

Programación

Decimocuarta semana - Clases

Febrero 2022

Cruz García, Iago



[Anotaciones previas](#)

[Ejercicios](#)

[Ejercicio 0](#)

[Ejercicio 1](#)

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 trabajar los conceptos del DOM de forma más abstracta y desvinculada que las semanas anteriores para poder reforzar otros conceptos ya vistos al mismo tiempo. Utiliza el código HTML que encontrarás a continuación para realizar los ejercicios. El contenido es el mínimo necesario, pero puedes agregar lo que necesites. **No utiliceis JQuery para estos ejercicios, coged práctica con las funciones del DOM.**

[Enlace al HTML](#)

Ejercicio 1

Añade al texto del párrafo con identificador "explicacion" la frase "Te damos la bienvenida al curso".

Pista: Para agregar texto sin borrar el otro tendrás que guardar el texto anterior y concatenar (sumar) el nuevo.

Ejercicio 2

Modifica la clase "standard" para que ambos párrafos tengan un ancho máximo.

Pista: Recuerda que se puede modificar el CSS usamos 'elemento'.style(...)

Ejercicio 3

Modifica la tabla con id "horario" para que tenga un borde alrededor, como tu quieras.

Ejercicio 4

En la tabla anterior, modifica la fila del recreo (busca cual es el ID en el HTML) para que el fondo sea de otro color.

Ejercicio 5

Modifica la clase "asignaturas" para que esté centrada en la página.

Ejercicio 6

Modifica el texto de bienvenida haciendo que la fuente sea diferente y subrayado.

Ejercicio 7

Modifica las filas de las columnas a tu conveniencia. Puedes hacer que el texto esté centrado, cursiva, negrita, fondo de otro color... tu decides.

Ejercicio 8

Añade un párrafo debajo de la tabla con el siguiente texto “Los horarios son provisionales, están dispuestos a cambios”.

Pista: Para añadir un elemento, utilizamos `createElement` y `appendChild`, visto en los apuntes.

Ejercicio 9

Añade a la lista de asignaturas de segundo las siguientes asignaturas:

- Diseño de interfaces web
- Despliegue de aplicaciones web
- Formación en centros de trabajo

Consulta el id en el HTML.

Pista: Para añadir, acuerdate de utilizar `appendChild` sobre el elemento ‘padre’ y para crear `createElement`.

Extra

Probad a modificar la página a vuestro gusto, desde JavaScript. Intentad poner un layout o agregar algún elemento como imágenes de forma dinámica. La implementación del diseño de páginas web es una de las tareas que engloba nuestra profesión y debemos coger soltura.

