# Geometrienachhilfe zu Dreiecken:

1. gleichseitiges Dreieck:  
   alle drei Seiten sind gleich groß:

a b

c

mathematisch: a = b = c  
in Java: a==b && b==c

Transitivgesetz (mathematisch geschrieben):

a = b und b = c => a = c

1. gleichschenklig:  
   mindestens zwei Seiten sind gleich groß:  
     
     
     
     
     
     
    a b

c  
  
in Java: a==b || a==c || b==c

1. rechtwinklig:  
   Es enthält einen rechten Winkel, d.h. einer mit 90°  
     
     
     
     
     
     
     
     
     
     
     
   mathematisch: a² + b² = c² für eine der drei Verteilungen

in Java: a\*a + b\*b == c\*c || a\*a + c\*c == b\*b || b\*b + c\*c == a\*a

Da die Quadrate für die drei Seiten jeweils dreimal berechnet werden müssen,

bietet es sich an, sie in Hilfsvariablen "aq" u.s.w. zu speichern.

1. ist gültig:

Nicht alle Kombinationen von 3 Seitenlängen ergeben ein gültiges Dreieck.

Beispiel: a=1, b=2, c=10

b

a

c

Ein Dreieck ist gültig, wenn gilt,  
dass die Summe von jeweils 2 Seiten größer ist als die dritte Seite.

Ein Dreieck ist gültig, wenn gilt:

a+b>c && a+c>b && b+c>a

Es macht Sinn,

von einem Dreieck zuerst zu prüfen,

ob es überhaupt gültig ist,

bevor wir nach den Eigenschaften sehen.

Weil wir mehrere Dreiecke brauchen,

damit alle Methoden mindestens einmal auf true- u. einmal auf false abbilden,

müssen wir mehrfach von verschiedenen Dreiecken

die gleichen Eigenschaften testen.

Würden wir dies einzeln machen,

hätten wir eine Codeduplizierung.

Deshalb macht es Sinn,

das Prüfen aller Eigenschaften in eine Methode zu packen

u. für jedes der zu untersuchenden Dreiecke aufzurufen.

Wir könnten Sie "eigenschaften()" nennen.

In ihr würde zuerst die Gültigkeit geprüft.

Nur wenn diese gegeben ist,

werden auch die anderen Eigenschaften geprüft.

Ausgaben könnten z.B. sein:

Für ein gleichschenkliges Dreieck:

Ist das Dreieck gueltig? true

Ist es gleichseitig? false

Ist es gleichschenklig? true

Ist es rechtwinklig? false

Für ein ungültiges Dreieck:

Ist das Dreieck gueltig? false