



Nivel Medio-Avanzado — Casos con enfoque lógico

◆ 1. Ranking de empleados

```
const empleados = [  
  { id: 1, nombre: "Laura", ventas: [300, 500, 100] },  
  { id: 2, nombre: "Carlos", ventas: [400, 200] },  
  { id: 3, nombre: "Ana", ventas: [100, 200, 300, 100] },  
];  
  
/// Enunciado:  
/// Calcular el total de ventas por empleado, luego ordenar y devolver  
/// los 2 mejores vendedores junto a su total.  
/// Resultado esperado:  
/// [{ nombre: "Laura", total: 900 }, { nombre: "Carlos", total: 600 }]
```

◆ 2. Validar estructura de usuarios activos

```
const usuarios = [  
  { id: 1, activo: true, datos: { nombre: "Juan", edad: 30 } },  
  { id: 2, activo: false, datos: { nombre: "Diana", edad: 22 } },  
  { id: 3, activo: true, datos: null },  
];  
  
/// Enunciado:  
/// Devolver los nombres de los usuarios activos y que tengan datos completos.  
/// Ignorar usuarios con datos incompletos o inactivos.
```



3. Buscar conflictos de horarios

```
const reuniones = [  
  { id: 1, horaInicio: "09:00", horaFin: "10:00" },  
  { id: 2, horaInicio: "09:30", horaFin: "11:00" },  
  { id: 3, horaInicio: "11:00", horaFin: "12:00" },  
];  
  
/// Enunciado:  
/// Detectar si existe algún solapamiento entre reuniones.  
/// Devolver true si hay conflicto, false si no lo hay.
```

◆ 4. Crear árbol de categorías

```
const categorias = [  
  { id: 1, nombre: "Tecnología", padre: null },  
  { id: 2, nombre: "Software", padre: 1 },  
  { id: 3, nombre: "Hardware", padre: 1 },  
  { id: 4, nombre: "Accesorios", padre: 3 },  
];  
  
/// Enunciado:  
/// Construir una estructura jerárquica en forma de árbol (nested JSON),  
/// donde cada categoría tenga su lista de "hijos".
```



◆ 5. Generar estadísticas de cursos

```
const cursos = [  
  {  
    nombre: "JS Avanzado",  
    estudiantes: [  
      { nombre: "Luis", completado: true },  
      { nombre: "Ana", completado: false },  
      { nombre: "Carla", completado: true },  
    ],  
  },  
  {  
    nombre: "Python",  
    estudiantes: [  
      { nombre: "Pedro", completado: true },  
      { nombre: "Lucía", completado: true },  
    ],  
  },  
];
```

/// Enunciado:

/// Devolver un resumen de cursos con porcentaje de finalización.

/// Ejemplo:

/// [{ nombre: "JS Avanzado", completado: "66.67%", ...}]



◆ 6. Normalizar información de pedidos

```
const pedidos = [  
  { cliente: "Maria", productos: [{ nombre: "Café", precio: 5 }, { nombre: "Pan", precio: 2 }] },  
  { cliente: "Juan", productos: [{ nombre: "Café", precio: 5 }, { nombre: "Té", precio: 4 }] },  
];
```

/// Enunciado:

/// Crear un resumen agrupado por producto:

/// Ejemplo esperado:

/// {

/// "Café": { totalPedidos: 2, totalIngresos: 10 },

/// "Pan": { totalPedidos: 1, totalIngresos: 2 },

/// "Té": { totalPedidos: 1, totalIngresos: 4 }

/// }