**Clase 1:**



Ejercicio:

Empecemos

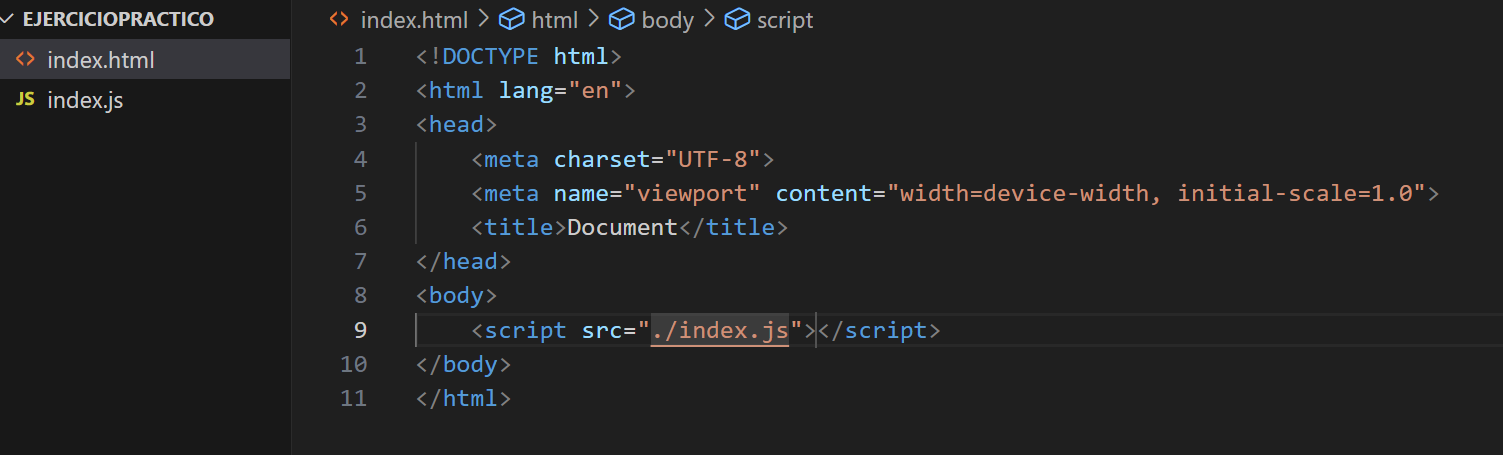
1. Abrir desde VSC (Visual Studio Code) nuestra carpeta de trabajoLogotipo

   Descripción generada automáticamente
2. Dentro de nuestra carpeta de trabajo creamos un archivo llamado index.html e insertamos una estructura base de HTML.

Texto

Descripción generada automáticamente

1. Creamos un archivo llamado index.js y lo vinculamos a nuestro archivo index.html



1. Dentro de nuestra estructura de trabajo creamos una carpeta llamada images y dentro de ella guardaremos las imágenes que requerimos.

OJO:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteSe deben renombrar las imágenes para poder trabajar con nombres cortos fáciles de usar.

1. Dentro de nuestra estructura de trabajo creamos una carpeta llamada data y dentro de ella creamos un archivo llamado data.js el cual tendrá la siguiente estructura.

Texto

Descripción generada automáticamente

1. Esta data la debemos exportar e importarla en el archivo index.js

Realicemos lo siguiente: En nuestro archivo index.html relacionamos nuestro archivo data y tanto al archivo data como a index.js les indicamos type=”module”



1. Dentro del archivo index.js importaremos el archivo data y verificaremos por consola la obtención de la data.

Texto

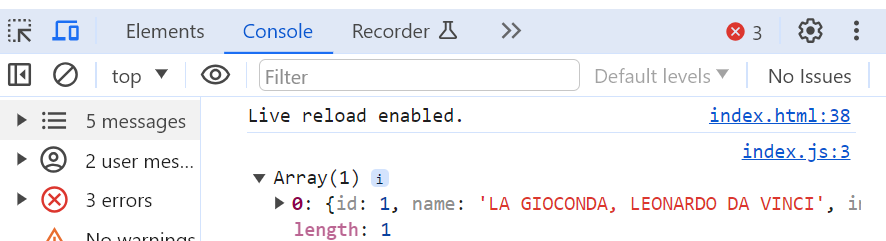
Descripción generada automáticamente

Ejecutamos con Open with Live Server

Captura de pantalla de un celular

Descripción generada automáticamente

En la consola nos debe de mostrar el arreglo con 1 objeto



Ya que tenemos nuestra data, empecemos a pintar la galería de pinturas en HTML.

**MOSTRAR PINTURAS EN HTML…**

1. Crear una división como contenedor principal con un id:“container”
2. Poner un título principal como el nombre: **Portafolio de pinturas famosas**
3. Agregar etiqueta de línea separadora **- <hr/>**

Texto

Descripción generada automáticamente

1. Agregar una tarjeta para mostrar las pinturas

Texto

Descripción generada automáticamente

1. Crear un archivo index.css y vincularlo en el archivo index.html.

Texto

Descripción generada automáticamente

1. En el archivo index.css agregar los siguientes estilos.

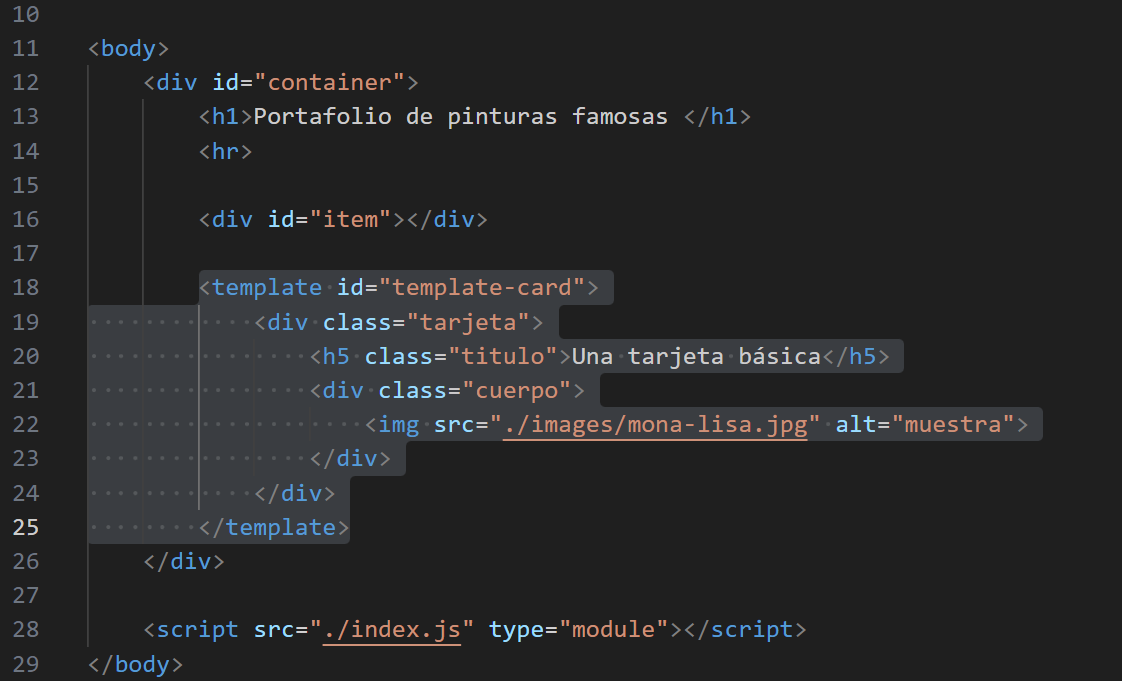


1. Agregar en el archivo HTML una etiqueta <div> que permita mostrar los elementos enviados desde el archivo index.js

Captura de pantalla de computadora

Descripción generada automáticamente

1. a toda la estructura de la tarjeta la vamos a contener un una itequeta llamada <template> y le vamos a poner un id=”template-card”



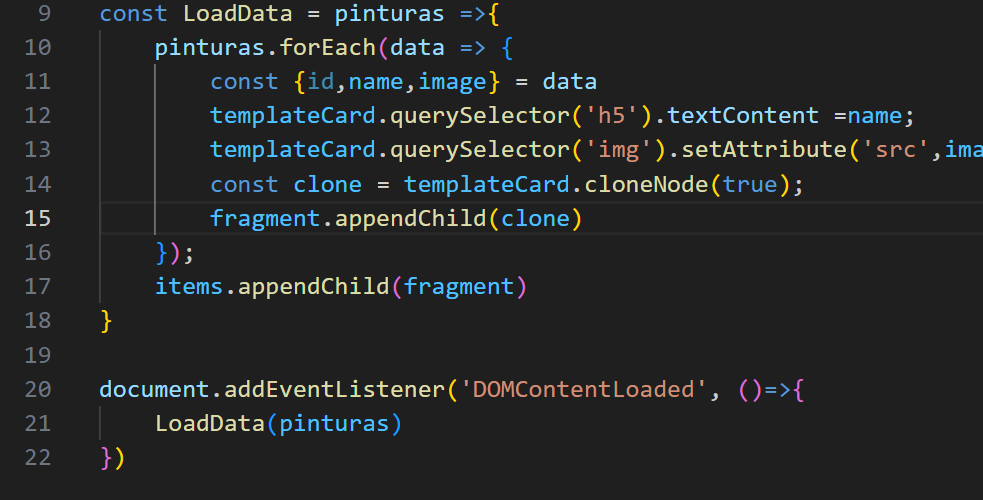
1. en el archivo index.js capturamos algunos elementos desde el HTML, tales como: la división con id ítems y el template con id template-card. Adicional adicionaremos un fragmento

Texto

Descripción generada automáticamente

Crearemos nuestra función para cargar la data y llamamos al DOMContentLoaded para el momento que se ejecute nuestra aplicación nos cargue la data con las pinturas en la estructura que definiremos

1. La función la llamaremos loadData y recibirá como parámetro la data que importamos desde el archivo data.js



Esta función recibirá las pinturas y la debemos recorrer, lo haremos con un foreach. Cuando ingresemos al foreach y nos quede el objeto, lo desestructuraremos y asignamos a la estructura del template los atributos. Clonamos los nodos del template card, empezamos a adicionar los nodos al fragmento (uno tras otro) y luego mostramos el fragmento en la división con id ítems

Texto original:

const LoadData = pinturas =>{

    pinturas.forEach(data => {

        const {id,name,image} = data

        templateCard.querySelector('h5').textContent =name;

        templateCard.querySelector('img').setAttribute('src',image)

        const clone = templateCard.cloneNode(true);

        fragment.appendChild(clone)

    });

    items.appendChild(fragment)

}

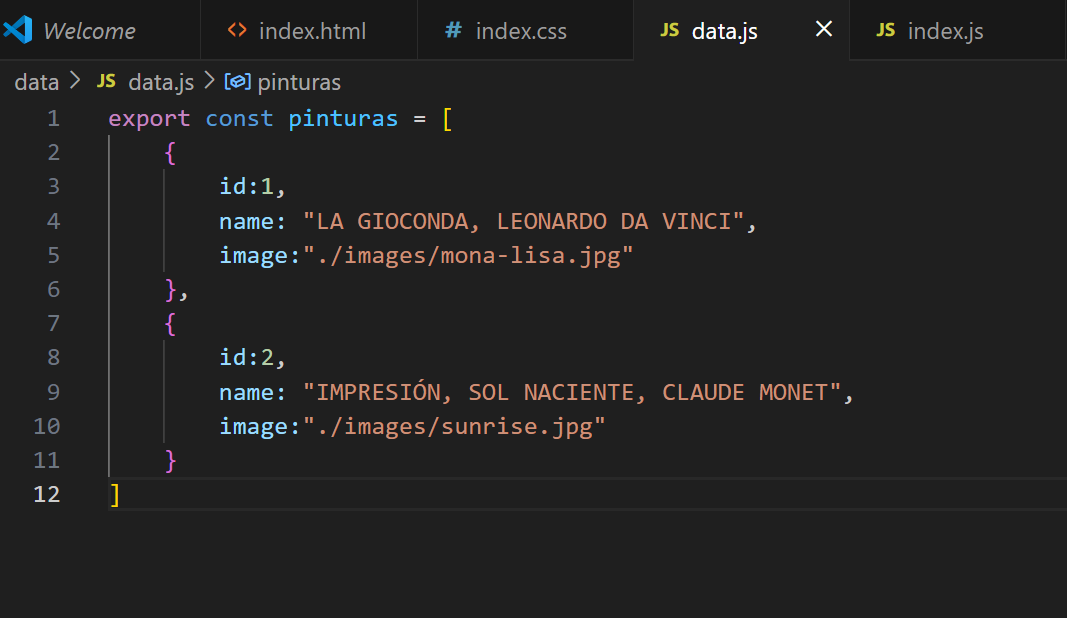
document.addEventListener('DOMContentLoaded', ()=>{

    LoadData(pinturas)

})

Para finalizar:

Vamos a agregar mas elementos en el archivo data.js de la siguiente manera.



Resultado Final:

Imagen que contiene Gráfico

Descripción generada automáticamente

Formulario de entrega: <https://forms.gle/ircE93xEyikCKZkQ6>