

Ejercicios de Arrays BÁSICOS para practicar

Ejercicio 1:

Dado un array de números, utiliza los métodos `filter` y `map` para obtener los números pares y elevarlos al cuadrado.

```
// Enunciado
const numeros = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];
```

Ejercicio 2:

Dado un array de strings, usa `reduce` para concatenar todos los strings en una sola frase con espacios entre ellos.

```
// Enunciado
const palabras = ["Hola", "mundo", "esto", "es", "JavaScript"];
```

Ejercicio 3:

Dado un array de números, usa `some` para verificar si existe algún número mayor a 100 y `every` para comprobar si todos los números son positivos.

```
// Enunciado
const numeros = [1, 50, 75, 99];
```

Ejercicio 4:

Dado un array de números desordenados, usa `sort` para ordenarlos de mayor a menor.

```
// Enunciado
const numeros = [5, 1, 8, 3, 10, 2];
```

Ejercicio 5:

Usa el método `find` para obtener el primer número divisible por 3 de un array, y `findIndex` para obtener su índice.

```
// Enunciado
const numeros = [4, 5, 9, 12, 7];
```

Ejercicio 6:

Dado un array de números, usa `forEach` para sumar todos los números.

```
// Enunciado
const numeros = [2, 4, 6, 8];
```

Ejercicio 7:

Dado un array de números, usa `slice` para obtener los primeros 3 números y `splice` para eliminar los dos últimos elementos del array original.

```
// Enunciado
const numeros = [10, 20, 30, 40, 50, 60];
```

Ejercicios de Objetos

Ejercicio 8:

Dado un objeto que representa un producto, usa `Object.keys` para obtener las claves del objeto y `Object.values` para obtener sus valores.

```
// Enunciado
const producto = { nombre: "Laptop", precio: 1000, stock: 5 };
```

Ejercicio 9:

Dado un objeto que representa un coche, usa `Object.entries` para convertirlo en un array de pares clave-valor.

```
// Enunciado
const coche = { marca: "Toyota", modelo: "Corolla", año: 2020 };
```

Ejercicio 10:

Dado un objeto de configuración, usa `Object.assign` para crear una copia del objeto con una propiedad adicional.

```
// Enunciado
const configuracion = { tema: "oscuro", idioma: "español" };
```

Ejercicio 11:

Crema un objeto que combine dos objetos dados usando el spread operator (`...`).

```
// Enunciado
const usuario = { nombre: "Ana", edad: 30 };
const detalles = { ciudad: "Madrid", ocupacion: "Ingeniera" };
```

Ejercicio 12:

Dado un objeto con varias propiedades, usa `delete` para eliminar una propiedad específica.

```
// Enunciado
const libro = { titulo: "1984", autor: "George Orwell", paginas: 328 };
```

Ejercicio 13:

Dado un objeto que representa una cuenta de usuario, usa `hasOwnProperty` para comprobar si tiene una propiedad específica.

```
// Enunciado
const cuenta = { usuario: "Juan", email: "juan@mail.com" };
```

Ejercicio 14:

Dado un objeto que representa un pedido, convierte todas sus claves a mayúsculas utilizando `Object.keys` y `reduce`.

```
// Enunciado
const pedido = { producto: "Silla", cantidad: 4, precio: 50 };
```

Ejercicios que mezclen Objetos y Arrays

Ejercicio 15:

Dado un array de objetos que representan estudiantes, usa `filter` y `map` para obtener los nombres de los estudiantes que tienen una calificación mayor o igual a 85. **Métodos sugeridos:** `filter`, `map`

```
// Enunciado
const estudiantes = [
  { nombre: "Carlos", calificacion: 80 },
  { nombre: "Ana", calificacion: 90 },
  { nombre: "Luis", calificacion: 88 },
  { nombre: "María", calificacion: 70 },
];
```

Ejercicio 16:

Dado un array de productos, usa **reduce** para calcular el costo total de todos los productos multiplicando el precio por la cantidad. **Métodos sugeridos:** **reduce**

```
// Enunciado
const productos = [
  { nombre: "Teclado", precio: 25, cantidad: 2 },
  { nombre: "Mouse", precio: 15, cantidad: 3 },
  { nombre: "Monitor", precio: 200, cantidad: 1 },
];
```

Ejercicio 17:

Dado un array de números y un objeto que clasifique esos números en pares e impares, usa **forEach** para llenar el objeto con los números correspondientes. **Métodos sugeridos:** **forEach**

```
// Enunciado
const numeros = [1, 2, 3, 4, 5, 6, 7, 8, 9];
const clasificacion = { pares: [], impares: [] };
```

Ejercicio 18:

Dado un objeto de empleados con sus salarios, usa **Object.entries** y **reduce** para calcular el salario promedio. **Métodos sugeridos:** **Object.entries**, **reduce**

```
// Enunciado
const empleados = {
  Juan: 1000,
  Maria: 1200,
  Pedro: 1500,
  Ana: 1100,
};
```

Ejercicio 19:

Dado un array de strings, usa **reduce** y **split** para contar cuántas veces aparece una letra específica en todos los strings. **Métodos sugeridos:** **reduce**, **split**

```
// Enunciado
const palabras = ["hola", "mundo", "javascript", "es", "genial"];
const letraABuscar = "a";
```

Ejercicio 20:

Dado un array de objetos que representan compras con propiedades `fecha` y `monto`, usa `sort` para ordenar las compras por fecha (más reciente primero). **Métodos sugeridos:** `sort`

```
// Enunciado
const compras = [
  { fecha: new Date(2023, 9, 21), monto: 200 },
  { fecha: new Date(2022, 1, 13), monto: 300 },
  { fecha: new Date(2023, 5, 10), monto: 150 },
];
```

Ejercicio 21:

Dado un array de objetos que representan tareas con una propiedad `completada`, usa `some` y `every` para verificar si hay alguna tarea incompleta y si todas están completas. **Métodos sugeridos:** `some`, `every`

```
// Enunciado
const tareas = [
  { tarea: "Lavar los platos", completada: true },
  { tarea: "Sacar la basura", completada: false },
  { tarea: "Comprar comida", completada: true },
];
```

Ejercicio 22:

Dado un array de objetos con propiedades `nombre` y `puntuacion`, usa `reduce` para encontrar el objeto con la puntuación más alta. **Métodos sugeridos:** `reduce`

```
// Enunciado
const jugadores = [
  { nombre: "Carlos", puntuacion: 120 },
  { nombre: "Ana", puntuacion: 180 },
  { nombre: "Luis", puntuacion: 150 },
];
```

Ejercicio 23:

Dado un objeto que contiene arrays como valores, usa `Object.values` y `flat` para obtener todos los elementos de esos arrays en un solo array. **Métodos sugeridos:** `Object.values`, `flat`

```
// Enunciado
const categorias = {
  frutas: ["manzana", "plátano", "pera"],
  verduras: ["lechuga", "tomate"],
  carnes: ["pollo", "cerdo"],
};
```

Ejercicio 24:

Dado un array de objetos con propiedades **nombre** y **edad**, usa **filter**, **map** y **reduce** para obtener la suma de las edades de las personas cuyo nombre empieza con una vocal. **Métodos sugeridos:** **filter**, **map**, **reduce**

```
// Enunciado
const personas = [
  { nombre: "Oscar", edad: 25 },
  { nombre: "Ana", edad: 30 },
  { nombre: "Luis", edad: 28 },
  { nombre: "Elena", edad: 22 },
];
```
