



# Certified Tech Developer

The Ultimate Degree

## Programación Imperativa

# Métodos de array

A esta altura de la materia utilizamos el bucle **for** para recorrer un array y realizar búsquedas, filtros, modificaciones, entre otras cosas. Ahora, luego de tener la teoría de métodos de arrays avanzados, afianzaremos los mismos con una serie de ejercicios.

## Introducción

Siguiendo con el JSON de la [mesa de trabajo anterior](#), hemos logrado utilizar y comprender los métodos de array como **forEach**, **filter**, **map** y **reduce**. Ahora trabajaremos sobre el mismo set de datos **arrayProfesionales**, pero haciendo uso de los métodos **slice**, **splice**, **sort** y **find**, por lo que podemos continuar utilizando el mismo **app.js**.

## Consignas

- A. Utilizar un método de array que retorne los primeros 5 profesionales del **arrayProfesionales**. No debe modificar el array original. Guardar el resultado de este método en una variable **primerosCincoProfesionales**, luego hacer la impresión de la misma.
- B. Utilizar un método de array que permita reemplazar el profesional con ID 14 por un **nuevoProfesional**. Es decir, se modificará el **arrayProfesionales**. Luego, imprimirlo para verificar el cambio.

```
const nuevoProfesional = {  
  identificador: 250,  
  estaHabilitado: true,  
  honorarioConsulta: 8000,  
  edad: 52,  
  nombre: 'Linus Torvalds',  
  especialidad: 'Junior Developer',  
  cantidadConsultas: 48,  
  puntuacion: 50  
}
```

- C. Ahora, necesitamos un método para ordenar el **arrayProfesionales** según la propiedad **honorarioConsulta** de menor a mayor. Mostrar el array ordenado.
- D. Utilizar un método para realizar una búsqueda sobre el **arrayProfesionales** del profesional con ID 15. Es decir, que retorne solo un profesional. Asignar este resultado a una variable, luego realizar la impresión de la misma.

## Extra

Ahora, podemos crear funciones para poder reutilizar todos los métodos que desarrollamos en estas dos mesas. También, crear una función que, utilizando el módulo **lecturaEscritura**, logre hacer la persistencia en el archivo JSON cada vez que modifiquemos el **arrayProfesionales**.