

Retos en JavaScript

Reto 001

Programar un procedimiento que pida dos números al usuario, mediante **prompt**, y luego muestre en pantalla el resultado de hacer estas cinco operaciones:

1. sumar los números
2. restar
3. multiplicar
4. dividir
5. elevar, por ejemplo $2^3=8$

Crear una función para cada una de las cinco operaciones anteriores, usando **function**.

Reto 002

Hacer otra versión del caso anterior, pero sin usar **prompt** y usando un formulario y un botón para capturar los datos que proporciona el usuario.

Reto 003

Crea un procedimiento que muestre en pantalla los números del 1 al 10, ambos incluidos, y luego los números pares entre 100 y 120, ambos incluidos. Usa **For** para conseguirlo. Cómo sería el código si en lugar de **For** se usa **While**. Y qué diferencia existe si en lugar de **While** se usa **Do...While**.

Reto 004

Crea un formulario con dos cajas de texto donde el usuario pueda poner el valor inicial y el valor final para hacer la serie, y tres botones de radio donde se pueda pedir que liste sólo los pares, o solo los impares, o todos. La opción 'todos' será la que aparezca por defecto. Crear un botón de Reset que ponga los botones de radio en la posición inicial por defecto, que consiste en que la opción 'todos' es la elegida.

Si el valor inicial es mayor que el final, se listarán los números en orden descendente.

Crear un botón para lanzar el procedimiento. Si los valores introducidos por el usuario no son válidos, mostrar un mensaje de error y permitir que se vuelva a poder introducir los valores. No serán valores válidos si se ponen números con decimales, o texto, o se deja vacío.

Reto 005

Deseamos que se generen números aleatorios enteros entre 1 y 100, ambos incluidos.

Estos números se van mostrando en pantalla hasta que salga uno que está entre 90 y 100, ambos incluidos, momento en el que ya no se muestran en pantalla más valores y el juego

termina. Por tanto, el último número mostrado en pantalla será un número que estará entre 90 y 100, ambos incluidos.

Reto 006

Hacer otra versión del caso anterior donde los números que detienen el juego no están entre 90 y 100, sino que son los pares entre 40 y 60, ambos incluidos. Los impares aunque estén entre 40 y 60, no detienen el juego.

Además queremos que se juegue esto veinte veces y que al final diga cuántos valores ha mostrado por pantalla la tirada que más valores ha llegado a mostrar.

Reto 007

Vamos a trabajar con arrays. Construir un **array** que contenga el nombre de 8 ciudades. Por ejemplo, las siguientes ciudades pueden ser válidas. Madrid, Londres, Dublin, Roma, Berlín, Amsterdam, París y Helsinki. Generar un dado de ocho posiciones, con las siguientes probabilidades: 15%, 9%, 14%, 5%, 23%, 16%, 8%, 10%, que corresponden a cada una de las ciudades respectivamente. Deseamos que aparezca en pantalla un botón y que al pulsarlo se genere un listado de 100 ciudades extraídas entre las ocho del array y que aparezcan de forma aleatoria cada una de ellas según su probabilidad. Para ello utilizar **switch...case**. Finalmente mostramos las ciudades y el número de veces que aparece cada una dentro del listado de 100 posiciones.

Reto 008

Deseamos hacer una aplicación que calcule la letra del DNI (Documento Nacional de Identidad) cuando el usuario proporciona el número de su DNI. En España el DNI se compone de un número y una letra. La letra se obtiene aplicando un algoritmo a los números. Se trata de un carácter de control. El algoritmo es el siguiente.

- Se calcula el resto de dividir el número del DNI entre 23
- Al resto se le suma 1
- La letra del DNI es la que ocupa la posición cuyo valor hemos calculado en el paso anterior dentro del siguiente listado: TRWAGMYFPDXBNJZSQVHLCKE.

Son 23 letras porque se ha quitado la I para no confundirla con el 1, se ha quitado la Ñ, se ha quitado la O para no confundirla con el 0, se ha quitado la U para no confundirla con la V.

Deseamos crear una caja de texto donde el usuario introduzca el número del DNI y un botón para lanzar el proceso. Programarlo de dos formas diferentes:

1. Usando un **array**
2. Usando funciones de cadena: cadena.**substr**(inicio[, longitud])

Reto 009

Crear un juego donde el usuario adivina un número secreto. El programa genera un número secreto entero entre 1 y 100. El usuario dice un número y la máquina responde con tres posibilidades.

- Si el número secreto es menor, lanza el mensaje “Es menor” y permite seguir intentándolo.
- Si el número secreto es mayor, lanza el mensaje “Es mayor” y permite seguir intentándolo.
- Si el número que dice el usuario coincide con el número secreto, se felicita al usuario y el juego termina.

Reto 010

Mismo juego que el caso anterior pero además queremos que al final nos diga cuántas jugadas han sido necesarias para ganar. Usar un **Do...While**.

Reto 011

Crear un Canvas donde vamos a representar el camino del borracho.

Vamos a calcular a qué distancia llega desde un origen después de haber dado n pasos.

Ver si la distancia alcanzada se aproxima a la raíz cuadrada de n , que es el valor teórico de esa distancia.



Un borracho en Montecarlo

Reto 012

Crear un Canvas donde se dibuje un cuadrado u otra figura geométrica y se le vea moverse por la pantalla y girar.

Reto 013

Veamos si se puede llegar a formar un triángulo dando la longitud de los tres lados. Crear un formulario donde se capturan las longitudes de los tres lados y un botón que lanza el procedimiento.

Un triángulo es factible si la suma de las longitudes de dos lados cualesquiera es siempre mayor a la longitud del lado restante. Ver el concepto de [desigualdad triangular](#) en la Wikipedia.

Si el triángulo es factible queremos que nos informe de ello y que también nos calcule el área.

Área del triángulo usando la [fórmula de Herón](#).

Si el triángulo no es factible queremos que nos informe de ello y nos diga el motivo. Por ejemplo, el triángulo de lados 4, 5 y 10 no es factible porque $4+5 \leq 10$.

Reto 014

Mismo caso que el anterior pero ahora deseamos que además se dibuje el triángulo en un Canvas en caso de ser factible.

Ver: <https://youtu.be/QsmFsl0kXM8>

Ver: <http://www.universoformulas.com/matematicas/trigonometria/resolucion-triangelos>

Ver: <https://www.dropbox.com/s/2l284mgjm1yjk2u/triangulo.xlsx?dl=1>

Reto 015

Hacer un programa para que la variable `n` se incremente de 1 en 1 empezando en 10 y finalizando en 30. Se mostrará el resultado en pantalla en una columna vertical.

Realizar varias versiones del programa o usar varios métodos para construirlo.

- Método 1: con `for`
- Método 2: con `while`
- Método 3: con `do...while`

Reto 016

Hacer un programa donde la variable `i` se incremente de 12 en 12, comenzando en 120 y finalizando en 360. Los valores se muestran en pantalla y cuando el número sea múltiplo de 36 queremos que aparezca en negrita y color rojo.

Reto 017

Generar posibles contraseñas de 11 dígitos donde los 8 primeros sean letras y los 2 últimos sean números, y las letras y los números se separan con una barra baja (`_`). Se generan de forma aleatoria. Las 8 letras están compuestas por 4 parejas. Dentro de cada pareja la primera letra es una consonante y la segunda es una vocal. La primera letra de todas es mayúscula y las 7 letras restantes son minúsculas.

Ejemplo: Pocomoco_88

Al pulsar un botón se generan series de 20 contraseñas que se muestran en pantalla.

Reto 018

Generar series de números aleatorios entre 1 y 100. Se muestran en pantalla y luego se muestra el máximo, el mínimo y la media.

Reto 019

Generar series de 30 números aleatorios entre 1 y 1000. Se muestran en pantalla en una columna vertical. Junto a esa columna se muestra otra con esos mismos números pero ordenados de menor a mayor.

1. Queremos que se lance la serie anterior al pulsar un botón
2. Crear un par de botones de radio donde se pueda elegir entre ordenar de menor a mayor o bien, ordenar de mayor a menor

Reto 020

Crear una serie de 20 palabras donde sus letras se obtienen de forma aleatoria. Mostrar en columna estas palabras y a su derecha en otra columna, mostrar esas mismas palabras pero ordenadas alfabéticamente.

Reto 021

Generar palabras de 6 caracteres formados por 3 parejas: $cvcvcv$, donde c representa una consonante y donde v una vocal. Hacer una lista de varias palabras separadas por coma, y probar la lista en una página de dominios de internet para ver si existe alguno de estos libre. Ejemplo: casato.com

Reto 022

Programar el algoritmo de Euclides para calcular el MCD (Máximo Común Divisor).

Reto 023

Programar la obtención del mcm (mínimo común múltiplo). Ver página 47 del manual de Casio FX-702P.

<http://www.lawebdelprogramador.com/descargar.php?id=147685&f=5663f85fe9cae-Casio-FX-702p-801p.pdf.zip>

Reto 024

Crear un CANVAS donde representar las funciones seno y coseno en dos colores diferentes.

Reto 025

Programar series de Fibonacci de longitud n .

Reto 026

Generar *cuadrados latinos* de tamaño $n \times n$.

Reto 027

Juego que ofrece a un jugador un *cuadrado latino* para que lo resuelva. También tiene un botón para resolverlo de forma automática.

Reto 028

Creación de un Sudoku de solución única. Son aquellos en los que se han de proporcionar al menos 17 datos inicialmente.

Reto 029

Resolución de Sudokus de solución única.

Reto 030

Tratamiento de las matrices bidimensionales en Javascript de orden $n \times 2$, y en general de orden $n \times m$.

Reto 031

Cálculo de un determinante de orden $n \times n$ en Javascript.

Reto 032

Cálculo de una matriz inversa de orden $n \times n$.

Reto 033

Reducir una matriz haciendo ceros por Gauss-Jordan.

Reto 034

Crear un Canvas para ilustrar el tiro parabólico calculando la distancia alcanzada en función del ángulo α y la velocidad inicial. Ver la posibilidad de añadir rozamiento con el aire o no. ¿Y si variamos g ?

Reto 035

Crear un juego donde los dos adversarios se disparan alternativamente balas de cañón variando el ángulo y la velocidad hasta que uno de ellos destruye al otro. Ver la posibilidad de incluir viento.

Reto 036

Ejercicio con matrices. Generar aleatoriamente la nota de 5 alumnos. Calcular la media e imprimir en pantalla la nota de cada uno y la desviación de cada una de las notas respecto a la media. Los que estén por debajo de la media que se impriman en rojo.
Probar a generar las notas siguiendo una distribución normal.

Reto 037

Vamos a crear un cubo 3D. Generar las notas de 5 alumnos en 6 asignaturas durante 4 cursos.

Reto 038

Reversión a la media. Usando matrices (arrays) crear un vector de un millón de datos aleatorios según una distribución normal. Ir calculando la media a medida que se van generando los valores, e ir la representando en un Canvas para ver como se tiende a la media a medida que aumenta el número de datos.

Reto 039

Crear un programa que escuche las teclas que se pulsa en el teclado. Ver si se puede crear un procedimiento que escriba en pantalla punto y raya para crear un código Morse. Ver si se puede emitir el sonido correspondiente.

Reto 040

Crear una caja de texto y manejar el foco de tal forma que al pulsar ENTER se ejecute un procedimiento. Por ejemplo, ponemos un número y al pulsar ENTER nos da su cuadrado.

Reto 041

Programar la criba de Eratóstenes que permite obtener los números primos entre 1 y n , dado el número n .

Reto 042

Reto 043

Reto 044

Reto 045

Reto 046

Reto 047

Reto 048

Reto 049

Reto 050