

1. Dada la siguiente colección:

```
let personas = [  
  {  
    id: 1,  
    nombre: "Ale",  
    edad: 15  
  },  
  {  
    id: 2,  
    nombre: "Javi",  
    edad: 83  
  },  
  {  
    id: 3,  
    nombre: "Luis",  
    edad: 26  
  },  
  {  
    id: 4,  
    nombre: "Dan",  
    edad: 16  
  },  
  {  
    id: 5,  
    nombre: "Tito",  
    edad: 22  
  },  
  {  
    id: 6,  
    nombre: "Marina",  
    edad: 76  
  },  
  {  
    id: 7,  
    nombre: "Susy",
```

```
    edad: 35
  },
  {
    id: 8,
    nombre: "John",
    edad: 25
  },
]
```

Realizar lo siguiente:

- Ordenar la colección de menor a mayor utilizando bubble sort.

```
function ordenarPoEdades(arreglo){
  for(let z=0 ; z < arreglo.length ; z++){
    for(let i=0 ; i < arreglo.length - 1 ; i++){
      if(arreglo[i].edad > arreglo[i+1].edad){

        let temporal = arreglo[i];
        arreglo[i] = arreglo[i+1];
        arreglo[i+1] = temporal;
      }
    }
  }
  return arreglo
}

let arregloOrdenado = ordenarPoEdades(personas);
console.log(arregloOrdenado);
```

- Crear una función que te permita imprimir por consola el nombre correspondiente a las siguientes edades:
 - 25
 - 76
 - 16
 - 15

Para buscar a las personas, utilizar el algoritmo de búsqueda binaria.

Tip: Así como ordenaste una colección con Bubblesort, fijándote en una propiedad numérica, podrás aplicar el algoritmo de búsqueda binaria de la misma manera. Solo tendrás que identificar cuál es la propiedad numérica que te interesa para la búsqueda.

```
const binarySearch = (list, item) => {
  let low = 0;
  let high = list.length - 1;

  while (low <= high) {
    const mid = Math.floor((low + high) / 2);
    const guess = list[mid].edad;

    if (guess === item) {
      return mid;
    }
    if (guess > item) {
      high = mid - 1;
    } else {
      low = mid + 1;
    }
  }
  return null;
};

let posicionDel25 = binarySearch(arregloOrdenado, 25);
console.log(posicionDel25);
```