







Frontender[1.0] JavaScript - Логические операторы


 YouTube	https://youtu.be/439EMrMTMf4
 Telegram	https://t.me/Dmitry_Kolotilshikov
 Github	https://github.com/DmitryKolotilshikov/
 Boosty	https://boosty.to/dmitry_ko
# Номер урока	8



Задачи к этому уроку тут https://boosty.to/dmitry_ko



<https://learn.javascript.ru/logical-operators>

 RU

JAVASCRIPT.RU

Учебник

Курсы

Форум

Тесты знаний

Скринкасты

Купить EPU

Раздел

Основы JavaScript

Навигация по уроку

|| (ИЛИ)

||= (Логическое присваивание ИЛИ)

&& (И)

&&= (Логическое присваивание И)

! (НЕ)

Задачи (10)

Комментарии

Поделиться

→ Язык JavaScript → Основы JavaScript

Логические операторы

В JavaScript есть семь логических операторов:

- `||` (ИЛИ)
 - `||=` (Оператор логического присваивания ИЛИ)
- `&&` (И)
 - `&&=` (Оператор логического присваивания И)
- `!` (НЕ)
- `??` (Оператор нулевого слияния)
 - `??=` (Оператор нулевого присваивания)

Здесь мы рассмотрим первые пять, операторы `??` и `??=` будут в следующей статье.

9 ноября 2023 г.

Frontender[1.0] JavaScript - Логические операторы

1



<https://javascript.info/logical-operators>

EN

JAVASCRIPT.INFO

Buy EPUE

Chapter

JavaScript Fundamentals

Lesson navigation

|| (OR)

OR "||" finds the first truthy value

&& (AND)

AND "&&" finds the first falsy value

→ The JavaScript language → JavaScript Fundamentals

June 5, 2022

Logical operators

There are four logical operators in JavaScript: `||` (OR), `&&` (AND), `!` (NOT), `??` (Nullish Coalescing). Here we cover the first three, the `??` operator is in the next article.

Although they are called “logical”, they can be applied to values of any type, not only boolean. Their result can also be of any type.

Let’s see the details.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <script src="script.js"></script>
  <title>Title</title>
</head>
<body>
</body>
</html>
```

```
// Logical operators

// || - Логическое ИЛИ
// || - Возвращает первое положительное truthy значение или последнее значение
// || - Находит первое истинное значение или возвращает последний операнд

let a;

a = 1 || 2; // 1
a = 0 || 2; // 2

a = true || false; // true (1 значение)
a = false || true; // true (2 значение)
a = false || false; // false (возвращает последнее значение)

a = false || false || true; // true (возвращает последнее значение)
a = null || undefined || 1; // вернет 1

console.log(a);

const age = 25;

if (age > 20 || age < 35) {
  console.log("Вы приняты!");
}

//-----
```

```

// && - Логическое И
// && - при проверках возвращает true, если оба аргумента истинны, а иначе – false
// или возвращает первое ложное значение как результат выражения

/*
Вычисляет операнды слева направо.
Каждый операнд преобразует в логическое значение. Если результат false,
останавливается и возвращает исходное значение этого операнда.
Если все операнды были истинными, возвращается последний.

Другими словами, && (И) возвращает первое ложное значение. Или последнее, если ничего не найдено.

Разница в том, что && (И) возвращает первое ложное значение, а ||(ИЛИ)– первое истинное.
*/

let b;

if (0 && 5) {
  // if (false) проверка отрицательная
  console.log("это сообщение никто не увидит");
}

b = 0 && 5; // результат выражения
b = true && "100%"; // вернет 100%

if (true && "100%") {
  // if (true) проверка положительная
  console.log("это сообщение увидит весь мир!");
}

console.log(b);

const check = 10 && 0 && 20 && "Hello";

console.log("check: ", check);

const hour = 10;
const minutes = 35;

if (hour === 10 && minutes > 30) {
  console.log("Заходите!");
}

//-----

// ! - Логическое отрицание. По сути меняет значение на противоположное
/*
Оператор принимает один аргумент и выполняет следующие действия:

1) Сначала приводит аргумент к логическому типу true/false.
2) Затем возвращает противоположное значение.
*/

let c;

c = !true; // false
c = !false; // true

```

```
c = !0; // true
c = "Hello"; // просто строка
c = !"Hello"; // false

// !! - для преобразования значений к логическому (boolean) типу

c = !!"Hello"; // true
c = !!""; // false
c = !!0; // false
c = !!1; // true
c = !!null; // false

// Boolean - встроенная функция

c = Boolean(1); // true
c = Boolean(null); // false

console.log(c);

//-----
// ?? - Возвращает правый операнд если левый равен null или undefined, иначе возвращает левый операнд

let d;

d = 1 ?? 2; // 1
d = null ?? 10; // 10
d = undefined ?? 10; // 10
d = false ?? 10; // false
d = true ?? 10; // true
d = '' ?? 10; // ''
d = 0 ?? 10; // 0

console.log(d);
```