

Ejercicio de examen

Duración: **180 minutos**. Leer TODO el enunciado antes de realizar la solución del ejercicio. Copiar los archivos entregado a una carpeta con nombre *ierApellidoNombre*. Mover el archivo js a su carpeta. **Solo se puede editar este archivo**.

El ejercicio consiste en hacer una *demo* de una aplicación, *timeApp*, para mostrar la hora, la fecha, la salida y puesta del sol con el número de horas solares de diversas localidades. La localidad se muestra en ①. La hora local de la localidad, con el formato de hh:mm:ss, en ②. La fecha, en castellano y con el formato de la imagen, en ③. Debajo, en ④, se muestran los datos del sol. Por último, en ⑤, se encuentran una serie de localidades con su hora local (sin los segundos). Los números identificativos aparecen como comentarios en el html.



Los datos de las ciudades se encuentran en el archivo *ciudades.json*. Cada ciudad se identifica por un *id* y un *nombre*. Un tercer campo indica la diferencia horaria con la UTC. Por último, el campo *mostrar a true*, indica que esa es la ciudad de la que se mostrará la información en ①②③y ④ nada mas cargarse la página. Los archivos que tienen la información de las salidas y puestas de sol (en función del día) se identifican por el id de la ciudad: *id.json*. Toda información horaria tendrá el formato de dos dígitos: 01:02:03.

Obligatorio

- Solución en forma de app con un único método expuesto: *iniciar*. La app tendrá formato IIFE y con ausencia de variables globales dentro y fuera del IIFE (salvo el nombre de la app). Uso de la cláusula 'use strict'. La creación/eliminación de nodos deberá hacerse creándolos con los métodos adecuados. NUNCA con *innerHTML*. No cumplir estas normas acarreará la nulidad de la prueba.

[1,5 puntos] JSON

La carga de los archivos JSON se realizará mediante CORS. Deberá haber solo una función de carga con dos argumentos: el archivo a cargar (sin ruta y sin terminación) y la función a llamar cuando se reciban los datos. El argumento de ésta última será un objeto JSON.

[4 puntos] Tiempo

- La información se debe refrescar, al menos, cada segundo.
- Nada más cargarse la página, se cargará la información de las ciudades por medio del método *iniciar* de la aplicación con argumento "ciudades" (ver HTML). Se debe detectar cuál de ellas es la que hay que mostrar (su campo *mostrar* tendrá un valor de "true"). De ésta ciudad, se deberá cargar su información sobre la salida y puesta del sol. En este último archivo, hay un campo con la fecha. Se debe buscar la fecha actual en el archivo y mostrar la información correspondiente. Las horas de sol se deben calcular en función de la hora de salida y la hora de puesta. No puede haber retardo al mostrarse la información de la ciudad.

[3 puntos] Ciudades

- La estructura html en ⑤ se muestra como comentario en el archivo js. Además, la información de cada campo del archivo se reflejará en la estructura html con el atributo *data-campo = valor*.
- Al pulsar sobre cualquiera de los elementos en ⑤, se mostrará la información de la ciudad correspondiente de forma inmediata.

[1,5 puntos] Funciones/métodos

- Partiendo de la hora local, devuelva la hora UTC que será la base para calcular las horas locales de las ciudades.
- Pasado un número como argumento, devuelva una cadena con dos caracteres. Si el número no es de dos dígitos, deberá añadir un cero por la izquierda.
- Pasados como argumentos la hora de salida y la de puesta de sol devuelva las horas de sol de la forma HHh MMm (por ejemplo, 05h 09m).
- Pasando la fecha de la ciudad actual, la devuelva con el formato de ③.