

Programmierkurs Tag 2 - Aufgaben

Aufgabe T2.1*

Welchen booleschen Wert ergeben die folgenden Ausdrücke:
(hier ist keine Programmierung gefragt - einfach mit Stift und Papier beantworten)

- (a) `true & false`
- (b) `true & true`
- (c) `false & false`
- (d) `false | false`
- (e) `false | true`
- (f) `true | true`
- (g) `false ^ false`
- (h) `true ^ false`
- (i) `true ^ true`
- (j) `true & !true`
- (k) `(true | false) & (false | (true & true & true | false)) | !false`
- (l) `1 == 2 && (3 != 4) || (1 < 2)`
- (m) `(7 < 5) ^ (2 == (9 / 4))`

Aufgabe T2.2*

Welche der folgenden Aussagen sind richtig (hier ist keine Programmierung gefragt - einfach mit Stift und Papier beantworten)

- (a) Wenn `(a || b)` true ist, ist mindestens eine der Variablen true.
- (b) Wenn `(a && b)` true ist, sind beide Variablen true.
- (c) Wenn `(a && b)` false ist, sind beide Variablen false.
- (d) Wenn `(a || b)` false ist, sind beide Variablen false.
- (e) Wenn `!(a && b)` false ist, ist keine der beiden Variablen true.
- (f) Wenn `!(a || b)` false ist, sind beide Variablen true.
- (g) `(a && b) || !(a && b)` ist immer true.

Aufgabe T2.3*

Berechne $1000\ 000\ 000\ 000 + 0.000\ 000\ 000\ 001$ mit Java. Versuche es einmal mit `double` und einmal mit `float` als Variablentyp. Wieso kommt dieses Ergebnis zustande? (Hinweis: Erinnere dich, wie du sicherstellen kannst, dass ein Wert einen bestimmten Datentyp hat.)

Aufgabe T2.4**

Gib eine Bedingung an, die für einen `int`-Wert `x` wahr wird, wenn dieser über 25 liegt und eine ungerade Zahl ist. (Lösungshinweis: Zur Prüfung, ob es sich um eine ungerade Zahl handelt, ist der Modulo-Operator `%` hilfreich.)

Aufgabe T2.5**

Ein Schaltjahr ist ein Jahr, dessen Jahreszahl restlos durch vier teilbar ist. Eine Ausnahme bilden die vollen Jahrhundertjahre, deren Jahreszahl durch 100 teilbar ist: Diese sind nur dann ein Schaltjahr, wenn ihre Jahreszahl auch durch 400 teilbar ist. Gib einen booleschen Ausdruck an, der für einen Integer-Wert `jahr` genau dann wahr wird, wenn es sich bei dem in `jahr` gespeicherten Wert um die Jahreszahl eines Schaltjahrs handelt.

Aufgabe T2.6*

Ändere das folgende Programm so, dass es Frauen und Männer bei der Begrüßung unterscheidet:

```
boolean male = true;
String name = "Peter";

System.out.println("Guten Tag, Herr" + name);
```

Aufgabe T2.7**

Mit Hilfe der p-q-Formel lassen sich quadratische Gleichungen der Form $x^2 + p \cdot x + q = 0$ lösen. Es gilt: $x_{1/2} = -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q}$. Eine reelle Lösung existiert nur dann, wenn die Diskriminante (der Ausdruck unter der Wurzel) nicht negativ ist.

Schreibe ein Programm, das das Ergebnis/die Ergebnisse für gegebene `p` und `q` berechnet und auf der Konsole ausgibt. Wenn es kein Ergebnis im Bereich der reellen Zahlen gibt, soll eine entsprechende Meldung auf der Konsole ausgegeben werden.

Um Wurzeln und Potenzen zu berechnen, kannst du die Methoden `sqrt()` und `pow()` der Klasse `Math` verwenden. Beide Methoden nehmen `double`-Werte als Parameter entgegen und geben einen `double`-Wert zurück, den man zum Beispiel in einer Variablen speichern kann.

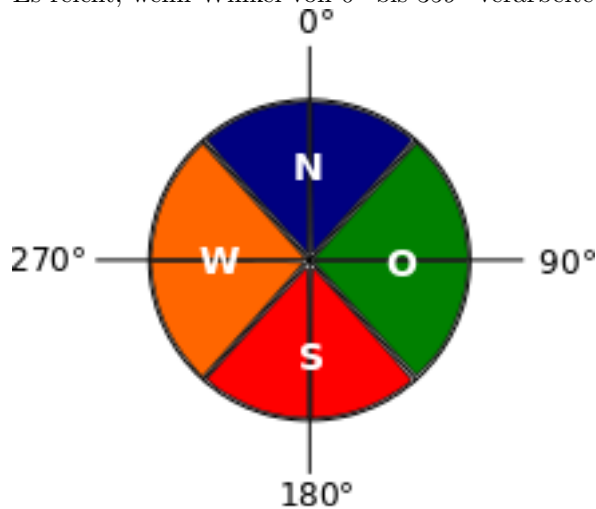
Bsp.:

- `Math.sqrt(4.0)` gibt den Wert 2.0 zurück
- `Math.pow(2.0, 3.0)` gibt den Wert 8.0 zurück

Aufgabe T2.8***

Schreibe ein Programm, das abhängig von einem Winkel (positiver Integer-Wert) eine der vier Himmelsrichtung Norden, Osten, Süden oder Westen ausgibt. Norden liegt dabei auf 0° .

Es reicht, wenn Winkel von 0° bis 359° verarbeitet werden können.



Aufgabe T2.9***

Schreibe ein Programm, das das gleiche Ziel ganz ohne if-Abfragen nur mit switch/case erreicht.

Lösungshinweise: Normiere den Winkelbereich auf die Anzahl der Himmelsrichtungen. Jede Himmelsrichtung wird dann durch einen von vier Integer-Werten repräsentiert, der als Switch-Variable dient. Dabei ist die ganzzahlige Division hilfreich.

Du solltest dir außerdem überlegen, was man gegen die „unschöne“ Aufteilung der Winkelbereiche tun kann. Jeder dieser Bereiche ist zwar 90° groß, aber Norden liegt zum Beispiel zwischen 315° und 45° , Süden zwischen 135° und 225° . Das ist zum Normieren etwas unpraktisch.

Vier case statements sollten reichen.