#### JavaScript: условия и ветвление



ORTDNIPRO.ORG/WEB

## 1. Тип Boolean

#### Тип boolean

```
let a = true;
 3
 4
        let b = false;
 5
 6
        // true boolean
        console.log(a, typeof(a));
 8
9
        // false boolean
        console.log(b, typeof(b));
10
11
```

Переменная типа boolean содержит один из всего 2 вариантов значения: истина (true) или ложь (false). Несмотря на простоту, этот тип применяется во многих конструкциях языка.

# 2. Oператор if/else

#### Условный оператор if/else

```
let age = prompt('Enter your age');
age = parseInt(age);

if(age >= 18) {
    console.log('Welcome!');
}else {
    console.log('Sorry...');
}
```

Оператор **if-else** позволяет задать условия, в зависимости от верности/не верности которого выполниться первая (**if**) или вторая (**else**) ветка кода (допускается возможность делать «многоэтажный» **if** используя конструкцию ...**else if...** 

#### Условный оператор if/else

```
let balance = 10_000;
 3
        console.log(`Ваш баланс: ${balance} грн.`);
 5
        let sum = +prompt('Сумма для снятия');
 6
 8
        if(sum <= balance){</pre>
            balance = balance - sum;
10
            console.log(`Выдано ${sum} грн.`);
11
        }else {
            console.log('Недостаточно средств!');
12
13
14
        console.log(`Ваш баланс: ${balance} грн.`);
15
16
```

Фигурные скобки в операторе **if-else** позволяет задать несколько строк кода для выполнения в случае истинности/ложности условия. После выполнение кода условного оператора, программа приступает к дальнейшему выполнению кода программы.

#### «Многоэтажный» if/else

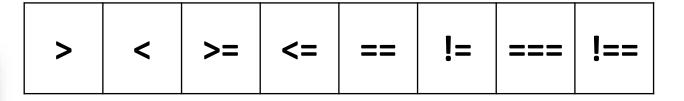
```
let team_A = 5;
3
        let team_B = 3;
 4
        if(team_A > team_B){
5
6
            console.log('Team A - WIN!');
        }else if(team_A < team_B){</pre>
            console.log('Team B - WIN!');
8
9
        }else {
            console.log('Draw...');
10
11
```

Если задача предполагает более чем два вариант развития событий то мы можем задействовать конструкцию if else if else...

## 3. Операторы сравнения

#### Операторы сравнения

(или откуда берётся boolean)



**Операторы сравнения** знакомы нам со школы. В **JavaScript** они решают те же задачи, проводят сравнения двух операндов и возвращают **true/false**.

#### Операторы сравнения

(или откуда берётся boolean)

```
2
3     let a = "Ivan";
4     let b = "Iven";
5
6     console.log(a > b);
7     console.log(a < b);
8     console.log(a == b);
9</pre>
```

Сравнение строк в **JavaScript** осуществляется посимвольно. Выполняется сравнение кодов символов.

### 4. Логические операторы

#### Логические операторы

(когда нужны «сложные» условия)

&& || !

**Логические операторы** позволяют скомбинировать несколько условий в одно. Например когда необходимо проверить, чтобы сумма была больше нуля, но меньше имеющегося баланса, при этом оба условия должны выполнятся одновременно.

#### Логические операторы

```
let age = +prompt('Введите возвраст (лет)');
3
       let height = +prompt('Введите рост (в см.)');
4
5
       if( (age < 12) && (height < 145) ){
6
           console.log('Нужно автокресло!');
       }else{
8
           console.log('Автокресло не нужно.');
9
10
```

Оператор **if/else** позволяет проверять несколько условий сразу, для их группировки существуют логические операторы **И** (&&) и **ИЛИ** (||).

#### Таблица истинности логических операторов

&&	False	True
False	False	False
True	False	True

П	False	True
False	False	True
True	True	True

Возможные варианты для работы оператора && (оператор И) и оператора || (оператор ИЛИ). Первому необходимо чтобы и левое и правое условие давали true и тогда результат будет true, второму достаточно, чтобы хоть один из операндов был true для итогового позитивного результата. Во всех остальных случаях операторы возвращают false.

## 5. Тернарный оператор

#### Тернарный оператор

```
let income = +prompt('Введите ваш доход (в грн.)');

let taxRate = income > 17 ? 19.5 : 0;

let taxSum = income * (taxRate / 100);

console.log(`Cумма налога ${taxSum} грн.`);
```

**Тернарный оператор** — сокращенная форма записи условного оператора. Особенно удобен ветвления между действиями которые можно задать в одно выражение.

### 6. Немного практики #1

**Задача:** У пользователя есть сумма на счету в банке. Он хочет выполнить перевод части денег, а банк хочет взять за это комиссию. Необходимо рассчитать сумму комиссии при тарифе в 1% (минимум 10 гривен, максимум 250 гривен).

### 7. Немного практики #2

Задача: Мы знаем день, месяц и год рождения человека. Мы также знаем сегодняшний день месяц и год, необходимо рассчитать сколько человеку полных лет на сегодняшний день.

## Домашнее задание

#### Домашнее задание #А1

**Задача:** Разработать скрипт который рассчитывает **индекс массы тела** пользователя.

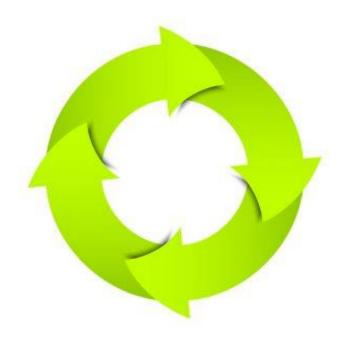
**Задача:** в зависимости от полученного результата выводить сообщение о том, что означает индекс (текстом, в соответствии с Wikipedia). *«Многоэтажных» іf поможет в этом*.

#### Домашнее задание #А2

**Задача**: Необходимо написать скрипт который **по номеру года** определяет **високосный год или нет**. Подсказка: оператор % может вам помочь (но он **не** берёт процент от числа).

## На следующем занятии

### JS: циклы



**Повторение** выполнения фрагментов **кода** и задачи решаемые циклами