

Übungsaufgabe JAVA 10 – Generics

1. Wiederholung Grundlagen

Eine Klasse **Position** soll mit zwei double-Werten eine Position im 2-Dimensionalen Raum darstellen. Sie soll diese double-Werte x und y als Parameter im Konstruktor übernehmen. Mittels einer Methode „*move(double x, double y)*“ sollen die Position x und y relativ verändert werden.

Erstellen Sie dann die Klassen **Circle**, **Square** und **Triangle**. Alle drei haben eine Position die der Konstruktor als Parameter erwartet. Der Konstruktor erwartet außerdem:

- einen Radius für Circle
- eine Seitenlänge für Square
- drei Werte für die Seitenlängen des Triangle

Alle drei Klassen sollen in der Lage sein ihre Fläche mit der Methode „*calculateArea()*“ zu berechnen.

2. Wiederholung Vererbung

Erstellen sie eine abstrakte Klasse **Figure**, die gemeinsame Fähigkeiten der Klassen **Circle**, **Square** und **Triangle** vereinigt. Mittels abstrakter Methode soll Figure die Implementierung von „*calculateArea*“ vorschreiben.

Dadurch verbleiben nur die Konstruktoren und die jew. Methode der Flächenberechnung in den ableitenden Klassen. Die Konstruktoren der drei ableitenden Klassen übergeben ihre Position per Aufruf von „parent“ an den Konstruktor von Figure.

3. Ein generischer Container

Entwickeln Sie eine generische Klasse **FigureContainer**, die entweder Circle, Square ODER Triangle Objekte aufnimmt. Intern soll eine ArrayList verwendet werden und folgende Methoden sollen mindestens zur Verfügung stehen:

- `add(T figure)` soll eine weitere ableitende Klasse von Figure aufnehmen
- `get(int id)` Zugriff anhand des Indexes (wie bei ArrayList üblich)
- `size()` Die Methode von ArrayList kann direkt verwendet werden

- `calculateAreaSum()` Soll die Flächen aller Figure-Objekte zusammen zurückgeben

- `randomPositions()` soll alle Flächen mittels „*move(x, y)*“ zufällig neu positionieren
diese Funktionalität könnte die Position-Klasse selbst können...

Hinweis: *Die schnellste Lösung ist eine Klasse, die von ArrayList ableitet und nur die letzten beiden Methoden implementiert. Dies nicht zu tun, kann als Übung sinnvoll sein...*