

Evgen959 / JavaScript Public[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#)[JavaScript / Homework / Les016 / example_01 / index.js](#)

Evgen959 newHomeworkL16

1 minute ago



116 lines (100 loc) · 3.71 KB

Code

Blame

Raw



```
1 console.log('----- №1 -----');
2 //№1 Написать функцию, которая принимает на вход массив имен и возвращает
3 // массив имен, начинающихся на букву "а" (большую или маленькую).
4 const arrayName = ["Владимер", "Артур", "Сергей", "Лев", "Анна", "Олександр",
5 "Катерина", "Ольга", "Андрей"];
6 const arrayNameFiltr = string => string[0].toUpperCase() === 'A';
7 // const arrayNameFiltr = (string) => {
8 //     // console.log(string[0]);
9 //     return (string[0] === 'A' || string[0] === 'a');
10 // }
11 const arrayNameFiltrA = arrayName.filter(arrayNameFiltr);
12 console.log(arrayNameFiltrA);
13
14
15 console.log('----- №2 -----');
16 //№2 Написать функцию, которая принимает на вход массив чисел и возвращает
17 // их сумму. Если получится использовать метод reduce - будет здорово.
18 // Завтра его рассмотрим.
19 const arrayNumbers = [6, 8, 5, 1, 9, 3];
20 const sumNumbers = arrayNumbers.reduce((accumulator, currentValue) => accumulator + curr
21 console.log(sumNumbers);
22
23
24 console.log('----- №3 -----');
25 //№3 Написать функцию, которая принимает на вход массив чисел и возвращает
26 // массив из тех же элементов, но в обратном порядке.
27 let arrayUnshift = [];
28 const arrayUnshiftForEach = (number) => {
29     arrayUnshift.unshift(number); // пушим в начало пустого массива
30 }
31 arrayNumbers.forEach(arrayUnshiftForEach);
32 console.log(arrayUnshift);
33
34
35 console.log('----- №4 -----');
36 //№4 Написать функцию, которая примет объект вида:
```

```
37 // {
38 //   'customer-id-1': {
39 //     name: 'William',
40 //     age: 54
41 //   },
42 //   'customer-id-2': {
43 //     name: 'Tom',
44 //     age: 17
45 //   }
46 // }
47 // и вернет массив объектов вида:
48
49 // [{name: 'William', age: 54, id: 'customer-id-1'}, {name: 'Tom', age: 17,
50 // id: 'customer-id-2'}]
51 const objectA = {
52   'customer-id-1': {
53     name: 'William',
54     age: 54
55   },
56   'customer-id-2': {
57     name: 'Tom',
58     age: 17
59   }
60 }
61 const objectInArray = Object.entries(objectA);
62 // console.log(objectInArray);
63 ✓ const arrayB = (arr) => {
64   return {...arr[1], id: arr[0]};
65   // arr[1].id = arr[0];
66   // return arr[1];
67 }
68 const objectInArrayEnd = objectInArray.map(arrayB);
69 console.log(objectInArrayEnd);
70
71
72 console.log('----- №5 -----');
73 //№5 Написать функцию, которая принимает на вход 2 объекта, и склеивает их в один,
74 // на выходе должен быть объект, который содержит все свойства обоих объектов.
75 // Если свойства(ключи) во входящих объектах повторяются, то взять значение из
76 // второго объекта.
77 // Входящие:
78
79 // {
80 //   key1: 'value1',
81 //   key2: 'value2',
82 //   key3: 'value3',
83 //   key4: 'value4',
84 //   key5: 'value5'
85 // };
86
87 // {
88 //   key3: 'new_value3',
```

```
89    //    key4: 'new_value4',
90    //    key5: 'new_value5',
91    //    key6: 'value6',
92    //    key7: 'value7'
93    // };
94
95    const object1 = {
96        key1: 'value1',
97        key2: 'value2',
98        key3: 'value3',
99        key4: 'value4',
100       key5: 'value5'
101    };
102    const object2 = {
103        key3: 'new_value3',
104        key4: 'new_value4',
105        key5: 'new_value5',
106        key6: 'value6',
107        key7: 'value7'
108    };
109
110    const objectNew = (obj1, obj2) => {
111        return {...obj1, ...obj2};
112    }
113    console.log(objectNew(object1, object2));
```