
Softwarepraktikum: Tanzeditor JDance

<http://www.informatik.uni-freiburg.de/proglang/teaching/aktuell/sopra>

Übungsblatt 1

Abgabe: 30.10.2002

Für JDance 3000 Version 2.0 ist ein brandneues Feature vorgesehen: eingegebenen Tänze sollen mit passenden Musiktiteln unterlegt werden können. Unser Kunde verlangt für die Vertragverhandlungen einen ersten Prototypen, der dieses Feature demonstrieren kann. Für diesen Prototypen muss eine Teilkomponente entwickelt werden, die Musiktitel-Daten verarbeiten kann. Die für uns relevanten Daten eines Musiktitels sind: *Interpret*, *Titel*, *Titellänge (mm:ss)*, *Tanzart*, *BPM (Schläge pro Minute)* und *URL*.

Aufgabe 1: Spezifikation der Schnittstellen

- (a) Schreibe ein Java-Interface `Title`, das Musikstücke in allgemeinerer Form, d.h. nur mit Attributen *Interpret*, *Titel*, *Titellänge* und *URL*, spezifiziert.
- (b) Erweitere die Schnittstelle `Title` mit Hilfe von Vererbung zur Schnittstelle `DanceTitle`, die zusätzlich die Komponenten für die restlichen Attribute *Tanzart* und *BPM* vorsieht.
- (c) Schreibe eine abstrakte Klasse `AbstractTitle`, die das Interface `Title` implementiert. Leite davon die Klasse `JDanceTitle` ab, die die Schnittstelle `DanceTitle` implementiert.

Repräsentiere das Attribut *BPM* durch `ints`, *URLs* als `java.net.URL`-Objekt, und alle anderen Attribute als `Strings`.

Aufgabe 2: Einlesen von Musiktiteln

Ein einfaches Datenaustauschformat, das von vielen Tabellenkalkulationsprogramme unterstützt wird, basiert auf folgenden Vereinbarungen:

- In jeder Zeile steht genau ein Datensatz.
- Benachbarte Datenfelder werden durch ein Semikolon ; getrennt.
- Ein Datenfeld kann leer sein (;).
- Datenfelder, die Leerzeichen oder Sonderzeichen enthalten, werden in doppelte Hochkommata (") gestellt.

Schreibe ein Programm, das Tanztitel, die in einer Datei dieses Formats abgespeichert sind, einliest, `JDanceTitle`-Objekte erzeugt, diese in einem `Vector` ablegt, aufsteigend nach Titeln sortiert, und am Bildschirm ausgibt. Gehe davon aus, dass die Datenfelder in den zu bearbeitenden Dateien immer in der oben angegebenen Reihenfolge vorliegen.

- Mit Hilfe der Klasse `java.io.BufferedReader` können Zeichen aus einem zwischengepufferten Eingabestrom gelesen werden. Die Methode `readLine()` dient dazu, eine komplette Zeile einzulesen. Die Klasse `java.io.FileReader` kann benutzt werden, um für Dateien Eingabeströme zu erzeugen.
- Die Klasse `java.util.StringTokenizer` ist nützlich, um Textzeilen in Felder zu zerlegen.
- Implementiere eine von `java.io.FileReader` abgeleitete Klasse `JDanceTitleFileReader`, die zusätzlich eine Methode `JDanceTitle readRecord()` besitzt, um einen weiteren Datensatz einzulesen.

Demonstriere deine Programms mit Hilfe der auf der Praktikumswebseite angegebenen Beispieldaten-Datei.

Aufgabe 3: GUI-Oberfläche

Entwerfe eine einfach graphische Benutzeroberfläche, die es erlaubt Tanztiteldateien einzulesen und anzuzeigen. Implementiere Deinen Entwurf, indem Du Dein Programm aus Aufgabe 2 um eine Benutzeroberfläche mit Hilfe von JFC/Swing-Komponenten erweiterst.

- Auf der Praktikumswebseite findest Du ein einfache Swing-Applikation, die Dir als Vorlage dienen kann.
- Es gibt mehrere Möglichkeiten um GUI-Elemente in einem Fenster anzuordnen: man benutzt einen so genannten “Layout Manager”. Benutze für Dein GUI das `java.awt.GridBagLayout` zur Anordnung.
- Benutze die Klasse `javax.swing.JTable`, um die Daten in Tabellform anzuordnen.