


Introducción

Una plataforma de streaming de películas nos contacta para ayudarles a facilitar sus tareas diarias con respecto al manejo de datos.

El listado de las películas lo obtendremos del [siguiente archivo con formato JSON](#) que contendrá un array de objetos literales que representan a dichas películas.

Adicionalmente nos proponen que utilicemos el siguiente [módulo de lectura de archivos JSON](#) como parte de nuestra solución.

 **Importante:** el método de escritura modifica el archivo JSON original, te recomendamos tener una copia a mano por si algo sale mal y necesitás restaurar el archivo.

La estructura de los datos

Como ya es costumbre, recibiremos un archivo en formato JSON con un array de objetos literales que representarán las películas de este sistema.

De cada película podremos saber:

- id → Es un **number** con un identificador único del registro
- título → Es un **string** con el nombre de la película
- director → Es un **string** con el nombre del director/a
- genero → Es un **string** con el nombre del género
- anio → Es un **number** que identifica el año de estreno
- duracion → Es un **number** que representa la duración en minutos
- fuePremiada → Es un **boolean** que nos indica si la película fue premiada

- `calificacionIMDB` → Es un **number** que representa su calificación según la IMDB (Internet Movie Database)

Consignas

1. Obtener el listado de las películas.

Tomando como base el [siguiente archivo JSON](#) y el siguiente [módulo de lectura de archivos JSON](#), deberán:

- Descargar ambos archivos y colocarlos en la raíz del proyecto.
- Utilizando [la plantilla](#) o en un nuevo archivo .js,
 - Importar el módulo de lectura de JSON
 - Ejecutar el método que les permita leer los datos del archivo JSON
 - Guardar el resultado en una variable

Resultado esperado: una variable conteniendo un array con todos las películas.

2. Crear un objeto literal que represente la aplicación

El objeto será la representación de nuestro sistema de gestión de películas, podemos llamarlo **gestionDePelículas** y contendrá todas las propiedades y métodos necesarios para resolver las consignas detalladas a continuación.

- A. Agregar una propiedad llamada ***películas*** que contenga las películas obtenidas en el punto 1.
- B. Agregar un método **listarPelículas** que reciba como parámetro un array de películas y los imprima por consola.

Este método deberá imprimir por consola el siguiente detalle de cada elemento:

- El título

- El director
- Su duración
- Su calificación
- Si ha sido premiada
 - i. “premiada” si **fuePremiada** es true
 - ii. “sin premios” si **fuePremiada** es false

Resultado esperado al ejecutar el método: un mensaje por consola por cada película con el siguiente formato:

*{titulo}, de {director}. Duración de {duración} minutos, {premiada}
({calificación} en IMDB)*

Ejemplos:

The Mist, de Frank Darabont. Duración de 126 minutos, premiada (7.1 / 10 en IMDB)

Tom and Huck, de Peter Hewitt. Duración de 97 minutos, sin premios (5.5 / 10 en IMDB)

- C. Agregar un método **buscarPorId** que permita buscar una película en función de su identificador.
- Este método recibirá por parámetro un **number** que represente el **id** a buscar
 - En caso de encontrar una película con el id buscado, devolverá el objeto literal que la representa.
 - En caso contrario devolverá *undefined*
- D. Agregar un método **peliculasPremiadas**, que retorne todas las películas premiadas, es decir, aquellas que tengan la propiedad **fuePremiada** en **true**.
- Este método no recibirá ningún parámetro.
 - Este método utilizará el listado de películas completo para filtrar.
 - Este método devolverá un array con las películas que tengan premios.
- E. Agregar un método **filtrarPorDuracion** que permita filtrar las películas que tengan una duración que esté entre el mínimo y máximo enviado.
- Este método recibirá por parámetro dos numbers que representan la duración mínima y máxima a buscar.

- Este método devolverá un array con todas las películas que tenga duración mayor o igual al primer parámetro y menor o igual al segundo parámetro.
- En caso de no encontrar ninguna película, devolverá un array vacío.

F. Agregar un método **ordenarPorCalificacion** que permita ordenar las películas recibidas de menor a mayor según su calificación.

- El método recibirá como parámetro un array de películas.
- Este método devolverá un array con todas las películas ordenadas por su calificación.

G. Agregar un método **duracionTotal** que permita calcular la duración total en minutos de todas las películas.

- Este método no recibirá ningún parámetro.
- Este método devolverá un string con el siguiente formato:

La duración en minutos de todas las películas sumadas es de {duracion} minutos

H. Agregar un método **premiarPelículaPorTitulo** que permita premiar una película en función de su título y guardar los cambios en el archivo JSON.

- Este método recibirá por parámetro un **string** que represente el **título** a buscar
- En caso de encontrar una película con el **título** buscado:
 - i. Cambiará el valor de la propiedad **fuePremiada** a **true**.
 - ii. Escribirá los cambios en el archivo JSON que contiene las películas.
 - iii. Devolverá la película modificada

- En caso contrario devolverá *undefined*

⚠ **Importante:** el método que te damos para escribir el archivo JSON reemplaza todo el archivo, así que le vas a tener que pasar el array completo de películas y no solo el elemento modificado 😊.

- I. Agregar un método **eliminarPorId** que permita eliminar una película en función de su identificador.
 - Este método recibirá por parámetro un **number** que represente el **id** a eliminar
 - Este método utilizará el método **buscarPorId**
 - En caso de encontrar una película con el id enviado
 - i. Eliminará la película del array
 - ii. Escribirá los cambios en el archivo JSON que contiene las películas.
 - iii. Devolverá el objeto literal con la película eliminada
 - En caso contrario devolverá *undefined*

⚠ **Importante:** el método que te damos para escribir el archivo JSON reemplaza todo el archivo, así que le vas a tener que pasar el array completo de películas sin el elemento eliminado 😊.