

Programación

Ejercicios

Octubre 2021

Cruz García, Iago



[Introducción](#)

[Diagrama de flujo](#)

[Inicio/Fin](#)

[Entrada](#)

[Salida](#)

[Proceso o función](#)

[Decisión](#)

[Ejemplo](#)

[Variables](#)

[Identificador](#)

[Tipos de datos](#)

[Anotaciones](#)

[Operadores](#)

[Operadores aritméticos](#)

[Operadores comparativos](#)

[Operadores lógicos](#)

[Conclusiones](#)

Anotaciones previas

Estos ejercicios son para familiarizarse con el lenguaje, la sintaxis y cómo resolverlos. Los primeros son sencillos y se va incrementando la dificultad. A continuación se presentan una serie de instrucciones que son necesarias para la resolución de los ejercicios:

- [alert\(parámetro\):](#) esta instrucción permite mostrar por pantalla un cartel con texto para mostrar la solución de algunos ejercicios.
- [console.log\(parámetro\):](#) esta instrucción permite mostrar en consola (F12 en el navegador) la solución de algunos ejercicios o trazar el código para comprobar que todo se ejecuta correctamente.
- [prompt\(texto, ejemplo\):](#) Muestra en pantalla un recuadro de **texto** y un cuadro para introducir texto con un **ejemplo**.
- Para poder ejecutar código JavaScript en Visual Studio Code debéis crear un fichero JavaScript (miScript.js) y un HTML básico (index.html por ejemplo) y dentro de la etiqueta <head> escribir los siguiente:
 - <script src="miScript.js"></script> comillas incluidas
- [Math.random\(\):](#) Genera un número al azar entre el 0 y el 1.
 - Para generar un número entre un máximo y un mínimo se utilizará $(\text{Math.random()} * \text{MAX}) + \text{MIN}$
 - Para generar un número entero, $\text{Math.floor}(\text{Math.random()})$

- Ahora que sabemos encapsular creando funciones o métodos, se pueden hacer los ejercicios en el mismo fichero, simplemente comentando las llamadas a métodos que no necesiteis.
- Ej:

```
ejercicio_1()  
//ejercicio_2()  
//ejercicio_3()
```

No dudar de consultar el foro de la asignatura en caso de atascarse con la solución.

Ejercicios

Ejercicio 1

Crea un programa que muestre por pantalla si un número introducido por el usuario es mayor o menor que 10

Ejercicio 2

Crea un programa que muestre por pantalla si una palabra que introduzca el usuario contiene la letra b minúscula.

Pista: `String.includes(texto)`

Ejercicio 3

Crea un programa que muestre por pantalla si el usuario es mayor o no de 18 años, introduciendo su fecha de nacimiento.

Pista: `Date.GetFullYear()`

Ejercicio 4

Crea un programa que muestre por pantalla una selección de productos al usuario (3 o 4) y dependiendo de lo que escriba el usuario, se muestre el precio del producto.

Ej.: Leche

Cacao

Avellanas

Azucar

“Cacao”

El cacao vale 2 €

Ejercicio 5

Crea un programa que muestre por pantalla distintas figuras geométricas (triángulo, rectángulo, pentágono y hexágono). Según lo que escoja el usuario, se le pedirá que introduzca unos valores clave para calcular el área de la figura seleccionada. Mostrar por pantalla el resultado.

Ejercicio 6

Crea un programa que pida una frase al usuario y compruebe si en esa frase contiene alguna b o v. Si existe una b, buscar si tiene alguna p la frase. Si tiene v, buscar si tiene u. Estas cláusulas no son excluyentes. Mostrar por pantalla si son ciertos todos los casos.

Pista: Se pueden concatenar múltiples operaciones lógicas. Ej: (A & B) & (C & D)

Ejercicio 7

Crea un programa que calcule si un número dado por el usuario es par o impar y lo muestre por pantalla.

Pista: El módulo o resto de la división indica si un número es par o no.

Ejercicio 8

Crea un programa que pida una cadena de texto al usuario. Calcular si la cadena contiene un número de caracteres par o impar y mostrarlo por pantalla.

Pista: Las cadenas de texto tienen la propiedad `.length` que devuelve el tamaño.

Ejercicio 9

Crea un programa que unifique todos los ejercicios anteriores, por orden numérico.