

1. Variables y Referencias Iniciales

```
const productos = [];
```

```
const form =  
document.getElementById('formulario-  
producto');
```

```
const tablaBody =  
document.getElementById('tabla-  
productos');
```

```
const estadísticas = {  
  totalProductos:
```

```
document.getElementById('total-
```

```
productos'),  
  totalStock:  
document.getElementById('total-stock'),  
  valorInventario:  
document.getElementById('valor-  
inventario')  
};  
const eliminarInput =  
document.getElementById('nombre-  
eliminar');  
const eliminarError =  
document.getElementById('error-  
eliminar');  
,
```

2. Validaciones

```
function validarNombre(nombre) {
```

```
if (!nombre.trim()) return "El nombre  
no puede quedar vacío.";
if (!/^[A-Z][a-  
záéíóúñ\s]*$/i.test(nombre)) return "El  
nombre debe comenzar con mayúscula y  
contener solo letras.";
return null;
}
```

```
function validarDescripcion(desc) {
  if (!desc.trim()) return "La descripción  
es obligatoria y debe ser válida.";
  if (!/^[a-zA-  
z0-9áéíóúñ\s.,]$/i.test(desc)) return "La  
descripción contiene caracteres  
inválidos.";
  return null;
}
```

```
function validarCategoria(cat) {
  if (!cat.trim()) return "La categoría es
```

```
obligatoria y debe tener formato válido.";
if (!/^[A-Z][a-záéíóúñ\s]*$/i.test(cat)) return "La
categoría debe comenzar con mayúscula y
contener solo letras.";
return null;
}
```

```
function validarPrecio(precio) {
  if (!/^\d+(\.\d{1,2})?$/i.test(precio) ||
  parseFloat(precio) < 0) return "Ingrese
un precio válido mayor o igual a cero.";
  return null;
}
```

```
function validarStock(stock) {
  if (!/^\d+$/i.test(stock)) return
  "Ingrese un stock válido (entero, 0 o
más).";
  return null;
}
```

3. Agregar o Actualizar Producto

`js

```
form.addEventListener('submit', (e) => {  
  e.preventDefault();
```

```
  const nombre =
```

```
  form.nombre.value.trim();
```

```
  const descripcion =
```

```
  form.descripcion.value.trim();
```

```
  const categoria =
```

```
  form.categoria.value.trim();
```

```
  const precio = form.precio.value.trim();
```

```
  const stock = form.stock.value.trim();
```

```
  const errores = {
```

```
    nombre: validarNombre(nombre),
```

descripcion:

```
validarDescripcion(descripcion),  
  categoria: validarCategoria(categoria),  
  precio: validarPrecio(precio),  
  stock: validarStock(stock)  
};
```

```
let hayErrores = false;  
for (let campo in errores) {  
  const errorEl =  
document.getElementById(error-$  
{campo});  
  if (errores[campo]) {  
    errorEl.textContent = errores[campo];  
    hayErrores = true;  
  } else {  
    errorEl.textContent = "";  
  }  
}
```

```
if (hayErrores) return;
```

```
const index = productos.findIndex(p =>
p.nombre === nombre);
const nuevoProducto = {
  nombre,
  descripción,
  categoría,
  precio: parseFloat(precio),
  stock: parseInt(stock)
};
```

```
if (index >= 0) {
  productos[index] = nuevoProducto;
} else {
  productos.push(nuevoProducto);
}
```

```
form.reset();
renderizarTabla();
actualizarEstadísticas();
});
```

4. Renderizar Tabla

`js

```
function renderizarTabla() {  
  tablaBody.innerHTML = "";  
  productos.forEach(p => {  
    const fila =  
document.createElement('tr');  
    fila.innerHTML = `  
      <td>${p.nombre}</td>  
      <td>${p.descripcion}</td>  
      <td>${p.categoria}</td>  
      <td>${p.precio.toFixed(2)}</td>  
      <td>${p.stock}</td>  
      <td>${(p.precio * 1.12).toFixed(2)}</  
td>  
    `;  
  });  
}
```



```
    tablaBody.appendChild(fila);  
  });  
}  
,
```



5. Eliminar Producto

```
`js  
document.getElementById('btn-  
eliminar').addEventListener('click', () =>  
{  
  const nombre =  
    eliminarInput.value.trim();  
    eliminarError.textContent = "";  
  
  if (!nombre) {  
    eliminarError.textContent = "Debe  
    ingresar un nombre."  
    return;  
  }  
})
```

```
}
```

```
const index = productos.findIndex(p =>
p.nombre === nombre);
if (index >= 0) {
  productos.splice(index, 1);
  renderizarTabla();
  actualizarEstadísticas();
  eliminarInput.value = "";
} else {
  eliminarError.textContent = "No se
encontró producto con ese nombre.";
}
});
`
```



6. Estadísticas

```
`js
```

```
function actualizarEstadísticas() {  
  const total = productos.length;  
  const stockTotal =  
    productos.reduce((acc, p) => acc +  
      p.stock, 0);  
  const valorTotal =  
    productos.reduce((acc, p) => acc +  
      (p.precio * p.stock), 0);  
  
  estadísticas.totalProductos.textContent  
    = total;  
  estadísticas.totalStock.textContent =  
    stockTotal;  
  estadísticas.valorInventario.textContent  
    = `${valorTotal.toFixed(2)}$`;  
}
```

¿Quieres que te ayude a integrar esto con

el HTML base o revisar cómo se vería el archivo index.html para que todo encaje perfectamente?