

Übung 5

Aufgabenstellung

In den letzten Unterrichtsstunden haben wir uns mit folgenden Themen beschäftigt:

- Vererbung
- Abstrakte Klassen und abstrakte Methoden
- Überschreiben von Methoden in abgeleiteten Klassen
- Aufruf von Methoden der Vaterklasse
- Polymorphismus

Im Folgenden sollen diese Programmiertechniken geübt werden. Es sollen einige Klassen betrachtet werden, die einen Teil einer **Artikelverwaltung** darstellen, wie sie z.B. ein Web-Versandhandel verwenden könnte.

Aufgabe 1

Erstellen Sie die Klasse **Artikel**. Ein Artikel sei durch drei Kenngrößen bestimmt: die Artikelnummer, die Artikelbezeichnung und den Verkaufspreis.

Diese Klasse beinhalte daher die folgenden Instanzvariablen: die **artikelNummer** als String, die **bezeichnung** als String und den **preis** als double-Wert.

Schreiben Sie die Klasse mit folgenden Eigenschaften, bzw. Methoden:

- setter- und getter-Methoden.
- Konstruktor
- eine Methode **toString**, die die Inhalte aller Instanzvariablen eines Objekts in einem String zurückliefert, der aber noch in eine Zeile passen soll.
- eine Methode **datenAusgeben**. Diese Methode soll die Inhalte der Instanzvariablen eines Objekts am Bildschirm ausgeben, und zwar jede Instanzvariable in einer eigenen Zeile.
- eine Methode **preisErhoehen**. Diese Methode hat als Parameter eine Zahl, die den Prozentsatz für eine Preiserhöhung darstellen soll. Wird die Methode also mit dem Parameter 5.5 aufgerufen, wird der Preis des Artikels um 5.5% erhöht.

Aufgabe 2

Erstellen Sie die Klasse **Buch**. Ein Buch ist auch ein Artikel. Es besitzt aber zusätzlich die Kenngröße Erscheinungsjahr. Diese wird in der Instanzvariablen **jahr** als int-Wert abgespeichert.

Ergänzen Sie die Klasse um folgende Eigenschaften, bzw. Methoden:

- die Klasse **Buch** ist eine von der Klasse **Artikel** abgeleitete Klasse.
- die setter- und getter-Methoden
- einen Konstruktor mit einer Parameterliste, so dass alle Instanzvariablen initialisiert werden können.
- eine Methode **toString**, die die Inhalte aller Instanzvariablen eines Objekts in einem String zurückliefert, der aber noch in eine Zeile passen soll.
- eine Methode **datenAusgeben**. Diese Methode soll die Inhalte der Instanzvariablen eines Objekts am Bildschirm ausgeben, und zwar jede Instanzvariable in einer eigenen Zeile.
- eine Methode **checkArtikelNummer**. Diese Methode soll die Artikelnummer daraufhin überprüfen, ob sie auch mit dem Buchstaben **B**, groß oder klein, anfängt, wie der Klassenname. Sie liefert einen entsprechenden boolean-Wert zurück.

Aufgabe 3

Erstellen Sie die Klasse **DVD**, die eine Film-DVD beschreibt.

Eine DVD ist auch ein Artikel. Sie besitzt aber zusätzlich die Kenngröße Spieldauer. Diese wird in der Instanzvariablen **dauer** als int-Wert abgespeichert, die Einheit in Minuten.

Ergänzen Sie die Klasse um folgende Eigenschaften, bzw. Methoden:

- die Klasse **DVD** ist eine von der Klasse **Artikel** abgeleitete Klasse.
- setter und getter-Methoden
- einen Konstruktor mit einer Parameterliste, so dass alle Instanzvariablen initialisiert werden können.
- eine Methode **datenAusgeben**. Diese Methode soll die Inhalte der Instanzvariablen eines Objekts am Bildschirm ausgeben, und zwar jede Instanzvariable in einer eigenen Zeile.
- eine Methode **laufzeit**. Diese Methode gibt die Spieldauer der DVD am Bildschirm aus, und zwar jetzt umgerechnet in Stunden und Minuten.
- eine Methode **checkArtikelNummer**. Diese Methode soll die Artikelnummer daraufhin überprüfen, ob sie auch mit dem Buchstaben **D**, groß oder klein, anfängt, wie der Klassenname. Sie liefert einen entsprechenden boolean-Wert zurück.

Aufgabe 4

Erstellen Sie die Klasse **ComputerBuch**. Ein ComputerBuch ist auch ein Buch.

Es besitzt aber zusätzlich die Kenngröße Informatik-Fachgebiet. Diese wird in der Instanzvariablen **gebiet** als String abgespeichert.

Ergänzen Sie die Klasse um folgende Eigenschaften, bzw. Methoden:

- die Klasse **ComputerBuch** ist eine von der Klasse **Buch** abgeleitete Klasse.
- Setter- und getter-Methoden
- einen Konstruktor mit einer Parameterliste, so dass alle Instanzvariablen initialisiert werden können.
- eine Methode **toString**, die die Inhalte aller Instanzvariablen eines Objekts in einem String zurückliefert, der aber noch in eine Zeile passen soll.
- eine Methode **datenAusgeben**. Diese Methode soll die Inhalte der Instanzvariablen eines Objekts am Bildschirm ausgeben, und zwar jede Instanzvariable in einer eigenen Zeile.
- eine Methode **checkArtikelNummer**. Diese Methode soll die Artikelnummer daraufhin überprüfen, ob sie auch mit dem Buchstaben **C**, groß oder klein, anfängt, wie der Klassenname. Sie liefert einen entsprechenden boolean-Wert zurück.

Aufgabe 5

Erstellen Sie die Klasse **ArtikelMain** zum Testen der erstellten Klassen und Methoden. Der eigentliche Testablauf in dieser Klasse soll folgendermaßen aussehen:

Zuerst wird ein Array von 3 Artikeln deklariert und angelegt.

Der erste Artikel in diesem Array sei ein Buch mit den Daten:

"B1234", "Warum", 38.00, 2000.

Der zweite Artikel in diesem Array sei eine DVD mit den Daten:

"d55", "Center Stage", 34.80, 111.

Der dritte Artikel in diesem Array sei ein ComputerBuch mit den Daten:

"F3000", "Alle Java-Klausuren", 999.99, 2001, "Java" .

Der Preis der DVD wird mit der entsprechenden Methode um 5 % erhöht.

Danach erfolgt eine Kontrollausgabe.

Die Spieldauer der DVD wird mit der entsprechenden Methode in Stunden und Minuten ausgegeben. (Vorsicht!)

Zum Schluss wird das Array mit einer Schleifenanweisung durchlaufen.

Dabei werden von jedem Array-Element sämtliche Daten ausgegeben und ob die Artikelnummer korrekt ist.

Aufgabe 6

Erstellen Sie die Klasse **SonderPaket**. Ein Sonderpaket ist einfach eine Zusammenfassung von Artikeln, die dann also im Paket preisgünstiger verkauft werden. Ein Sonderpaket enthält als Instanzvariable einen Array von Artikeln. Diese Instanzvariable habe den Namen **inhalt**.

Ergänzen Sie die Klasse um folgende Eigenschaften, bzw. Methoden:

- setter- und getter-Methoden
- einen Konstruktor mit einer Parameterliste, so dass die Instanzvariable initialisiert werden kann.
- eine Methode **preisBerechnen()**. Diese Methode soll den Preis des Pakets ausrechnen und als double-Wert zurückliefern. Der Preis berechnet sich wie folgt:
Summe der Preise der einzelnen Artikel minus 15%.

Ergänzen Sie noch die Klasse ArtikelMain am Ende um einen Test der Klasse SonderPaket: Aus den drei bereits vorhandenen Artikeln soll ein SonderPaket gebildet und dessen Preis ausgegeben werden.
