

# Programación

Decimosexta semana

Febrero 2022

Cruz García, Iago



[Anotaciones previas](#)

[Ejercicios](#)

[Ejercicio 0](#)

# 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
- 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.

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

# Ejercicios

**IMPORTANTE:** A partir de ahora algunos ejercicios deben hacerse en múltiples ficheros .js, por lo que en vez de entregar todos en un mismo main.js, será necesario dividirlos en directorios. Se aconseja la estructura de “PrácticaX\_ejercicio1” y dentro el index.html y los ficheros .js necesarios.

Para estos ficheros, lo mejor es agrupar aquellas funciones o métodos que realicen tareas similares (entradas.js o salidas.js por ejemplo). En caso de duda, no importa que un método quede aislado en un fichero.

El fichero que realice las llamadas a los métodos, que aune toda la funcionalidad, debe nombrarse como main.js y no debe tener más que un método que se llame igual y una llamada a este mismo.

## Ejercicio 0

Vamos a utilizar como base el ejercicio del Boletín 9 de lenguajes de marcas.

```
<html>

<head></head>

<body>
    <form
action="http://moodle.danielcastelao.org/daw/action_page.php"
method="GET">
        <label for="nif">NIF: <input name="nif" id="nif" type="text"></label>
        <br>
        <br>
        <label for="nombre">Nombre: <input name="nombre" id="nombre"
type="text"></label>
        <br>
        <br>
        <label for="apellidos">Apellidos: <input name="apellidos"
id="apellidos" type="text"></label>
        <br>
        <br>
        <label>Edad</label>
        <br>
        <label for="edad"><input name="edad" class="radio" type="radio">
15-19</label>
        <br>
        <label for="edad"><input name="edad" class="radio" type="radio">
20-29</label>
        <br>
        <label for="edad"><input name="edad" class="radio" type="radio">
30-40</label>
        <br>
        <br>
        <label for="estado">Estado civil<br>
```

```

<select name="estado" id="estado" multiple="multiple">
<option value="soltero">Soltero</option>
<option value="casado">Casado</option>
<option value="divorciado">Divorciado</option>
</select>
</label>
<br>
<br>
<label>Aficiones</label>
<br>
<label for="cine"><input type="checkbox" class="check" name="cine"
value="cine">Cine</label>
<br>
<label for="deportes"><input type="checkbox" class="check"
name="deportes" value="deportes" checked>Deportes</label>
<br>
<label for="lectura"><input type="checkbox" class="check"
name="lectura" value="lectura">Lectura</label>
<br>
<label for="senderismo"><input type="checkbox" class="check"
name="senderismo" value="senderismo">Senderismo</label>
<br>
<label for="comentarios">Comentarios:</label>
<br>
<textarea id="comentarios" rows="8" cols="35">Escribe aquí tus
comentarios...</textarea>
<br>
<br>
<label for="envio"><input type="button" id="envio" name="envio"
value="Enviar"></label>
</form>
</body>
<footer></footer>

</html>

```

Para la resolución de los ejercicios podéis añadir lo que necesiteis en el HTML pero no podéis modificar los elementos ya creados.

## Ejercicio 1

Crea un evento que evite que el usuario ponga números en el apartado de nombre y en el apartado de apellidos cuando se presione el botón de “enviar”.

## Ejercicio 2

Añade un evento que al presionar el botón “enviar” evite que en el campo de DNI haya menos de 9 caracteres.

## Ejercicio 3

Crea un evento que evite que el usuario escoja más de dos aficiones.

**Pista:** Crea un atributo “valor” que aumente en 1 cada vez que se escoja una opción y disminuya cuando se deseccione. Cuando se clique en uno de los checkbox que compruebe “valor”.

## Ejercicio 4

Crea un evento que modifique los valores de las aficiones dependiendo de la edad. Utiliza los valores que quieras como "Pintura", "Música", etc.

## Ejercicio 5

Crea un evento que controle el número de caracteres máximos que puede introducirse en la caja de comentarios, hasta 250 caracteres.

## Ejercicio 6

Existe un tipo de botón en los formularios que permite reiniciar los valores del mismo. Crea un botón simple y crea un evento que haga que los campos vuelvan al estado por defecto.

## Extra

Modifica el formulario a tu gusto para practicar el uso de eventos. Algunas de las funcionalidades que puedes añadir son:

- Crea un campo de texto que indique si el DNI tiene el formato correcto (8 números y una letra) estilo "El DNI no coincide con el patrón requerido" y que solo aparezca cuando esto ocurra.



- Añade un campo de ocupación y que se modifiquen los valores dependiendo de la edad (por ejemplo, menos de 18 años no puede ser Jubilado)
- Un botón para rellenar al azar los campos excepto el de comentarios. Tened en cuenta que tiene que modificar otros valores como el de "valor" para el máximo de aficiones.