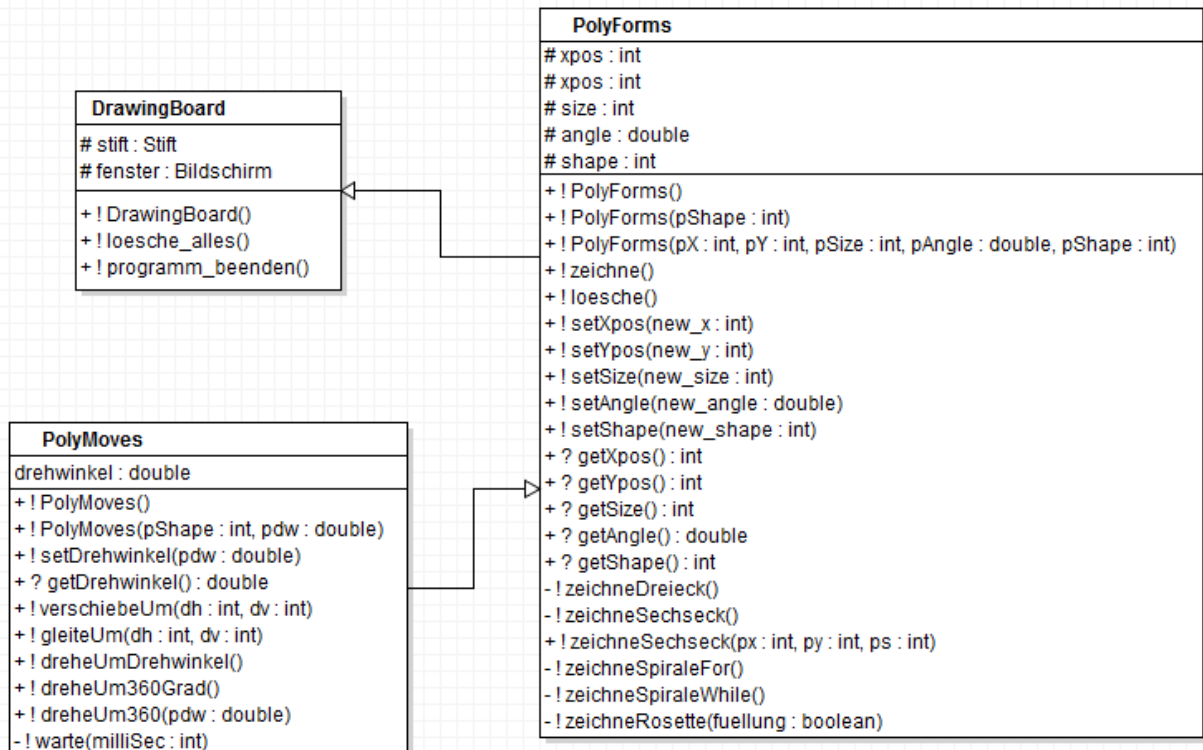


Thema: Objektorientiertes Entwerfen / Modellieren / Implementieren mit Java

Als Ausgangspunkt des Teilprojektes **PolyMoves** sollte eine möglichst vollständige und lauffähige Implementierung der Klassen *DrawingBoard* und *PolyForms* entsprechend den vorhergehenden Aufgabenstellungen vorliegen.

Beide Klassen sind auch in den Kursdateien zum Download verfügbar, sodass in Verbindung mit den gegebenen Musterlösungen hier jeder den gleichen (Wieder-)Aufsetzpunkt im Projekt hat.



Die neue Klasse **PolyMoves** wird als *Spezialisierung* (\Rightarrow Ist-Beziehung / Vererbung) der Klasse *PolyForms* angelegt. Sie erweitert diese um verschiedene Dienste, mit denen die jeweils gezeichnete Figur im Ausgabefenster verschoben oder gedreht werden kann. Die Klassenvariable bzw. das Attribut *drehwinkel* beinhaltet dabei den Wert in Grad, den *ein* Drehschritt in einer Drehbewegung umfasst. Als Standardwert wird dafür -2° vorgegeben.

Hinweis: Der formale Methodenparameter *pdw* dient der Übergabe eines neuen Drehwinkels.

Das obige **UML-Klassendiagramm** beschreibt die Struktur der Klasse **PolyMoves** vollständig.

Arbeitsauftrag 1:

Erzeuge in *BlueJ* die Klasse **PolyMoves** und übersetze hierin die in obigem Klassendiagramm dargestellte Klassenstruktur vollständig in *Java-Quellcode*.

Alle Konstruktoren sowie set-/get-Methoden sollten *vollständig* implementiert sein. Die bereits aus vorgehenden Projekten bekannte *warte(...)*-Methode wird unten noch einmal als Quellcode zur Verfügung gestellt. Alle weiteren Methoden müssen vorhanden sein, weisen aber zunächst nur einen leeren Methodenrumpf auf, der im Verlauf des Teilprojektes vervollständigt werden wird.

Notiere deinen Namen hinter dem *@author*-Tag im Quellcode der Klasse, mache eine Bildschirmkopie des *BlueJ*-Editorfensters mit dem gesamten Quellcode der Klasse und stelle diese in der Cloud ein (alternativ kopiere den gesamten Quellcode per copy&paste in einen Texteditor und stelle eine entsprechende .txt- oder .pdf-Datei ein, keine .java-Datei bitte).

Java-Quellcode der warte(...)-Methode:

```
private void warte(int milliSec) {  
    try {  
        Thread.sleep(milliSec);  
    }  
    catch (InterruptedException e) { /* Fehler ignorieren */ }  
}
```