

ОПЕРАТОРЫ

Арифметические операторы

+ сложение

- вычитание

* умножение

/ деление

% взятие остатка от деления

Оператор сложения + используется для:

1. сложения чисел
2. конкатенации строк
3. приведения значения к типу `int`, если это переменная типа `byte`, `short` или `char`

Арифметические операции

1. При делении целых чисел, остаток отбрасывается. Чтобы результат представлял число с плавающей точкой, один из операндов также должен быть числом с плавающей точкой.
2. Если один операнд имеет тип `long`, то тип всего выражения повышается до `long`.
3. Если один операнд имеет тип `float`, то тип всего выражения повышается до `float`.
4. Если любой из операндов имеет тип `double`, типом результата будет `double`.

Операторы присваивания

= `int a = 12` а присвоили значение 12

+= `a += 12` краткая форма от `a = a + 12`

-= `a -= 12` краткая форма от `a = a - 12`

***=** `a *= 12` краткая форма от `a = a * 12`

/= `a /= 12` краткая форма от `a = a / 12`

%= `a %= 12` краткая форма от `a = a % 12`

Инкремент и декремент

- `i++` инкремент (постфиксная форма) сначала возвращает значение, потом увеличивает
- `++i` инкремент (префиксная форма) сначала увеличивает, а потом возвращает значение
- `i--` декремент (постфиксная форма) сначала возвращает значение, потом уменьшает
- `--i` декремент (префиксная форма) сначала уменьшает, а потом возвращает значение

В операциях сравнения сравниваются два операнда, и возвращается значение типа `boolean`: `true`, если выражение верно, `false`, если выражение неверно.

Операторы сравнения

`>` больше

`<` меньше

`==` равно

`>=` больше или равно

`<=` меньше или равно

`!=` не равно

Возвращают значение типа boolean

Логические операторы

&&	<code>a && b</code>	вернет true, если a и b - true (если a - false, b не вычисляется)
-------------------	-----------------------------	--

 	<code>a b</code>	вернет true, если a или b - true (если a - true, b не вычисляется)
-----------	---------------------	---

&	<code>a & b</code>	вернет true, если a и b - true (b оценивается в любом случае)
--------------	------------------------	--

 	<code>a b</code>	вернет true, если a или b - true (b оценивается в любом случае)
----------	--------------------	--

Возвращают значение типа `boolean`

Логические операторы

! !a меняет значение a на противоположное

^ a^b вернет true, если либо a, либо b true
(но не одновременно)

Тернарный оператор используется для оценки выражений типа boolean

Цель тернарного оператора заключается в том, чтобы решить, какое значение должно быть присвоено переменной.

переменная x = выражение ? выражение1 : выражение2;

Если выражение равно true, то вычисляется выражение1 и его результат становится результатом выполнения всего оператора.

В противном случае вычисляется выражение2. Выражение1 и выражение2 должны возвращать значение одинакового (или совместимого) типа.

Примеры с тернарным оператором

```
1 boolean isActive = true;  
2 int code = isActive ? 1 : 0; // 1  
3  
4 int n = 9;  
5 int res = n % 2 == 0 ? n / 2 : n * 2; // 18
```