

EJERCICIOS

J2



LOOPS!

1. Solo 25 términos

- Realizar un programa que imprima 25 términos de la serie 11 - 22 - 33 - 44, etc. (No se ingresan valores por teclado).

2. Múltiplos de 8

- Mostrar los múltiplos de 8 hasta el valor 500. Debe aparecer en pantalla 8 -16 -24, etc.

3. Cuantos tuvieron mejores notas

- Escribir un programa que lea 10 notas de alumnos y nos informe cuántos tienen notas mayores o iguales a 7 y cuántos menores.

4. Promedio de estaturas

- Se ingresan un conjunto de 5 alturas de personas por teclado. Mostrar la altura promedio de las personas.

5. Reportes de salarios

- En una empresa trabajan 5 empleados cuyos sueldos oscilan entre \$100 y \$500, realizar un programa que lea los sueldos que cobra cada empleado e informe cuántos empleados cobran entre \$100 y \$300 y cuántos cobran más de \$300. Además el programa deberá informar el importe que gasta la empresa en sueldos al personal.

6. De diez en diez

- Mostrar los múltiplos de 10 hasta el valor 1500. Debe aparecer en consola 10 - 20 -30 etc.

7. Dos listas. Obtener el acumulado mayor.

- Realizar un programa que permita cargar dos listas de 3 valores cada una. Informar con un mensaje cuál de las dos listas tiene un valor acumulado mayor (mensajes 'Lista 1 mayor', 'Lista 2 mayor', 'Listas iguales')

Tener en cuenta que puede haber dos o más estructuras repetitivas en un algoritmo.

8. *¿Par o impar?*

- Desarrollar un programa que permita cargar 5 números enteros y luego nos informe cuántos valores fueron pares y cuántos impares.

Emplear el operador "%" en la condición de la estructura condicional.

```
if (valor%2==0)
```

El operador "%" retorna el resto de la división valor / 2. Por ejemplo: 12 % 2, retorna 0; 13 % 2, retorna 1, porque el resto de dividir 13 en 2 es 1.

9. *¿El triángulo de las bermudas?*

- Confeccionar un programa que lea 3 pares de datos, cada par de datos corresponde a la medida de la base y la altura de un triángulo. El programa deberá informar:
 - a) De cada triángulo la medida de su base, su altura y su superficie.
 - b) La cantidad de triángulos cuya superficie es mayor a 12.

10. Los últimos serán los primeros

- Desarrollar un programa que solicite la carga de 10 números e imprima la suma de los últimos 5 valores ingresados.

11. Tabla del 5

- Desarrollar un programa que muestre la tabla de multiplicar del 5 (del 5 al 50).

12. Tabla del

- Confeccionar un programa que permita ingresar un valor del 1 al 10 y nos muestre la tabla de multiplicar del mismo (los primeros 12 términos)

Ejemplo: Si ingreso 3 deberá aparecer en pantalla los valores 3, 6, 9, hasta el 36.

12. Los cuatro triángulos

- Realizar un programa que lea los lados de 4 triángulos, e informar:
 - a) De cada uno de ellos, qué tipo de triángulo es: equilátero (tres lados iguales), isósceles (dos lados iguales), o escaleno (ningún lado igual)
 - b) Cantidad de triángulos de cada tipo.
 - c) Tipo de triángulo del que hay menor cantidad.

13. *Más coordenadas*

- Escribir un programa que pida ingresar coordenadas (x,y) que representan puntos en el plano. Informar cuántos puntos se han ingresado en el primer, segundo, tercer y cuarto cuadrante. Al comenzar el programa se pide que se ingrese la cantidad de puntos a procesar.

14. Positivos, Negativos o múltiplos de 15?

- Se realiza la carga de 10 valores enteros por teclado. Se desea conocer:
 - a) La cantidad de valores negativos ingresados.
 - b) La cantidad de valores positivos ingresados.
 - c) La cantidad de múltiplos de 15.
 - d) El valor acumulado de los números ingresados que son pares.

15. Promedio de edades por turnos...

- Se cuenta con la siguiente información:
 - a) Las edades de 5 estudiantes del turno mañana.
 - b) Las edades de 6 estudiantes del turno tarde.
 - c) Las edades de 11 estudiantes del turno noche.
 - d) Las edades de cada estudiante debe ingresar por teclado.
- Obtener el promedio de las edades de cada turno (tres promedios).
- Imprimir dichos promedios (promedio de cada turno).
- Mostrar por pantalla un mensaje que indique cuál de los tres turnos tiene un promedio de edades mayor.

Ejercicio de ciclos

Dado el arreglo ['H', 'o', 'l', 'a', '!']

- a) Imprimir todo el arreglo en una misma palabra
- b) Agregar espacios entre las letras e imprimirlas nuevamente para que se lea
'H o l a !'
- c) Repetir el A y B con el método de ciclos no utilizado anteriormente.

Arreglos

1. A sumar elementos en un arreglo

- Desarrollar un programa que permita ingresar un vector de 8 elementos, e informe:
El valor acumulado de todos los elementos del vector.
El valor acumulado de los elementos del vector que sean mayores a 36.
Cantidad de valores mayores a 50.

2. Suma de arreglos

- Realizar un programa que pida la carga de dos vectores numéricos. Obtener la suma de los dos vectores, dicho resultado guardarlo en un tercer vector del mismo tamaño. Sumar componente a componente.

El tamaño del vector es a elección.

Ejercicio de arreglos

Dado el arreglo `[true, 'hola', 4, null]`

Implementar una función por cada ítem y probarlo con arreglo de ejemplo

- a) Imprimir el tercer elemento del arreglo
- b) Cambiar el último valor por un `3`
- c) Remover los últimos dos elementos
- d) Agregar un `6` y un `'chau'` al final
- e) Insertar en la tercera posición del arreglo un `5`

Mas Arreglos!

1. Rellenemos!

- Rellena un array(arreglo) con los números del 1 al 10. Al finalizar muéstralo por consola

2. Relleno aleatorio

- Rellena un array con números aleatorios. Muéstralo por consola

Nota: investiga `Math.floor()` y `Math.random()` para que sirven 😊

3. *Copiar array*

- Del array anterior, créale una copia (debe ser otro array exactamente con los mismos valores). Muéstralo en consola

Nota: Investiga sobre `array.slice()`

4. Ordenemos

- Genera un array de números aleatorios, ordénalos de menor a mayor y muéstrales en consola

Objetos

1. Tu Entorno

- Mira a tu alrededor y escoge 6 cosas de tu entorno(libro, sillas, tasa, etc.). Una vez elegidas, crea objetos JS que puedan simularlos y guardar sus características.

2. Deportista!

- Para este ejercicio contamos con un objeto literal deportista ya definido, que tiene los siguientes atributos: nombre, energía, experiencia. Queremos poder pedirle al deportista que entrene. Para esto, nuestro trabajo va a ser completar la función entrenarHoras. La función entrenarHoras tiene las siguientes tres características: Recibe por parámetro la cantidad de horas. Resta a su energía (this.energia) la cantidad de horas x 5. Incrementa su experiencia (this.experiencia) la cantidad de horas x 2. Una vez definida la función, podemos ejecutar el código y ver cómo va variando la energía y experiencia del deportista por consola.

```
let deportista = {  
  energia: 100,  
  experiencia: 10,  
  nombre: "M. Jordan",  
  entrenarHoras: ???  
};
```

2. Array de objetos!

- Vamos a crear una lista en un arreglo el cual guarde los datos de los alumnos como objetos, que contendrán los datos como, nombre, apellido, teléfono, nacimiento y dirección.

Una vez terminada la lista imprimirla en consola.