

Tu Primera Lista de la Compra con JS

Una guía visual para entender cómo crear y manejar datos en JavaScript.

1

Preparando el Escenario 🛒

Antes de nada, necesitamos un "carrito" (un array vacío) y las "fichas de producto" (objetos) que vamos a meter dentro.

1.1 El Carrito Vacío

Creamos una lista vacía (Array) para guardar los productos.

```
let shoppingList = []
```

1.2 Las Fichas de Producto

Creamos un objeto para cada producto con sus detalles.

```
let firstProduct = {
  item: "tomates",
  quantity: 3,
  purchased: false
}
```

1.3 ¡Al Carrito!

Usamos `.push()` para añadir cada producto al final de la lista.

```
shoppingList.push(firstProduct)
```

2

Las Herramientas Mágicas 🛠️

Para no repetir código, creamos funciones: máquinas reutilizables que realizan tareas específicas. Les damos "ingredientes" (parámetros) y nos devuelven un resultado.

La Máquina de AÑADIR: `addItem`

Esta función crea un nuevo producto y lo añade a la lista que le pasemos.

Ingredientes: Una lista (`list`), un nombre (`newItem`), una cantidad (`newQuantity`).

Receta:

1. Crea una nueva "ficha de producto" (`newProduct`).
2. La añade (`.push()`) al final de la lista.
3. Devuelve (`.return`) la lista actualizada.

```
function addItem (list, newItem, newQuantity) {
  let newProduct = {
    item: newItem,
    quantity: newQuantity,
    purchased: false
  }
  list.push(newProduct)
  return list
}
```

La Máquina de QUITAR: `removeItem`

Elimina un producto de la lista basándose en su posición (índice).

Ingredientes: Una lista (`list`) y una posición (`index`).

Receta:

1. Usa `.splice(index, 1)` para cortar y quitar un (1) elemento desde la posición `index`.
2. Devuelve (`.return`) la lista sin el elemento.

```
function removeItem(list, index) {
  list.splice(index, 1)
  return list
}
```

La Máquina de MODIFICAR: `updateItem`

Encuentra un producto por su posición y actualiza sus datos.

Ingredientes: Una lista (`list`), una posición (`index`), y los nuevos datos (`newItem`, `newQuantity`).

Receta:

1. Encuentra el producto con `list[index]`.
2. Asigna (`.=`) el `newItem` a la propiedad `item` del producto.
3. Asigna (`.=`) la `newQuantity` a la propiedad `quantity`.
4. Devuelve (`.return`) la lista con el producto modificado.

```
function updateItem(list, index, newItem, newQuantity) {
  let updatedProduct = list[index];
  updatedProduct.item = newItem;
  updatedProduct.quantity = newQuantity;
  return list;
}
```

3

¡Manos a la Obra! ✅

Finalmente, usamos nuestras funciones para manipular la lista. Usamos `console.table()` para ver los resultados de forma clara en la consola del navegador.

```
// 1. Añadimos un producto
shoppingList = addItem(shoppingList, "sandia", 3);
console.table(shoppingList); // Muestra la lista con 5 productos

// 2. Eliminamos el producto en la posición 3 ("repollo")
shoppingList = removeItem(shoppingList, 3);
console.table(shoppingList); // Muestra la lista con 4 productos

// 3. Modificamos el producto en la posición 1 ("patatas")
shoppingList = updateItem(shoppingList, 1, "cebolla", 6);
console.table(shoppingList); // Muestra "cebolla" en lugar de "patatas"
```

¡Felicidades! Has creado y manipulado datos en JavaScript como un profesional.