





Workshop

Урок 1





Регламент

-  Получаем задание
-  Выполняем задание определенное время
-  Проверяем правильность выполнения
-  Переходим к новому заданию



Задания





Задание 1 (тайминг 20 минут)

1. Создайте функцию `mergeArrays`, которая принимает два массива и возвращает новый массив, содержащий все элементы из обоих массивов. Используйте `spread` оператор для объединения массивов.
`mergeArrays([1, 2, 3], [4, 5, 6]);` // Ожидаемый результат: `[1, 2, 3, 4, 5, 6]`
2. Создайте функцию `removeDuplicates`, которая принимает любое количество аргументов и возвращает новый массив, содержащий только уникальные значения. Используйте `rest` оператор для сбора всех аргументов в массив а затем `filter` для определения дубликатов.
`removeDuplicates(1, 2, 3, 2, 4, 1, 5);` // Ожидаемый результат: `[1, 2, 3, 4, 5]`





Задание 2 (Чистые функции 25 минут)

Текст задания

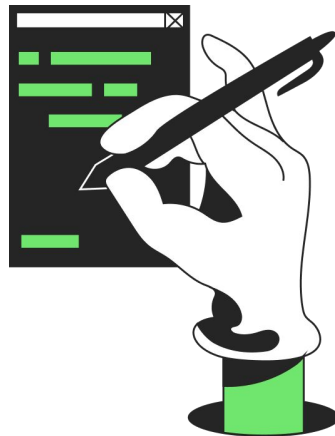
1. Напишите функцию `getEvenNumbers`, которая принимает массив чисел в качестве аргумента и возвращает новый массив, содержащий только четные числа.
2. Задача: Напишите функцию `calculateAverage`, которая принимает массив чисел в качестве аргумента и возвращает среднее значение этих чисел.
3. Напишите функцию `capitalizeFirstLetter`, которая принимает строку в качестве аргумента и возвращает новую строку, в которой первая буква каждого слова является заглавной.





Задание 3 (Замыкания 20 минут)

1. Напишите функцию `createCalculator`, которая принимает начальное значение и возвращает объект с двумя методами: `add` и `subtract`. Метод `add` должен увеличивать значение на переданное число, а метод `subtract` должен уменьшать значение на переданное число. Значение должно быть доступно только через методы объекта, а не напрямую.





Задание 4 (Лексический контекст 15 минут)

1. Напишите функцию `createGreeting`, которая принимает имя пользователя и возвращает функцию, которая будет выводить приветствие с использованием этого имени.

// Пример использования:

```
const greeting = createGreeting('John');  
greeting(); // Ожидаемый результат: "Hello, John!"
```



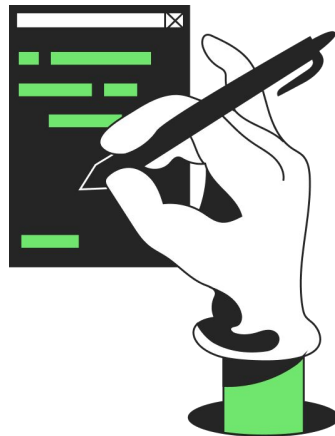


Задание 5 (тайминг 15 минут)

1. Задача: Напишите функцию `createPasswordChecker`, которая принимает допустимую длину пароля в качестве аргумента и возвращает функцию для проверки введенного пароля. Возвращаемая функция должна принимать пароль и возвращать `true`, если его длина соответствует допустимой, и `false` в противном случае.

// Пример использования:

```
const isValidPassword = createPasswordChecker(8);  
console.log(isValidPassword('password')); // Ожидаемый результат:  
false  
console.log(isValidPassword('secret')); // Ожидаемый результат: true
```





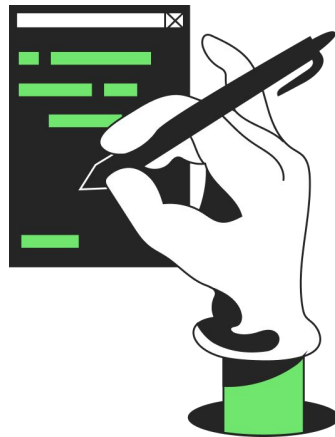
Задание 6 (Рекурсия 25 минут)

1. Напишите рекурсивную функцию `sumDigits`, которая принимает положительное целое число в качестве аргумента и возвращает сумму его цифр.

// Пример использования:

```
console.log(sumDigits(123)); // Ожидаемый результат: 6 (1 + 2 + 3)
```


```
console.log(sumDigits(456789)); // Ожидаемый результат: 39 (4 + 5 + 6 + 7 + 8 + 9)
```





Итоги



Спасибо 
за внимание

