

Übung: 4 – State & Strategy Pattern

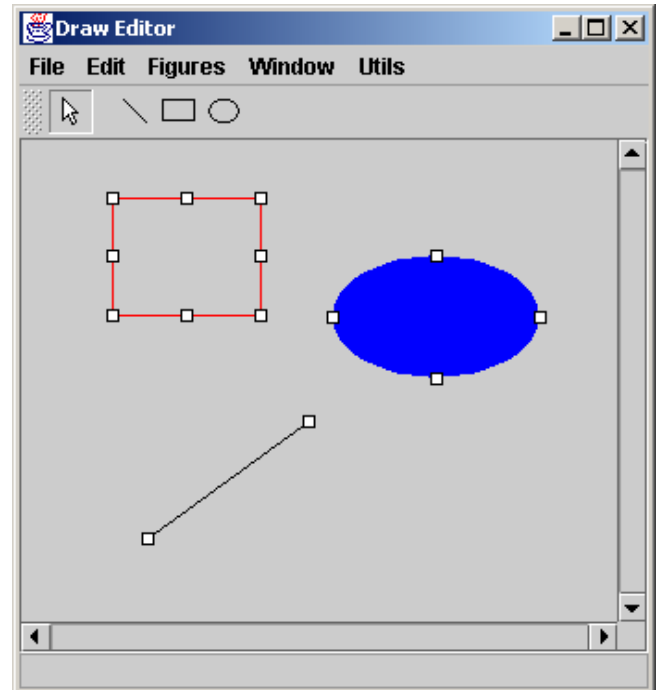
Mit der letzten Übung haben sie sich in das JDraw Framework eingearbeitet, das Modell und erste Figuren entwickelt und die Aktualisierung der Sichten sichergestellt. Gewisse Methoden aus dem Interface *Figure* haben Sie jedoch noch nicht ausprogrammiert. In dieser Woche sollen Sie die Methode *getHandles* realisieren.

Mit der Methode *getHandles* gibt eine Figur ihre Handles zurück. Handles werden verwendet, um das Aussehen einer Figur zu verändern. In der nebenstehenden Figur bietet die Linie zwei und das Rechteck acht Handles an.

Handles sind Objekte die das Interface *jdrawing.framework. FigureHandle* implementieren. Analog zur Figurenklasse kennt auch dieses Interface die Methoden *draw*, *contains* und *getBounds*.

Mit der Methode *getCursor* kann ein neuer Cursor definiert werden. Dieser wird angezeigt, wenn sich die Maus gerade über einem Handle befindet.

Die Methode *getOwner* soll für jeden Handle eine Referenz auf die zugehörige Figur zuruckliefern.



Wenn die Maus in einem Handle gedrückt wird, so werden die Methoden *startInteraction*, *dragInteraction* und *stopInteraction* aufgerufen. Die zugehörige Figur soll dabei angepasst werden. Da während dem *drawInteraction* typischerweise die Methode *setBounds* der Figur aufgerufen wird, die ihrerseits einen *FigureEvent* an das Modell schickt, wird die View automatisch nachgeführt. Ein expliziter Aufruf der Methode *repaint* auf der View ist sollte nicht nötig sein. Der zusätzliche *MouseEvent* Parameter kann verwendet werden, um Modifier-Tasten auszulesen, und der *DrawView* Parameter könnte verwendet werden, um in der Statuszeile Text anzuzeigen.

Überlegen Sie sich, wo sie hier sinnvoll abstrakte Basisklassen einsetzen können.