

ACTIVIDAD PRÁCTICA 4 (AP4)

Título

Ejercicios básicos de programación (Arrays)

Objetivos

- Conocer el uso de Arrays / arreglos / vectores de una dimensión de forma básica para utilizarlas en cualquier lenguaje de programación.
- Saber recorrer y aplicar métodos propios de manipulación de Arrays utilizando JavaScript.

Temporalización

Previsto: Utilización de dos sesiones presenciales + trabajo en casa.

Proceso de desarrollo

Crea un directorio de proyecto donde puedas almacenar los ejercicios que se proponen a continuación.

Al finalizar las sesiones de clase dedicadas a esta práctica, se publicará una propuesta de solución para cada ejercicio planteado de tal forma que el alumnado pueda realizar una comparativa / corrección de forma autónoma.

Ante de comenzar, unas reflexiones interesantes... Estas personas deberían ser programadores, ¿no?:

“ Los resultados que consigues estarán en proporción directa al esfuerzo que aplicas ”

Denis Waitley

El esfuerzo constante – no la fuerza o la inteligencia – es la clave para liberar nuestro potencial

Winston Churchill

PARTE 1) LECTURA Y ESTUDIO DE DOCUMENTACIÓN.

Debes realizar una lectura pausada y un estudio posterior del **manual básico. Manejo de Arrays con JavaScript**, disponible en la sección de RECURSOS DIDÁCTICOS de este bloque. La comprensión y el estudio de dicho documento es básico para afrontar la realización de los ejercicios propuestos en la parte 2 de la actividad.

Repito... Es **sumamente importante, el estudio y comprensión del manual básico**, ya que desarrolla conceptos que habrá que poner en práctica en los ejercicios propuestos posteriormente. No es posible ni recomendable abordar las siguientes partes de la actividad sin haber completado correctamente esta sección.

PARTE 2) REALIZACIÓN DE EJERCICIOS PRÁCTICOS.

EJERCICIO 1. Desarrollar un programa en el que se pida al usuario un vector de 5 números enteros e indique en pantalla la media aritmética de todos sus elementos.

EJERCICIO 2. Crea un array de 10 posiciones de números (con valores pedidos por teclado). Visualiza el índice y el valor al que corresponde. Haz dos funciones, una para rellenar los valores del array y otra para mostrar sus elementos.

EJERCICIO 3. Programa que pida por teclado 10 números y los almacene en un array. Una vez introducidos los números, debe informar posteriormente sobre cuantos números son PARES y cuantos IMPARES.

EJERCICIO 4. El orden de clasificación de un concurso es el siguiente:

1º. Ana. - 2º Oswaldo – 3º. Raul – 4º. Celia - 5º. María – 6º. Antonio

Realiza un programa que:

- 1). Almacene dicha clasificación en un array e imprima la clasificación inicial.
- 2) Se supone que el concurso continúa y se van modificando las posiciones anteriores, de tal forma que debemos de cambiar el estado del array atendiendo a las siguientes situaciones:
 - Celia adelanta a Raúl.
 - Antonio es descalificado y se elimina del concurso.
 - Detrás de Ana y antes de Oswaldo se clasifican dos nuevos Concursantes: Roberto y Amaya, en ese orden.
 - Hay una nueva participante que pasa a encabezar la clasificación: Marta.
- 3) Finaliza el programa visualizando la clasificación actualizada y comprueba que se ha hecho correctamente los cambios indicados en el punto 2.

EJERCICIO 5. Desarrollar un programa en el que se pidan al usuario dos arrays de números enteros e indique en pantalla el producto de los elementos que ocupan la misma posición. En la petición de cada array, la introducción de elementos terminará cuando se suministre el valor cero (o). Si los Arrays proporcionados no tienen la misma longitud, se ha de escribir en pantalla un mensaje indicativo y no se realizará el proceso.

EJERCICIO 6. Crea dos Arrays de números positivos entre 0 y 20 que sean generados de forma aleatoria. El tamaño de los Arrays debe ser de 10 elementos. Después, crea una función que reciba como parámetros los dos Arrays creados y devuelva un nuevo Array con la suma de la posición cero del array1 con el del array2 y así sucesivamente hasta su último elemento. Por último, muestra (en bruto) el contenido de cada array (array1, array2 y el resultante de la suma de los elementos de ambos).

EJERCICIO 7. Desarrollar un programa en el que se pidan al usuario dos vectores de marcas de coches y motos (Array coches y Array motos). En cada caso, el programa deberá preguntar si se desean introducir más marcas y finalizará la introducción de datos cuando se indique 'N'. Una vez introducidos los datos, se debe construir el vector resultado de “concatenar” los vectores coches y motos, es decir, poner los elementos de motos a continuación de los de coches en un nuevo vector (transportes), y finalmente se visualicen (en bruto) todos los elementos de la concatenación.

EJERCICIO 8. Desarrollar un programa en el que se pidan al usuario un array de números enteros -hasta que se introduzca el número -1 que no formará parte del array-. Tras visualizar todos los números introducidos, debe indicar si dicho vector es capicúa, es decir, la secuencia de sus elementos es igual vista de delante hacia atrás y de detrás hacia delante.

EJERCICIO 9. Crea un array de números de un tamaño especificado por teclado. El array contendrá los números primos a partir del 1 y hasta completar el tamaño especificado del array. Utiliza una función que calcule si el número pasado por parámetro es primo o no.

EJERCICIO 10. Crea un array de caracteres que contenga de la 'A' a la 'Z' (solo las mayúsculas). Después, ve pidiendo posiciones del array por teclado y si la posición es correcta, la letra correspondiente a esa posición se añadirá a un nuevo array que se mostrará al final del programa en forma de string o cadena. Se dejará de insertar posiciones cuando se introduzca un -1. Por ejemplo, si escribo los siguientes números:

```
0 // Añadirá la 'A'
5 // Añadirá la 'F'
25 // Añadirá la 'Z'
50 // No hará nada ya que la posición no pertenece al array.
-1 // Finalizará y visualizará "Cadena resultante: A,F,Z"
```

EJERCICIO 11. Desarrollar un programa en el que se pida al usuario un vector de 10 números enteros (v) y se construya un nuevo vector (u) con los mismos elementos que (v) pero ordenados de menor a mayor. El programa ha de escribir en pantalla el nuevo vector donde aparezcan los elementos ordenados.

EJERCICIO 12. Escribir un programa que simule el lanzamiento de dos dados. Haz uso de la función *Math.random* para obtener números aleatorios entre 1 y 6 para cada uno de los lanzamientos de los dos dados. Debes sumar el resultado de lanzar los dos dados en cada "tirada" y anotar en un array el número de apariciones de dicha suma, repitiendo 36.000 veces esta operación. Los resultados de cada tirada estarán entre el valor 2 y el 12. ¿Cuántas veces ha salido cada valor? ¿Qué porcentaje representa respecto al total?

EJERCICIO 13. En España cada persona está identificada con un Numero de Identificación Fiscal (NIF) en el que figura un número y una letra, por ejemplo: 56999545W.

Crea un programa que calcule la letra de un NIF a partir del número de DNI que introduzca el usuario. Es decir, se debe pedir el DNI sin la letra por teclado y el programa nos devolverá el DNI completo (con la letra), es decir, el NIF.

Para calcular la letra a partir del DNI se sigue el siguiente algoritmo: se debe tomar el resto de dividir nuestro número de DNI entre 23. El resultado debe estar por tanto entre 0 y 22. La letra correspondiente será a la que apunte dicho índice en la siguiente tabla:

RESTO	0	1	2	3	4	5	6	7	8	9	10	11	12
LETRA	T	R	W	A	G	M	Y	F	P	D	X	B	N
RESTO	13	14	15	16	17	18	19	20	21	22			
LETRA	J	Z	S	Q	V	H	L	C	K	E			

Crea una función *obtenerLetra(numeroDNI)* donde según el resultado de la anterior fórmula busque en un array de caracteres la posición que corresponda a la letra.

Por ejemplo, si introducimos el DNI 20267079, el resto de dividirlo por 23 sería 8, luego la letra sería la P, que es la que ocupa esa posición en la tabla de caracteres.

EJERCICIO 14. Desarrolla un programa que genere 10 fechas válidas (al azar) del actual año con el siguiente formato:

```
Fecha 1: 23 de Marzo de 2018
Fecha 2: 15 de Enero de 2018
...
Fecha 10: 23 de Diciembre de 2018
```

Una fecha será válida si el día indicado es correcto para el mes generado (por ejemplo, el "30 de Febrero" no es una fecha correcta), el mes está entre "Enero" y "Diciembre", y el año es el 2018.

Piensa la lógica de programación más sencilla que se te ocurra para resolver el problema. En tu programa debes utilizar al menos el siguiente array:

```
let meses = ["Enero", "Febrero", "Marzo", "Abril", "Mayo", "Junio", "Julio", "Agosto", "Septiembre", "Octubre", "Noviembre", "Diciembre"];
```

Evaluación

Esta actividad práctica no es una actividad de evaluación. Su realización forma parte del 10% de evaluación correspondiente al seguimiento e interés de la asignatura por parte del alumnado. Además, supone un refuerzo / estudio de cara a la realización de las actividades de evaluación entregables y las pruebas teórico-prácticas.

Recursos

Disponibles en plataforma (Recursos didácticos del Bloque1):

- Transparencias / Materiales didácticos / Links de interés.

Anotaciones
