

Übung 10: Undo & Redo

Der Grafikeditor soll in dieser Übung mit Undo/Redo Funktionalität erweitert werden. Dazu müssen alle Operationen wie Einfügen und Löschen einer Figur sowie das Verschieben und Skalieren einer Figur als Kommandos bereitgestellt werden.

Im Framework ist dazu das Interface DrawCommand vorgegeben.

```
public interface DrawCommand {  
    // Executes a command  
    void redo();  
  
    // Undoes the action performed by execute.  
    void undo();  
}
```

Mit der Methode redo wird ein Kommando ausgeführt und mit undo wird es rückgängig gemacht. Bei einer Einfügeoperation muss z.B. eine Klasse InsertCommand programmiert werden, deren redo Methode die Figur einfügt, und deren undo Methode die Figur wieder aus dem Modell entfernt.

Das Modell delegiert die Verwaltung der Kommandos an ein separates Objekt, welches das Interface DrawCommandHandler implementiert. Der Zugriff auf diesen Handler erfolgt über die Methode getDrawCommandHandler im Modell. In Ihrem Modell gibt diese Methode aktuell einen leeren Handler (eine Instanz der Klasse EmptyDrawCommandHandler) zurück.

```
public interface DrawCommandHandler {  
  
    // adds a command to the command chain  
    public void addCommand(DrawCommand cmd);  
  
    // clears the command chain  
    public void clearHistory();  
  
    // undoes or redoes commands in the command chain  
    public void undo();  
    public void redo();  
  
    // Indicates whether an undo or redo operation is possible.  
    public boolean undoPossible();  
    public boolean redoPossible();  
  
    // Method to open/close a script.  
    public void beginScript();  
    public void endScript();  
}
```

Bei den Verschiebe- oder Skalierungsoperationen hat man das Problem, dass diese aus vielen kleinen Mausbewegungen (und damit aus vielen kleinen Kommandos) bestehen. Wünschenswert wäre, wenn man diese Operationen zu einer Makrooperation zusammenfassen könnte. Dazu sind im Interface DrawCommandHandler die Methoden beginScript und endScript vorgesehen. Operationen, die zwischen diesen beiden Methodenaufrufen mit addCommand registriert werden, sollen intern zu einer einzigen Operation zusammengefasst werden, die mit *einem* undo-Befehl rückgängig gemacht werden kann.

Die Undo/Redo-Menüeinträge sind bereits vorgesehen, und die Klasse StdDrawView verwendet für Löschoperationen und Verschiebeoperationen bereits Commands, die im DrawCommandHandler registriert werden. Die beginScript und endScript Methoden werden ebenfalls in den MotionEventListener-Methoden der Klasse StdDrawView aufgerufen (z.B. für das Verschieben von Figuren).

Aufgaben:

1. Implementierung des DrawCommandHandlers in Ihrem Modell
2. Umwandlung des Hinzufügens einer Figur zum Modell in den DrawTools in einen Command.
3. Die Handles müssen die Änderung der Figur ebenfalls als Command registrieren

Abgabe: 09. Januar 2018