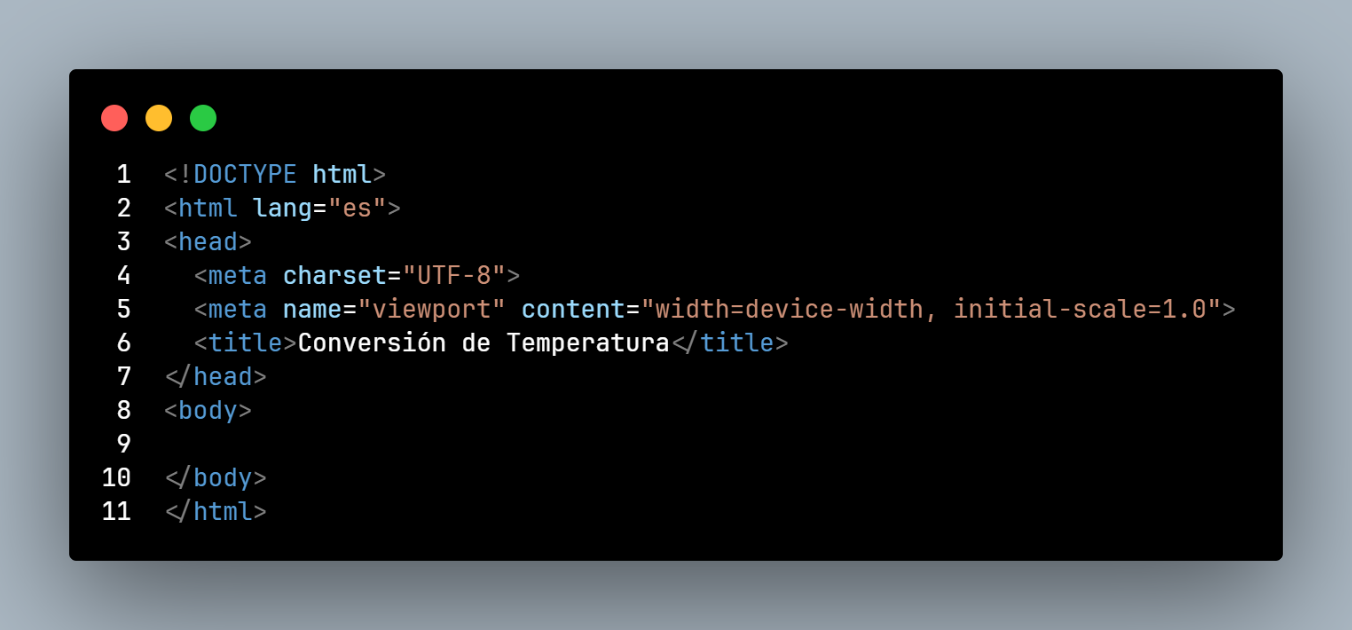
1. **Crear subcarpetas**



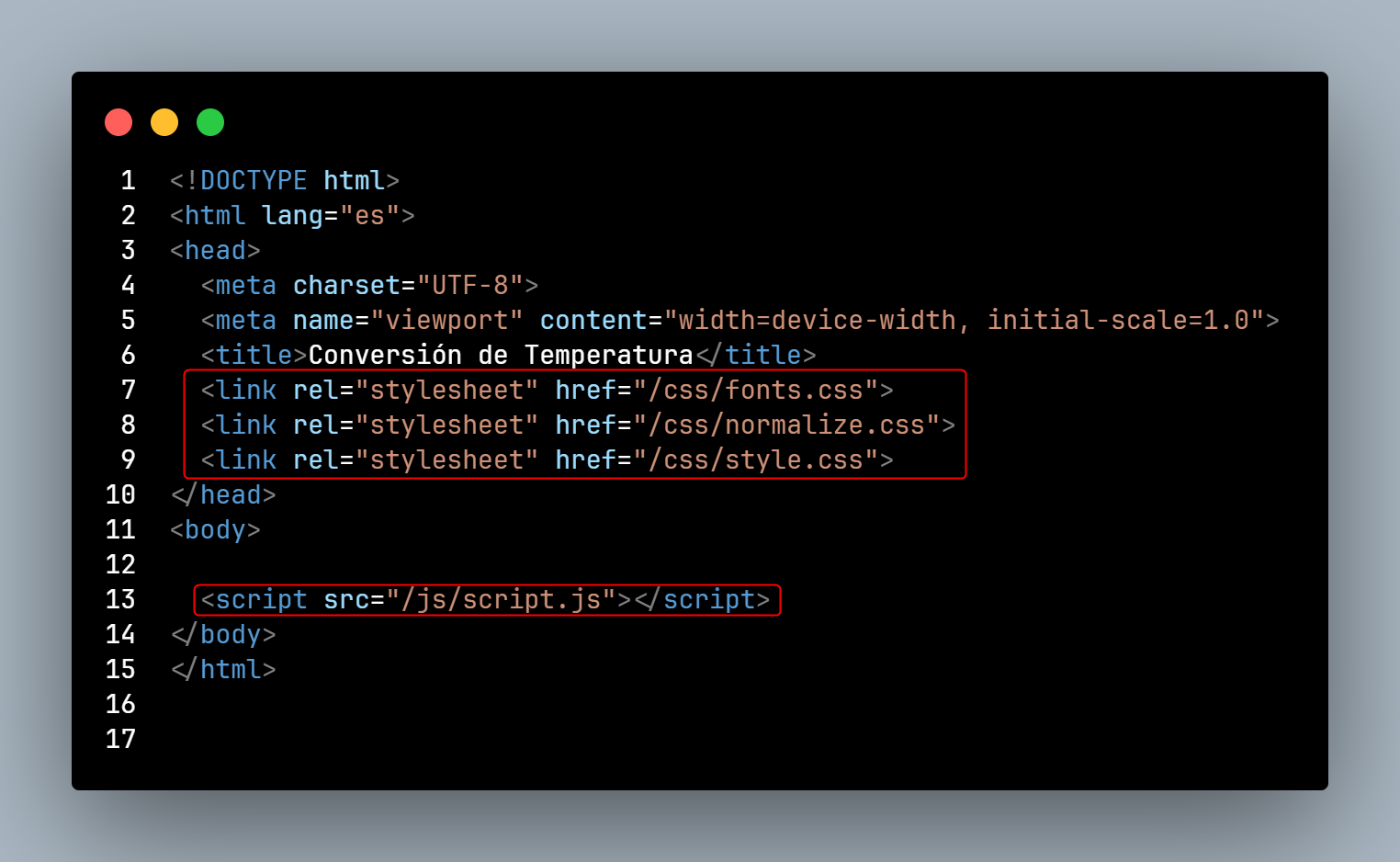
1. Crear archivos y guardarlos en sus respectivas carpetas



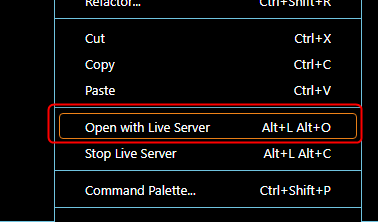
1. **Crear estructura HTML básica**



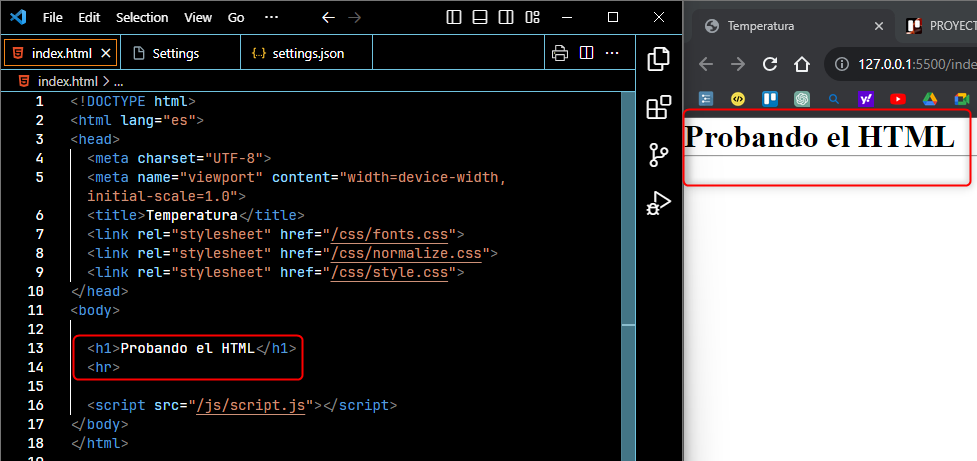
1. **Enlazar archivos .css y script.js al index.html y realizar pruebas.**

****

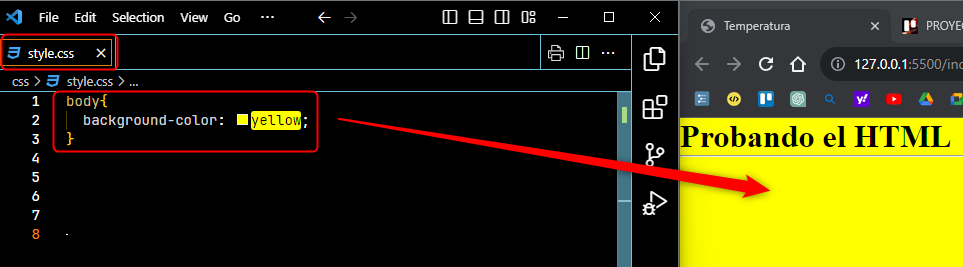
Ejecutamos el live server haciendo click derecho dentro de archivo index.html en cualquier espacio vacío.

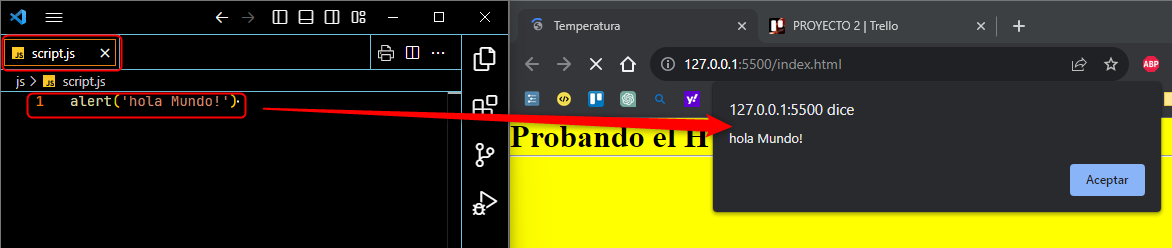
****

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***

1. **Agregar Favicon**

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

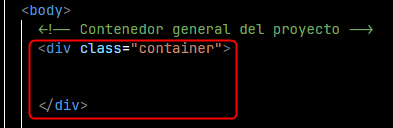


1. **Maquetación HTML**

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.

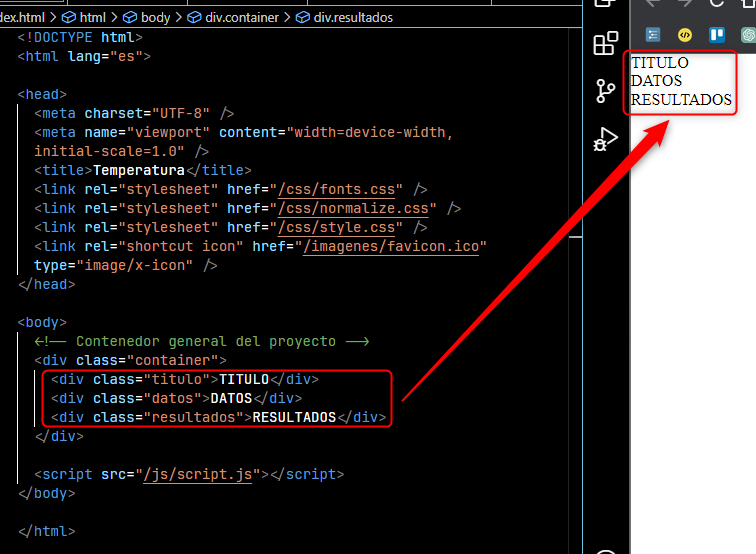


Dentro del contenedor general agregaremos contenedores para

## Titulo

## Datos

## Resultados

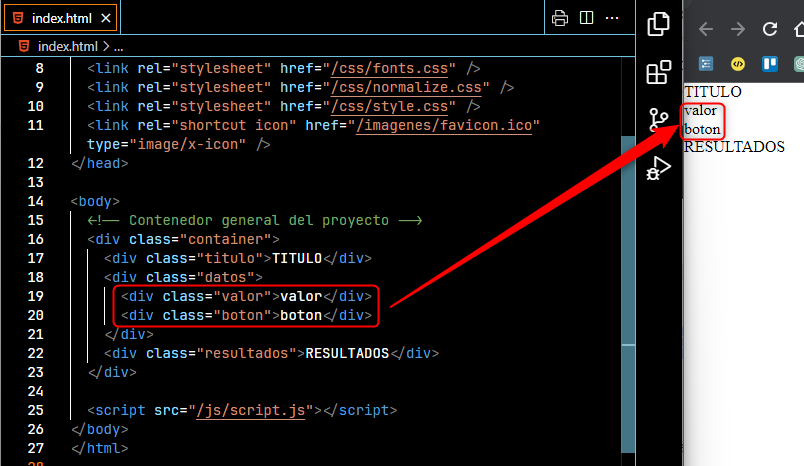
A su vez agregamos un texto descriptivo para ir ubicándolos en el navegador

Dentro del contenedor de datos deben ir dos contenedores más:

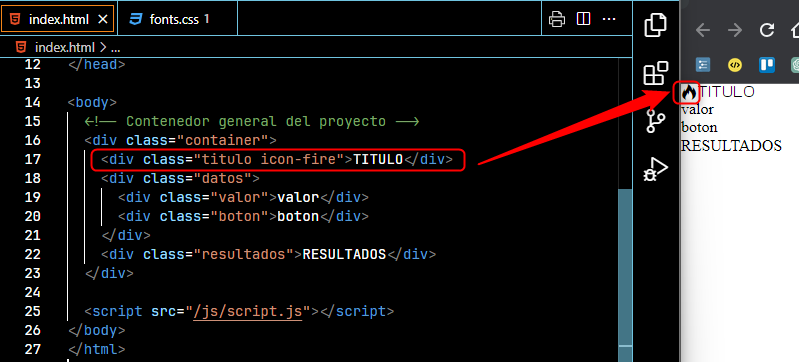
## Valor

## 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.



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***



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.***

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



Con esto hemos terminado la maquetación HTML.

1. **Maquetación CSS**

**Fuente General**

****

## 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**



## Agrego la fuente a todo el documento

## Indico el tamaño base para todas las fuentes

**Contenedor General (.container)**



## Agregamos borde para visualizar el área del contenedor

## Se establece el ancho máximo del contenedor

## Se ubica 50px hacia abajo y se centra horizontalmente

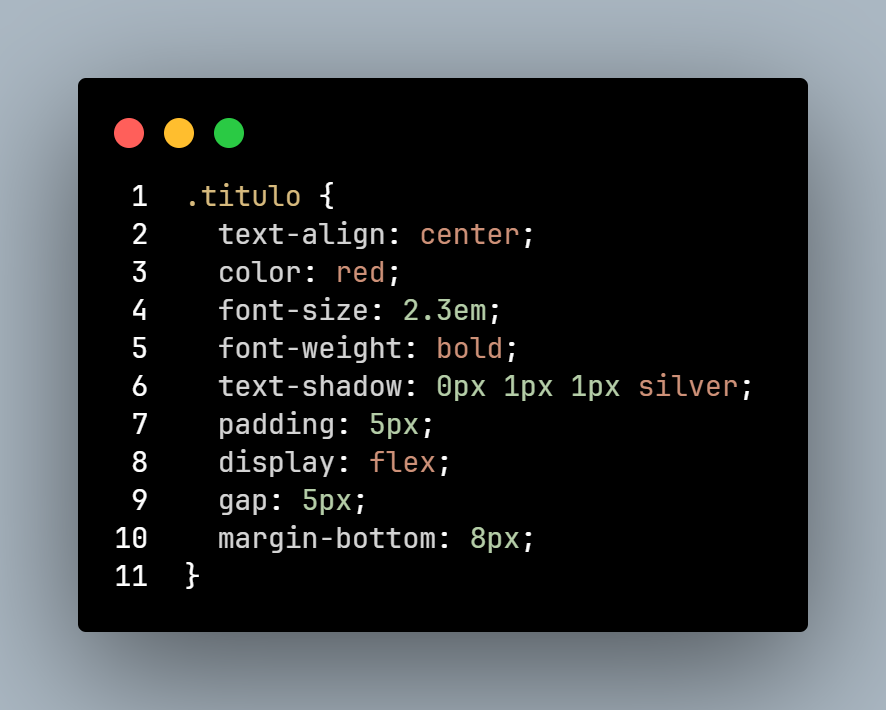
## Se centran los elementos internos del contenedor (titulo, datos y resultados)

## Se coloca color de fondo

## Se redondean las esquinas

## Se agrega margen interno

**Título (.titulo)**



## 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

## Se coloca un espaciado entre el icono y el titulo

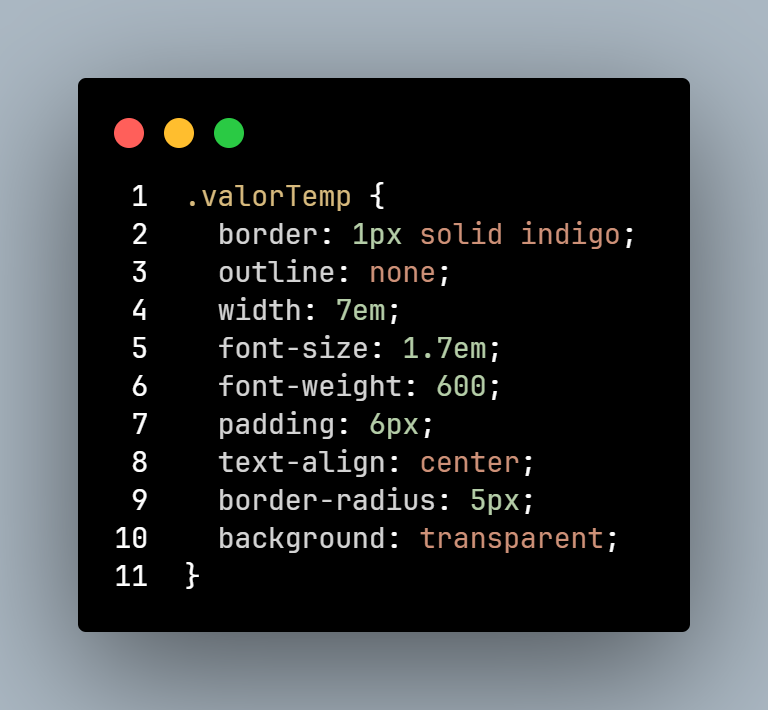
## Se coloca un espacio debajo del título para separarlo de los datos

**Datos**



* 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**

****

* 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**

****

**Boton de Conversión**

****

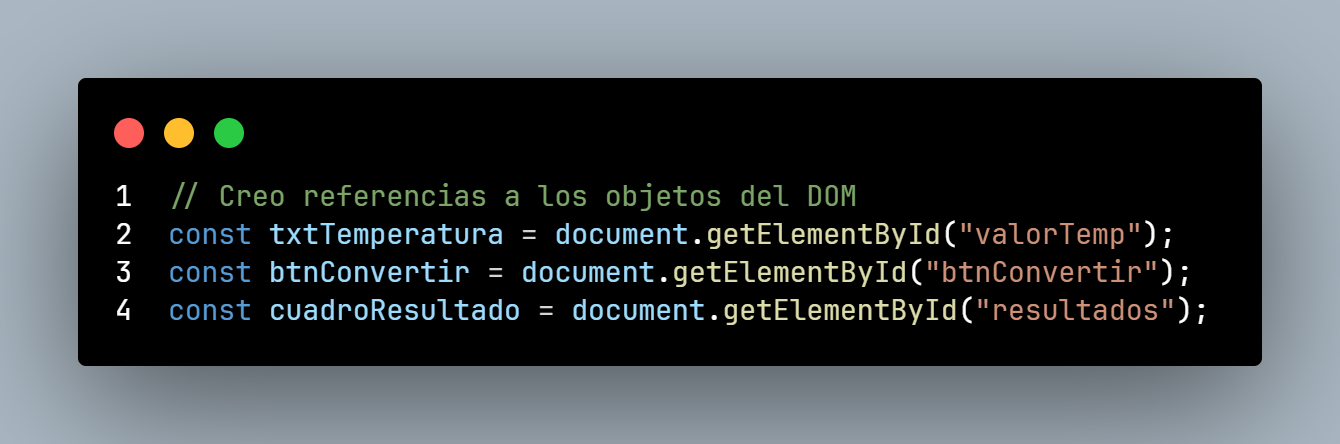
* Coloco borde
* Coloco tamaño de texto
* Asigno peso o intensidad del texto
* Color de texto
* Coloco espacio interno para agrandar botón
* Redondeo esquinas del botón
* Pongo cursor de manito al pasar el mouse por encima
* Quito fondo para que quede transparente

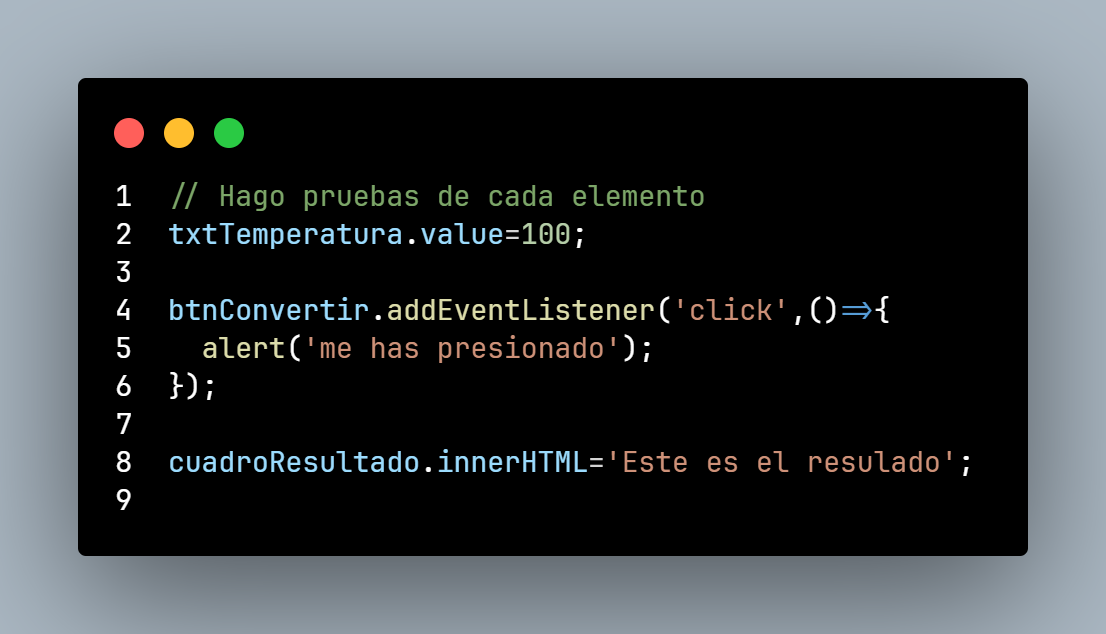
**Resultados**

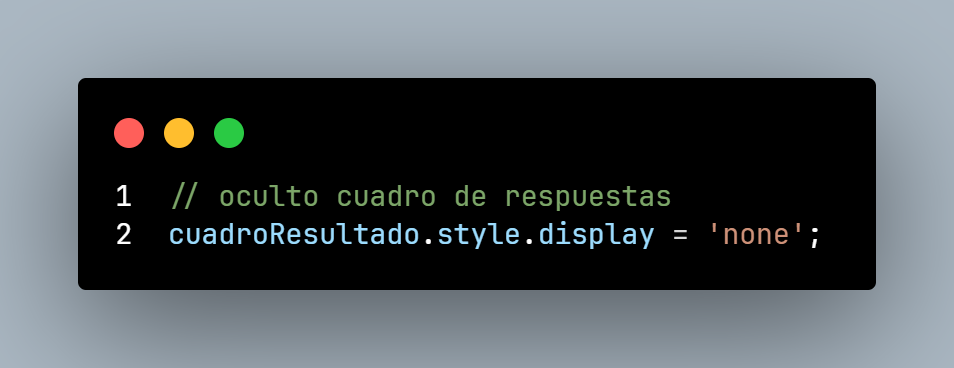


* Aplico ancho del contenedor
* Coloco espacio interior para separar de los bordes
* Centro el texto que se vaya a mostrar
* Aplico color de fondo
* Tamaño de texto
* Intensidad del texto
* Redondeo bordes

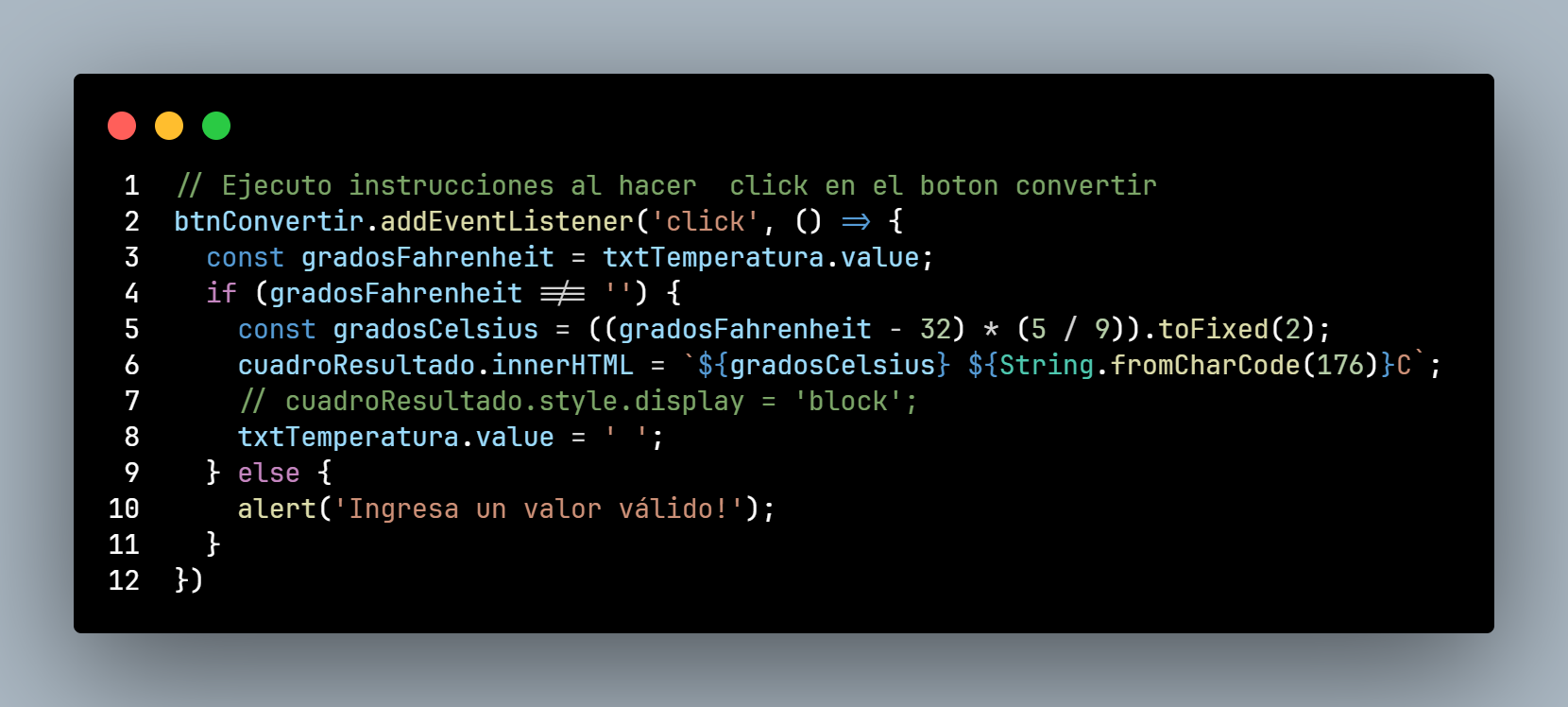
1. **Codificación JavaScript**











1. **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** |

1. **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**

****

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

****

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