

# Comenzando con JavaScript

**NOTA:** los objetivos de aprendizaje de esta tarea buscan las siguientes metas:

- a) Diferenciar entre tipos de datos JavaScript. Identificar secuencias de escape.
- b) Comprender comportamiento datos booleanos y operadores relacionales.
- c) Manejar operadores de bit en JavaScript. Manejar operadores aritméticos.
- D) Combinar bucles y sentencias condicionales. Manejar uso de String.
- e) Manejar desestructuración de Arrays y funciones.

## Arrays, listas, vectores o arreglos de una dimensión BLOQUE 1.

### 1. Construye un programa que compruebe los tipos de datos que devuelve el operador "typeof"

(Página 71) Construye un programa que inicialice de forma estática los diferentes valores posibles que puede devolver el operador "typeof" de JavaScript y los muestre en una ventana emergente "alert(...);"

### 2. Construye un programa que pinte una cara haciendo uso de las secuencias de escape

(Página 77) Construye un programa que pinte una cara haciendo uso de las secuencias de escape de la página 77. Esta secuencia se debe mostrar en una ventana emergente "alert(...);"

### 3. Construye un programa que permita probar los valores Booleanos

(Página 78) Construye un programa que a partir de los posibles valores que se pueden considerar verdaderos o falsos, cuando se interpretan como booleanos, muestre como evalúa JavaScript esos valores. Los valores se deben poder probar a través de la etiqueta HTML "Select". Los posibles valores deben estar precargados en la etiqueta "Select" indicada anteriormente.

### 4. Construye un programa que permita probar los operadores relacionales

(Página 79) Construye un programa que permita comparar diferentes valores, incluyendo la comparación estricta. Para ello se debe hacer uso de tres etiqueta "Select", la primera debe contener el primer valor, la segunda el tipo de operador relacional y la tercera el segundo valor. Al pulsar en texto que indique "comparar" se debe realizar la comparación. Los valores deben estar precargados en las etiquetas "select".

### 5. Construye un programa que calcule la dirección de red

(Página 83) Construye un programa que a partir de una dirección IP y una máscara de red escritas en decimal, permita convertir ambos números en binario y calcule la dirección de red.



## 6. Construye un programa que permita realizar operaciones matemáticas

**IMPORTANTE:** no se puede utilizar el método `eval()`, ni cualquier otro método que automatice el proceso. El objetivo del ejercicio es evaluar los caracteres de la cadena y almacenarlos en variables para realizar las operaciones haciendo uso de sentencias condicionales, bucles, etc.

**(Páginas 79 y 120)** Construye un programa que permita realizar operaciones matemáticas a través de los operadores aritméticos (página 79), el programa debe analizar una cadena de entrada, que puede tener una preferencia de operación con paréntesis. Algunos ejemplos de las cadenas de entrada que puede recibir el programa son las siguientes:

$3+(5*4)$  ,  $6*(3+2)$  ,  $8\%(2+3)$  ,  $3+4$  ,  $(3-1)*3$  ,  $25/(3+2)$

## 7. Construye un programa que permita mostrar triángulos, cuadrados y rectángulos de asteriscos

**(Página 96)** Construye un programa que le solicite al usuario el tipo de figura a mostrar, y el número de longitud que desea mostrar y en base a estos datos muestre un triángulo, cuadrado o rectángulo. Hay que tener en cuenta que para el cuadrado solo hace falta solicitar al usuario un lado, puesto que son los 4 lados iguales.

## 8. Construye un programa que permita modificar una letra en otra letra de una cadena de texto

**(Página 122)** Construye un programa que permita solicitar al usuario una cadena de texto, sobre la cadena de texto el programa debe permitir al usuario:

- Solicitar al usuario dos caracteres, y sustituir el primer carácter por el segundo en la cadena de texto que se solicito al usuario.

## 9. Construye un programa que reciba una cadena con un nombre y apellidos y lo separe

**(Página 125)** Construye un programa que reciba por teclado en una sola entrada, el nombre y los apellidos de una persona. El programa debe realizar las siguientes acciones:

- El programa debe convertir la primera letra del nombre y los apellidos en mayúscula.
- Debe dejar en minúscula o convertir en minúscula las letras que no sean la primera.
- Debe separar en tres valores el nombre, primer apellido y segundo apellido.

## 10. Construye un programa que permita calcular la media de los valores que indique el usuario

**(Página 140 y 174)** Construye un programa que permita al usuario introducir diferentes valores numéricos hasta que introduzca la letra "n". Se tiene que tener en cuenta lo siguiente:

- Si el usuario introduce un valor que no es numérico se le debe solicitar al usuario de nuevo indicándole que ese valor no es correcto.
- Se deben introducir valores hasta que el usuario introduzca la letra "n"
- La media se debe calcular de todos los valores introducidos, por lo que se deben ir contando el número de valores que ha introducido el usuario.