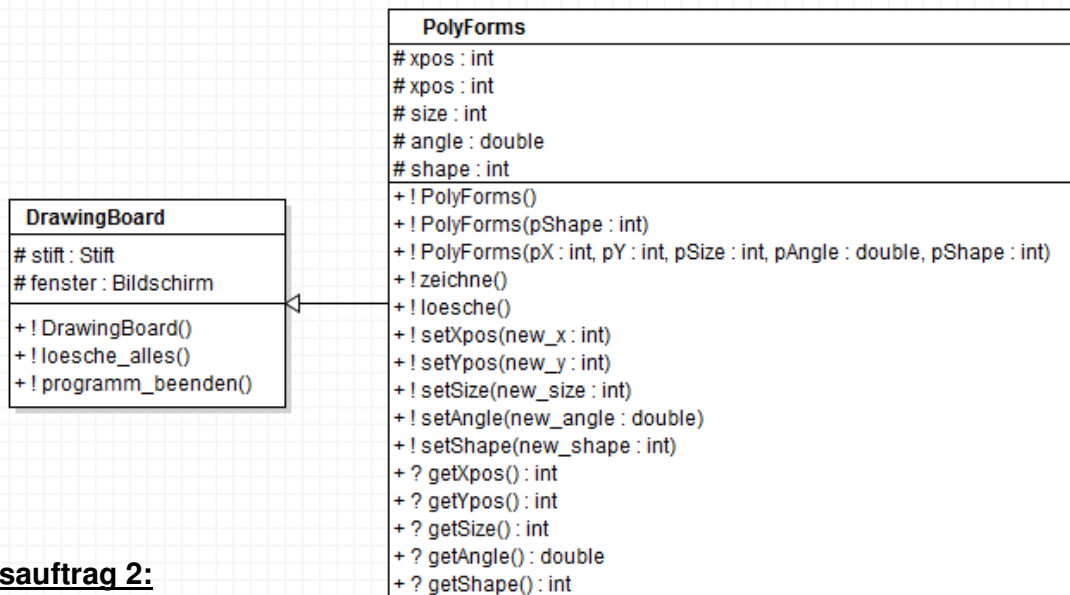


Thema: Objektorientiertes Entwerfen / Modellieren / Implementieren mit Java

Als Ausgangspunkt gegeben ist der *UML*-Entwurf der Klasse **PolyForms** entsprechend der Aufgabenstellung aus Arbeitsblatt 1:



Arbeitsauftrag 2:

Teil a):

Übersetze den oben dargestellten *UML*-Entwurf der Klasse **PolyForms** in **Java-Quellcode** (auf dem Papier!).

Halte dich dabei exakt an die Vorgaben aus dem *UML*-Diagramm, da dies die Basis aller weiteren Ergänzungen ist und nur so vergleichbare Ergebnisse im Kurs erreicht werden können.

Achte auf Vollständigkeit und möglichst korrekte Syntax. Die Methode *zeichne()* besteht zunächst nur aus einem leeren Methodenrumpf, **alle** anderen Methoden müssen vollständig umgesetzt und funktional sein. Achte auch auf die Umsetzung der *Ist*-Beziehung zur Klasse *DrawingBoard*.

Teil b):

Nach der Besprechung / Diskussion mindestens einer Lösung im Plenum ergänze / korrigiere deinen Quellcode nach Bedarf.

Lege dann ein neues Projekt **PolyForms** in **BlueJ** an und erstelle die Klassen **DrawingBoard** und **PolyForms** hierin. Beseitige alle Tipp- und sonstigen Programmierfehler im Quellcode und bringe das Programm in einen übersetzbaren und lauffähigen Zustand.

Teil c):

Erprobe die Funktionalität deines Programms.

Erzeuge hierzu eine Instanz der Klasse **PolyForms** durch Aufruf des gewünschten Konstruktors aus dem Kontextmenü der Klasse im *BlueJ*-Klassendiagramm.

Es sollte ein leeres Bildschirmfenster angezeigt werden. Die öffentlichen Methoden der Klasse können nun über das Kontextmenü der Klasseninstanz / des Objektes (Rechtsklick auf das rote Instanz-Icon unten links im *BlueJ*-Fenster) aufgerufen werden.

Teste verschiedene *set*- und *get*-Methoden sowie auch die aus der Klasse *DrawingBoard* vererbten Methoden.

