****

**Programación Imperativa**

Segundo Parcial - TEMA 1

## Objetivo

Integrar los conocimientos adquiridos hasta el momento en la cursada.  
Practicar la lectura, comprensión e interpretación de consignas y producir código que resuelva los enunciados propuestos.

## **Buenas prácticas**

A la hora de la resolución no solo es importante que el código funcione correctamente, sino también que sea legible y respete las buenas formas.

* Que las variables, métodos y funciones tengan nombres descriptivos
* Que se utilicen nombres en castellano o en inglés pero no ambos
* Que se utilice camelCase donde corresponda
* Que se utilicen los métodos más adecuados para resolver cada problema
* Que el código produzca el resultado esperado a partir de los datos suministrados

## Introducción

Una empresa que reúne Prestadores de Servicios de Salud (PSS) necesita crear un pequeño sistema que le permita facilitar la gestión de éstos.

La empresa tiene una base de información que podremos encontrar en un archivo con formato JSON que contendrá un array de objetos literales que representan las propiedades de cada profesional.

### **Para comenzar:**

* Te damos un [archivo .zip](https://drive.google.com/file/d/1JxpUavBcEqqVZronMl7PNZK4f0si0ctH/view?usp=sharing) que contiene una carpeta que debes descargar.
* Encontrarás algo de código ya escrito donde se requiere el módulo de lecturaEscritura que es el responsable de leer y guardar los cambios que sucedan en los datos de la cartilla de profesionales.
* También un objeto literal que representa a la empresa con algo de código ya escrito.
* Finalmente una estructura de logs e invocaciones de métodos que debes respetar indicando qué consigna está siendo ejecutada.

## Comprobaciones previas:

### **1. Ejecutar**

Comprobar que al ejecutar app.js se imprime en la consola el listado de profesionales.

### **2. Analizar el objeto literal que representa a la empresa**

Este objeto literal contiene una **propiedad** con el listado de profesionales obtenido en la línea 2 y algunas **funcionalidades** básicas que nos ayudarán a comprobar que los nuevos métodos solicitados están funcionando correctamente. Mira dentro de ellos e intenta comprender qué y cómo lo hacen aún cuando no lo logres plenamente.

## Consignas:

1. Vemos aquí una propiedad que contiene toda nuestra base de datos.   
   Responder en un comentario cuál es esa propiedad.
2. También vemos algunas funciones. Responda con qué terminología se las denomina.
3. Completa este punto 3 utilizando el método ya dado que muestra **un solo profesional**. Solo falta elegir uno cualquiera del array con notación de corchetes.
4. Agregar (crear) un método **filtrarHabilitados** que permita filtrar profesionales que estén habilitados, es decir, aquellos cuya propiedad *estaHabilitado* sea *true*.
   1. Este método deberá retornar un nuevo array de profesionales habilitados.
   2. Ese array retornado puedes guardarlo en una nueva variable y utilizar el método ya dado de **listarTodos** pasándole por argumento dicho array.
5. Agregar un método **buscarPorNombre** que permita buscar dentro de la propiedad de nuestro objeto profesionales, a un profesional por la propiedad nombre, la misma deberá ser igual a un nombre que se envía como argumento al momento de invocarlo.
   * Este método deberá retornar un profesional.
   * Debe mostrarse por consola utilizando el método ya dado que muestra **un solo profesional**.
6. Agregar un método **incrementarHonorarios** que permite cambiar la propiedad *honorariosConsulta* incrementando el valor que ya trae la propiedad, el mismo deberá realizar los siguiente pasos.
   1. Recibe dos parámetros, el nombre del profesional y el porcentaje que se desea de incremento.
   2. Reutilizar el método **buscarPorNombre** para obtener el profesional.
   3. Una vez encontrado el profesional deberá cambiar el valor de la propiedad *honorarioConsulta* según el porcentaje enviado como parámetro.
   4. Utilizando la función *escribirJson()* del módulo ya dado de lecturaEscritura tenemos que sobreescribir el archivo JSON de profesionales para persistir los datos modificados.
   5. Al finalizar deberá retornar al profesional en cuestión.