

# EJERCICIOS DE JAVASCRIPT

1. Dadas las variables `a`, `b`, `c`, `d` y `e` de tipo `int`, inicializadas con los valores `-9`, `21`, `-35`, `51` y `33` respectivamente, crea una sola expresión que realice las operaciones siguientes:
  - Sumar los valores almacenados en `a` y `b` y multiplicar el resultado por el valor almacenado en `e`.
  - Dividir el resultado de la operación anterior entre el resto de dividir el valor almacenado en `d` entre el valor almacenado en `c`.
  - Almacenar en una variable `f` el resultado parcial que se obtiene de la operación descrita en el primer punto.

Ejecutar con Node.js.

2. Dadas las variables `a`, `b`, `c` y `d` de tipo `int`, escribe dos expresiones distintas, pero que ambas retornen `true` si `a` es menor que `b` o `c` es distinto de `d`. Ejecutar con Node.js.
3. Crea un script que muestre la suma de los 50 primeros números impares. Ejecutar con Node.js.
4. Crea una página web que permita generar el triángulo de Floyd para un número de filas que el usuario introducirá a través de un formulario. Por ejemplo, si el número de filas es 4 el triángulo de Floyd será:

```
1
2 3
4 5 6
7 8 9 10
```

5. Crea una página web que permita al usuario introducir número `n` y use JavaScript para dibujar en la misma triángulos usando asteriscos. Por ejemplo, para `n=4` se podrían dibujar los triángulos siguientes:

```
****      ****      *          *          *          ****
***       ***       **         **         ***       *****
**        **        ***        ***        *****      ***
*         *         ****       ****       ****       *
```

La página debe permitir la elección del tipo de triángulo que se dibujará.

6. Crea una función que calcule el coste total en euros de un viaje sumando los gastos de alojamiento, alimentación y entretenimiento.

Ponla a prueba obteniendo los datos de entrada desde un formulario HTML. El resultado retornado se mostrará en la misma página web. Validar los datos de entrada mediante HTML o mediante JavaScript.

7. Crea un script que reciba una cadena que represente un color HTML (p. ej. "#10A5BF") y convierta sus componentes RGB a componentes en el espacio YIQ, según las expresiones matemáticas siguientes:

$$y = 0,299r + 0,587g + 0,114b$$

$$i = 0,596r - 0,275g - 0,321b$$

$$q = 0,212r - 0,528g + 0,311b$$

Se obtendrán los datos de entrada desde un formulario HTML y se mostrarán los resultados en la misma página. Validar el dato de entrada mediante HTML o mediante JavaScript.

8. Crea una función que calcule la fuerza gravitatoria entre dos masas usando la fórmula de la gravitación universal

$$F = G \times \frac{m1 \times m2}{d^2}$$

$G$  es la constante de gravitación universal, cuyo valor es:

$$G = 6.67384 \times 10^{-11} \frac{Nm^2}{kg^2}$$

Ponla a prueba obteniendo el dato de entrada desde un formulario HTML. El valor retornado se mostrará en la misma página web. Validar el dato de entrada mediante HTML o mediante JavaScript.

9. Crea dos funciones que calculen el perímetro y el área de un círculo partir del radio usando las expresiones siguientes:

$$\text{perímetro} = 2\pi r \quad \text{area} = \pi r^2$$

Ponlas a prueba obteniendo el dato de entrada desde un formulario HTML. El valor retornado se mostrará en la misma página web. Validar el dato de entrada mediante HTML o mediante JavaScript.

10. Crea una página web que use JavaScript para convertir grados Celsius a Fahrenheit o viceversa. El resultado de cada conversión se mostrará con una precisión de tres decimales. Validar los datos de entrada mediante HTML o mediante JavaScript.

11. Crea un script para ejecutar con Node.js que genere los primeros  $N$  términos de la secuencia de Fibonacci y los muestre en la consola, siendo  $N$  un número introducido por teclado.

12. Crea una página web que permita que el usuario introduzca dos números enteros y utilice JavaScript para mostrar en la misma todos los números primos que hay entre ellos y cuántos son. Validar los datos de entrada mediante HTML o mediante JavaScript.

13. Crea una página web que permita que el usuario introduzca un número entero  $N$  mayor que cero y, usando JavaScript, muestre en la misma el resultado de calcular la

suma  $\frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{N}$ . Validar el dato de entrada mediante HTML o mediante JavaScript.

14. Crea una página web que permita jugar al siguiente juego de adivinación:

- La página genera un número aleatorio entre 1000 y N, siendo N un número entero aleatorio entre 10.000 y 100.000.
- En la página se mostrará el mensaje “he pensado un número entre 1000 y N, adivina cuál es”.
- El usuario tendrá que introducir números mediante un formulario hasta que acierte el que ha pensado el ordenador.
- Cada vez que introduzca un número y no acierte la página le dirá si es mayor o menor que el que ha pensado y le volverá a preguntar.
- Una vez acertado el número, la página dirá cuántos intentos han sido necesarios para adivinar el número y permitirá reiniciar el juego.

15. Crea una página web que permita al usuario introducir dos cadenas y muestre el número de veces que la segunda está contenida en la primera.

16. Crea una página web que permita al usuario introducir su nombre y muestre el resultado de sustituir los caracteres centrales (todos menos el primero y el último) por asteriscos. Se ha de tener en cuenta que cada nombre puede estar precedido y/o seguido de un número arbitrario de espacios en blanco y que, de ser así, estos se deben eliminar antes de obtener el resultado.

17. Crea una página web que permita al usuario introducir secuencias de caracteres con el formato siguiente:

*LetraNúmeroCarácter*

Por ejemplo, las siguientes son cadenas de este tipo: M135t, b57X, n1335\$ o L91. Cada secuencia se procesará para generar un valor numérico de la forma siguiente:

- Si la primera letra es mayúscula, el número que hay entre ella y el último carácter se multiplicará por el valor numérico de dicho carácter. En caso contrario, en lugar de la multiplicación se realizará una división.
- Si el último carácter es alfabético, se calculará la raíz cuadrada del resultado anterior. Si es un dígito, se calculará su logaritmo neperiano. En caso contrario se calculará el resultado de elevar el número de Euler a dicho resultado.

Se asume que no se cometen errores en la entrada de datos y todas las secuencias son válidas.

18. Asigna a una variable una expresión de función que acepte un array y un número entero **n** y, usando el método **slice** del array, retorne:

- Los **n** primeros elementos del array si **n** es mayor o igual que cero.
- El array completo si **n** es mayor o igual que la longitud del array.

- El primer elemento del array si no recibe el parámetro `n`.
- `undefined` si no recibe ningún parámetro.
- Un array vacío si `n` es negativo.

19. Crea una función que acepte un array y retorne:

- `undefined` si la función no recibe nada o recibe un array vacío.
- El elemento que se repite con mayor frecuencia.

20. Crea una función que acepte un array y desordene (baraje) sus elementos.

21. Crea una función que acepte dos arrays y retorne otro array que contenga en cada posición la suma de los elementos que se encuentra en la misma posición de los arrays recibidos.

22. Crea una función que acepte un array de cualquier número de dimensiones y lo aplane retornando el resultado de almacenar todos sus elementos en un nuevo array unidimensional.

23. Crea las siguientes funciones:

- Función que acepte dos arrays y retorne su unión.
- Función que acepte dos arrays y retorne su intersección.
- Función que acepte dos arrays y retorne su diferencia.

24. Crea una función que acepte un array y elimine en él todos los elementos que se puedan interpretar como false.

Ejemplo:

```
[NaN, 0, 23, false, -7, undefined, null, 47, ""] ⇒ [ 23, -7, 47]
```

25. Crea una función que acepte un array de objetos de tipo {autor: '...', título: '...', id: n} y ordene su contenido por autor.

26. Crea una función que acepte dos arrays y retorne un nuevo array que contenga la mezcla de ambos sin duplicados.

27. Insertar en un documento HTML un script de JavaScript que inserte en el documento (objeto *document*) una tabla en la que se muestren los 25 primeros naturales y su factorial:

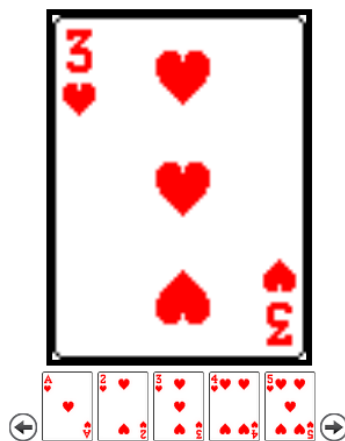
n	n!
1	1
2	2
3	6
4	24
5	120
6	720
7	5040
8	40320
9	362880
10	3628800
11	39916800
12	479001600
13	6227020800
14	87178291200
15	1307674368000
16	20922789888000
17	355687428096000
18	6402373705728000
19	121645100408832000
20	2432902008176640000
21	51090942171709440000
22	1.1240007277776077e+21
23	2.585201673888498e+22
24	6.204484017332394e+23
25	1.551121004330986e+25

28. Crear una página web con un botón cuya pulsación muestre en una ventana emergente el nombre del navegador, su versión y si está o no habilitado JavaScript (usar el objeto *navigator*).
29. Crear una página web que contenga una lista de colores desplegable. Al seleccionar un color el fondo de la página cambiará a ese color.
30. Crear una página web que muestre un cronómetro con tres botones: iniciar, parar y reset.
31. Crear una página web que contenga una imagen de forma que, al pasar el ratón por encima de ella, la imagen cambie, restaurando la original cuando el ratón deje de estar encima.
32. Crear una página web con dos botones: Añadir campo y Eliminar campo. Pulsando el primero se añadirá a la página un campo compuesto por una etiqueta y un cuadro de texto, hasta un máximo de 10. El texto de la etiqueta será "campo *n*", siendo *n* el número de orden en que se creó el campo. Pulsando en el segundo se eliminará el último campo añadido.
33. Crear una página web que incluya dos cajas de texto, con dos botones que permitan pasar el contenido de cada uno de los Text al otro.

**Introduce un texto en una caja y pulsa uno de los botones**

34. Crear una página web que muestre una galería de imágenes con el formato que se muestra en el ejemplo siguiente:



Las flechas permitirán seleccionar la imagen que se mostrará en grande. La imagen que se muestra en grande siempre estará en el centro de las miniaturas, excepto si es la primera, segunda, penúltima o última de la lista. La miniatura que se esté mostrando se resaltará de alguna forma.