

Elektronikschule Tettnang	Softwareentwicklung	Datum:	Klasse:
	Beziehungen zwischen Klassen Teil 2	Name:	

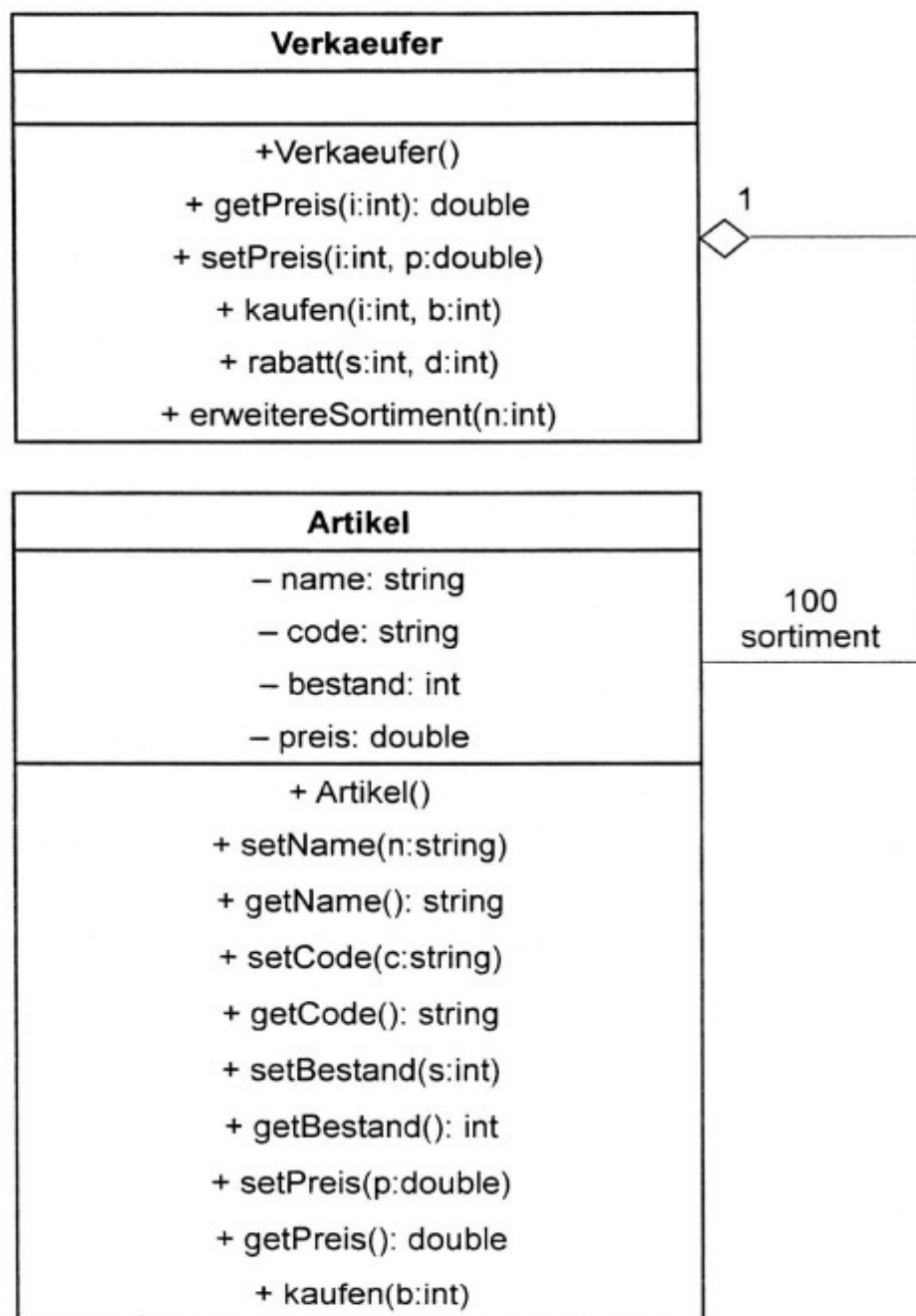


Aufgabe 1:

Sie sind Mitarbeiter der Firma eComConsult GmbH, die sich auf Hard- und Software-spezifische Problemstellungen von Online-Händlern spezialisiert hat. Ein neuer Kunde beauftragt Ihre Firma mit der ganzheitlichen Unterstützung und Beratung bei der Gründung eines neuen Online-Vertriebs für IT-Produkte.

Die Warenwirtschafts-Software des zukünftigen Online-Vertriebs soll programmiert werden. Der Projektleiter beauftragt Sie als Teammitglied mit der Implementierung der Software-Bestandteile für das Artikel-Sortiment und dessen Preisgestaltung.

Er übergibt Ihnen folgendes UML-Klassendiagramm als Vorgabe:



Elektronikschule Tettnang	Softwareentwicklung	Datum:	Klasse:
	Beziehungen zwischen Klassen Teil 2	Name:	

a)

Erstellen Sie für die Klasse **Artikel** die Methode **kaufen**, die zur Erhöhung des Bestandes eines Artikels dienen soll. Hierzu wird ihr ein ganzzahliger Parameter übergeben, um den der Bestand eines Artikels erhöht wird.

b)

Erstellen Sie eine Klasse **Verkaeuer** (siehe UML-Klassendiagramm). Die Klasse **Verkaeuer** soll die folgenden Methoden besitzen:

Einen Konstruktor **Verkaeuer**, der sämtliche Elemente des Arrays **sortiment** mit Objekten vom Typ **Artikel** initialisiert.

Eine Methode **getPreis**, die als Parameter einen ganzzahligen Wert *i* erhält. **getPreis** soll als Rückgabewert den Preis des Artikels liefern, der innerhalb des Arrays **sortiment** den Index *i* hat.

Eine Methode **setPreis**, die als Parameter einen ganzzahligen Wert *i* und den Fließkommawert *p* erhält. **setPreis** soll den Preis des Artikels, der innerhalb des Arrays **sortiment** den Index *i* hat, auf *p* setzen.

Eine Methode **kaufen**, die als Parameter zwei ganzzahlige Werte *i* und *b* erhält. **kaufen** soll den Bestand des Artikels, der innerhalb des Arrays **sortiment** den Index *i* hat, um *b* erhöhen.

Eine Methode **rabatt**, die den Preis jedes Artikels innerhalb des Arrays **sortiment**, dessen Bestand größer als *s* Exemplare ist, um den Prozentwert *d* reduziert.

Eine Methode **erweitereSortiment**, die das aktuelle Array **sortiment** um *n* Elemente erweitert. Übernehmen sie vorhandene Artikel und initialisieren sie die neu hinzugekommenen Elemente mit Objekten vom Typ **Artikel**.

Hinweise:

Um die Ausgabe in einem geeigneten Währungsformat müssen Sie sich nicht kümmern.
Deklarieren und Implementieren Sie jeweils vollständig die Klasse und die Methoden.
Sie können davon ausgehen, dass keine ungültigen Parameter übergeben werden.