

Einführung in die Programmierung für Kunst und Multimedia

Übungsblatt 9

Ende der Abgabefrist: 08.01.2014 12:00 Uhr

Hinweise zur Abgabe:

Übungsblätter dürfen NICHT in Teams abgegeben werden, da Sie sich durch eine erfolgreiche Bearbeitung einen Bonus für die Klausur verdienen können. Es ist zwar sinnvoll in kleinen Teams die Aufgaben zu diskutieren, die Lösungen müssen aber von jedem Studenten EINZELN bearbeitet werden. Bitte beachten Sie, dass abgeschriebene Lösungen mit 0 Punkten bewertet werden!

Sammeln Sie die Lösungen zu diesem Übungsblatt in einem zip-Archiv loesung09.zip. Dieses zip-Archiv können Sie schließlich in UniWorX abgeben.

Wichtig: Achten Sie bitte darauf, dass Ihre Lösungsdateien den korrekten Namen und das korrekte Format haben! Beides wird in der Angabe explizit angegeben. Dateien im falschen Format oder mit falschem Namen werden im Allgemeinen nicht korrigiert.

Aufgabe 9-1 Erweiterung der Banksimulation

20 Punkte

Diese Aufgabe baut auf der bisherigen Banksimulation (Übungsblätter 6 und 8) auf. Musterlösungen dazu finden Sie zum Download auf der Homepage. Sie können sowohl auf diese als auch auf Ihre eigene Lösung aufbauen – beachten Sie dann jedoch, dass Ihre bisherige Lösung die früheren Anforderungen vollständig und fehlerfrei erfüllt.

In der Übung haben Sie das MVC-Pattern, das Observer-Pattern und deren Implementierung kennengelernt. Da die Banksimulation eine visuelle Ausgabe bekommen soll, sollen diese beiden Pattern umgesetzt werden.

1. Erstellen Sie dazu in Ihrem *src*-Ordner drei Packages: *model*, *view* und *controller*. Verschieben Sie Ihre Klassen in das *model*-Package. Lediglich die Klasse *Banksimulation.java* bleibt, als Programm aufrufende Klasse, ohne Package im *default package*. Erstellen Sie außerdem im *controller*-Package eine Klasse *Controller* sowie im *view*-Package eine Klasse *View*. Einen leeren Beispielaufbau dazu finden Sie auch auf der Homepage zum Download.
2. Die Klasse *Banksimulation.java* dient nun ausschließlich nur noch zum Start des Programms. In der *main*-Methode wird dazu lediglich ein neuer *Controller* erzeugt.

```
public static void main(String[] args) {  
    new Controller();  
}
```

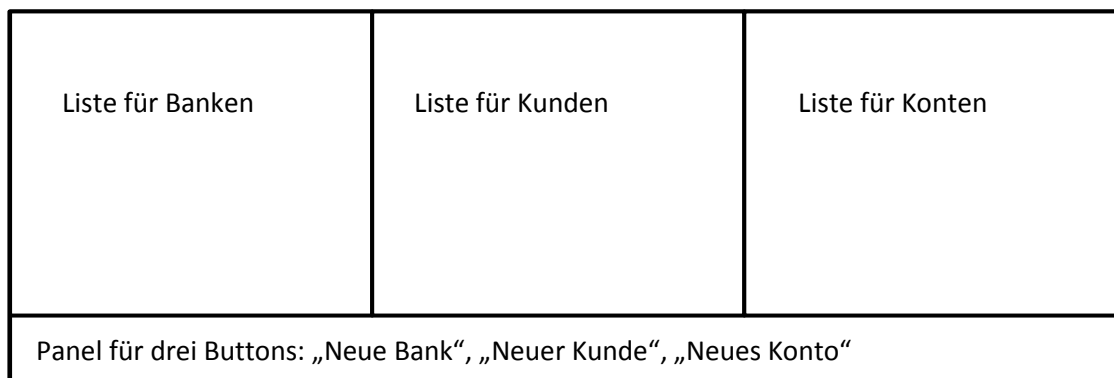
Diese Klasse enthält sonst keinen weiteren Inhalt.

3. Sie sollen nun das MVC-Pattern gemeinsam mit dem Observer-Pattern implementieren. Die Klasse Bank ist dabei das Observable, die View der Observer. Fügen Sie die dazu notwendigen Änderungen an die Klassen hinzu und verknüpfen Sie beide in Ihrem Controller. Der Controller wird in der Klasse Banksimulation.java aufgerufen, sie können daher den fürs Observer-Pattern notwendigen Code in den Konstruktor des Controllers schreiben.
4. In der Übung haben Sie den grundlegenden Aufbau von UI-Fenstern in Java mit Swing kennengelernt. Die Ansicht der Banksimulation soll aus drei Listen sowie mehreren Buttons bestehen. In einer Liste werden Banken, in einer Liste Kunden und in einer Liste Konten angezeigt. Durch die Buttons können neue Elemente den Listen hinzugefügt werden.

Ihre View verfügt dabei in diesem Übungsblatt über keine Funktion, es sollen lediglich die leeren Listen und nicht funktionierenden Buttons erstellt und angeordnet werden.

Lassen Sie Ihre View von der Bibliotheksklasse `JFrame` erben und erstellen Sie den grundsätzlichen Aufbau Ihres Fensters im Konstruktor der Klasse `View`. Denken Sie dabei auch an Funktionen zum Beenden des Programms und an eine sinnvolle Fenstergröße- sowie Position. Das Fenster sollte dabei mindestens 600x400 Pixel groß sein und an keiner Ecke des Bildschirms kleben.

Machen Sie sich mit den verschiedenen Layout-Möglichkeiten in Swing vertraut, und entscheiden Sie sich für ein sinnvolles Layout, um vier Elemente in der folgenden Form anzeigen zu können:



5. Vergessen Sie nicht, Ihren Code ausreichend ausführlich zu kommentieren und zu dokumentieren. Erzeugen Sie wie in Aufgabe 8-1 eine Javadoc Ihrer Banksimulation im Ordner `doc` und geben Sie diese mit ab.

Bitte geben Sie Ihr vollständiges Eclipse-Projekt als zip-Archiv ab. Verwenden Sie dazu einfach den gesamten Ordner Ihres Projektes im workspace. Bitte achten Sie unbedingt darauf, dass Ihre Lösung fehlerfrei und kompilierbar ist. **Es werden ausschließlich lauffähige Lösungen im korrekten Format bewertet.**

Abgabe

Zulässiges Dateiformat für die Lösungen dieses Übungsblattes ist JAVA sowie die Dateien Ihrer Javadoc. Bitte geben Sie Ihre Lösung als ZIP-Datei bis zum 18.12.13 12:00 Uhr in UniWorX (<https://uniworx.ifi.lmu.de>) ab.

Hinweis: Verspätete Abgaben, Abgaben im falschen Dateiformat und nicht lauffähige Java-Dateien werden nicht bewertet.