**Universidad Tecnológica Metropolitana**

**4F**

**ACTIVIDAD # 1-4**

**Nombre alumno:**

**Fernando David Sanchez Sacnhez**

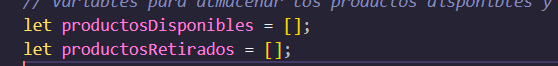
**Nombre del profesor(a): Ruth Dominguez**

**Fecha de entrega: Mérida, Yucatán a martes09 de octubre de 2024**

Practica 1.

**1. Declaración de Variables Globales**

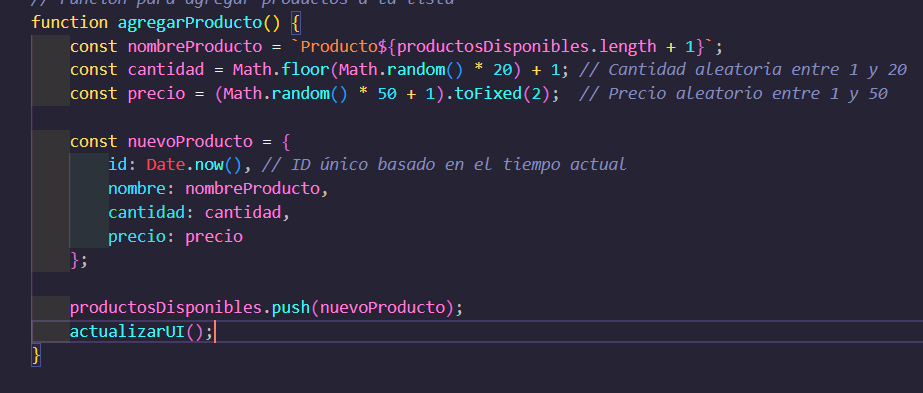
El código comienza definiendo dos arreglos vacíos para almacenar los productos disponibles y los productos retirados:



* productosDisponibles: Almacena los productos que actualmente están disponibles.
* productosRetirados: Almacena los productos que han sido retirados de la lista de disponibles.

**2. Función agregarProducto**

Esta función se encarga de agregar un nuevo producto a la lista de productos disponibles. A continuación, se detalla cada paso:



**Paso 1:** Genera un nombre de producto automático (Producto1, Producto2, etc.), basado en la longitud del arreglo productosDisponibles.



**Paso 2:** Genera una **cantidad aleatoria** entre 1 y 20 para el producto.

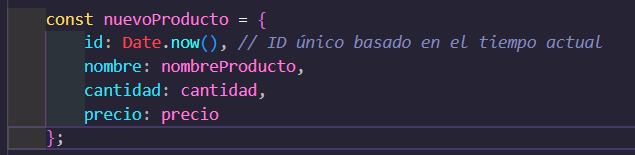


**Paso 3:** Genera un **precio aleatorio** entre 1 y 50 con dos decimales.

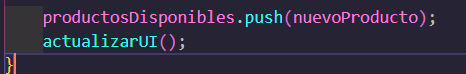


**Paso 4:** Crea un objeto nuevoProducto con las siguientes propiedades:

* id: ID único basado en el tiempo actual (Date.now()).
* nombre: Nombre generado del producto.
* cantidad: Cantidad aleatoria.
* precio: Precio aleatorio.

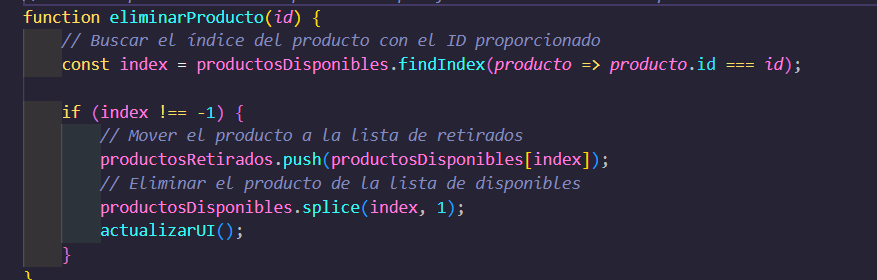


**Paso 5:** Añade el nuevo producto a la lista productosDisponibles y llama a actualizarUI para reflejar los cambios en la interfaz.



**3. Función eliminarProducto**

Esta función elimina un producto de la lista de productosDisponibles y lo mueve a la lista de productosRetirados usando su ID como referencia.



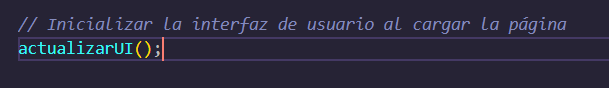
**4. Función actualizarUI**

Esta función actualiza la interfaz de usuario para reflejar la lista de productos disponibles y retirados, así como las cantidades y precios.



**5. Inicialización de la Interfaz**

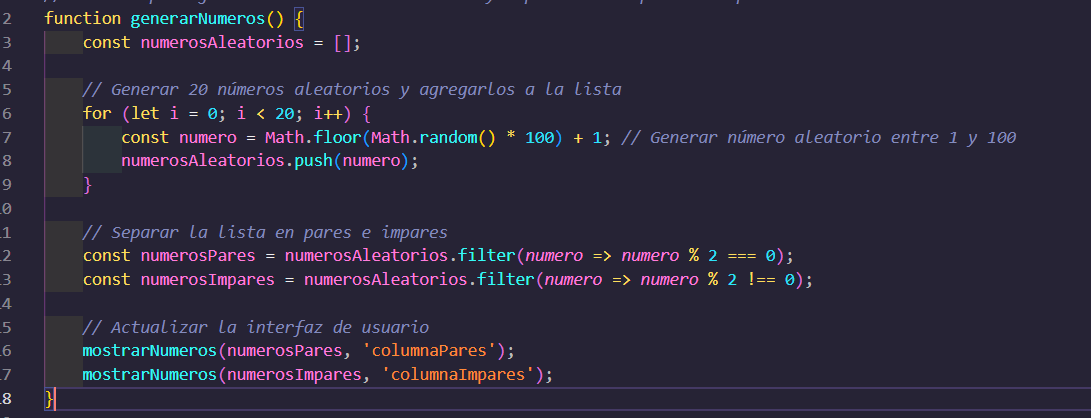
Por último, el código llama a actualizarUI() para inicializar la interfaz al cargar la página:



Practica 2.

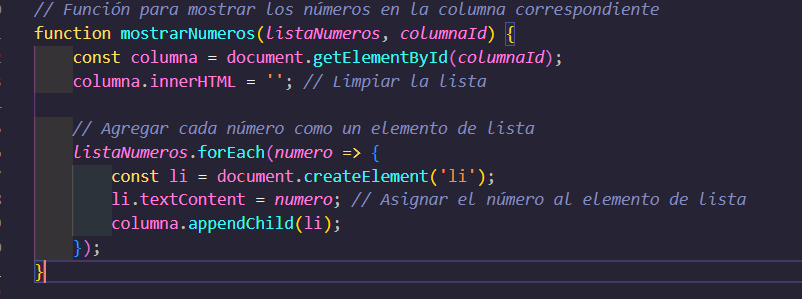
**1. Función generarNumeros**

Esta función es la encargada de generar los 20 números aleatorios y separarlos en números pares e impares, para luego actualizar la interfaz de usuario.



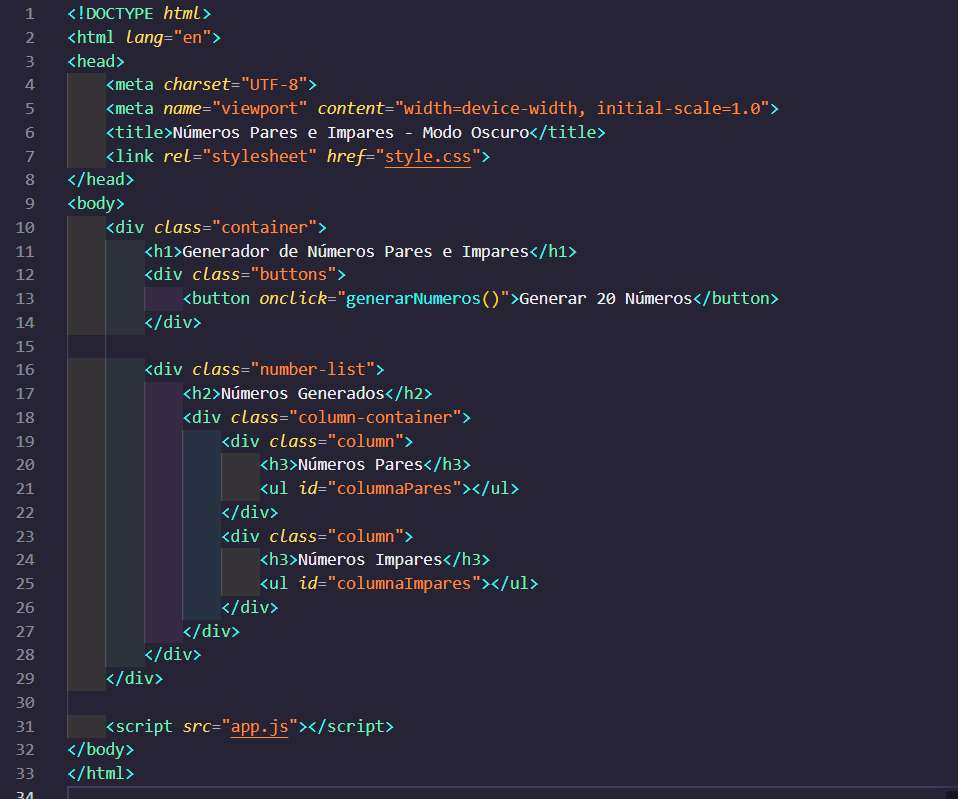
**2. Función mostrarNumeros**

Esta función se encarga de mostrar una lista de números en un contenedor HTML específico, que es indicado por el parámetro columnaId.



**3. Estructura Esperada del HTML**

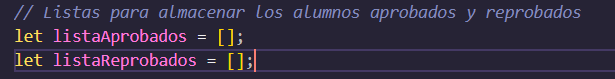
Para que este código funcione correctamente, se deben definir dos columnas en el archivo HTML con los id columnaPares y columnaImpares. Un ejemplo de cómo podría verse el HTML es el siguiente:



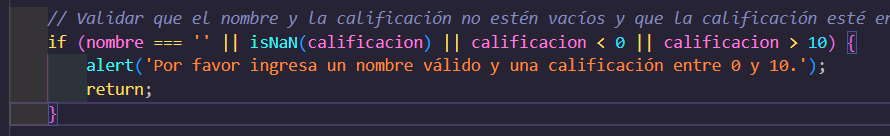
Practica 3.

**1. Declaración de Listas Globales**

El código comienza con la creación de dos arreglos vacíos para almacenar los alumnos aprobados y los reprobados:

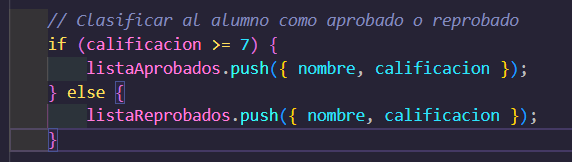


**Paso 2:** Valida los datos ingresados. Si el nombre está vacío o la calificación no es un número válido o está fuera del rango de 0 a 10, muestra un mensaje de alerta y termina la función.

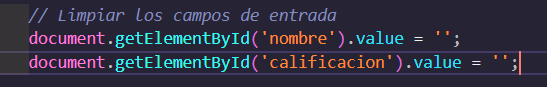


**Paso 3:** Clasifica al alumno según su calificación:

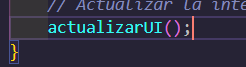
* Si la calificación es igual o mayor a 7, se agrega a listaAprobados.
* Si la calificación es menor a 7, se agrega a listaReprobados



**Paso 4:** Limpia los campos de entrada de texto (nombre y calificacion) después de agregar el alumno.



**Paso 5:** Llama a la función actualizarUI para mostrar las listas de aprobados y reprobados en la interfaz de usuario.



**3. Función actualizarUI**

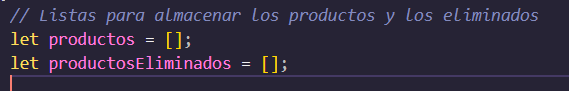
Esta función es la encargada de actualizar la interfaz de usuario para mostrar las listas de alumnos aprobados y reprobados en sus respectivos contenedores.



Practica 4.

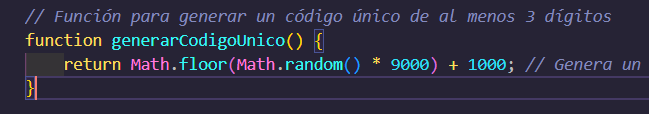
**1. Declaración de Listas Globales**

El código comienza con la declaración de dos arreglos vacíos para almacenar los productos disponibles y los productos que han sido eliminados.



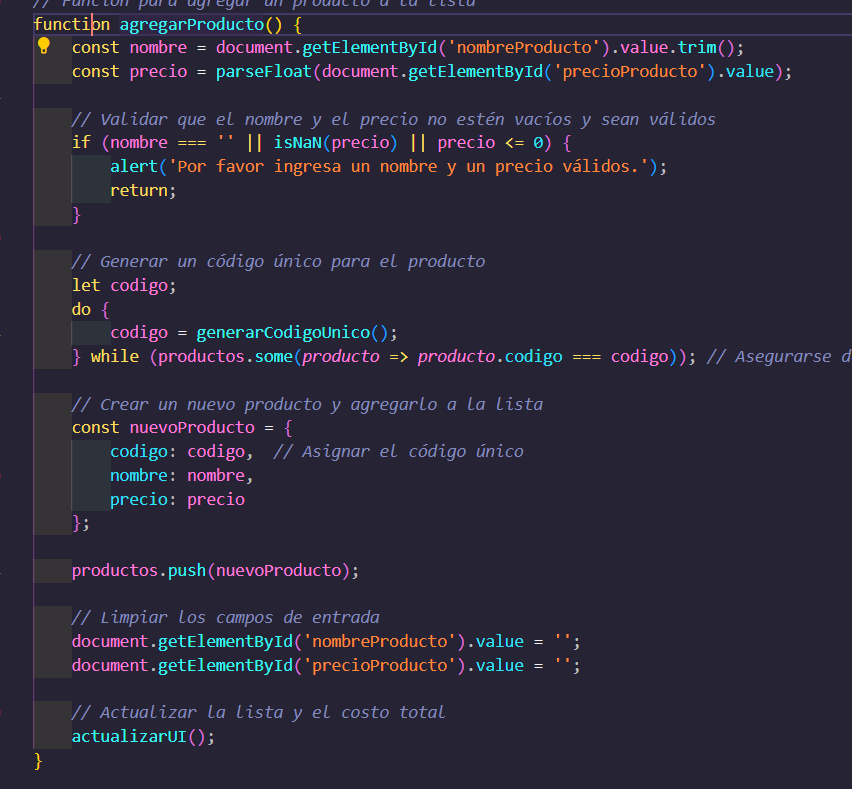
**2. Función generarCodigoUnico**

Esta función genera un código numérico único de 4 dígitos aleatorios para cada producto.



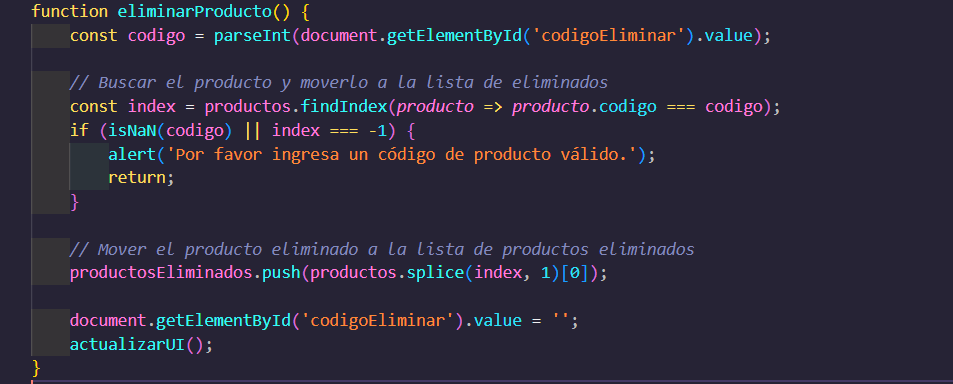
**3. Función agregarProducto**

Esta función permite agregar un nuevo producto a la lista de productos con un nombre y un precio proporcionado por el usuario.



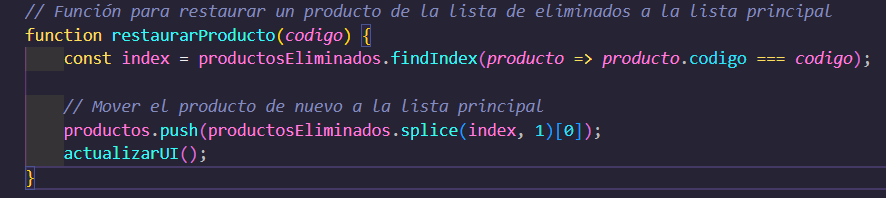
**4. Función eliminarProducto**

Esta función permite eliminar un producto de la lista de productos usando su código único y moverlo a la lista de productosEliminados.



**5. Función restaurarProducto**

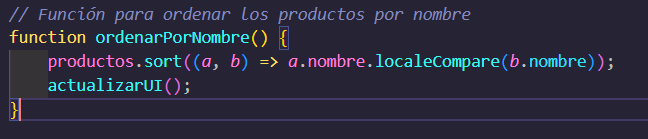
Esta función permite restaurar un producto eliminado a la lista principal de productos usando su código único.



.

**6. Función ordenarPorNombre**

Esta función permite ordenar la lista de productos por su nombre en orden alfabético.



**7. Función actualizarUI**

Esta función se encarga de actualizar la interfaz de usuario para mostrar las listas de productos y productos eliminados, así como calcular el costo total de los productos actuales.



**8. Inicialización**

Llama a actualizarUI() para actualizar la interfaz al cargar la página.

