

# JS: Operators

# НАШИ ПРАВИЛА



Включенная камера



Вопросы по поднятой руке



Не перебиваем друг друга



Все вопросы, не связанные с тематикой курса (орг-вопросы и т. д.), должны быть направлены куратору



Подготовьте свое рабочее окружение для возможной демонстрации экрана (закройте лишние соцсети и прочие приложения)

# Повторим;)

Какие способы объявления переменных есть в JavaScript?

Сколько примитивных типов в JavaScript?

Что входит в тип Object

С помощью какого оператора можно проверить тип переменной?

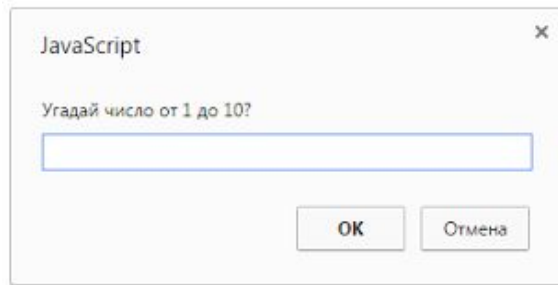
# ЦЕЛЬ

Изучить различные виды операторов в JavaScript

# ПЛАН ЗАНЯТИЯ

- Math operators
- Type conversion

# Встроенные функции. Взаимодействие с пользователем



A screenshot of a JavaScript alert dialog box. The title bar reads "JavaScript" with a close button (X) on the right. The main text inside the box says "Угадай число от 1 до 10?". Below the text is a single-line text input field. At the bottom of the dialog are two buttons: "ОК" (OK) and "Отмена" (Cancel).

# console.log()

Чтобы вывести что-то на консоль из нашего кода, существует функция `console.log`.

Обычный пользователь сайта не увидит такой вывод, так как он в консоли. Чтобы увидеть его, либо откройте консольную панель инструментов разработчика, либо нажмите Esc, находясь в другой панели: это откроет консоль внизу.

```
let message = "Привет, мир!";  
console.log(message);
```

# alert()

- alert используется для вывода всплывающего диалогового окна с сообщением.
- Принимает один параметр - текст сообщения.

Она показывает сообщение и ждёт, пока пользователь нажмёт кнопку «OK».

```
1 alert("Hello");
```



# prompt()

- `prompt` используется для вывода всплывающего диалогового окна с полем для ввода текста.
- Принимает два параметра: текст сообщения и необязательное значение по умолчанию для поля ввода.

```
1 let age = prompt('Сколько тебе лет?', 100);  
2
```

# confirm()

Функция `confirm` отображает модальное окно с текстом вопроса `question` и двумя кнопками: ОК и Отмена.

```
1 let isBoss = confirm("Ты здесь главный?");  
2  
3 alert( isBoss ); // true, если нажата ОК
```

# Выражения

```
= += -= *= /= %= << >> =  
x != y; x %= 2;  
x = 5 + 12;
```

## Expressions

```
true || false j = i++;  
var result = 5 + 4 * 12 / 4;
```

**В JavaScript поддерживаются следующие математические операторы:**

**Сложение +,**

**Вычитание −,**

**Умножение \*,**

**Деление /,**

**Взятие остатка от деления %,**

**Возведение в степень \*\*.**

## Сложение

+

Объединение чисел в одно целое.

```
let a = 5;  
let b = 10;  
let result = a + b;  
console.log(result); // Выведет 15
```

## Вычитание

-

Вычитает правое число от левого.

```
let a = 10;  
let b = 2;  
let result = a - b;  
console.log(result); // Выведет 8
```

## Умножение

\*

Умножает два числа вместе.

```
let a = 3;  
let b = 7;  
let result = a * b;  
console.log(result); // Выведет 21
```

## Деление

/

Делит левое число на правое.

```
let a = 10;  
let b = 2;  
let result = a / b;  
console.log(result); // Выведет 5
```



## Взятие остатка %

Оператор взятия остатка %, несмотря на обозначение, никакого отношения к процентам не имеет.

Результат  $a \% b$  – это остаток от целочисленного деления  $a$  на  $b$ .

Например:

```
1 alert( 5 % 2 ); // 1, остаток от деления 5 на 2
2 alert( 8 % 3 ); // 2, остаток от деления 8 на 3
3 alert( 8 % 4 ); // 0, остаток от деления 8 на 4
```

## Возведение в степень \*\*

Оператор возведения в степень  $a ** b$  возводит  $a$  в степень  $b$ .  
В школьной математике мы записываем это как  $a^b$ .

Например:

```
1 alert( 2 ** 2 ); // 22 = 4
2 alert( 2 ** 3 ); // 23 = 8
3 alert( 2 ** 4 ); // 24 = 16
```

## Сложение строк при помощи +

Давайте рассмотрим специальные возможности операторов JavaScript, которые выходят за рамки школьной арифметики.

Обычно при помощи плюса '+' складывают числа.

Но если оператор '+' применить к строкам, то он их объединяет в одну:

```
1 let s = "моя" + "строка";  
2 alert(s); // моястрока
```

# Преобразование типов

## Сложение строк и чисел при помощи +

Обратите внимание, если хотя бы один операнд является строкой, то второй будет также преобразован в строку.

Например:

```
1 alert( '1' + 2 ); // "12"  
2 alert( 2 + '1' ); // "21"
```

Какой будет результат?

```
1 alert(2 + 2 + '1' );
```

Другие арифметические операторы работают только с числами и всегда преобразуют операнды в числа.

Например, вычитание и деление:

```
1 alert( 6 - '2' ); // 4, '2' приводится к числу
2 alert( '6' / '2' ); // 3, оба операнда приводятся к числам
```

## Строковое преобразование

Строковое преобразование происходит, когда требуется представление чего-либо в виде строки, мы можем использовать функцию **String(value)**, чтобы преобразовать значение к строке:

```
1 let value = true;
2 alert(typeof value); // boolean
3
4 value = String(value); // теперь value это строка "true"
5 alert(typeof value); // string
```

## Численное преобразование

Мы можем использовать функцию **Number(value)**, чтобы явно преобразовать value к числу:

```
1 let str = "123";  
2 alert(typeof str); // string  
3  
4 let num = Number(str); // становится числом 123  
5  
6 alert(typeof num); // number
```



## Логическое преобразование

Логическое преобразование может быть выполнено явно с помощью функции **Boolean(value)**.

```
1 alert( Boolean(1) ); // true
2 alert( Boolean(0) ); // false
3
4 alert( Boolean("Привет!") ); // true
5 alert( Boolean("") ); // false
```



# **Ваша новая IT-профессия – Ваш новый уровень жизни**

Программирование с нуля в  
немецкой школе AIT TR GmbH