

Java

wprowadzenie
operatory

Czego się dowiesz?

- Jakie operatory są dostępne w Javie?
- Jak z nich korzystać?
- Jaka jest kolejność wykonywania operatorów?

Czym jest operator?

Operator służy do wykonywania operacji typu dodawanie, porównanie, przypisanie.

W Javie wyróżniamy kilka grup operatorów:

- arytmetyczne
- relacji
- logiczne
- bitowe*

* więcej w materiałach dodatkowych

Operatory arytmetyczne

Używane są w wyrażeniach matematycznych.

- $+$, $-$, $*$, $/$ // dodawanie, odejmowanie, mnożenie, dzielenie
- $\%$ // reszta z dzielenia
- $++$, $--$ // inkrementacja oraz dekrementacja
- $+=$, $-=$, $*=$, $/=$ // przypisanie z dodawaniem, odejmowaniem,
// mnożeniem, dzieleniem
- $\%=$ //przypisanie z resztą z dzielenia

Podstawowe operatory

$2 + 2 + 4 + 12$ // suma liczb, to 20
 $4 - 100$ // różnica liczb, to -96
 $15 * 2 * 1$ // iloczyn liczb, to 30
 $20 / 2 / 2$ // iloraz liczb, to 5

Wszystkie operacje zostały wykonane od lewej do prawej, tzn. że np. w wyrażeniu $2 + 2 + 4 + 12$, wykonały się pośrednie operacje:

$2 + 2$ // 4
 $4 + 4$ // 8
 $8 + 12$ // 20

Uwaga: dzielenie dwóch liczb całkowitych zwraca liczbę całkowitą, a więc $5/2 = 2$.

Operator reszty z dzielenia

Reszta z dzielenia (modulo, %) to operator dwuargumentowy. Oblicza resztę z dzielenia argumentu pierwszego przez argument drugi.

np.

$5\%2$ // to 1, dlaczego? $5/2$, to 2 całości. $5-(2*2)$, to 1.

$127\%128$ // to 127, dlaczego? $127/128$, to 0 całości. $127-(0*128)$, to 127

$-15\%5$ // to 0, dlaczego? $-15/5$, to -3 całości. $-15-(-3*5)$, to 0

$1000\%1234$ // to 1000, dlaczego? $1000/1234$, to 0 całości. $1000-(0*1234)$, to 1000

Gdy $x\%y$, to $x/y = z$. Resztą z dzielenia będzie wynik operacji $x - (z*y)$.

(In/De)krementacja

Inkrementacja (++), to zwiększenie liczby o 1.

Dekrementacja(--), to zmniejszenie liczby o 1.

np.

```
int i = 10;
```

`i++`, jest tożsame z `i = i + 1`; Czyli wynikiem będzie `i` równe 11.

```
int j = 28;
```

`j--`, jest tożsame z `j = j - 1`; Czyli wynikiem będzie `j` równe 27.

* więcej w materiałach dodatkowych. Różnice pomiędzy post i pre inkrementacją/dekrementacją

Operatory z przypisaniem

Operatory: +=, -=, *=, /=, %=, poza obliczeniem wartości przypisują ją do zmiennej.

np.

```
int i = 92;
```

```
i+=1, tożsame z i++, oraz i=i+1;    // czyli zwiększenie i przypisanie
```

```
i+=100, tożsame z i = i+100;      // wskazanej wartości
```

analogicznie

```
i*=4, tożsame z i=i*4;
```

```
i%=123, tożsame z i =i%123;
```


Operatory relacji

Ta grupa operatorów określa związek pomiędzy dwoma operandami.

- < // mniejszy
- > // większy
- == // równy
- != // różny
- <= // mniejszy lub równy
- >= // większy lub równy

Operatory relacji, zwracają jedną z wartości boolean (true lub false).

Operatory relacji

Przykłady:

234 < 123	// false
123 < 234	// true
234 > 123	// true
5 <= 7	// true
5 <= 5	// true
9 >= 10	// false
8 == 9	// false
9 == 9	// true
6 != 6	// false

Przykłady:

'c' == 'z'	// false
'c' == 'c'	// true
true == true	// true
true == false	// false
52.1 != 52	// true
52.1 >= 52	// true
'a' > 'z'	// false
'k' <= 'c'	// false

Operatory logiczne

Operandami operatorów logicznych mogą być tylko wartości typu boolean. Najczęściej wykorzystujemy je do łączenia kilku warunków.

- `&`, `|`, `^` // koniunkcja, alternatywa, różnica symetryczna
- `&&`, `||` // koniunkcja oraz alternatywa ze skracaniem
- `!` // negacja
- `&=`, `|=`, `^=` // koniunkcja, alternatywa i różnica z przypisaniem
- `==`, `!=` // równość, różność
- `?:` // operator trójargumentowy

Koniunkcja

Koniunkcja (&) - inaczej iloczyn logiczny, to operator dwuargumentowy. Wyrażenie jest prawdziwe, gdy oba argumenty są prawdziwe.

true & true // true

true & false // false

false & true // false

false & false // false

2 < 3 & 6 != 1 // true

50%5 ==0 & 'z' < 'a' // false

7 == 8 & 9 !=8 // false

-12 < 12 & 23.2 >= 23.1 // false

Koniunkcja ze skracaniem

Koniunkcja "ze skracaniem" (**&&**), różni się od standardowej koniunkcji sposobem działania. Wiemy, że koniunkcja wymaga dwóch argumentów prawdziwych, aby całość była prawdziwa.

Koniunkcja ze skracaniem kończy sprawdzanie już po pierwszym argumencie, gdy okaże się, że jest on fałszywy. Skoro choć jeden jest fałszywy, to już wiemy, że całość jest fałszywa, więc nie ma potrzeby sprawdzania drugiego argumentu.

Używamy więc koniunkcji ze skracaniem w celu przyśpieszenia sprawdzania warunków.

Alternatywa

Alternatywa (|) - inaczej suma logiczna, to operator dwuargumentowy. Wyrażenie jest prawdziwe, gdy przynajmniej jeden argument jest prawdziwy.

true | true // true

true | false // true

false | true // true

false | false // false

2 < 3 | 6 != 1 // true

50%5 ==0 | 'z' < 'a' // true

7 == 8 | 9 !=8 // true

-12 < 12 | 23.2 >= 23.1 // false

Alternatywa ze skracaniem

Alternatywa "ze skracaniem" (\parallel), różni się od standardowej alternatywy sposobem działania. Wiemy, że alternatywa jest prawdziwa, gdy choć jeden argument jest prawdziwy.

Alternatywa ze skracaniem kończy sprawdzanie już po pierwszym argumencie, gdy okaże się, że jest on prawdziwy. Skoro choć jeden jest prawdziwy, to już wiemy, że całość jest prawdziwa, więc nie ma potrzeby sprawdzania drugiego argumentu.

Używamy więc alternatywy ze skracaniem w celu przyśpieszenia sprawdzania warunków.

Różnica symetryczna

Różnica symetryczna (^) - inaczej alternatywa rozłączna, to operator dwuargumentowy. Wyrażenie jest prawdziwe, gdy argumenty są różne od siebie.

true ^ true // false

true ^ false // true

false ^ true // true

false ^ false // false

2 < 3 ^ 6 != 1 // false

50%5 == 0 ^ 'z' < 'a' // true

7 == 8 ^ 9 != 8 // true

-12 < 12 ^ 23.2 >= 23.1 // false

Negacja

Negacja (!) - inaczej zaprzeczenie, to operator jednoargumentowy. Wyrażenie jest prawdziwe, gdy argument jest fałszywy. Wyrażenie jest fałszywe, gdy argument jest prawdziwy.

`! true // false`

`! false // true`

`!(2==2) // false`

`!('c' == 'z') // true`

Operator trójargumentowy

Operator trójargumentowy (?) - jest skróconą formą instrukcji warunkowej if-then-else, którą poznamy w kolejnych lekcjach.

Definicja:

wyrażenie1 ? wyrażenie2 : wyrażenie3

wyrażenie1 - jest wyrażeniem logicznym, zwraca true/false.

wyrażenie2 - jest zwracane gdy wyrażenie1 jest true

wyrażenie3 - jest zwracane gdy wyrażenie1 jest false

Operator trójargumentowy

Przykład 1.

```
int wiek = 14;
```

```
boolean isAdult = (wiek >= 18) ? true : false;
```

Przykład 2.

```
int smaller = 2 < 5 ? 2 : 5;
```

Podsumowanie

W Javie wyodrębniamy kilka grup operatorów.

Operatory arytmetyczne służą do wykonywania operacji matematycznych na wartościach. Operatory relacji pozwalają porównywać wartości, a operatory logiczne łączyć relacje. Operatory bitowe pozostawiam jako dodatek.

Operatory są jednym z najczęściej stosowanych elementów języka Java.

Zadanie 1

Napisz program, który przy pomocy typów prostych przelicza stopnie Fahrenheita TF na stopnie Celsjusza TC. Temperatuty te są powiązane następującą zależnością:
 $TC = (TF - 32) * 5/9.$

Zadanie 2*

Napisz program, który obliczy obwód koła dla promienia równego 7.

Wzór na obwód koła: $2 \cdot \pi \cdot \text{promień}$.

Liczba $\pi = 3.14$;

* wykonaj samodzielnie

Zadanie 3*

Napisz program, w którym przy pomocy zmiennych opiszesz swoją ostatnią podróż.

Wykorzystaj następujące zmienne:

- kilometers
- minutes
- days
- accountBalance

* wykonaj samodzielnie

Zadanie 4*

Utwórz trzy zmienne „całkowitoliczbowe”. Dwie pierwsze zainicjalizuj dowolnymi liczbami, a do trzeciej przypisz wynik dodawania dwóch pierwszych. Następnie dodaj liczby do siebie i wyświetl wynik w konsoli.

* wykonaj samodzielnie

Zadanie 5*

Dane są wartości `int a = 7`, `int b = 9`, `int c = 12`, `int triangleRegion`. Policz obwód trójkąta. Wyświetl wynik w konsoli.

* wykonaj samodzielnie

Zadanie 6*

Napisz program, który przy pomocy logicznego operatora trój-argumentowego, wyznacza wartość absolutną liczby x .

* wykonaj samodzielnie

Zadanie 7*

Napisz program, który przy użyciu operatora modulo, wylicza resztę z dzielenia wartości 47 przez 10 oraz 35.5 przez 6.

* wykonaj samodzielnie

Materiały dodatkowe

Zamiana liczb dziesiętnych na binarne

<https://www.youtube.com/watch?v=rsxT4FfRBaM>

Operatory bitowe

<https://www.baeldung.com/java-bitwise-operators>

Różnica pomiędzy post i pre inkrementacją/dekrementacją

<https://www.baeldung.com/java-unary-operators>

Kolejność wykonywania operatorów

<https://introcs.cs.princeton.edu/java/11precedence/>