

UD4

EJERCICIOS: Map, Filter y Reduce

Resumen Gráfico

map([, , ], cook)
=> [, , ]

filter([, , , isVegetarian)
=> [, ]

reduce([, , eat)
=> 

¡Todos los ejercicios deben de realizarse haciendo uso de las funciones map, filter y reduce!

Ejercicios

Ejercicio 1

Crea una función `multiplicarNumeros (object)` que multiplique todas las variables “numéricas” de un objeto por 2. Por ejemplo para la instancia de objeto siguiente:

```
// antes de la llamada
let menu = {
  width: 200,
  height: 300,
  title: "Mi menu"
};

multiplicarNumeros(menu);

// Después de la llamada
// menu = { width: 400, height: 600, title: "Mi menu" };
```

- La función no debe de devolver ningún tipo de valor.
- Debes de mirar si los valores son numéricos o no antes de multiplicarle su valor por 2.

Ejercicio 2

Tenemos un objeto que guarda los salarios de un equipo de trabajo:

```
let salarios = {
  Juan: 10000,
  Alicia: 16000,
  Lucas: 13000
}
```

Escribe un script para sumar estos salarios y mostrar el resultado de la suma.

Ejercicio 3

Dado el siguiente array de objetos:

```
const iceCreams = [
  { flavor: 'pineapple', color: 'white' },
  { flavor: 'strawberry', color: 'red' },
  { flavor: 'watermelon', color: 'red' },
```

```

    { flavor: 'kiwi', color: 'green' },
    { flavor: 'mango', color: 'yellow' },
    { flavor: 'pear', color: 'green' }
  ];

```

3.1. ¿Cómo podría obtener un array con solo esos helados que son de color rojo ?

3.2. ¿Cómo podríamos obtener un array que solo contenga los sabores de los helados tal como [“pineapple”, “watermelon”, “kiwi”,etc] ?

Ejercicio 4

Dado la siguiente estructura de datos sobre pilotos y años de experiencia:

| | |
|---|---|
| <pre> let pilotos = [{ id: 10, name: "Poe Dameron", years: 14, }, { id: 2, name: "Temmin 'Snap' Wexley", years: 30, },]; </pre> | <pre> { id: 41, name: "Tallissan Lintra", years: 16, }, { id: 99, name: "Ello Asty", years: 22, }]; </pre> |
|---|---|

4.1. ¿Cómo podríamos obtener los años totales de experiencia de todos los pilotos juntos?

4.2. Realice la misma operación del apartado anterior (1.1) pero haciendo uso del operador “arrow”:

4.3 Ahora vamos a tratar de obtener el piloto que con más experiencia. (Esta hay que darle un par de vueltas)

Ejercicio 5

Dado la siguiente estructura de datos sobre pilotos con id, nombre y su división (faction):

| | |
|--|---|
| <pre> var pilots = [{ id: 2, name: "Wedge Antilles", faction: "Rebels", }, { </pre> | <pre> { id: 40, name: "Iden Versio", faction: "Empire", }, { id: 66, </pre> |
|--|---|

```

    id: 8,
    name: "Ciena Ree",
    faction: "Empire",
  },

```

```

    name: "Thane Kyrell",
    faction: "Rebels",
  }
];

```

5.1. ¿Cómo podría obtener los pilotos que fueran solo Rebeldes?

5.2. Realice la misma operación del apartado 2.1 pero haciendo uso del operador “arrow”.

Ejercicio 6

Dado la siguiente estructura de datos sobre personajes

```

var personnel = [
  {
    id: 5,
    name: "Luke Skywalker",
    pilotingScore: 98,
    shootingScore: 56,
    isForceUser: true,
  },
  {
    id: 82,
    name: "Sabine Wren",
    pilotingScore: 73,
    shootingScore: 99,
    isForceUser: false,
  },
  {
    id: 22,
    name: "Zeb Orellios",
    pilotingScore: 20,
    shootingScore: 59,
    isForceUser: false,
  },

```

```

  {
    id: 15,
    name: "Ezra Bridger",
    pilotingScore: 43,
    shootingScore: 67,
    isForceUser: true,
  },
  {
    id: 11,
    name: "Caleb Dume",
    pilotingScore: 71,
    shootingScore: 85,
    isForceUser: true,
  },
];

```

6.1. Intente obtener el total de todas las puntuaciones (tanto pilotingScore como shootingScore juntas) de todos aquellos personajes que sean solo usuarios de la fuerza (isForceUser == true). (CONSEJO: Aquí va a necesitar encadenar diversos operadores).

6.2. Realice lo anterior pero haciendo uso esta vez del operador arrow (=>).