<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Lista - Rosas Eternas</title>

<link rel="stylesheet" href="mo.css">

<!-- Script para mostrar/ocultar detalles de productos -->

<script>

function toggleDetails(id) {

const detalles = document.getElementById(id);

if (detalles.style.display === "none") {

detalles.style.display = "block";

} else {

detalles.style.display = "none";

}

}

</script>

<!-- Estilo para las imágenes de categoría -->

<style>

.categoria img {

width: 200px; /\* Tamaño fijo para las imágenes \*/

height: 200px;

object-fit: cover; /\* Ajuste de imagen para cubrir el espacio \*/

border-radius: 8px; /\* Bordes redondeados \*/

}

</style>

</head>

<body>

<!-- Título principal de la página -->

<center><h1>CATALOGO</h1></center>

<!-- Encabezado con logo y navegación -->

<header>

<div class="contenedor-img">

<img src="melancolia.png"> <!-- Logo de la tienda -->

</div>

<h1>Rosas Eternas</h1>

<p>La belleza que perdura</p>

<!-- Menú de navegación -->

<nav>

<ul class="navbar">

<li><a href="cool.HTML">Inicio</a></li>

<li><a href="index.html">Productos</a></li>

<li>♥♥♥♥♥♥</li> <!-- Separador decorativo -->

<li><a href="contacto.html">Contacto</a></li>

</ul>

</nav>

</header>

<!-- Sección principal de productos -->

<section id="Lista">

<div class="categorias-lista">

<!-- Cada "categoria" representa un producto -->

<div class="categoria">

<img src="ar.jpg" alt="Ramo de rosas convinadas">

<h3>Ramo de rosas convinadas</h3>

<p class="precio">$380.00</p>

<!-- Botones para ver detalles y agregar al carrito -->

<button onclick="toggleDetails('detalles-1')">Ver detalles</button>

<button onclick="agregarAlCarrito('Ramo de rosas convinadas', 380)">Agregar al carrito</button>

<!-- Detalles ocultos del producto -->

<div id="detalles-1" class="detalles">

<center>

<p><strong>Descripción:</strong> 6 rosas,2 rojas,2 rosas,2 blancas...</p>

<p><strong>Material:</strong>liston rojo,blanco y rosado,palos de madera...</p>

<p><strong>Ramos disponibles:</strong> 10 </p>

</center>

</div>

</div>

<!-- Más productos con la misma estructura... -->

<!-- [Se omiten otros productos por brevedad, pero siguen el mismo patrón] -->

</div>

</section>

<!-- Separador decorativo -->

<center><h1>♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥</h1></center>

<!-- Sección del carrito de compras -->

<section id="carrito">

<h2>Carrito de Compras</h2>

<ul id="lista-carrito"></ul> <!-- Aquí se listarán los productos agregados -->

<p><strong>Total:</strong> $<span id="total">0.00</span></p> <!-- Total a pagar -->

<!-- Formulario de método de pago -->

<h3>Método de Pago</h3>

<form id="metodo-pago">

<label><input type="radio" name="pago" value="Crédito" required> Tarjeta de Crédito</label><br>

<label><input type="radio" name="pago" value="Débito"> Tarjeta de Débito</label><br>

<label><input type="radio" name="pago" value="Transferencia"> Transferencia Bancaria</label><br><br>

<button type="submit">Confirmar Compra</button>

</form>

<!-- Mensaje que aparecerá después de la compra -->

<p id="mensaje-compra" style="color: pink; font-weight: bold;"></p>

</section>

<!-- Script principal para funcionalidad del carrito -->

<script>

// Función para alternar visibilidad de detalles

function toggleDetails(id) {

const detalles = document.getElementById(id);

detalles.style.display = detalles.style.display === 'block' ? 'none' : 'block';

}

// Variables para manejar el carrito

let total = 0;

const listaCarrito = document.getElementById("lista-carrito");

const totalElement = document.getElementById("total");

const mensajeCompra = document.getElementById("mensaje-compra");

// Función para agregar productos al carrito

function agregarAlCarrito(nombre, precio) {

const li = document.createElement("li");

const texto = document.createElement("span");

texto.textContent = `${nombre} - $${precio.toFixed(2)}`;

// Botón para eliminar productos del carrito

const botonEliminar = document.createElement("button");

botonEliminar.textContent = "Eliminar";

botonEliminar.style.marginLeft = "10px";

botonEliminar.onclick = function() {

listaCarrito.removeChild(li);

total -= precio;

totalElement.textContent = total.toFixed(2);

};

li.appendChild(texto);

li.appendChild(botonEliminar);

listaCarrito.appendChild(li);

// Actualizar total

total += precio;

totalElement.textContent = total.toFixed(2);

}

// Evento para confirmar la compra

document.getElementById("metodo-pago").addEventListener("submit", function(e) {

e.preventDefault();

const metodo = document.querySelector('input[name="pago"]:checked');

if (metodo && listaCarrito.children.length > 0) {

mensajeCompra.textContent = `¡Compra realizada con éxito usando ${metodo.value}!`;

// Limpiar carrito después de la compra

listaCarrito.innerHTML = "";

total = 0;

totalElement.textContent = total.toFixed(2);

// Resetear selección de método de pago

document.querySelectorAll('input[name="pago"]').forEach(radio => {

radio.checked = false;

});

} else if (listaCarrito.children.length === 0) {

mensajeCompra.textContent = "⚠ No hay productos en el carrito.";

}

});

</script>

</body>

</html>

### Comentarios principales:

1. \*\*Estructura HTML\*\*: Documento bien formado con DOCTYPE, head y body.

2. \*\*Responsive Design\*\*: Uso de meta viewport para diseño adaptable.

3. \*\*Función toggleDetails()\*\*: Alterna la visibilidad de los detalles del producto.

4. \*\*Estilo de imágenes\*\*: CSS para mantener consistencia en el tamaño de las imágenes de productos.

5. \*\*Sección de productos\*\*: Cada producto tiene:

- Imagen

- Nombre

- Precio

- Botones para detalles y agregar al carrito

- Detalles ocultos (descripción, materiales, disponibilidad)

6. \*\*Carrito de compras\*\*:

- Lista dinámica de productos

- Cálculo automático del total

- Opciones de pago

- Confirmación de compra

7. \*\*Funcionalidad JavaScript\*\*:

- Agregar/eliminar productos

- Calcular total

- Manejar el proceso de compra

8. \*\*Feedback visual\*\*: Mensajes de confirmación/error para el usuario.

El código está bien estructurado y ofrece una experiencia completa de e-commerce para una tienda de arreglos florales.