



PRACTICA: OBJETOS Y ARREGLOS

ESTUDIANTE: PÉREZ MELLADO JENNYFER

GRUPO: 211

MAESTRO: IVAN ISAY GUERRA LOPEZ

MATERIA: DESARROLLO DE APPS MÓVILES

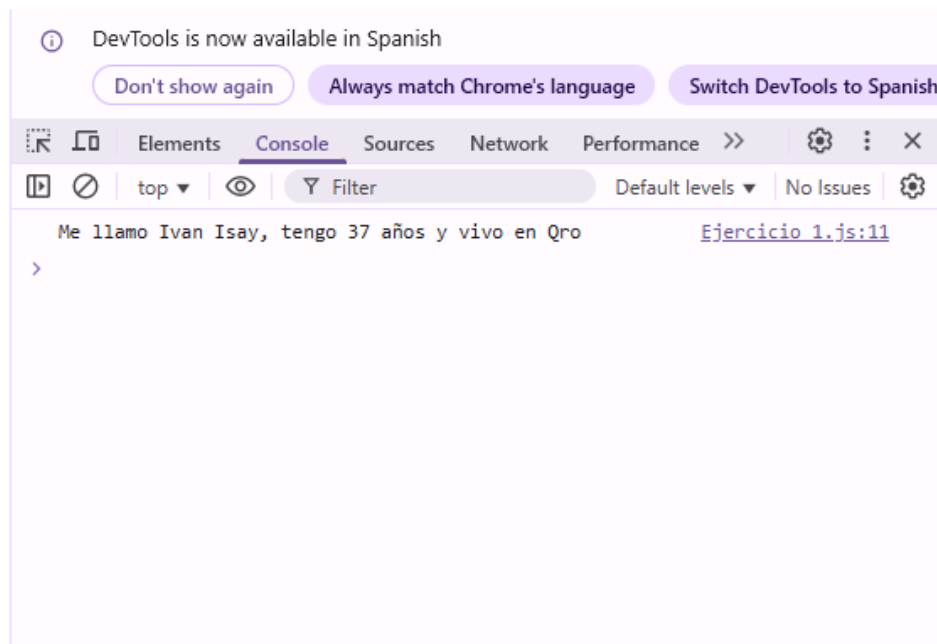
FECHA: 22 - 09 - 2025

En esta practica utilizamos y aplicamos conceptos de objetos, arreglos y manipulación de estos mismos, a continuación, presentare y explicare la resolución de los tres ejercicios dados.

Ejercicio 1:

```
JS Ejercicio_1.js > ...
1  const persona = {
2    nombre: "Ivan Isay",
3    edad: 37,
4    direccion: {
5      ciudad: "Qro",
6      pais: "MX"
7    }
8  };
9
10 const {nombre, edad, direccion: {ciudad, pais}} = persona;
11 console.log("Me llamo " + persona.nombre + ", tengo " + persona.edad + " años y vivo en " + persona.direccion.ciudad);
```

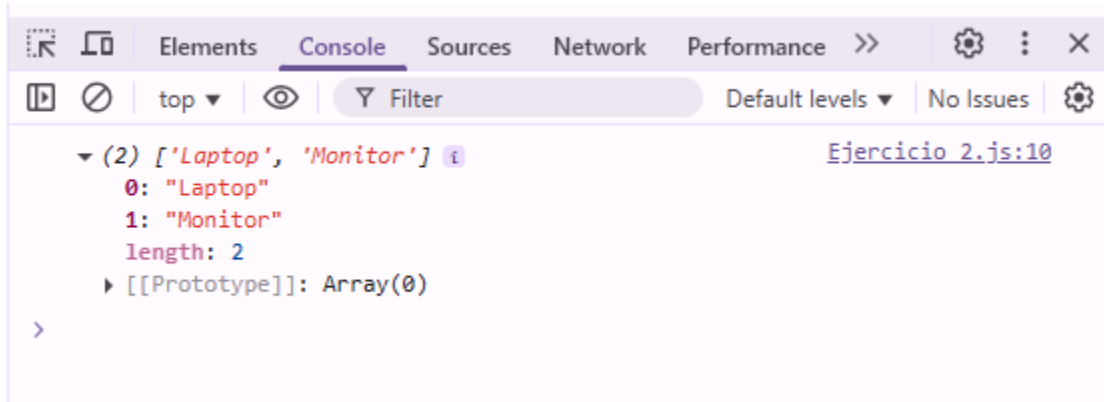
En este ejercicio nos entregaron un objeto con ciertos atributos, por lo que el objetivo se centraba en usar la deestructuración, par de este forma asignar una variable a cada atributo, incluso a aquellos atributos pertenecientes a otros atributos (Atributos anidados) para así utilizarlos en una frase que utilizara estas variables, para ello tal como se indico use la deestructuración con la forma en que se nos presentó, solo agregando unas llaves extras para agregar aquellos atributos extra que en conjunto completan otro atributo, de esta forma se puede llamar cada variable mediante el objeto.atributo, y en el caso específico de atributos anidados, utilizando la formula objeto.atributo.atributo, por lo que el resultado final visual en la consola es el siguiente:



Ejercicio 2:

```
JS Ejercicio_2.js > ...
1  const productos = [
2    {nombre: "Laptop", precio: 12000},
3    {nombre: "Mouse", precio: 250},
4    {nombre: "Teclado", precio: 750},
5    {nombre: "Monitor", precio: 3000}
6  ]
7
8  const nombres = productos.filter(producto => producto.precio > 1000).map(producto => producto.nombre);
9
10 console.log(nombres);
```

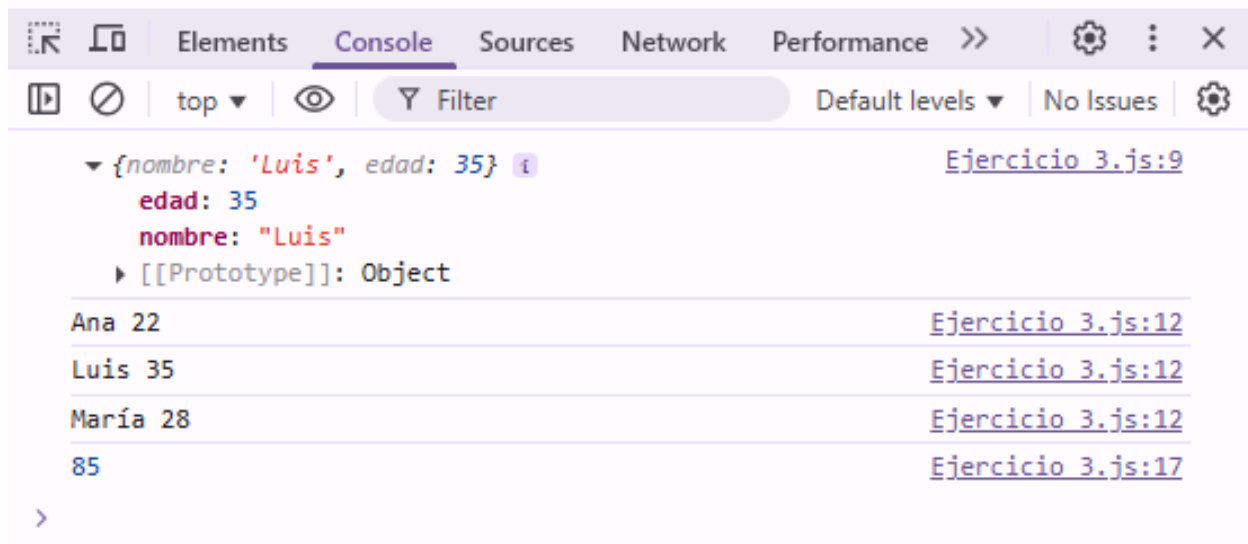
En este ejercicio, de igual forma teníamos un objeto, o de cierta forma, una lista de objetos, todos pertenecientes a la misma categoría y con los mismos atributos, en este caso el objetivo era filtrar aquellos productos con un precio mayor a 1000 y en la consola solo debía mostrarse el nombre del producto, para esto primero se utiliza un `.filter` para filtrar los productos de forma específica que nos de aquellos que cumplan con la característica de tener un precio mayor a 1000, y ahí mismo utilizando un `.map` para que el programa solo nos entregue los nombres y no los precios, por lo que el resultado final es el siguiente:



Ejercicio 3:

```
JS Ejercicio_3.js > ...
1  const personas = [
2    {nombre: "Ana", edad: 22},
3    {nombre: "Luis", edad: 35},
4    {nombre: "María", edad: 28}
5  ];
6
7  const nombre = personas.find(personas => personas.nombre === "Luis");
8
9  console.log(nombre);
10
11 personas.forEach(persona => {
12   console.log(persona.nombre + " " + persona.edad);
13 })
14
15 const suma = personas.map(personas => personas.edad).reduce((acumulador, edad) => acumulador + edad, 0);
16
17 console.log(suma);
```

En el tercer ejercicio, tenemos tres instrucciones que aborde de forma aislada, primero encontrando de forma aislada el objeto el cual coincidiera con el nombre de “Luis”, para esto utilice el `.find`, luego de esto para imprimir cada objeto con sus atributos, utilice el `.forEach`, y por ultimo para sumar las edades, utilice dos cosas, un `.map` para separar, es decir crear un array solo con las edades y `.reduce` para sumar estas mismas y obtener un total, una vez hecho esto, en la consola podemos observar lo siguiente:



Conclusión:

En esta práctica, pude aplicar muchos métodos para la manipulación de objetos y arreglos, al solucionar cada ejercicio por mí misma, entendí cada uno de ellos más, el cómo utilizarlos, el cómo puedo usarlos juntos para obtener cosas específicas, etc.; de esta forma entendí mucho mejor estos métodos para seguir implementándolos en otras prácticas.