Тут будут задачки на закрепления второго урока :)

1. Добавление и удаление элементов:

-Создайте массив с именами пользователей.

-Используйте push для добавления нового пользователя.

-Используйте pop для удаления последнего пользователя.

-Используйте shift и unshift для добавления и удаления пользователя в начале массива.

1. Работа с порядком элементов:

Критерий для сортировки - возраст. Надо отсортировать по возрастанию.

let users = [

{ name: 'Alice', age: 25 },

{ name: 'Bob', age: 20 },

{ name: 'Charlie', age: 22 },

];

ожидаемый результат:

console.log(users); //[{ name: 'Bob', age: 20 },{ name: 'Charlie', age: 22 },{ name: 'Alice', age: 25 }]

Дополнительная задача реализовать сортировку по дате, дата по возрастанию

Исходный массив объектов:

let events = [

{ name: 'Event 1', date: new Date('2024-03-08T10:30:00') },

{ name: 'Event 2', date: new Date('2024-03-09T14:00:00') },

{ name: 'Event 3', date: new Date('2024-03-07T09:00:00') },

];

Ожидаемый результат:

console.log(events); // [{ name: 'Event 3', date: 2024-03-07T07:00:00.000Z },{ name: 'Event 1', date: 2024-03-08T08:30:00.000Z },{ name: 'Event 2', date: 2024-03-09T12:00:00.000Z }]

3)Работа со строками:

Предположим, у вас есть приложение для управления списком задач, где пользователи могут вводить несколько задач в одну строку, разделяя их запятыми. Вам нужно разбить эту строку на отдельные задачи для обработки и отображения на интерфейсе.

let tasksString = 'Помыть посуду, Погулять с собакой, Сделать покупки';

ожидаемй результат:

console.log(tasksArray); // Вывод: ["Помыть посуду", "Погулять с собакой", "Сделать покупки"]

4)Объединение используя concat:

Предположим, у вас есть приложение для управления списками задач, и вы хотите объединить два списка задач в один для удобного отображения на странице.

// Исходные списки задач

let todoList1 = ['Покормить кота', 'Помыть посуду', 'Сделать уроки'];

let todoList2 = ['Погулять с собакой', 'Сходить в магазин', 'Позвонить маме'];

и вывести после объединения в таком формате:

Общий список задач:

1. Покормить кота

2. Помыть посуду

3. Сделать уроки

4. Погулять с собакой

5. Сходить в магазин

6. Позвонить маме

5) filter:

Отфильтровать активных пользователей (isActive === true)

Исходный массив

let users = [

{ id: 1, name: 'Alice', age: 28, isActive: true },

{ id: 2, name: 'Bob', age: 35, isActive: false },

{ id: 3, name: 'Charlie', age: 22, isActive: true },

{ id: 4, name: 'David', age: 40, isActive: true },

{ id: 5, name: 'Eva', age: 32, isActive: false },

];

ожидаемый результатат:

console.log(activeUsers); // [{ id: 1, name: 'Alice', age: 28, isActive: true },{ id: 3, name: 'Charlie', age: 22, isActive: true },{ id: 4, name: 'David', age: 40, isActive: true }]

6) find:

Поиск юзера по имени Charlie:

let users = [

{ id: 1, name: 'Alice', age: 28, isActive: true },

{ id: 2, name: 'Bob', age: 35, isActive: false },

{ id: 3, name: 'Charlie', age: 22, isActive: true },

{ id: 4, name: 'David', age: 40, isActive: true },

{ id: 5, name: 'Eva', age: 32, isActive: false },

];

Ожидаемый результат:

console.log(foundUser); //{ id: 3, name: 'Charlie', age: 22, isActive: true }

7)includes:

Предположим, у вас есть приложение для учета товаров в магазине, и вам нужно проверить, есть ли определенный товар в наличии с использованием метода includes().

Исходный массив товаров:

let inventory = ['Футболка', 'Джинсы', 'Кроссовки', 'Рюкзак', 'Часы'];

Что надо найти

let itemToCheck = 'Кроссовки';

Если Кроссовки есть в корзине, то вывести в консоль “Кроссовки есть в наличии”, если товара нет в корзине, то вывести - “Товара нет на складе”

Далее реализуйте эту задачу используя indexOf() и eсли Кроссовки есть в корзине, то вывести в консоль “Кроссовки есть в наличии”, если товара нет в корзине, то вывести - “Товара нет на складе”