Заметки по коду

Часть №1

`use strict`;

// create map

let mapTest = new Map();

// добавляем элементы, если элемент уже существовал

// он просто перезаписывается

// можно добавлять строки и числа как ключи

mapTest.set('temp', 11);

mapTest.set('temp', 12);

mapTest.set(5, 1);

// получаем элементы из map по ключу

console.log(mapTest.get(5));

console.log(mapTest.get('temp'));

// узнаём размер map, size это

// свойство, а не метод

console.log(mapTest.size);

// очищаем map

// mapTest.clear();

// console.log(mapTest);

// узнаём есть ли в множестве

// ключ который мы передали в метод

console.log(mapTest.has(5));

продолжение часть №2

`use strict`;

// создадим map и добавим в него значения поумолчанию

let mapEat = new Map([['milk', 250], [{eat : {orn: 454}}, 400], ['bread', 45]]);

// перебираем значения элементов из map ы цикле

// map можно перебрать только через of

// в map перебор происходит в порядке вставки элементов

for (let elem of mapEat){

    console.log(elem);

}

// перебираем map по ключам

for(let elem of mapEat.keys()){

    console.log(elem);

}

// перебираем map по значениям

for(let elem of mapEat.values()){

    console.log(elem);

}

// у map есть встроенный метод forEach

// первый параметр это значение второй параметр это ключ

mapEat.forEach((value, key, map) => {

    console.log(key, value);

});

Продолжение часть №3

`use strict`;

// множество можно создать с помощью преобразования

// из объекта в map, с помощью метода который преобразует множество пар

// объекта в набор массивов с ключ значением, который

// потом передаётся в конструктор map

let objTest = {

    name : 'Kostan',

    age : 32,

    money : 'dollar'

};

let mapEat = new Map(Object.entries(objTest));

console.log(mapEat, typeof(mapEat));

// теперь преобразуем map в объект обратно

// используем особый метод, который принимает внутрь массив

// из массивов, который преобразуется в объект

let obj = Object.fromEntries(mapEat.entries());

console.log(obj.name);