

# PRÁCTICA 01 - DESTACADAS: Gestión de Clientes en Lista de Espera

---

"Administrando mi fila"  : Manejo de listas con Arrays"

## Objetivos de Aprendizaje:

- Comprender cómo se utiliza un **array** para almacenar datos de forma secuencial.
- Aplicar los métodos `.push()` y `.pop()` para **agregar** y **eliminar elementos** del array.
- Relacionar los conceptos con un **contexto real**: la gestión de una fila de espera de clientes en una tienda o sistema de atención.

## Descripción de la práctica:

El estudiante simulará un sistema de **turnos**, donde podrá:

- Ingresar nuevos clientes en una lista
- Atender al último cliente ingresado
- Ver la lista actualizada de clientes en espera

## Código de la práctica:

```
1 const readline = require("readline-sync");
2
3 let filaDeEspera = [];
4
5 let opcion;
6
7 do {
8     console.log("\n📋 MENÚ");
9     console.log("1. + Agregar cliente");
10    console.log("2. - Atender último cliente");
11    console.log("3. 📋 Ver lista de espera");
```

```
12 console.log("4. └ Salir");
13
14 opcion = readline.question("Selecciona una opción: ");
15
16 switch (opcion) {
17     case "1":
18         const cliente = readline.question("Nombre del cliente: ");
19         filaDeEspera.push(cliente);
20         console.log(`✓ ${cliente} ha sido agregado a la lista.`);
21         break;
22
23     case "2":
24         if (filaDeEspera.length > 0) {
25             const atendido = filaDeEspera.pop();
26             console.log(`👤 • $ Atendiendo a: ${atendido}`);
27         } else {
28             console.log("✗ No hay clientes en la lista.");
29         }
30         break;
31
32     case "3":
33         console.log("📋 Lista actual:");
34         filaDeEspera.forEach((c, i) => {
35             console.log(`${i + 1}. ${c}`);
36         });
37         break;
38
39     case "4":
40         console.log("👋 Cerrando sistema...");
```

41 break;

```
42
43     default:
44         console.log("✗ Opción inválida.");
45     }
46
47 } while (opcion !== "4");
```

 Al finalizar la práctica el estudiante:

-  Comprenderá el funcionamiento básico de un array y cómo simular **estructuras dinámicas** (como una fila o cola).
-  Dominará los métodos `.push()`, `.pop()` y `.forEach()` para manipular datos reales.
-  Aplicará la lógica de programación a un **caso de uso cotidiano**, integrando entrada del usuario, lógica condicional y estructura repetitiva.
-  Estará listo para crear sistemas más complejos como listas de tareas, pedidos, asistentes virtuales o formularios dinámicos.



## PRÁCTICA 2: Encuesta de Opinión Exprés



---

### "Voz del público: Analizando respuestas con Arrays"

 Objetivos de Aprendizaje:

- Reforzar el uso de **arrays de cadenas (string)** para almacenar información dinámica.
- Aplicar métodos como `.push()`, `.length`, y `.forEach()` para analizar datos.
- Simular una situación real de **recopilación de encuestas** y procesamiento de resultados.

 Código de la práctica:

```
1 const readline = require("readline-sync");
2
3 let respuestas = [];
4 let respuesta;
5
6 console.log("📝 ENCUESTA: ¿Qué opinas de nuestro servicio?");
7
8 do {
```

```

9  respuesta = readline.question("Escribe tu opinión o escribe
10 'fin' para terminar: ");
11  if (respuesta.toLowerCase() !== "fin") {
12      respuestas.push(respuesta);
13  }
14
15 console.log("\n📊 RESULTADOS:");
16 console.log(`Total de respuestas: ${respuestas.length}`);
17
18 respuestas.forEach((resp, index) => {
19     console.log(`${index + 1}. ${resp}`);
20 });

```

### Al finalizar la práctica el estudiante:

-  Sabrá cómo **almacenar texto de usuarios** de forma dinámica usando `.push()`
-  Practicará `.length` para obtener cantidad de elementos
-  Usará `.forEach()` para mostrar datos en consola con formato
-  Relacionará arrays con **aplicaciones reales**: encuestas, formularios, feedbacks

## PRÁCTICA 3: Control de Temperatura

---

### "Monitoreo climático: Registrando temperaturas con JavaScript"

#### Objetivos de Aprendizaje:

- Usar arrays numéricos para **almacenar y analizar datos físicos** (temperatura).
- Aplicar `.push()`, `.splice()` y bucles para capturar y modificar registros.
- Comprender cómo manejar una lista dinámica con valores numéricos reales.

## Código de la práctica:

```
1 const readline = require("readline-sync");
2
3 let temperaturas = [];
4
5 for (let i = 1; i <= 5; i++) {
6   let temp = parseFloat(readline.question(`\ud83d\udcbb Ingresá la
7   temperatura del día ${i}: `));
8   temperaturas.push(temp);
9 }
10
11 console.log("\n\ud83c\udcda Temperaturas registradas:");
12 console.log(temperaturas);
13
14 // Corrección opcional de temperatura
15 let corregir = readline.question("\ud83d\udcbb ¿Deseas corregir una
16   temperatura? (s/n): ");
17
18 if (corregir.toLowerCase() === "s") {
19   let indice = parseInt(readline.question("\ud83d\udcbb ¿Qué día quierés
20     corregir? (1-5): ")) - 1;
21   let nuevaTemp = parseFloat(readline.question(`\ud83d\udcbb Nueva
22     temperatura: `));
23   temperaturas.splice(indice, 1, nuevaTemp);
24 }
25
26 console.log("\n\ud83c\udcda Temperaturas finales:");
27 temperaturas.forEach((t, i) => {
28   console.log(`\ud83d\udcbb Día ${i + 1}: ${t}°C`);
29 });
30
```

## Al finalizar la práctica el estudiante:

- Aprenderá a trabajar con **arrays numéricos reales**
- Usará `.splice()` para **corregir valores** en una posición específica
- Integrará entrada del usuario, control de flujo y manipulación de datos
- Conectará la programación con una **aplicación científica o meteorológica**