

UD2

Ejercicios: Introducción al lenguaje JavaScript

Ejercicio 1	2
Ejercicio 2	2
Ejercicio 3	3
Ejercicio 4	3
Ejercicio 5	3
Ejercicio 6	3
Ejercicio 7	3
Ejercicio 8	3
Ejercicio 9	3
Ejercicio 10	4
Ejercicio 11	4
Ejercicio 12	4
Ejercicio 13	5
Ejercicio 14	5
Ejercicio 15	5
Ejercicio 16	6
Ejercicio 17	6
Ejercicio 18	6

Ejercicio 1

Modificar el siguiente script para que:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>El primer script</title>

<script type="text/javascript">
    console.log("Hola Mundo!");
</script>
</head>

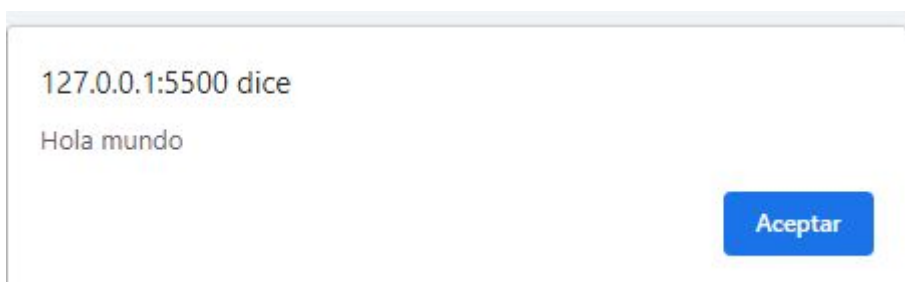
<body>
<p>Esta página contiene el primer script</p>
</body>
</html>
```

1. Todo el código JavaScript se encuentre en un archivo externo llamado **codigo.js** y el script siga funcionando de la misma manera.
2. Después del primer mensaje, se debe mostrar otro mensaje que diga "Soy el primer script".
3. Añadir algunos comentarios que expliquen el funcionamiento del código.
4. Añadir en la página HTML5 un mensaje de aviso para los navegadores que NO tengan activado el soporte de JavaScript.

Ejercicio 2

Modificar el primer script del ejercicio anterior para que:

- 2.1. El mensaje que se muestra al usuario se almacene en una variable llamada mensaje y el funcionamiento del script sea el mismo.
- 2.2. El mensaje mostrado sea parecido al de la siguiente imagen:



- 2.3. Modificar el programa para que pida primeramente tu nombre por pantalla, y luego sea capaz de mostrar un mensaje como el anterior pero que te salude.

Ejercicio 3

Crear un array llamado meses y que almacene el nombre de los doce meses del año. Mostrar por pantalla los doce nombres utilizando la función **alert()**.

Ejercicio 4

Realizar un programa que lea por teclado dos números, si el primero es mayor al segundo mostrar la suma y la diferencia de ambos, y en caso contrario informar el producto y la división del primero respecto al segundo.

Ejercicio 5

Confeccionar una función a la cual le envíe tres enteros y los muestre ordenados de menor a mayor.

Ejercicio 6

Crear un array con los valores: [0,7,2,3,4,2,6,2,8,2]. Luego borrar las componentes donde hay un 2 y en dichas posiciones, al mismo tiempo insertar 2 elementos con el valor 1. Es decir, cada vez que se encuentre un 2, lo borras y los sustituyes por 2 posiciones en el array conteniendo ambas el valor 1.

El resultado tendría que ser: [0,7,1,1,3,4,1,1,6,1,1,8,1,1]

Debes de hacer uso del método splice tanto para el borrado como para insertar.

Ejercicio 7

Crear un vector de 10 elementos con valores “aleatorios” comprendidos entre 0 y 1000. Buscar el menor y luego generar otro vector a partir de dicha posición hasta el final. Para manipular el vector debes de hacer uso de la función “slice”. También debes de buscar por internet como generar números random en javascript comprendidos en un rango determinado.

Ejercicio 8

Crear un vector con 10 elementos enteros con valores aleatorios. Ordenarlos de menor a mayor. Luego invertir el vector para verlo de mayor a menor. **Importante:** Para las 2 ordenaciones debes de emplear 1 función/método específica que ya existen para tal cometido.

Ejercicio 9

A partir del siguiente array que se proporciona:

```
var valores = [true, 5, false, "hola", "adios", 2];
```

1. Determinar cuál de los dos elementos **de texto** es mayor.
2. Utilizando exclusivamente los dos valores **booleanos** del array, determinar los operadores necesarios para obtener un resultado true y otro resultado false.
3. Determinar el resultado de las **cinco operaciones matemáticas** realizadas con los **dos elementos numéricos**.

Ejercicio 10

Escribe una función en JavaScript que sea capaz de clonar un array. Por ejemplo, para los siguientes datos de ejemplo:

```
console.log( arrayClonado([1, 2, 4, 0]));
```

```
console.log( arrayClonado([1, 2, [4, 0]])); // La dificultad es que dentro de una de las posiciones del array, puede que haya otro array y dentro ese, otro, y otro, y otro...
```

Las salidas por consola serían:

```
[1, 2, 4, 0]
```

```
[1, 2, [4, 0]]
```

Ejercicio 11

Escribe un simple programa JavaScript que obtenga los siguientes resultados por consola.

Array inicia:

```
[ "Rojo", "Verde", "Blanco", "Negro" ]
```

Salida por consola:

```
"Rojo,Verde,Blanco,Negro"
```

```
"Rojo + Verde + Blanco + Negro"
```

Ejercicio 12

Escribe una función capaz de fusionar 2 arrays y que elimine todos los elementos duplicados.

Datos de ejemplo:

```
let array1 = [1, 2, 3];
let array2 = [2, 30, 1];
console.log( fusionarArrays( array1 , array2));
```

Salida: [3, 2, 30, 1]

Ejercicio 13

El cálculo de la letra del **Documento Nacional de Identidad (DNI)** es un proceso matemático sencillo que se basa en obtener el resto de la división entera del número de DNI y el número 23. A partir del resto de la división, se obtiene la letra seleccionándola dentro de un array de letras.

El array de letras es:

```
var letras = ['T', 'R', 'W', 'A', 'G', 'M', 'Y', 'F', 'P', 'D', 'X', 'B', 'N', 'J', 'Z', 'S', 'Q', 'V', 'H', 'L', 'C', 'K', 'E', 'T'];
```

Por tanto si el resto de la división es 0, la letra del DNI es la T y si el resto es 3 la letra es la A.

Con estos datos, elaborar un pequeño script que:

1. Almacene en una variable el número de DNI indicado por el usuario y en otra variable la letra del DNI que se ha indicado. (Pista: si se quiere pedir directamente al usuario que indique su número y su letra, se puede utilizar la función **prompt()**)
2. En primer lugar (y en una sola instrucción) se debe comprobar si el número es mayor que 0 o menor que 999999999. Si ese es el caso, se muestra un mensaje al usuario indicando que el número proporcionado no es válido y el programa no muestra más mensajes.
3. Si el número es válido, se calcula la letra que le corresponde según el método explicado anteriormente.
4. Una vez calculada la letra, se debe comparar con la letra indicada por el usuario. Si no coinciden, se muestra un mensaje al usuario diciéndole que la letra que ha indicado no es correcta. En otro caso, se muestra un mensaje indicando que el número y la letra de DNI son correctos.

Ejercicio 14

El factorial de un número entero n es una operación matemática que consiste en multiplicar todos los factores $n \times (n-1) \times (n-2) \times \dots \times 1$. Así, el factorial de 5 (escrito como 5!) es igual a: **$5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$**

Utilizando la estructura for, crear un script que calcule el factorial de un número entero.

Ejercicio 15

Realiza las siguientes operaciones con un vector/array:

1. Crear un array con las siguientes elementos: “Sandía” y “Melón”.
2. Añade “Kiwi” al final del array.
3. Reemplaza el valor justo en la mitad del array por “Melocotón”. Ten cuidado porque el código para encontrar el elemento del medio debería de funcionar para arrays de cualquier tamaño.
4. Saca el primer elemento del vector y muéstralo (el valor será eliminado del vector).
5. Añade por el principio “Pera”.
6. Muestra todo el vector.

7. Muestra todo el vector ordenado alfabéticamente.

Ejercicio 16

Código de rotación

El objetivo de esta actividad es utilizar alguna de las funciones predefinidas del lenguaje JavaScript. Puede utilizar las funciones: `charAt`, `indexOf`, `alert`, `length`, `prompt` y `toUpperCase`.

El dictador Julio César utilizaba un código cuando quería mantener en secreto un mensaje. El cifrado consistía en sustituir la primera letra del alfabeto (A) por la cuarta (D) y así sucesivamente con las otras, es decir, B, por la quinta (E) la tercera (C) para la sexta (F) ...

El alfabeto latino que utilizaba Julio César constaba de 21 letras, por lo tanto la sustitución de letras para cifrar o descifrar mensajes queda descrita como sigue:

Alfabeto original: **ABCDEFGHIJKLMNOPQRSTUVWXYZ**

Alfabeto cifrado: **DEFGHIJKLMNOPQRSTUVWXYZA**

Este tipo de cifrado es un código de rotación, donde la clave de rotación es 3. Escribe un programa que permita cifrar utilizando un código de rotación.

→ Para otro año, debería añadir un ejemplo de entrada y de salida

Ejercicio 17

Escribe un programa JavaScript que sume todos los múltiplos de 3 y de 5 que estén por debajo de 1000.

Ejercicio 18

Escribe un programa Javascript que encuentre todos los “números de armstrong” de 3 dígitos.

Un número de Armstrong de 3 dígitos es un entero tal que la suma de sus cubos de sus dígitos es igual al número en sí. Ejemplo: 371 es un número de Armstrong porque $3^3 + 7^3 + 1^3 = 371$