

Введение в JavaScript. Основные конструкции языка, переменные, типы данных.

Операторы сравнения

Операторы сравнения

- Возвращают логическое значение

```
2 > 1 // true
2 < 1 // false
2 >= 1 // true
2 <= 1 // false
2 == 2 // true
2 != 1 // true
```

```
let isBigger = 2 > 1;
```

Операторы сравнения

- Работает и со строками

```
'я' > 'А'
```

```
'Стул' > 'Стол'
```

```
'Стул' < 'стол'
```

Операторы сравнения

- При сравнении разных типов

```
'2' > 1 // true  
'2' == 2 // true  
'01' == 1 // true  
'1' != 1 // false
```

Строгое сравнение

- Не выполняет преобразование типов

```
'2' === 2 // false  
2 === 2 // true  
'1' !== 1 // true  
1 !== 1 // false
```

Условные операторы

Условные операторы if и «?»

- Позволяют выполнить разный код в зависимости от условий.

Условные операторы if и «?»

true



```
if (time > 18) {  
    alert('Наш менеджер свяжется с вами завтра');  
}
```

Условные операторы if и «?»

```
if (0) { // false  
    alert('Не работает');  
}
```

```
if (1) { // true  
    alert('Сработает');  
}
```

Условные операторы if и «?»

```
if (time > 18) {  
    alert('Наш менеджер свяжется с вами завтра');  
} else {  
    alert('В ближайшее время с вами свяжется наш менеджер');  
}
```

Условные операторы if и «?»

```
if (time > 18) {  
    alert('Наш менеджер свяжется с вами завтра');  
} else {  
    alert('В ближайшее время с вами свяжется наш менеджер');  
}
```

Условные операторы if и «?»

```
if (time > 18) {  
    alert('Наш менеджер свяжется с вами завтра');  
} else if (time < 9) {  
    alert('Наш менеджер свяжется с вами в течении дня');  
} else {  
    alert('В ближайшее время с вами свяжется наш менеджер');  
}
```

Условные операторы if и «?»»

С помощью if:

```
let theme;  
  
if (time > 21) {  
  theme = 'dark';  
} else {  
  theme = 'light';  
}
```

С помощью «?»»:

```
const theme = (time > 21) ? 'dark' : 'light';
```

Условные операторы if и «?»

```
const theme = (time > 21) ? 'dark' : 'light';
```

A diagram illustrating the components of a ternary operator. The code snippet is displayed on a dark background. The expression (time > 21) is underlined in red, with a red arrow pointing down to it from above. The string 'dark' is also underlined in red, with a red arrow pointing down to it from above. The string 'light' is underlined in red. The colon and the question mark are also present in the expression.

Выражения и инструкции

Выражения (Expressions)

```
const theme = (time > 21) ? 'dark' : 'light';  
const num = 5 + 10 + 15;  
const isBiggerThenZero = 100 > 0;  
const text = `Sum: ${num}`;
```


Выражения и инструкции

Инструкция (Statement)

```
if (time > 18) {  
    alert('Наш менеджер свяжется с вами завтра');  
}
```

```
let theme = if (time > 21) { // ERROR  
    // some code  
}
```

Конструкция «switch»

```
const theme = 'dark';

if (theme === 'light') {
  alert('Светлая тема');
} else if (theme === 'dark') {
  alert('Тёмная тема');
} else {
  alert('Неизвестная тема');
}
```

Конструкция «switch»

```
switch (theme) {  
    case 'light':  
        alert('Светлая тема');  
        break;  
    case 'dark':  
        alert('Тёмная тема');  
        break;  
    default:  
        alert('Неизвестная тема');  
        break;  
}
```

Логические операторы

Логические операторы

- || - ИЛИ
- && - И
- ! - НЕ

Оператор «||»

```
if (age < 0 || age > 150) { // Если возраст меньше 0 ИЛИ больше 150
    alert('неверный возраст');
}
```

```
if (0 || 1) { // Работает как if (false || true)
    alert('Привет!');
}
```

```
if (age < 0 || age > 150 || isRobot) {
    alert('Неверные данные');
}
```

Оператор «&&»

```
if (time < 9 && time > 18) { // Если время меньше 9 И больше 18  
    alert('Рабочий день закончен');  
}
```

```
if (1 && 0) { // Работает как if (true && false)  
    alert('Не сработает');  
}
```

```
if (time < 9 && time > 18 && isOfficeClosed)  
    alert('Рабочий день закончен');  
}
```

Оператор «!»

```
if (!isWorkTime) {  
    alert('Рабочий день окончен');  
}
```

```
!!0 // false  
!!1 // true
```


Ссылки

- Условные операторы
 - <https://learn.javascript.ru/logical-ops>
 - [https://ru.hexlet.io/courses/introduction to programming/lessons/boolean/theory unit](https://ru.hexlet.io/courses/introduction%20to%20programming/lessons/boolean/theory_unit)
 - <https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Statements/if...else>
 - <https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Operators...>
- Выражения и инструкции
 - [https://ru.hexlet.io/courses/introduction to programming/lessons/expressions/theory unit](https://ru.hexlet.io/courses/introduction%20to%20programming/lessons/expressions/theory_unit)
 - <https://medium.com/launch-school/javascript-expressions-and-statements-4d32ac9c0e74>
- Логические операторы
 - <https://learn.javascript.ru/logical-operators>
 - <https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Operators...>
 - <https://mariusschulz.com/blog/the-and-and-or-operators-in-javascript>

Ссылки

- Конструкция switch
 - <https://learn.javascript.ru/switch>
- Примитивы как объекты
 - <https://learn.javascript.ru/primitives-methods>
 - <https://javascriptweblog.wordpress.com/2010/09/27/the-secret-life-of-javascript-primitives/>
- <https://developer.mozilla.org/ru/docs/Web/JavaScript> -
Справочник по JavaScript
- <https://learn.javascript.ru/> - Учебник по JS