



Σavante

Desarrollo de Aplicaciones Web
Desarrollo Web Entornos Cliente

Tema 6.
Objetos y Arrays

Diego Rodero Pulido



Objetivo

El objetivo de este ejercicio es comprender y utilizar los objetos y arrays en JavaScript, incluyendo su creación, modificación y borrado, así como manejar elementos del DOM para interactuar con estos objetos.

Enunciado

Completa 10 ejercicios progresivos que cubren desde conceptos básicos de objetos y arrays hasta el desarrollo de una aplicación completa de gestión. Cada ejercicio está diseñado para construir sobre los conocimientos del anterior.

Objetivos de aprendizaje

1. **Crear y manipular objetos** en JavaScript con propiedades y métodos
2. **Trabajar con arrays** usando métodos básicos y avanzados
3. **Combinar objetos y arrays** para crear estructuras de datos complejas
4. **Aplicar métodos funcionales** como map, filter y reduce
5. **Desarrollar aplicaciones completas** usando objetos y arrays
6. **Manipular el DOM** para mostrar resultados dinámicamente

Recursos necesarios

- **Archivos proporcionados:**
 - ejercicio_1_alumno.html - Interfaz web con Bootstrap 5
 - ejercicio_1_alumno.js - Archivo JavaScript con estructura de ejercicios
- **Tecnologías a utilizar:**
 - HTML5
 - CSS3 (Bootstrap 5)
 - JavaScript vanilla (ES5)
 - Manipulación del DOM



Entrega

- El código del ejercicio, se subirá al repositorio del alumno de la asignatura, con una carpeta Tarea6, y con los dos archivos (ejercicio_1_alumno.html y ejercicio_1_alumno.js)
- Un PDF con capturas de pantalla mostrando:
 - El resultado después de realizar cada uno de los ejercicios.
 - En total, son 10 capturas.

En el documento PDF se indicará la **URL del repositorio del alumno**.

Configuración del Entorno

Normas de Desarrollo

- **NO modificar** el archivo HTML ni los nombres de funciones
- **Completar** los comentarios marcados con TODO
- **Usar sintaxis tradicional** de JavaScript (function, var, no arrow functions)
- **Mostrar resultados** en el DOM usando innerHTML, NO en la consola
- **Testar cada ejercicio** antes de continuar con el siguiente

Descripción de los ejercicios

Ejercicio 1: Crear Objeto Estudiante

Objetivo: Crear un objeto con propiedades básicas y un método

Conceptos: Propiedades de objetos, métodos, this

Instrucciones: Debes crear un objeto llamado "estudiante" que contenga las propiedades: nombre, apellidos, edad y curso. Además, debe incluir un método llamado mostrarInfo() que genere y retorne código HTML con toda la información del estudiante formateada de manera clara y legible.

Tareas a realizar:

- Crear objeto estudiante con nombre, apellidos, edad, curso



- Implementar método mostrarInfo() que retorne HTML
- Mostrar la información en el DOM

Ejercicio 1 - Crear Objeto Estudiante

Objetivo: Crear un objeto estudiante con propiedades (nombre, apellidos, edad, curso) y un método mostrarInfo() que devuelva HTML con la información.

Crear y Mostrar Estudiante

Ejercicio 2: Gestión de Array de Colores

Objetivo: Manipular arrays con métodos básicos

Conceptos: push(), pop(), bucles, arrays

Instrucciones: Trabaja con un array predefinido de colores. Implementa tres funciones: una para agregar nuevos colores al final del array usando push(), otra para eliminar el último color usando pop(), y una tercera para mostrar todos los colores en una lista HTML ordenada.

Tareas a realizar:

- Agregar colores al array usando push()
- Eliminar el último color con pop()
- Mostrar lista completa de colores



Ejercicio 2 - Gestión de Array de Colores

Objetivo: Trabajar con un array de colores usando métodos `push()`, `pop()` y mostrar el contenido dinámicamente.

Agregar ColorEliminar ÚltimoMostrar Colores

Ejercicio 3: Catálogo de Productos

Objetivo: Trabajar con arrays de objetos

Conceptos: Arrays de objetos, `sort()`, `filter()`

Instrucciones: Crea un array que contenga al menos 5 objetos producto, donde cada producto tenga las propiedades: nombre, precio y categoría. Implementa funciones para mostrar todos los productos, ordenarlos por precio de menor a mayor, y filtrar solo aquellos productos que cuesten más de 50 euros.

Tareas a realizar:

- Crear array de productos (nombre, precio, categoría)
- Ordenar productos por precio
- Filtrar productos caros (>50€)



Ejercicio 3 - Catálogo de Productos

Objetivo: Crear y gestionar un array de objetos producto (nombre, precio, categoría).
Practicar ordenamiento y filtrado.

Cargar Productos

Ordenar por Precio

Filtrar Productos > 50€

Ejercicio 4: Estudiante con Notas

Objetivo: Combinar objetos con arrays anidados

Conceptos: Objetos complejos, métodos de cálculo, validación

Instrucciones: Desarrolla un objeto estudiante más complejo que incluya un array de notas como propiedad. El objeto debe tener métodos para: agregar nuevas notas (con validación de 0 a 10), calcular el promedio de todas las notas, y mostrar todas las notas junto con el promedio calculado.

Tareas a realizar:

- Crear objeto con array de notas
- Implementar métodos para agregar notas
- Calcular promedio de calificaciones

Ejercicio 4 - Estudiante con Notas

Objetivo: Crear un objeto estudiante que contenga un array de notas y métodos para agregar notas y calcular el promedio.

Ingresa una nota (0-10)

Agregar Nota

Calcular Promedio

Mostrar Notas



Ejercicio 5: Gestión de Empleados

Objetivo: Aplicar filtros y búsquedas complejas

Conceptos: Filtrado avanzado, búsqueda por criterios

Instrucciones: Construye un sistema de gestión con un array de objetos empleado. Cada empleado debe tener: nombre, departamento, salario y antigüedad. Implementa funciones para cargar empleados de ejemplo, buscar empleados por departamento específico, y filtrar empleados que tengan un salario superior a 3000 euros.

Tareas a realizar:

- Crear sistema de empleados
- Buscar por departamento
- Filtrar por salario

Ejercicio 5 - Gestión de Empleados

Objetivo: Gestionar un array de objetos empleado con métodos de búsqueda y filtrado por departamento y salario.

Cargar EmpleadosBuscar por DepartamentoSalario > 3000€

Ejercicio 6: Métodos Avanzados de Arrays

Objetivo: Dominar métodos de manipulación de arrays

Conceptos: splice(), slice(), concat(), find(), indexOf()

Instrucciones: Practica el uso de métodos avanzados de arrays trabajando con un array de ciudades españolas. Implementa funciones que demuestren el uso de: splice() para eliminar elementos del medio del array, slice() para extraer porciones sin modificar el original, y find()/indexOf() para localizar elementos específicos.

Tareas a realizar:



- Eliminar elementos con splice()
- Extraer porciones con slice()
- Buscar elementos específicos

Ejercicio 6 - Métodos Avanzados de Arrays

Objetivo: Practicar splice, slice, concat, find, indexOf con un array de ciudades españolas.

Crear Array Ciudades

Eliminar del Medio

Extraer con Slice

Buscar Madrid

Ejercicio 7: Constructor de Vehículos

Objetivo: Crear múltiples objetos con función constructora

Conceptos: Funciones constructoras, this, instanciación

Instrucciones: Crea una función constructora llamada Vehiculo que permita crear múltiples objetos vehículo con propiedades como marca, modelo, año y velocidad. Cada vehículo debe tener métodos para acelerar (incrementar velocidad) y mostrar su información. Instancia varios vehículos y demuestra el funcionamiento de sus métodos.

Tareas a realizar:

- Crear función constructora Vehiculo
- Instanciar múltiples objetos
- Implementar métodos compartidos

Ejercicio 7 - Constructor de Vehículos

Objetivo: Crear múltiples objetos vehículo usando función constructora con propiedades y métodos.

Crear Vehículos

Acelerar Todos

Mostrar Info



Ejercicio 8: Matriz de Números

Objetivo: Trabajar con arrays multidimensionales

Conceptos: Arrays 2D, bucles anidados, operaciones matemáticas

Instrucciones: Desarrolla funciones para trabajar con arrays bidimensionales (matrices). Crea una matriz de 3x3 con números aleatorios, implementa una función para calcular la suma de la diagonal principal, y otra para mostrar la matriz completa en formato de tabla HTML con estilos apropiados.

Tareas a realizar:

- Crear matriz 3x3
- Calcular suma de diagonal
- Mostrar matriz en tabla HTML

Ejercicio 8 - Matriz de Números

Objetivo: Trabajar con arrays multidimensionales (matriz 3x3) y realizar operaciones matemáticas.

Crear Matriz 3x3Sumar DiagonalMostrar Matriz

Ejercicio 9: Métodos Funcionales

Objetivo: Aplicar programación funcional básica

Conceptos: map(), filter(), reduce()

Instrucciones: Domina los métodos funcionales más importantes de JavaScript trabajando con un array de números del 1 al 10. Implementa funciones separadas que demuestren: map() para duplicar todos los números, filter() para obtener solo los números pares, y reduce() para calcular la suma total de todos los elementos.

Tareas a realizar:

- Transformar datos con map()



- Filtrar elementos con filter()
- Reducir arrays con reduce()

Ejercicio 9 - Métodos Funcionales

Objetivo: Dominar los métodos map, filter y reduce con un array de números.

Crear Array Números

Duplicar (map)

Filtrar Pares

Sumar Todo (reduce)

Ejercicio 10: Biblioteca de Libros (Proyecto Final)

Objetivo: Desarrollar una aplicación completa CRUD

Conceptos: Todas las técnicas anteriores integradas

Instrucciones: Desarrolla una aplicación completa de gestión de biblioteca. Cada libro debe tener: título, autor, año de publicación y género. La aplicación debe permitir: agregar nuevos libros con validación de datos, mostrar toda la biblioteca, ordenar libros alfabéticamente por título, filtrar libros por género seleccionado, y mostrar solo libros publicados después del año 2020.

Tareas a realizar:

- Sistema completo de gestión
- Operaciones CRUD (Create, Read, Update, Delete)
- Múltiples filtros y ordenamientos
- Validación de datos



Ejercicio 10 - Biblioteca de Libros (Proyecto Final)

Objetivo: Crear una aplicación completa de gestión de biblioteca con funcionalidades CRUD (Crear, Leer, Actualizar, Eliminar).

Criterios de evaluación

Puntuación por Ejercicio:

- Ejercicios 1-3: **1 punto cada uno** (3 puntos total)
- Ejercicios 4-6: **1.5 puntos cada uno** (4.5 puntos total)
- Ejercicio 7-8: **2 puntos cada uno** (4 puntos total)
- Ejercicio 9: **2.5 puntos**
- Ejercicio 10: **6 puntos**
- **Total: 20 puntos**

Aspectos Evaluados:

Funcionalidad (60%): Los ejercicios funcionan correctamente

Código limpio (20%): Sintaxis correcta y comentarios adecuados

Creatividad (10%): Mejoras o características adicionales

Presentación (10%): Interfaz ordenada y profesional