

Einstieg in Java und OOP

Christian Silberbauer

Übungsblatt 5

Folgende Aufgaben beziehen sich weitgehend auf die GUI-Folien der Vorlesung. Speichern Sie bitte die einzelnen Versionen der KundenVerwaltung in verschiedenen Projekten.

Aufgabe 1

Implementieren Sie die KundenVerwaltung v1 aus der Vorlesung. Stichwörter:

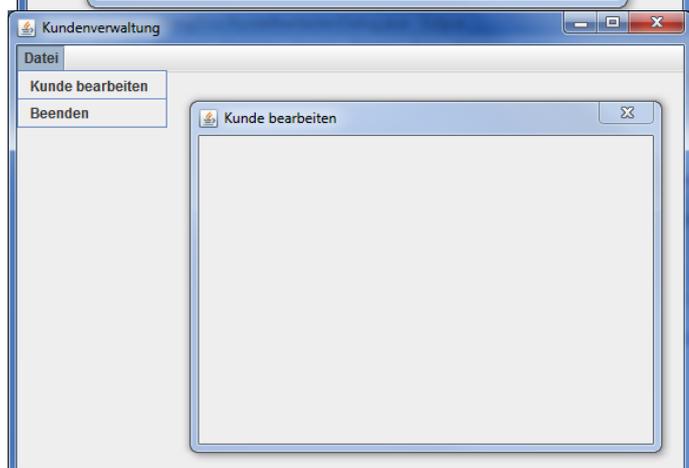
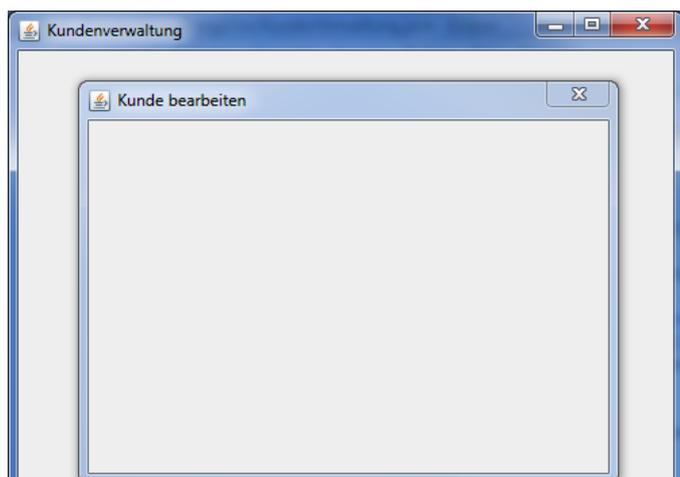
- JDialog
- setDefaultCloseOperation()
- setVisible()



Aufgabe 2

Implementieren Sie die KundenVerwaltung v2 aus der Vorlesung. Stichwörter:

- setModal()
- JFrame
- Fenster öffnen



Aufgabe 3

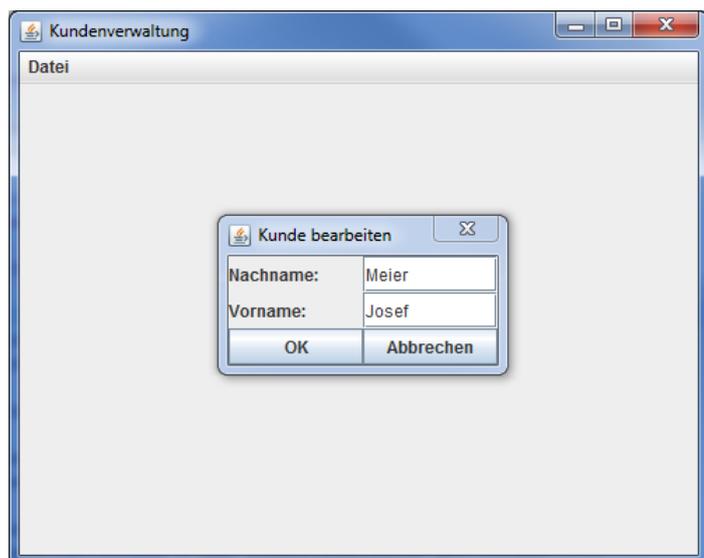
Implementieren Sie die Kundenverwaltung v3 aus der Vorlesung. Stichwörter:

- JMenuBar
- JMenu
- JMenuItem
- JSeparator
- ActionListener

Aufgabe 4

Implementieren Sie die Kundenverwaltung v4 aus der Vorlesung. Stichwörter:

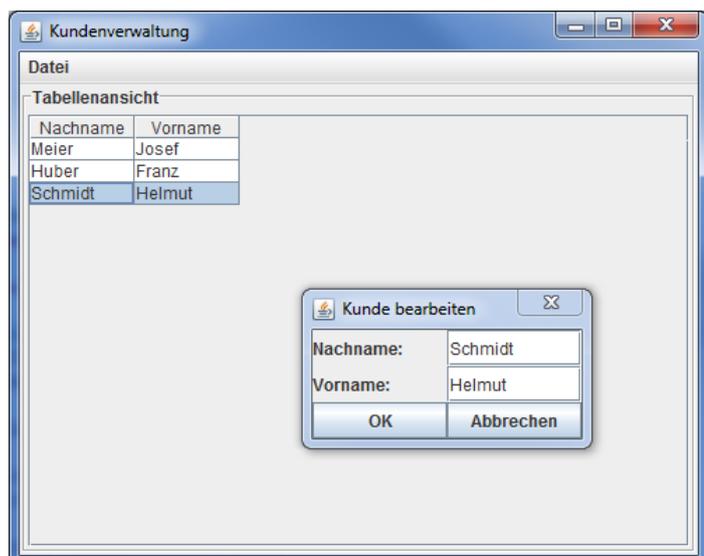
- JTextField
- JButton
- LayoutManager
- Inter-Dialog-Kommunikation



Aufgabe 5

Implementieren Sie die Kundenverwaltung v5 aus der Vorlesung. Stichwörter:

- JTable
- TableModel
- ListSelectionModel
- MouseListener
- ScrollPane
- TitlePane



Die Anwendung so wie sie auf den Vorlesungsfolien beschrieben ist, wirft eine Exception, wenn über das Menü der „Kunde bearbeiten“-Dialog geöffnet wird und in der Tabelle noch kein Eintrag ausgewählt wurde. Beheben Sie das Problem, indem Sie den entsprechenden Menüeintrag deaktivieren (disable), sofern keine Zeile selektiert ist.

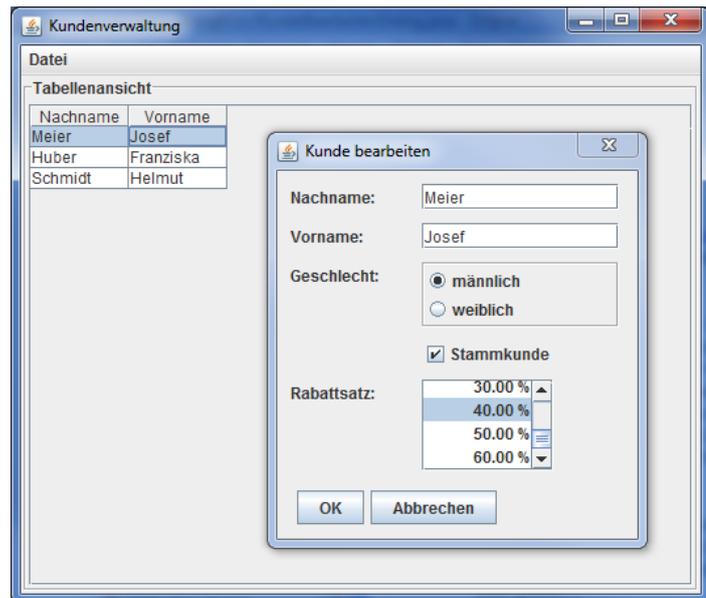
Lösungshinweise für die Erweiterung:

- Entsprechendes `JMenuItem` als Attribut speichern und dieses zunächst mit `setEnabled(false)` deaktivieren.
- Für das `SelectionModel` einen `ListSelectionListener` registrieren. Falls der ausgewählte Index ungleich `-1` ist, das `JMenuItem` aktivieren.

Aufgabe 6a

Implementieren Sie die Kundenverwaltung v6 aus der Vorlesung. Stichwörter:

- `JCheckBox`
- `JRadioButton`
- `JList`
- `ListModel`
- `ListCellRenderer`



Fügen Sie zudem dem „Kunden bearbeiten“-Dialog ein `JScrollPane` hinzu, sodass durch Verkleinern des Fensters Scrollbalken erscheinen.

Lösungshinweise für die Erweiterung:

- Erzeugen Sie zunächst ein `JPanel`, dem sämtliche Komponenten des Dialogs hinzugefügt werden. Dieses `JPanel` verwendet keinen `LayoutManager`.
- Setzen Sie hierbei mit `setPreferredSize()` die bevorzugte Größe (`setSize()` funktioniert nicht!)
- Fügen Sie dieses `JPanel` einem `JScrollPane` hinzu, das daraufhin dem `ContentPane` hinzugefügt wird.
- Das `ContentPane` selbst sollte das `BorderLayout` verwenden (Default), sodass das `JScrollPane` den gesamten Platz verwendet.
- Durch `pack()` vor `setVisible()` wird die initiale Fenstergröße automatisch bestimmt (anhand der bevorzugten Größe des `JPanel`s im `ContentPane`).

Aufgabe 6b

Ergänzen Sie die Kundenverwaltung der vorhergehenden Aufgabe um die Felder Geburtsdatum, Straße, Postleitzahl und Wohnort.

Verwenden Sie zur Speicherung der Kundendaten eine Collection vom Typ `List<Kunde>`, statt wie bisher einen Array.

Zeigen Sie sämtliche Felder wie folgt in der Tabellenansicht an. Aus dem Geschlecht soll in der Tabellenansicht der Wert für die Anrede „Herr“ oder „Frau“ abgeleitet werden.

Anrede	Nachname	Vorname	Geburtsdatum	Straße	Plz	Wohnort	Stammkunde?	Rabattsatz
Herr	Meier	Josef	03.05.1987	Universitätsstraße 31	93053	Regensburg	<input checked="" type="checkbox"/>	40.00 %
Frau	Huber	Franziska	13.12.1976			München	<input type="checkbox"/>	0.00 %
Herr	Schmidt	Helmut	23.12.1918	Hamburger Straße 233	20095	Hamburg	<input type="checkbox"/>	0.00 %

Stammkunden erhalten i.d.R. mindestens 30% Rabatt. Wird im `KundeBearbeitenDialog` ein Kunde zum Stammkunden gemacht, so soll folgende Meldung erscheinen, wenn der aktuelle Rabattsatz unter 30% liegt:

Wählt der Anwender hier „Ja“, dann wird der Rabatt auf 30% gesetzt.

Lösungshinweise:

- Für das Geburtsdatum empfiehlt sich ein `JFormattedTextField`, parametrisiert mit einem `SimpleDateFormat`.
- `JTable` generiert in der Spalte „Stammkunde?“ eigenständig `Checkboxen`, wenn hierfür die Methode `getColumnClass()` des `TableModel`s den Typ `Boolean` zurückgibt. Zudem muss hier für das Geburtsdatum `Date` zurückgegeben werden, ansonsten `String`.
- Um die Hinweismeldung beim Klick auf „Stammdaten“ zu erzeugen, wird ein `ItemListener` benötigt.