

Tenemos el siguiente programa HTML:

```
<html>
<head>
<title>retos JavaScript</title>
</head>
<body>
<h1>retos JavaScript</h1>
<script>
var ciudades = ["Sevilla", "Vigo", "Leon",["Caceres",
"Marbella"],["Algatocin", "Mijas", "Estepona"], "LosBarrios"]];

var ciudadakas = ["Chevrolet", "Gaucin", "Leon",["Caceres",
"SanRoque"],["Algatocin", "Mijas", "Algeciras"],
"LosBarrios"],["LaLinea", "Mijas", "LasRozas",["Majadahonda",
"Boadilla", "Mostoles"]]];

var newciudades = ["Sevilla", "Vigo", "Leon",["Caceres",
"Marbella"],["Algatocin", "Vigo", "Estepona"], "LosBarrios","Vigo"]];

coleguillas = [{firstName:"Jonny", lastName:"Rambo",
car:"Leon"},{firstName:"Arnold", lastName:"Suarchenager",
car:"Caceres"},{firstName:"Chuck", lastName:"Norris",
car:"Caceres"},{firstName:"Yakie", lastName:"Chang",
car:"Leon"},{firstName:"Bruce", lastName:"Lee",
car:"Caceres"},{firstName:"Steven", lastName:"Seagal",
car:"Estepona"}];

let maleficas=["Caceres","Estepona"]

//EJERCICIO 1

//EJERCICIO 2

//EJERCICIO 3

//EJERCICIO 4

</script>
</body>
</html>
```

Se pide:

Ejercicio 1: Ver si una ciudad es trendy. Una ciudad es trendy si esta es utilizada por más de un colega. Crea en este mismo ejercicio un array trendy con dichas ciudades. Muestra el array trendy por consola. Se pide que lo realices en 3-4 líneas de código. (3 PUNTOS)

Ejercicio 2: Muestra por consola las ciudades del array newciudades que gusten a los colegas, que están obviamente en newciudades pero que no estén en el array maléficas. Muestra el resultado por consola. Puede aparecer alguna ciudad repetida (no importa). Realiza este programa en una línea de código. (3 PUNTOS)

Ejercicio 3: Se quiere conocer la ciudad más trendy de todas usada por los colegas (la más preferida). Realiza este programa en 2 líneas de código. (2,5 PUNTOS)

Ejercicio 4: ¿Cuántas veces aparece la palabra Vigo en newciudades en una línea? (1,5 PUNTOS)

Importante. No puedes utilizar en ningún ejercicio el método includes() ni utilizar sets. No puedes utilizar for.

Se valorará de tu programa:

- El programa realiza la funcionalidad requerida.
- Que no haya errores de consola.
- La eficiencia.
- La modularidad (alta cohesión y bajo acoplamiento).
- Que el código sea limpio, flexible, reutilizable y mantenible.
- Sigue los principios SOLID.
- Otros criterios.