Массивы Arrays

[] – массив содержится в квадратных скобках

Let firstArray = [1,2,3];

Let secondArray = new Array(5,6,7,8); второй способ создания массива

Let array = []; создание пустого массива

onst clients = ["Mango", "Poly", "Ajax"];

// Указывая в скобках индекс элемента мы получаем его значение

console.log( clients[0] ); // Mango

console.log( clients[1] ); // Poly

console.log( clients[2] ); // Ajax

const clients = ["Mango", "Poly", "Ajax"];

// Значение элемента можно заменить

clients[0] = 'Chelsey';

console.log( clients[0] ); // Chelsey

// Или добавить

clients[3] = 'Alex';

console.log( clients[3] ); // Alex

console.log( clients ); // ["Mango", "Poly", "Ajax", "Alex"]

методы работы с массивом

push pop shift unshift

push - позволяет добавить один или несколько элементов в конец массива.

Push можно зписать много ячеек

pop - удаляет элемент в конце массива и возвращает его.

shift - удаляет элемент из начала массива и возвращает его (элемента) значение.

unshift - добавляет элемент в начало массива.

Название масиива .indexOf(x) - возвращает первый индекс, в котором элемент x был найден в массиве, или число -1, если такого элемента нет. Используйте indexOf тогда, когда необходимо получить непосредсвенно индекс элемента.

Название массива.includes(x) - определяет, содержит ли массив некоторый элемент x, возвращая true или false соответственно. Используйте includes тогда, когда необходимо проверить есть ли элемент в массиве и не важен его порядковый номер.

Синтаксис метода **Array.slice** одинаков для строк и для массивов. Его просто запомнить. Он позволяет извлекать элементы подмножества массива и добавлять их в новый массив. В подавляющем большинстве случаев используется для создания копии части или целого исходного массива.

array.splice - позволяет вставлять новые элементы в произвольное место массива. Однако можно использовать этот метод для удаления и замены существующих элементов. Это швейцарский нож для работы с массивами, в том случае если исходный массив необходимо изменить.

Чтобы удалить элементы в массиве, вы передаете два аргумента методу splice следующим образом: Array.splice(position, num).

Аргумент position указывает позицию (индекс) первого элемента для удаления, а аргумент num определяет количество удаляемых элемента.

// Предположим, у нас есть массив оценок,

// который содержит пять чисел от 1 до 5.

const scores = [1, 2, 3, 4, 5];

// Следующая операция удаляет три элемента массива,

// начиная с первого элемента (индекс 0).

const deletedScores = scores.splice(0, 3);

// Теперь массив scores содержит два элемента.

console.log(scores); // [4, 5]

// А массив deletedScores содержит три удаленных элемента.

console.log(deletedScores); // [1, 2, 3]

Циклы / Loops

Основные понятия

**Тело цикла -** последовательность инструкций, предназначенная для многократного исполнения.

**Итерация** - единичное выполнение тела цикла.

**Счётчик цикла** - переменная, хранящая текущий номер итерации

Для **while** обязательным являются счетчик, проверка, увеличитель счетчика в теле цикла.

Выполняет код лишь в случае успешной проверки.

**Do while** отличается тем, что выполнит код как минимум один раз, потому, что он выполняет код и лишь после этого делает проверку.

**Создание счетчика, Проверка счетчика, Модификатор счетчика.**

**For(let I = 0; i<=10; i++){ тело цикла }**