

# Введение в Javascript. Массивы и циклы

Frontend. Урок 20

# Вопросы на повторение

1. Что такое условный оператор в JS ?
2. Какие типы условных операторов есть в JS?
3. С помощью какого ключевого слова можно перечислить варианты внутри блока switch?

# Постановка проблемы

Представим, что нам нужно выполнить определенный код 2 раза - мы можем его скопировать и вставить ниже, а теперь представим, что нам нужно переиспользовать этот код 10 - 15 -100 раз подряд. Согласитесь, что копировать этот код будет не лучшей практикой.

Представьте, что нам нужно хранить большое количество переменных, например имена студентов нашей группы и уметь из программного кода получать список и обращаться к каждому элементу этого списка.

Как можно по-другому решить эту проблему?

# Циклы в JavaScript

Циклы - это конструкции в программировании, которые позволяют выполнять один и тот же блок кода несколько раз. JavaScript предоставляет несколько типов циклов:

**Цикл for:**

```
for (let i = 0; i < 5; i++) {  
  console.log(i);  
}
```

- **let i = 0** - инициализация переменной **i** с начальным значением 0.
- **i < 5** - условие, при котором цикл будет выполняться.
- **i++** - выражение, выполняемое после каждой итерации цикла (увеличение **i** на 1).

# Цикл while

Цикл while отличается от цикла for тем, что количество итераций неизвестно заранее. Продолжительность цикла зависит от продолжительности истинности условия

```
let i = 0;
while (i < 5) {
  console.log(i);
  i++;
}
```

**while (i < 5)** - цикл будет выполняться, пока условие истинно.

# Цикл do...while

```
let i = 0;  
do {  
  console.log(i);  
  i++;  
} while (i < 5);
```

В отличие от цикла while, цикл **do...while** гарантирует, что **блок кода выполнится хотя бы один раз**, даже если условие изначально ложно.

# Массивы в JavaScript

- **Массив** - это упорядоченная коллекция элементов. В JavaScript массивы могут содержать элементы разных типов данных:

```
let fruits = ['яблоко', 'апельсин', 'банан', 'капибара'];
```

```
// Доступ к элементам массива по индексу
```

```
console.log(fruits[0]); // 'яблоко'
```

```
// Изменение элемента массива
```

```
fruits[1] = 'груша';
```

# Базовые методы и свойства массивов

- push

- Метод push используется для добавления элемента в конец массива.

Например:

```
let fruits = ['яблоко', 'апельсин', 'банан', 'капибара'];  
fruits.push('киви');  
console.log(fruits); // ['яблоко', 'апельсин', 'банан', 'капибара',  
                        'киви']
```

- length

- Свойство length возвращает количество элементов в массиве:

```
console.log(fruits.length); // 5
```



# Базовые методы и свойства массивов

- for...of
  - Цикл for...of предоставляет более краткую форму для перебора элементов массива:

```
for (let fruit of fruits) {  
  console.log(fruit);  
}
```

# Вопросы на закрепление

- 1) Что такое массив?
- 2) Где и для чего мы можем использовать циклы?

# Практика А

## Задача: Работа с массивом чисел

- Создайте массив чисел `numbers` с несколькими произвольными значениями.
- Используя цикл `for`, выведите каждый элемент массива в консоль.
- Используя цикл `while`, найдите сумму всех элементов массива и выведите результат.

# Практика В

## Задача: Работа с массивом строк

- Создайте массив строк **fruits** с названиями нескольких фруктов.
- Используя цикл `for...of`, выведите каждый элемент массива в верхнем регистре в консоль.
- Используя методы массивов, добавьте новый фрукт в конец массива.
- Проверьте, является ли массив после добавления фрукта длиннее определенной длины. Если да, выведите "Массив слишком длинный", иначе "Массив подходит по длине".

# Практика С

## Задача: Работа с массивом объектов

- Создайте массив объектов `students`, представляющий учеников с именем и возрастом.
- Используя цикл `for...of`, выведите информацию о каждом ученике в консоль (например, "Имя: Alice, Возраст: 22").
- Используя методы массивов, добавьте нового ученика в массив.
- Проверьте, есть ли ученик с определенным именем в массиве. Если да, выведите "Ученик найден", иначе "Ученик не найден".

```
• let students = [  
  { name: 'Alice', age: 22 },  
  { name: 'Bob', age: 24 },  
  { name: 'Charlie', age: 21 }  
];
```