

9. Übung

Abgabe: bis 18.01.2022, 6 Uhr

Aufgabe 1 - Unterklassen von Artikel

Erweitern Sie Ihre **korrigierten** Klassen Artikel und Lager um die nachfolgend beschriebenen Funktionalitäten. Von der Klasse Artikel sind die Unterklassen CD, Video und Buch abzuleiten mit den folgenden zusätzlichen Attributen. Beachten Sie dabei die Namen der Attribute, die in Klammern angegeben sind.

CD

Zusätzliche Attribute:

- Interpret (interpret)
- Titel (titel)
- Anzahl der Musiktitel (anzahlTitel)

Folgende Bedingungen gelten für die Attribute:

- interpret: nicht leer
- titel: nicht leer
- anzahlTitel: > 0

Die Klasse CD sollte mindestens den folgenden Konstruktor haben: *CD(int artikelNr, int bestand, double preis, String interpret, String titel, int anzahlTitel)*

Video

Zusätzliche Attribute:

- Titel (titel)
- Spieldauer in Minuten (spieldauer)
- Erscheinungsjahr (jahr)

Folgende Bedingungen gelten für die Attribute:

- titel: nicht leer
- spieldauer: > 0
- jahr: zwischen 1900 und 2022

Die Klasse Video sollte mindestens den folgenden Konstruktor haben: *Video(int artikelNr, int bestand, double preis, String titel, int spieldauer, int jahr)*

Buch

Zusätzliche Attribute:

- Titel (titel)
- Autor (autor)
- Verlag (verlag)

Folgende Bedingungen gelten für die Attribute:

- titel: nicht leer
- autor: nicht leer
- verlag: nicht leer

Die Klasse Buch sollte mindestens den folgenden Konstruktor haben: *Buch(int artikelNr, int bestand, double preis, String autor, String titel, String verlag)*

Weitere Funktionen

Sie dürfen natürlich gerne weitere Konstruktoren definieren, falls Sie diese benötigen. Das Attribut *art*, das in der Klasse Artikel definiert ist, ist für diese Klassen jeweils mit dem Text „Medien“ zu belegen. Implementieren Sie für die Unterklasse jeweils eine sinnvolle equals()- und hashCode()-Methode.

In der Klasse Artikel und ihren Unterklassen ist eine Methode *getBeschreibung()* zu definieren. Sie erzeugt eine Beschreibung als String, die je nach Unterklasse anders zusammengesetzt ist. Im folgenden ist die Rückgabe der Methode *getBeschreibung()* für die jeweiligen Klassen zu sehen:

- Artikel: *art*
- Video: *titel*
- CD: *interpret: titel*
- Buch: *autor: titel*

Die Dialogklasse aus der 5. Übung ist zu erweitern um die Möglichkeit, CD-, Video- und Buch-Objekte erfassen zu können. Über die Methode *legeAnArtikel()* lassen sich nun Objekte vom Typ Artikel, CD, Video und Buch in das Lager einbuchen. Zusätzlich ist eine Methode *ausgebenBestandsListe()* zum Ausgeben der Lagerbestände zu erstellen, deren Ausgabe **ungefähr** so aussehen soll:

ArtNr	Beschreibung	Preis	Bestand	Gesamt
4711	Stroustrup: The C++ Programming Language	49,95	51	2547,45
5712	Santana: Supernatural	10,90	10	109,00
6713	Star Wars I - VIII	40,00	15	600,00
Gesamtwert:				3256,45

Die *toString()*-Methoden der Klassen sollen ebenfalls jeweils realisiert werden und zwar so, dass die *toString*-Methode von Lager den kompletten Lagerinhalt inklusive der unterschiedlichen Artikel-Typen aufbereitet.

Aufgabe 2 - Automatische Tests

Schreiben Sie Testklassen mit JUnit für die oben beschriebenen Unterklassen von Artikel. Die Methode *ausgebenBestandsListe()* braucht nicht automatisch getestet zu werden.