

## Parcial 1: Trabajo práctico domiciliario

### Consignas para la entrega del parcial:

- Para la entrega del presente parcial, la carpeta archivos debe ser comprimida en un único archivo de extensión .ZIP o .RAR, respetando el siguiente formato para asignarle el nombre de archivo:
  - Si la entrega es individual:
    - **p1-parcial-1-dwm2ah-[apellido]-[nombre].[zip o rar]**
  - Si la entrega es grupal:
    - **p1-parcial-1- dwm2ah-[apellido alumn@ 1]-[apellido alumn@ 2].[zip o rar]**
  - TODO EN MINÚSCULAS Y UTILIZANDO EL GUIÓN MEDIO COMO SEPARADOR.
  - EN CASO DE SER ENTREGA GRUPAL, LOS APELLIDOS EN ORDEN ALFABÉTICO.
  - Ejemplos:
    - **p1-parcial-1-dwm2ah-potente-alejandro.zip**
    - **p1-parcial-1-dwm2ah-potente-perez.zip**
  - ***En caso de incumplir con el formato del nombre, se restarán automáticamente dos puntos, pudiendo desaprobar por este error el parcial.***
- La entrega será mediante **Classroom**, en la actividad correspondiente que genere el docente.
- **Fecha de entrega:**
  - La fecha de entrega será la que indique el docente en la actividad del **Classroom**.

## Consigna del trabajo

Se le pide crear un programa usando código JavaScript que permita cargar los datos de discos de una colección e ir mostrando un listado de estos.

Este parcial es **DOMICILIARIO, INDIVIDUAL o GRUPAL DE A 2 (DOS) PERSONAS**.

De no aprobarse/presentarse la entrega, se pasará a un recuperatorio al final del cuatrimestre.

Se le entrega al alumno una carpeta con los siguientes archivos:

- **index.html:** contiene la estructura básica del parcial, incluidos los botones que ya están llamando a las funciones necesarias para la acción del código. Se puede modificar libremente.
- **estilo.css:** contiene los estilos para darle un mínimo de "diseño" al parcial. Se puede modificar libremente.
- **parcial.js:** contiene algunas líneas de código para tomar como base para el desarrollo, entre las que se encuentran:
  - Las funciones *Cargar* y *Mostrar*, que serán explicadas a continuación.
  - Un array *discos*.
  - Un modelo de la estructura del objeto disco para tomar como referencia, por tal motivo están comentados.
  - Puede comenzar con un código desde cero, no es obligatorio partir de la referencia.
- **favicon.ico:** un favicon para mostrar en el navegador. Se puede cambiar libremente.

## Desarrollo mínimo y obligatorio, nota 4

Cualquier faltante de los requisitos que se presentan a continuación, hará que la entrega se considere desaprobada, aunque haya hecho cosas no solicitadas o del punto siguiente. Esta condición es mínima e innegociable.

- Al hacer clic en el botón "Cargar nuevo disco" se debe disparar la función **Cargar** que solicita al usuario la siguiente información de un disco:
  - Nombre del disco.
  - Autor o banda del disco.
  - Código numérico único del disco:
    - Si el usuario ingresa un código numérico que ya fue cargado, se deberá pedir otro código (hasta ingresar uno que no haya sido utilizado anteriormente).
  - Todas las pistas del disco, donde cada pista tiene un nombre y una duración:
    - La cantidad de pistas a ingresar NO debe ser fija.
    - No se debe preguntar cuántas pistas hay que cargar, el dato se desconoce.
    - Al finalizar la carga de una pista, se debe confirmar si se desea ingresar otra más.
    - Cuando el usuario decide dejar de cargar pistas, finaliza la acción de la función.
  - En todo caso debe validarse que los datos ingresados sean válidos:
    - El nombre del disco, autor/banda y nombre de la pista no pueden quedar vacíos.
    - El código numérico único del disco no puede ser menor a 1, ni mayor a 999.
    - La duración de cada pista debe estar entre 0 y 7200 (segundos) inclusive.
- Al hacer clic en el botón "Mostrar discos", la función **Mostrar** debe mostrar cada disco con su nombre, autor/banda, código único y todas las pistas con sus respectivos datos: nombre y duración.
- Hay que destacar visualmente con rojo las duraciones mayores a 180.

## **Agregados para llegar a 10**

Entre las entregas que cumplan el punto anterior, incrementarán su nota los desarrollos que:

- Usen al menos otras dos (2) funciones a lo largo de todo el proceso:
  - Deben recibir argumentos y/o devolver valores para que cuenten como funciones para sumar puntos.
- Muestren un mensaje (con el método alert) ante cada error cometido (haciendo referencia a las validaciones).
- Informen cada vez que se muestran los discos, cuántos se llevan cargados (contador).
- Informen por cada disco:
  - La cantidad de pistas que tiene cada disco.
  - La duración total del disco.
  - El promedio de duración de cada disco.
  - La pista con mayor duración de cada disco.
- Que destaquen la duración total más alta entre todos los discos.
- Que se pueda mostrar la información de un disco específico mediante su código numérico.
- Formateen toda la muestra de datos con código HTML + CSS.

---

## **Consideraciones finales**

- Se valorará el uso de objetos y/o clases para el desarrollo del programa (no es obligatorio).
- Se descontarán puntos por errores semánticos en el uso de los elementos HTML.
- Se descontarán puntos por el uso incorrecto de las estructuras de programación.
- La lógica del programa también será evaluada.