

Aufgabe: Filmverwaltung

Thomas.Klumpp@stud.hs-mannheim.de

An der Hochschule Mannheim soll das Freizeitangebot für die Studenten erweitert werden. Hierzu hat der Hochschulrat beschlossen, eine eigene Videothek speziell für Studenten, Lehrpersonal und Mitarbeiter einzurichten. Da jedoch zum Sommersemester 2012 das Land Baden-Württemberg keine Studiengebühren mehr erhebt, ist das zur Verfügung stehende Budget sehr gering. Für die Herrichtung der benötigten Räumlichkeiten sowie für diverse DVD & Blue-ray Discs hat das Geld gerade noch gereicht, für eine Software zur Verwaltung der Medien und zur Organisation der Ausleihe aber leider nicht mehr. Es wurde daher beschlossen die Aufgabe zur Entwicklung einer solchen Software an die GDI-Studenten des ersten Semesters zu übertragen.

Jetzt sind Sie an der Reihe:

Schreiben Sie eine Filmverwaltungssoftware für die neue Hochschulvideothek!

(a)

Erstellen Sie eine Klasse *Movie* und eine Klasse *Customer* mit zunächst folgenden Attributen:

Ein *Movie* hat:

- title
- mediaType
- length
- ageRating
- productionYear
- lendTo

Ein *Customer* hat:

- firstName
- lastName
- adress
- postalCode
- city
- gender
- dateOfBirth

Überlegen Sie sich geeignete Datentypen für diese Attribute (müssen während dem Testat begründet werden können!) und sorgen Sie dafür, dass von außen kein direkter Zugriff auf diese Attribute möglich ist.

Implementieren Sie in den beiden Klassen jeweils die *toString()*-Methode. Es sollen alle Felder wie im folgenden Beispiel auf der Konsole ausgegeben werden (Tabulator bei Adresse beachten).

Beispiel <i>Movie</i>	Beispiel <i>Customer</i>
Filmtitel: Pulp Fiction Medientyp: Blue-ray Filmlänge in Minuten: 148 Altersfreigabe FSK: 16 Produktionsjahr: 1994 Zur Zeit ausgeliehen an: Max Mustermann	Max Mustermann Adresse: Musterstr. 1 1234 Musterstadt Geschlecht: männlich Geburtsdatum (TT.MM.JJJJ): 01.01.2012

„Zur Zeit ausgeliehen an“ soll nur erscheinen, wenn der Film gerade ausgeliehen ist (also wenn `lendTo` gesetzt ist).

(b)

Erstellen Sie eine Klasse *MovieManagementDatabase* die als Hauptprogramm dient und sämtliche Operationen steuert.

In dieser Klasse sollen alle Filme und Kunden in Arrays gespeichert werden. Es sollen maximal 5 Millionen Filme und 10 Tausend Kunden gespeichert werden können. Es soll später möglich sein die beiden Zahlenwerte an **einer** zentralen Stelle im Programmcode ändern zu können!

Implementieren Sie nun die folgenden Methoden in dieser Klasse:

public Movie createNewMovie()

Fragt die benötigten Parameter vom Benutzer über die Konsole ab, erstellt einen neuen Film und liefert ihn zurück.

public Customer createNewCustomer()

Fragt die benötigten Parameter vom Benutzer über die Konsole ab, erstellt einen neuen Kunden und liefert ihn zurück.

public void borrowMovie()

Diese Methode ermöglicht es einen Film an einen Kunden auszuleihen.

Es werden folgende Schritte durchgeführt:

1. Abfrage welcher Film ausgeliehen werden soll
Es kann davon ausgegangen werden, dass es in der Videothek einen Film mit demselben Namen maximal 1x auf DVD **und** 1x auf Blue-ray geben kann.
2. Suche im Array nach dem Film
3. Abfrage an wen der Film ausgeliehen werden soll
Es kann davon ausgegangen werden, dass ein Kundenname nur einmal vorkommt
4. Suche im Array nach dem Kunden
5. Attribut `lendTo` in Film mit dem Kunden befüllen

public void returnMovie()

Diese Methode ermöglicht es einen Film zurückzugeben.

Es werden folgende Schritte durchgeführt:

1. Abfrage welcher Film zurückgegeben werden soll
Es kann davon ausgegangen werden, dass es in der Videothek einen Film mit demselben Namen maximal 1x auf DVD und 1x auf Blue-ray geben kann.
2. Suche im Array nach dem Film (Denken Sie daran keinen Code zu duplizieren!)
3. Attribut `lendTo` in Film leeren

public void printAllMovies()

Druckt alle Filme in einem menschenlesbaren Format auf die Konsole. Hierbei bitte auf eine deutliche Trennung der einzelnen Datensätze achten (ggf. Trennzeichen einführen).

public void printAllCustomers()

Druckt alle Kunden in einem menschenlesbaren Format auf die Konsole. Hierbei bitte auf eine deutliche Trennung der einzelnen Datensätze achten (ggf. Trennzeichen einführen).

(c) Legen Sie in der Klasse *MovieManagementDatabase* eine weitere Methode *public void callMainMenu()* an, die ein zentrales Bedienmenü mit folgenden Aktionen bietet:

1. Neuen Film anlegen
2. Neuen Kunden anlegen
3. Film ausleihen
4. Film zurückgeben
5. Auflistung aller Filme
6. Auflistung aller Kunden
7. Programm beenden

Das Menü soll mit einer Switch-Anweisung realisiert werden.

Die *main()*-Methode soll lediglich den Menüaufruf beinhalten.

(d)

Fügen Sie dem Programm einen weiteren Menüpunkt „Erzeuge Statistik“ an. Erzeugen Sie hierfür eine Methode *public void printStatistics(...)*.

Das ... in der Argumentliste bedeutet, dass es Ihnen frei gestellt ist, ob Sie der Methode Argumente übergeben möchten oder nicht.

Durch die Methode sollen die folgenden Daten ausgegeben werden:

Anzahl der Kunden

Anzahl der Filme

Anzahl der Filme die eine Jugendfreigabe haben (< 18)

Anzahl der Filme die keine Jugendfreigabe haben (>18)

(e) Zusatzaufgabe

Überlegen Sie sich eine Möglichkeit wie Sie verhindern können, dass jemand aus einer anderen Klasse heraus eine Instanz der Klasse *MovieManagementDatabase* erstellen kann. In der *main()*-Methode der Klasse muss die Instanziierung natürlich weiterhin möglich sein!

Implementieren Sie anschließend ihren gefundenen Mechanismus!

Hinweise zur Implementierung

- Das Geschlecht soll mit m=männlich oder w=weiblich gespeichert werden
- Das Geburtsdatum soll mit 10 Ziffern gespeichert werden. Falls das Format durch den Benutzer nicht eingehalten wird soll darauf reagiert werden.
- Fehleingaben durch den Benutzer sollen abgefangen werden und sinnvoll darauf reagiert werden. Es dürfen keine Felder der Klassen *Movie* und *Customer* leer bleiben (also auch kein leerer String!) bis auf *endTo*
- Lassen Sie sich nur die Getter und Setter von Eclipse generieren, die Sie auch wirklich benötigen!
- Vermeiden Sie einen Arrayüberlauf: 5 Millionen Filme und 10 Tausend Kunden sind nicht unendlich!

Allgemeine Hinweise

- Benennen Sie alle ihre Variablennamen auf Englisch
- Kommentieren Sie zumindest **jede** Methode mit JavaDoc in englischer Sprache
- JavaDoc darf gerne auch bei Attributen einer Klasse eingesetzt werden
- Fügen Sie ggf. weitere sinnvolle Kommentare in Ihrem Quelltext ein, die ebenso konsequent in Englisch zu halten sind.
- Schreiben Sie keinen Code doppelt, verwenden Sie ggf. eigene weitere Methoden.