

Тип данных boolean и логические операторы — введение в Java 005



Boolean в Java — это логический тип данных. Переменная типа `boolean` может принимать всего два значения — это правда или ложь — **true & false**. Эти два значения обозначаются в других языках и часто выдаются на экран как **1** и **0**, но всё же не равны этим значениям: например, выражение `boolean b = 0;` приведёт к ошибке при компиляции программы. Но мы можем сравнить переменные или выполнить логическую операцию с типом данных `boolean`:

```
int a = 1, b = 2;
boolean bool = a <= b;
```

Говоря о булевых или логических типах данных, нам придётся освежить свои воспоминания о булевой алгебре и возможных логических операциях.

Таблица истинности

a	b	a & b	a b	a ^ b	! a
false	false	false	false	false	true
false	true	false	true	true	true
true	false	false	true	true	false
true	true	true	true	false	false

Давайте представим себе пример из жизни: мы ищем на сайте все статьи, которые мы опубликовали и комментировали. Должны быть два совпадения — это вариант **a & b**. Или мы ищем все статьи, в которых есть упоминание слов "алгебра" или "математика" — это **a | b**. А отыскать все статьи, которые написаны **не** нами, можно, применив логический оператор **!a**.

И так далее, и так далее. Ключевые слова для повторения школьного материала: таблица истинности, булева алгебра, теория множеств, конъюнкция, дизъюнкция.

С помощью таблицы с результатами логических операций можно перепроверить работу Java:

```
public class NewBoolean {
    public static void main(String[] args) {
        boolean a, b, c;
        a = true;
        b = false;
        c = a & b;
        System.out.println(c); // returns false because only one of the two required values is true

        int d = 1, f = 2;
        boolean bool = d <= f;

        int i = 10;
        int j = 9;
        System.out.println(i > j); // returns true, because 10 is higher than 9

        System.out.println(10 == 15); // returns false, because 10 is not equal to 15

        int x = 10;
        System.out.println(x == 10); // returns true, because the value of x is equal to 10
    }
}
```

Надо понимать, что значение вашего чекбокса (галочки) на сайте Facebook при регистрации — "с правилами ознакомлен и согласен" — и есть значение булевой переменной в программе.

Логические операторы, которые поддерживаются Java

Логический операторы	Значение
==	Проверка на соответствие (допустим, что a равен b)
!=	Не равно (если a не равно b , то)
!	Отрицание, логическое не
&	Логическое И, and
	Логическое или, or
^	Исключительное или, XOR
&&	Укороченный &
\	Укороченный или

Амперсанд — это название знака **&**.

Мы ещё раз рассмотрим данные операторы в [12-ом уроке](#). Пока мы должны понимать, что с арифметическими операторами всё немного сложнее, чем хотелось бы.

Дополнительные материалы

- 1. [Булева алгебра — самое важное](#)
- 2. [Булева алгебра](#)
- 3. [Теория множеств](#)

4. [Очень неплохая статья](#)

5. [simulator](#) — симулятор логических выражений в электротехнике. Очень помог мне в своё время вспомнить булеву алгебру.

Домашнее задание

- Что выдаст программа, если запросить значения a, b, c, d, e, f?
- Посчитайте сначала в уме и проговорите вслух, что делает каждая строчка.

```
boolean a = (7+8)*5 > 7+8*5;  
boolean b = (7+8)*4 != 7+4*5;  
boolean c = 3+4 > 9+1 & 16-5 > 3*4;  
boolean e = 16/2 < 6+2 | 4+5 <= 4*5;  
boolean f = !(3*4 < 7+8);
```

Напишите программу и проверьте свои результаты.

Сравните строчку вывода со своей:

```
System.out.println(a + "\n" + b + "\n" + c + "\n" + d + "\n" + e + "\n" + f);
```

-
- Повторения String: Найдите метод из официальной библиотеки и покажите его работу. По возможности используйте printf.

Булева алгебра

Булева алгебра, ударение на первый слог. Булева.

Принципиально основы булевой алгебры не должны были давать в школе. Программисты учат её в институте.

Давайте я попробую на пальцах рассказать основы и то, что нам понадобится на минимальном уровне.

Дизъюнкция

Когда мама **или** папа дают мне деньги на карманные расходы, то я бегу и покупаю себе мороженное.

Знакомая ситуация, деньги можно получить в трёх случаях из четырёх. В одном случае же деньги может дать и мама, и папа, тогда и друга можно угостить мороженым.

Это дизъюнкция.

Дизъюнкция - логическое сложение, логическое **или**, включающее или, просто "или"(англ. **OR**; нем. ODER) В Java операторы "|" и "||"

```
boolean a = false, b = true, c;  
c = a | b;
```

Пример в технике; дублирование выключателя или кнопки, дверной звонок и звонок у калитки вызывают одну и ту же реакцию - включается мелодия звонка.

Конъюнкция

Конъюнкция - логическое **и**, логическое умножение, просто **"И"**, **"AND"**, **"&"**.

В Java оператор **"&"** и **"&&"**.

```
boolean a = false, b = true, c;  
c = a & b;
```

Если светит солнце **"И"** у меня выходной, то я иду купаться на озеро.

Пример из жизни. Ядерный чемоданчик могут активировать только два офицера одновременно. По отдельности чемодан остаётся неактивным.

Антиваленц

"XOR", эксклюзивное или, **"EOR"**, **"EXOR"**. В Java оператор **"^"**.

```
boolean a = false, b = true, c;  
c = a ^ b;
```

Только на горных лыжах в Австрии или на лошадях у бабушки в деревне я забываю про свою работу.

Или ты сядишься за математику или я расскажу всё отцу.

ИЛИ - ИЛИ. Исключительное или.

Лампочка в больнице может работать от городского электричества или от дизельного генератора в подвале. Но не от двух источников одновременно.

Отрицание

Negation. NOT. В Java оператор **"!"**.

```
boolean a = false, b;  
b = !a;
```

Давайте представим огромный станок по продольной распилке леса. В цеху есть две кнопки. Зелёная и красная. При включении зелёной пила должна работать. При нажатии на красную пила должна остановиться.

Домашнее задание по булевой алгебре, не по java:

Задача. Петя, Вася и Маша остались дома одни. Кто-то из них ел варенье. На вопрос мамы, кто это сделал, они сказали:

1. Петя: "Я не ел. Маша тоже не ела."
2. Вася: "Маша действительно не ела. Это сделал Петя"
3. Маша: "Вася врет. Это он съел."

Выясните, кто ел варенье, если известно, что двое из них оба раза сказали правду, а третий один раз соврал, а один раз сказал правду.

[Ответ к задаче для собственной сверки](#)