

PRUEBA INDIVIDUAL

GESTIÓN DE PRODUCTOS AVANZADA

Generación : Generación 5

Módulo: Módulo 1

Curso: Programando con Javascript

Descripción : Esta prueba marca el cierre del Módulo 1, donde se trabajó principalmente con tecnologías base del desarrollo web como HTML, CSS y JavaScript. A lo largo del módulo, los estudiantes aprendieron a construir interfaces web funcionales, aplicar estilos personalizados y programar la lógica necesaria para validar información, actualizar contenidos en pantalla.

INDICACIONES

- Se permite utilizar documentación y apuntes de clase para resolver la prueba.
- No está permitido el uso de inteligencia artificial para generar código durante la prueba.
- Cualquier evidencia de uso de inteligencia artificial resultará en una penalización inmediata.
- La prueba tiene una duración máxima de **90 minutos**.
- Cada cambio en los datos debe reflejarse instantáneamente en la interfaz para facilitar la validación visual.
- **El diseño también será evaluado, por lo que se espera que partan del estilo base proporcionado y lo mejoren con creatividad y buen gusto.**

Contexto:

En esta etapa, se entregarán archivos base que no están completamente desarrollados. Los estudiantes deberán completarlos aplicando todo lo aprendido a lo largo del módulo, siguiendo cuidadosamente las indicaciones.

El reto consiste en asumir el rol de desarrollador dentro de un equipo que construirá las funcionalidades clave para una tienda online. El sistema debe permitir gestionar categorías, administrar un carrito de compras, registrar datos del cliente y controlar las ventas de manera dinámica y segura.

Además, la interfaz debe ser amigable, con validaciones claras y respuestas visuales inmediatas, garantizando una experiencia de usuario efectiva y profesional..

Instrucciones detalladas:

1. Agregar y actualizar productos

- Captura los valores de los campos:

PRUEBA INDIVIDUAL

- **Nombre**
- **Descripción**
- **Categoría**
- **Precio**
- **Stock**
- Cada campo debe pasar por las siguientes validaciones **antes de permitir guardar**:
 - **Nombre:**
 - No puede estar vacío ni contener sólo espacios.
 - Sólo debe contener letras (no números, símbolos ni caracteres especiales).
 - La primera letra debe ser mayúscula y el resto minúsculas.
 - Mostrar mensaje de error bajo el campo si no cumple, por ejemplo:
 - *“El nombre no puede quedar vacío.”*
 - *“El nombre debe contener solo letras.”*
 - *“La primera letra debe ser mayúscula.”*
 - **Descripción:**
 - No puede estar vacía.
 - No debe contener caracteres especiales raros.
 - Mostrar mensaje como: *“La descripción es obligatoria y debe ser válida.”*
 - **Categoría:**
 - No puede estar vacía.
 - Sólo letras y espacios permitidos.
 - Primera letra mayúscula, resto minúsculas.
 - Mostrar mensaje similar a: *“La categoría es obligatoria y debe tener formato válido.”*
 - **Precio:**
 - Debe ser número decimal positivo (≥ 0).
 - Mostrar error si no es válido: *“Ingrese un precio válido mayor o igual a cero.”*
 - **Stock:**

PRUEBA INDIVIDUAL

- Debe ser número entero positivo o cero.
- Mostrar error si es incorrecto: *“Ingrese un stock válido (entero, 0 o más).”*
- Solo cuando **todos los campos estén correctos y sin errores**, se podrá guardar (agregar o actualizar).
- Si el producto ya existe (comparando nombre exacto), se debe actualizar la información. Si no, agregarlo.
- Al guardar, limpiar los campos y mensajes de error.

2. Mostrar productos registrados

- Actualizar dinámicamente la tabla de productos con:
 - Nombre
 - Descripción
 - Categoría
 - Precio
 - Stock
 - Precio con IVA (12%) calculado automáticamente y mostrado con dos decimales
- La tabla debe reflejar todos los productos actuales **en tiempo real** después de cada operación (agregar, actualizar o eliminar).

3. Eliminar producto mediante recuadro

- Se debe crear un nuevo recuadro donde se pueda eliminar el producto dentro de gestionar productos
- Debe existir un campo de texto (cuadro para ingresar nombre del producto a eliminar) ubicado al lado o debajo de la tabla de productos.
- Debe haber un botón “Eliminar Producto” que, al presionar:
 - Verifique que el nombre ingresado no esté vacío ni sólo espacios.
 - Busque el producto que coincida exactamente con el nombre ingresado (sensible a mayúsculas).
 - Si existe, eliminarlo y actualizar tabla y estadísticas.
 - Si no existe, mostrar un mensaje de error claro cerca del cuadro de texto, por ejemplo:
“No se encontró producto con ese nombre.”

4. Estadísticas dinámicas

- Debe mostrarse en pantalla, en un área dedicada, las siguientes estadísticas y se deben **actualizar automáticamente después de cualquier cambio** (agregar, actualizar o eliminar producto):
 - Total de productos registrados (cantidad)
 - Suma total del stock de todos los productos
 - Valor total del inventario (suma de precio * stock por producto) mostrado con dos decimales
- Estas estadísticas ayudan a la gestión y control del inventario, por lo que deben ser siempre precisas y reflejar el estado actual.

5. Mensajes de error y experiencia visual

- Todos los mensajes de error de validación deben mostrarse justo debajo del campo al que corresponden, utilizando los elementos `.error` ya definidos en el HTML.
- Los mensajes deben ser claros, concisos y deben desaparecer cuando el error se corrige.
- **Se espera que mejoren la interfaz visual usando estilos CSS para que los mensajes se destaquen sin saturar la página.**

Archivos entregados para esta prueba:

- `index.html` — Contiene la estructura, campos con áreas para errores, tabla de productos y recuadro para eliminación.
- `styles.css` — Estilos básicos y espacio para mensajes de error.
- `utilitarios.js` — Funciones básicas para manipular elementos y valores en DOM.
- `scripts.js` — Deben completar las funciones para validaciones.