



JavaScript: warunki i rozgałęzienia

dimon.work/kurs.html

1. Тип **Boolean**

Typ **boolean**

```
1
2   let a = true;
3
4   let b = false;
5
6   // true boolean
7   console.log(a, typeof(a));
8
9   // false boolean
10  console.log(b, typeof(b));
11
```

Zmienna typu **boolean** zawiera tylko jedną z dwóch opcji wartości: **true** (**prawda**) lub **false** (**fałsz**). Pomimo swojej prostoty, typ ten jest używany w wielu konstrukcjach językowych.

2. Operator **if/else**

Operator warunkowy if/else

```
1
2   let age = prompt('Enter your age');
3       age = parseInt(age);
4
5   if(age >= 18) {
6       console.log('Welcome!');
7   }else {
8       console.log('Sorry...');
9   }
10
```

Operator **if-else** pozwala na określenie warunków, w zależności od poprawności/niepoprawności których zostanie wykonana pierwsza (**if**) lub druga (**else**) gałąź kodu (możliwe jest utworzenie „wielopiętrowego” if przy użyciu konstrukcji **...else if...**).

Operator warunkowy if/else

```
1
2   let balance = 10_000;
3
4   console.log(`Ваш баланс: ${balance} грн.`);
5
6   let sum = +prompt('Сумма для снятия');
7
8   if(sum <= balance){
9       balance = balance - sum;
10      console.log(`Выдано ${sum} грн.`);
11  }else {
12      console.log('Недостаточно средств!');
13  }
14
15  console.log(`Ваш баланс: ${balance} грн.`);
16
```

Nawiasy klamrowe w operatorze **if-else** umożliwiają określenie kilku wierszy kodu, które zostaną wykonane, jeśli warunek jest prawdziwy/fałszywy. Po wykonaniu kodu instrukcji warunkowej program przechodzi do dalszego wykonywania kodu programu.

«Wielopiętrowy» if/else

```
1
2   let team_A = 5;
3   let team_B = 3;
4
5   if(team_A > team_B){
6       console.log('Team A - WIN!');
7   }else if(team_A < team_B){
8       console.log('Team B - WIN!');
9   }else {
10      console.log('Draw...');
11  }
12
```

Jeśli zadanie obejmuje więcej niż dwa warianty zdarzeń, możemy użyć instrukcji **if else if else if else....**

3. Operatory porównania

Operatory porównania

(lub skąd pochodzi boolean)

>	<	>=	<=	==	!=
---	---	----	----	----	----

Operatory porównania są nam znane ze szkoły. W **JavaScript** rozwiązują one ten sam problem, porównując dwa operandy i zwracając wartość **prawda/fałsz**.

```
2
3   var a = 8;
4   var b = 7;
5
6   var c = a >= b;
7
8   console.log(c, typeof(c));
9
```

Operatory porównania

(lub skąd pochodzi 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
```

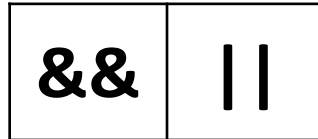
>	<	>=	<=	==	!=
---	---	----	----	----	----

Porównywanie ciągów znaków w **JavaScript** odbywa się znak po znaku. Porównywane są kody znaków.

4. Operatory logiczne

Operatory logiczne

(kombinowanie warunków)



Operatory logiczne umożliwiają łączenie kilku warunków w jeden. Na przykład, gdy konieczne jest sprawdzenie, czy suma jest większa od zera, ale mniejsza od dostępnego salda, oba warunki muszą być spełnione w tym samym czasie.

Operatory logiczne

```
1
2   let age      = +prompt('Введите возраст (лет)');
3   let height   = +prompt('Введите рост (в см.)');
4
5   if( (age < 12) && (height < 145) ){
6       console.log('Нужно автокресло!');
7   }else{
8       console.log('Автокресло не нужно. ');
9   }
10
```

Operator **if/else** umożliwia sprawdzenie kilku warunków jednocześnie, a do ich grupowania służą operatory logiczne **AND (&&)** i **OR (||)**.

Tabela prawdy operatorów logicznych

&&	False	True
False	False	False
True	False	True

 	False	True
False	False	True
True	True	True

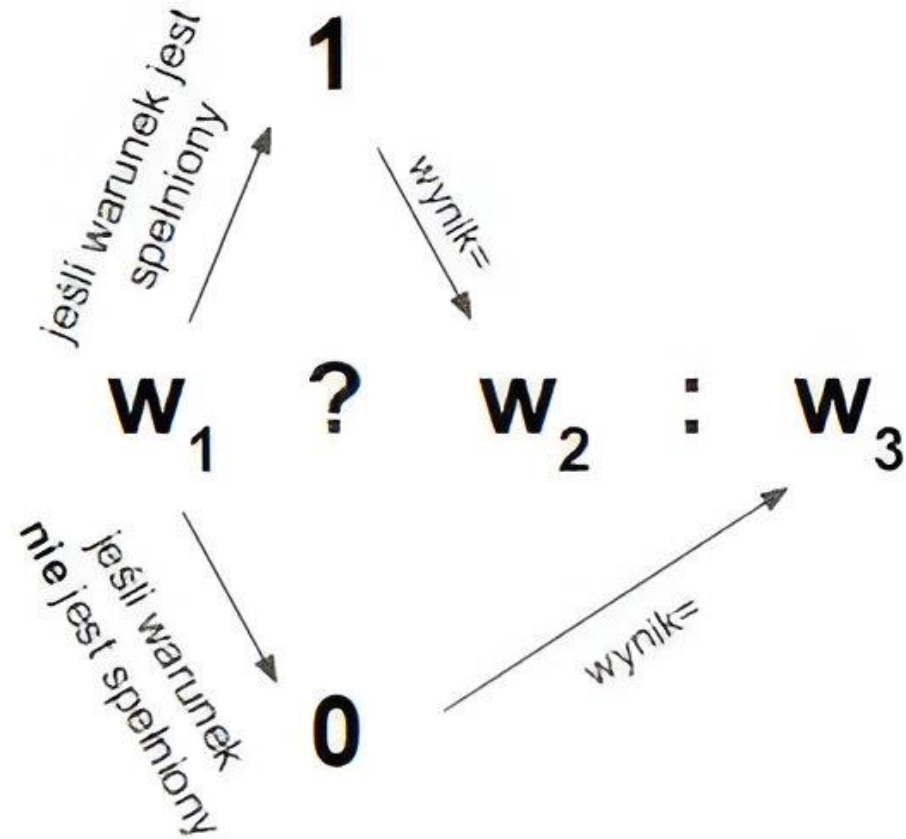
Możliwe warianty działania **operatora &&** (operator AND) i **operatora ||** (operator OR). Pierwszy z nich wymaga, aby zarówno lewy, jak i prawy warunek były prawdziwe, a następnie wynik będzie **true**, drugi wymaga, aby co najmniej jeden z operandów był prawdziwy, aby ostateczny wynik był **true**. We wszystkich innych przypadkach operatory zwracają wartość **false**.

5. Operator warunkowy (Ternarny)

Operator trójskładnikowy

```
1
2   let income = +prompt('Введите ваш доход (в грн.)');
3
4   let taxRate = income > 17 ? 19.5 : 0;
5
6   let taxSum = income * (taxRate / 100);
7
8   console.log(`Сумма налога ${taxSum} грн.`);
9
```

Operator trójskładnikowy jest skróconą formą zapisu operatora warunkowego. Szczególnie wygodne jest rozgałęzianie działań, które można określić w jednym wyrażeniu.



W_1 - warunek do sprawdzenia

6. Trochę praktyki #1

Zadanie: Użytkownik ma pewną sumę pieniędzy na swoim koncie bankowym. Chce przelać część pieniędzy, a bank chce pobrać opłatę. Konieczne jest obliczenie kwoty prowizji według stawki 1% (minimum 10 PLN, maksimum 250 PLN).

7. Trochę praktyki #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

Zadanie domowe #A1

Zadanie: Opracuj skrypt obliczający wskaźnik masy ciała użytkownika.

Zadanie: w zależności od wyniku, wyświetla komunikat o tym, co oznacza indeks (w tekście, zgodnie z Wikipedią). Pomoże w tym „wielopiętrowy” if.

Zadanie domowe #A2

Zadanie: Konieczne jest napisanie skryptu, który określa rok **przestępny** lub nie **na podstawie numeru roku**.

Wskazówka: operator % może ci pomóc (ale nie pobiera procentu liczby)

Na następnych zajęciach

JS: cykle



Powtarzanie wykonywania fragmentów **kodu** i zadania rozwiązywane przez pętle