

Micro desafío 1:

Paso 1: Estructura de carpetas

- Crea una carpeta para el proyecto llamada "dhAutos".
- Dentro de esta carpeta, crea una subcarpeta llamada "datos".
- Dentro de la subcarpeta "datos", crea tres archivos: "marca1.json", "marca2.json" y "marca3.json". Cada archivo debe contener un array de objetos literales con información de los autos en stock. Para cada auto, incluye los siguientes detalles:
 - Para la marca "Toyota":
 - Marca (Toyota)
 - Modelo (Corolla, Camry, RAV4, Prius, Supra)
 - Precio (Indicar el precio)
 - Stock (Indicar la cantidad)
 - Para la marca "Ford":
 - Marca (Ford)
 - Modelo (Mustang, Focus, Escape, Explorer, Fusion)
 - Precio (Indicar el precio)
 - Stock (Indicar la cantidad)
 - Para la marca "Chevrolet":
 - Marca (Chevrolet)
 - Modelo (Cruze, Malibu, Equinox, Silverado, Camaro)
 - Precio (Indicar el precio)
 - Stock (Indicar la cantidad)

Paso 2: Crear el archivo "concesionaria.js"

Crea un archivo llamado "concesionaria.js" en la carpeta raíz del proyecto. En este archivo, deberás construir una función llamada "*importar*". Esta función recibirá como parámetro el nombre de la marca de los autos. Tendrá la responsabilidad de leer el archivo en formato JSON y devolver un array de objetos de esa marca.

No olvides que este archivo es un módulo propio, por lo tanto, una vez creado, recuerda incluir la última línea de código necesaria para exportar la función.

Micro desafío 2:

Paso 1: Instrucciones para "app.js"

- Crea un nuevo archivo llamado "app.js".
- En este archivo, importa el módulo creado previamente ("concesionaria.js"). Crea una variable para cada una de las marcas de autos ("toyota", "ford", "chevrolet") e invoca la función "importar" pasando como argumento cada una de las marcas. Crea un único array llamado "unifiedAutos" que contenga todas las listas de autos de las diferentes marcas. Utiliza el operador de propagación (Spread Operator). Crea un objeto literal llamado "concesionaria" que tenga como primer atributo "autos", y que contenga la lista de todos los autos importados.

Paso 2: Funcionalidades solicitadas por el cliente:

- Desarrollar la funcionalidad "listAutos". Esta función tendrá la responsabilidad de mostrar al usuario todos los datos de los diferentes autos almacenados. Puedes utilizar los métodos `forEach()`
- Crear una funcionalidad "autosByBrand" que reciba por parámetro la marca del auto. Tendrá la responsabilidad de retornar todos aquellos autos que correspondan a la marca recibida por parámetro. Deberíamos utilizar el método `filter`.
- Comprueba las funcionalidades utilizando `console.log()` e invocando a cada una de ellas pasando los referidos parámetros.

Micro desafío 3 (EXTRA):

1. En la carpeta raíz del proyecto, crear un nuevo archivo (`extra.js`).
2. Crear un array de números y asignarle un total de diez (10) valores numéricos.
3. Crear variables que contengan los valores numéricos dispuestos en el array en la posición (0, 2 y 4) y con el resto de los valores numéricos crear un nuevo array. Utiliza los recursos aprendidos en esta clase: Destructuring y Spread operator.
4. Crear un objeto literal para almacenar los datos de nuestra mascota, considerando los atributos: a. nombre b. tipo de mascota (Perro - Gato) c. color d. raza Haciendo uso de la deestructuración, crea una variable por cada una de las claves del objeto literal y al final mostrar al usuario:

Hola les presento a mi mascota su nombre es: Bony, es un hermoso Perro, de color: Dorado y su raza es: Golden retriever.