

EVALUACIÓN	PARCIAL	GRUPO	M1E	FECHA	06/07/2022
MATERIA	Programación 1				
CARRERA	AP/ATI				
CONDICIONES	- Puntos: 50 - Duración y mecanismo de entrega: 2 horas, realización y entrega presenciales. - Sin material y realizado en forma individual. - Consultas: Exclusivamente de interpretación y/o alcance de letra.				

Una empresa nos solicita el desarrollo de una aplicación para gestionar canciones. Se desea poder tener información de las canciones que se tienen, los usuarios que las escuchan, etc. De cada canción se desea guardar la siguiente información: id (un autonumérico único), título, cantante, género (pop, rock, rap, cumbia, reggaeton, clásica y Jazz) y duración (en segundos). Las canciones se encuentran guardadas en un array indexado como el que se presenta a continuación:

```
let canciones = [ ... {id: 1, titulo: "Believer", cantante: "Imagine Dragons", genero: "rock", duracion: 204}, {id: 2, titulo: "Yendo a la casa de Damián", cantante: "El cuarteto de nos", genero: "rock", duracion: 254}, ...];
```

De cada usuario, se guarda: id (autonumérico único), correo, playlist, canción favorita. En un array indexado "usuarios".

Al momento que un usuario ingresa al sistema los datos del mismo se guardan en una variable global "usuarioLogueado"

Se pide:

1. Crear las estructuras (clases) necesarias con sus tipos de datos para el correcto funcionamiento del sistema. **(3 puntos)**
  - a. Respetar la estructura y los tipos de datos en las partes siguientes. **(2 puntos)**
2. Suponiendo que en el HTML se tiene una etiqueta `<select id="cancionesDisponibles"></select>` . Crear la función `completarComboCanciones()`, la cual deberá insertar en el select antes mencionado, las opciones correspondientes para poder seleccionar cualquier canción del array de canciones que no se encuentran dentro de la playlist del usuario. Las opciones deberán tener como value el id de la canción y como texto a mostrar, el nombre de la misma. **(10 puntos)**
3. Crear la interfaz HTML (sólo el contenido dentro del body) necesaria para poder agregar una canción a la playlist del usuario logueado. La canción se seleccionará de la etiqueta select antes mencionada, debajo debe mostrar en párrafo con id="totalTime" la duración total de la playlist de la forma **HH:MM:SS (5 puntos)**
4. Crear la función `existeEnPlaylist(idCancion, idUsuario)` que recibe el Id de una canción y el Id de un usuario retorne 'true' si se encuentra en la playlist del usuario. Se debe evitar seguir buscando una vez que se encuentra el objeto y no está permitido cortar la ejecución con break o return en ningún caso. **(7 puntos)**

5. Crear la función *formatearTiempo(segundos)* que dado una cantidad de segundos retorna un string con la duración en el formato **HH:MM:SS (6 puntos)**
6. Crear la funcionalidad js del botón “Añadir a mi playlist” que agrega la canción a la playlist del usuario. Para poder guardarlo se deberá validar que la canción no se encuentre en la playlist actualmente, y actualizar la duración total de la playlist de canciones del usuario. **(6 puntos)**
7. Escriba el código de la función *reporteDeCanciones()*, la cual deberá retornar un HTML que sea una tabla donde se listen todas las canciones. De cada canción, se debe mostrar el nombre de la canción, cantante, género la cantidad de usuarios que la tienen como favorita, la cantidad usuarios que la tienen en su playlist. **(11 puntos)**

El HTML obtenido debería crear una tabla con la siguiente forma:

Título de canción	Cantante	Género	Cantidad de favoritos	Veces en playlists
Believer	Imagine Dragons	Rock	10	22
Yendo a la casa de Damián	El Cuarteto de Nos	Rock	25	130

Se valorará la utilización de nombres de variables y funciones mnemotécnicos, la optimización de la solución y que se respeten las estructuras que se solicitan utilizar.

En caso de no poder resolver alguna de las partes, se puede suponer como completada cuando se solicite en algún otro punto posterior.

Toda función auxiliar utilizada debe ser codificada y aparecer en la entrega realizada.

Se debe aclarar qué ejercicio se está resolviendo en cada parte.