

**Aufgabe 45:** (2+3=5 Punkte) Modellieren und implementieren Sie Klassen, die im Kontext eines Zooverwaltungssystems verwendet werden können. In einer ersten Anforderungsanalyse wurde erkannt, dass dieses System drei Arten von Dingen speichern können soll: Tiere, Habitate und Tierpflegepersonal.

Zu einem Tier sollen die folgenden Informationen gespeichert werden können:

- sein Namen
- seine Rasse
- sein Geschlecht
- sein Futter
- seine Fütterungszeit
- sein Habitat

Zu einem Habitat sollen die folgenden Informationen gespeichert werden können:

- seine Größe in  $m^2$
- sein Typ (Käfig/Aquarium/...)
- die Anzahl seiner Türen
- den zuständigen Tierpfleger

Zu einem Tierpfleger sollen folgende Informationen gespeichert werden können:

- sein Name
- sein Vorname
- seine Adresse

Darüber hinaus soll den Tieren, Habitaten und Tierpflegern jeweils eine eindeutige Nummer zugeordnet werden. (Um die Vergabe dieser Nummern soll es jedoch in dieser Aufgabe nicht gehen.)

**Hinweis:** Für (a) und (b) sind jeweils separate Lösungen abzugeben.

- (a) Geben Sie zu den oben abgebildeten Klassen ein Klassendiagramm in der UML an, das alle Attribute, Methoden und Beziehungen zwischen Klassen abbildet.

Implementieren Sie anschließend die Klassen in Java.

- (b) In einer zweiten Anforderungsanalyse ergibt sich jedoch die folgende Verfeinerung des Begriffs „Habitat“ aus der vorigen Aufgabe:

Es soll auf Typebene unterschieden werden können, ob ein Habitat ein Aquarium oder ein Gehege ist.

Erzeugen Sie entsprechende Klassen durch Vererbung. Überlegen Sie dazu welche der oben definierten Attribute noch auf den jeweiligen Habitattyp zutreffen oder spezifisch für einen der beiden Typen sind.

Ändern Sie die Klasse Habitat entsprechend und sorgen Sie dafür, das keine Objekte der Klasse Habitat mehr erzeugt werden können (Es gebe nur Aquarien oder Käfige).

Zu jedem Habitat soll seine Länge und Breite, zu jedem Aquarium zusätzlich seine Tiefe gespeichert werden. Außerdem soll den beiden verschiedenen Habitattypen (Käfig/Aquarium) eine Methode zur Verfügung stehen, die jeweils Auskunft über die zugehörigen Attribute gibt.

Modellieren und implementieren Sie die Klassen wie in Aufgabenteil (a) beschrieben.

---

**Aufgabe 46:** (BA, 1+2+2=5 Punkte) Machen Sie sich mit der Java-API (URL siehe Kapitel 08) vertraut und betrachten Sie die beiden Klassen `TextArea` und `TextField`.

- Zeichnen Sie die Vererbungshierarchie dieser beiden Klassen bis zur Wurzelklasse `Object`.
- Beschreiben Sie, wodurch sich diese beiden Klassen unterscheiden.
- Beschreiben Sie für jede dieser beiden Klassen, welche wesentlichen Eigenschaften/Verhaltensweisen von den Oberklassen geerbt werden. Geben Sie hierbei an, von welcher Oberklasse jeweils die geerbte Eigenschaft/Verhaltensweise ursprünglich vorgegeben wurde.

**Aufgabe 47:** (BA, 5 Punkte) Die im folgenden Text umgangssprachlich(!) beschriebene Pizzeria soll in UML modelliert werden.

- Eine Pizzeria verkauft Menüs.
- Ein Menü ist entweder ein Pizzamenü oder ein Nudelmenü.
- Bei einem Pizzamenü soll notiert werden können, ob es ein Mitnahmemenü ist. Es besteht aus einer Pizza, die einen Namen und einen Durchmesser hat, und einem Getränk.
- Ein Nudelmenü besteht aus einem Nudelgericht, das einen Namen hat, und einem Getränk.
- Ein Getränk soll einen Namen haben und keine oder genau drei Eiswürfel enthalten können.
- Pizza und Nudeln werden jeweils frisch erzeugt, können also nicht existieren, ohne zu einem entsprechenden Menü zu gehören. Eiswürfel und Getränke sind immer vorrätig.
- Zu einem Menü gehört immer genau eine Beilage. Eine Beilage ist entweder ein kleiner Salat oder eine Tüte Pizzabrötchen.

Zeichnen Sie ein UML-Diagramm und begründen Sie unter Bezug auf die obigen Beschreibung die Arten der von Ihnen modellierten Beziehungen zwischen Ihren Klassen. Es sind nur die in der obigen Beschreibung gegebenen Attribute und Beziehungen abzubilden. Für die Attribute sind zusätzlich Selektoren anzugeben; für eventuelle Assoziationen oder Kompositionen ist dies nicht notwendig.

**Aufgabe 48:** (5 Punkte) Modellieren Sie ein Schaltnetz, d.h. eine Kombination mehrerer logischer Schaltbausteine unter Verwendung der im LearnWeb bereit gestellten Klassen. Sie können sich auf die Modellierung von AND-, OR- und NOT-Bausteinen beschränken. Geben Sie ein UML-Klassendiagramm an und implementieren Sie die modellierten Klassen in Java. Nutzen Sie hierbei soweit wie möglich die in Kapitel 11 vorgestellten objektorientierten Konstrukte; hierdurch entsteht ggfs. die Notwendigkeit, neue Klassen hinzuzufügen.

Erläutern Sie die von Ihnen getroffenen Modellierungsentscheidungen und kommentieren Sie Ihre Programmtexte.

**Hinweis:** Ihre Implementierung muss mindestens die in der Rahmendatei vorgegebenen Tests bestehen, die die nachfolgend angegebenen Schaltnetze beschreiben.

