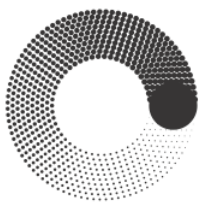


**федеральное государственное автономное образовательное
учреждение высшего образования**



МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет информационных технологий

Кафедра Информатики и информационных технологий

направление подготовки

09.03.02 «Информационные системы и технологии»

ЛАБОРАТОРНАЯ РАБОТА № 1

Дисциплина: Функциональное программирование

Тема: Основы JS для функционального программирования

Выполнил: студент группы **__221-3711__**

_____ Чернышов_Н._А. _____
(Фамилия И.О.)

Дата, подпись _____
(Дата) (Подпись)

Проверил: _____
(Фамилия И.О., степень, звание) **(Оценка)**

Дата, подпись _____
(Дата) (Подпись)

Замечания: _____

Москва 2025

1. Разработайте набор чистых функций для работы с массивами:

- Функция, которая принимает массив чисел и возвращает новый массив, содержащий только четные числа.
- Функция, которая принимает массив чисел и возвращает новый массив, содержащий квадраты этих чисел.
- Функция, которая принимает массив объектов и возвращает новый массив, содержащий только объекты с определенным свойством.
- Функция, которая принимает массив чисел и возвращает их сумму.

2. Создайте функцию высшего порядка, которая принимает функцию и массив в качестве аргументов и применяет функцию к каждому элементу массива, возвращая новый массив с результатами.

3. Используя разработанные функции, выполните следующие математические операции:

- Найдите сумму квадратов всех чётных чисел в заданном массиве.
- Найдите среднее арифметическое всех чисел, больших заданного значения, в заданном массиве объектов.

1. Разработайте набор чистых функций для работы с массивами:

- Функция, которая принимает массив чисел и возвращает новый массив, содержащий только четные числа.

```
2. export function extractEvenNumbers(numberArray) {  
3.   const evenNumbersList = [];  
4.   for (let i = 0; i < numberArray.length; i++) {  
5.     if (numberArray[i] % 2 === 0) {  
6.       evenNumbersList.push(numberArray[i]);  
7.     }  
8.   }  
9.   return evenNumbersList;  
10. }
```

Данная функция предназначена для извлечения четных чисел из массива. Для этого сначала создается пустой массив, куда будут помещены отобранные значения. Затем, используя цикл `for`, функция перебирает элементы исходного

массива, проверяя каждый элемент на четность. Если элемент оказывается четным, он добавляется в созданный ранее массив с помощью метода `push`. В конечном итоге функция возвращает новый массив, содержащий только четные числа из исходного массива.

- Функция, которая принимает массив чисел и возвращает новый массив, содержащий квадраты этих чисел.

```
2. export function calculateSquares(numberArray) {  
3.   const squareList = [];  
4.   for (let i = 0; i < numberArray.length; i++) {  
5.     squareList.push(numberArray[i] * numberArray[i]);  
6.   }  
7.   return squareList;  
8. }
```

Функция обрабатывает входной массив, преобразуя каждый его элемент. В начале работы функции создается пустой массив, который будет содержать результаты преобразования. Затем, с помощью цикла `for`, функция перебирает элементы исходного массива. Для каждого элемента вычисляется его квадрат, и полученное значение добавляется в новый массив. В результате функция возвращает новый массив, состоящий из квадратов элементов исходного массива.

- Функция, которая принимает массив объектов и возвращает новый массив, содержащий только объекты с определенным свойством.

```
9. export function filterGreaterThan(array, threshold) {  
10. return array.filter(item => item > threshold);  
11. }
```

Функция использует метод `filter` для отбора элементов из входного массива. Метод `filter` создает новый массив, содержащий только те элементы исходного массива, которые удовлетворяют условию, заданному стрелочной функцией. Стрелочная функция применяется к каждому элементу массива, и если она возвращает `true`, элемент включается в новый массив.

- Функция, которая принимает массив чисел и возвращает их сумму.

```
12. export function calculateArraySum(numberArray) {  
13. let totalSum = 0;  
14. for (let i = 0; i < numberArray.length; i++) {  
15.   totalSum += numberArray[i];  
16. }  
17. return totalSum;  
18. }
```

```
18. }
```

Функция предназначена для вычисления суммы элементов массива. Для этого внутри функции объявляется переменная, которая изначально устанавливается в ноль и будет накапливать сумму. Затем, используя цикл `for`, функция проходит по каждому элементу входного массива и добавляет его значение к переменной-аккумулятору суммы.

2. **Создайте функцию высшего порядка, которая принимает функцию и массив в качестве аргументов и применяет функцию к каждому элементу массива, возвращая новый массив с результатами.**

```
3. export function transformArray(array, transformationFunction) {  
4.   return array.map(transformationFunction);  
5. }
```

Функция выполняет преобразование элементов массива. Она использует метод `map`, который проходит по каждому элементу входного массива и применяет к нему функцию `fn`. Результат применения функции к каждому элементу становится элементом нового массива. В итоге функция возвращает новый массив, содержащий преобразованные элементы.

```
export function multiplyByTwo(number) {  
  return number * 2;  
}
```

Результат:

```
(6) [2, 4, 6, 8, 10, 12]
```

4. **Используя разработанные функции, выполните следующие математические операции:**

- Найдите сумму квадратов всех чётных чисел в заданном массиве.

```
5. const evenArray = extractEvenNumbers(numberArray)  
6. const squaresArray = calculateSquares(evenArray)  
7. let totalSumOfSquares = calculateArraySum(squaresArray)  
8. console.log("Сумма квадратов всех чётных чисел: " + totalSumOfSquares)
```

Вызываю все нужные функции.

Результат:

Сумма квадратов всех чётных чисел: 120

- Найдите среднее арифметическое всех чисел, больших заданного значения, в заданном массиве объектов.

```
6. const filteredArray = filterGreaterThan(numberArray, 3)
7. let averageOfFiltered = calculateArraySum(filteredArray) /
  filteredArray.length
8. console.log("Среднее арифметическое всех чисел, больших заданного значения: "
  + averageOfFiltered)
```

Вызываю нужные функции.

Результат:

Среднее арифметическое всех чисел, больших заданного значения: 6.5

Ссылка на гит: <https://github.com/НАНАНАНАНААНАНААА/Labs.git>