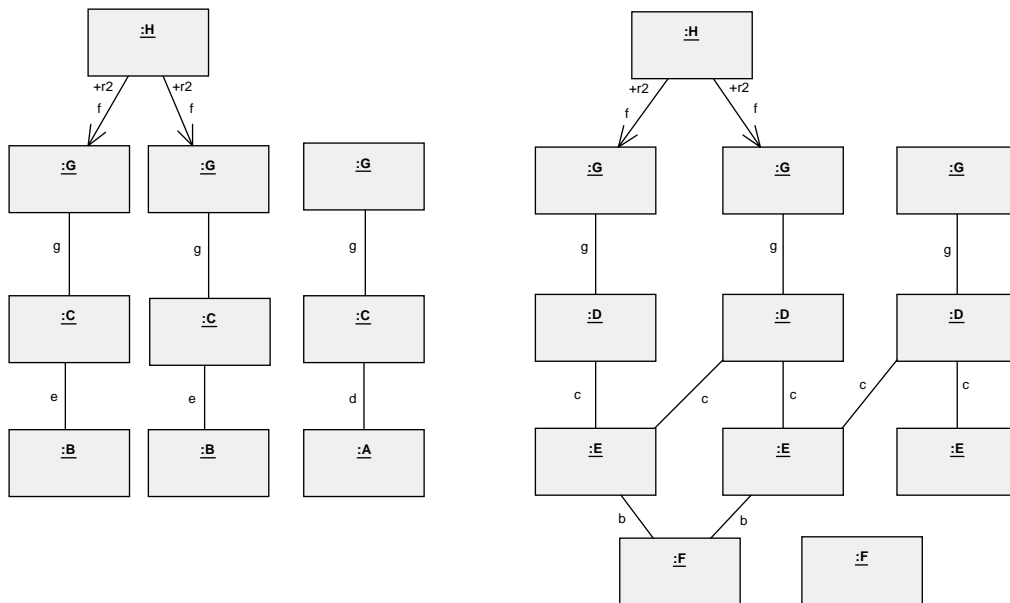


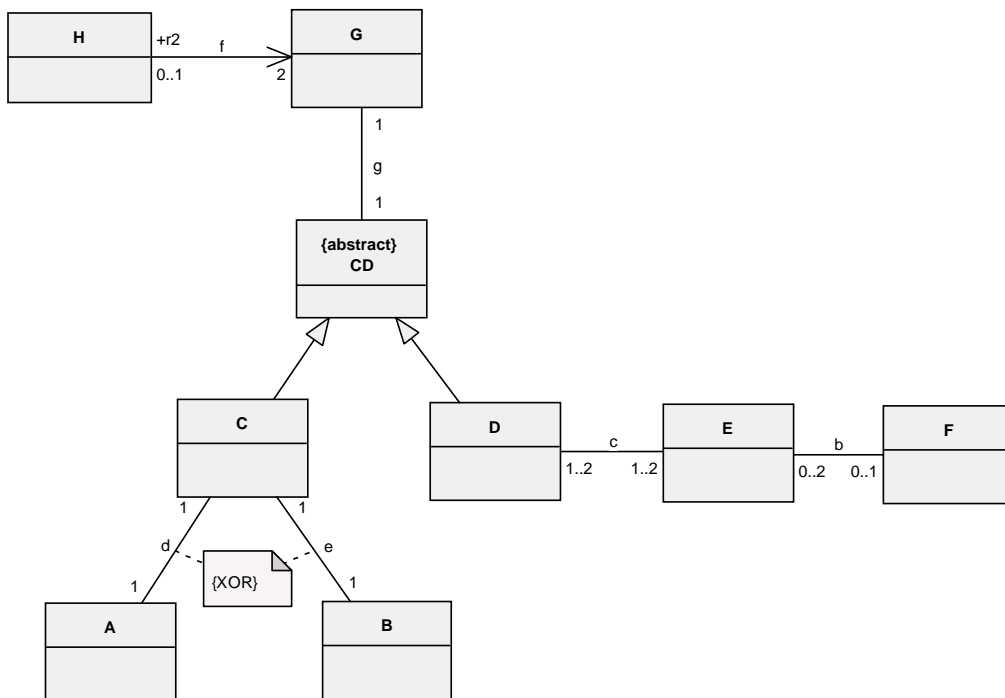
Aufgabe 1: Objektdiagramm

Entwerfen Sie zwei Klassendiagramme, zu denen nachfolgende Objektdiagramme konform sind. Wählen Sie die Kardinalitäten an den Assoziationsenden möglichst genau. Sie können davon ausgehen, dass diese Objektdiagramme die höchstzulässige Anzahl an Beziehungen mit Objekten einer anderen Klasse darstellen. Weiters sollen mögliche Generalisierungen bzw. XOR-Beziehungen erkannt werden.

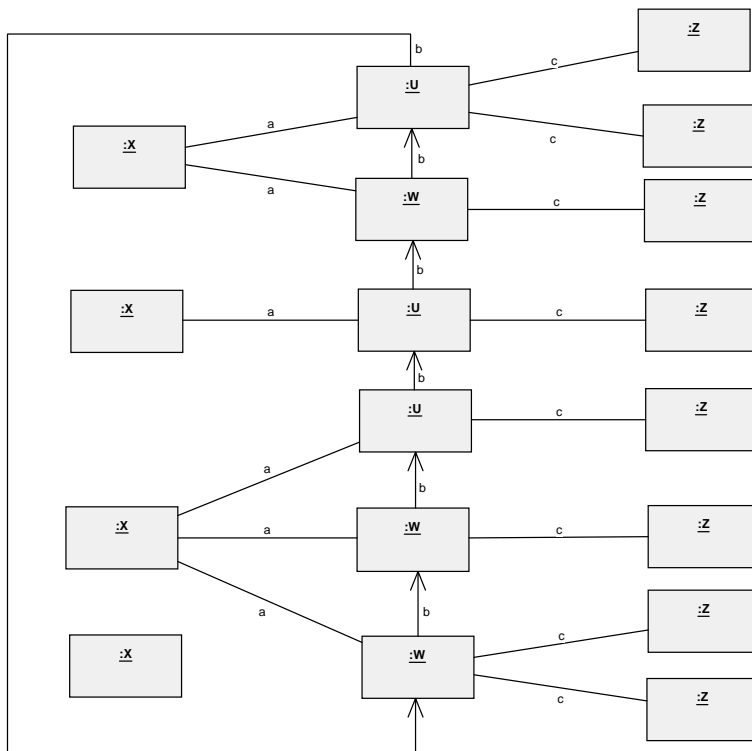
• Objektdiagramm 1:



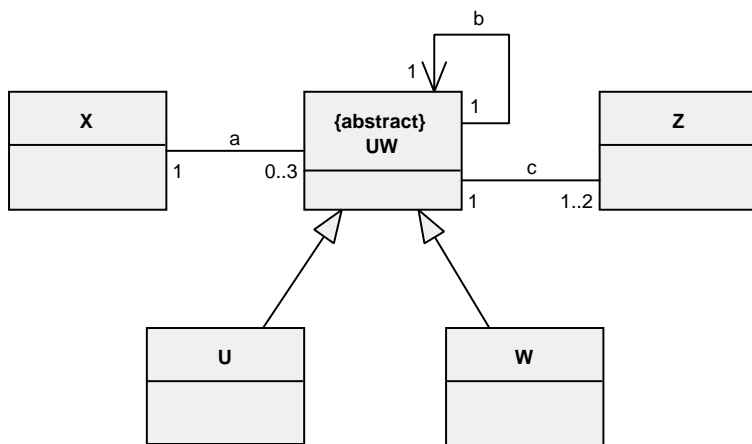
Klassendiagramm 1:



• Objektdiagramm 2:



Klassendiagramm 2:



Aufgabe 2: Vergleich von Klassendiagrammausschnitten

Erklären Sie den Unterschied zwischen folgenden Klassendiagrammausschnitten.

a)



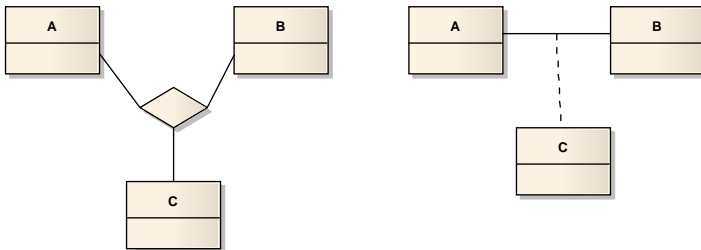
b)



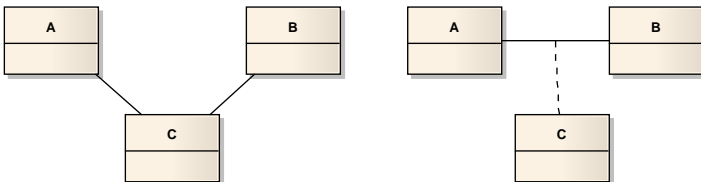
c)



d)



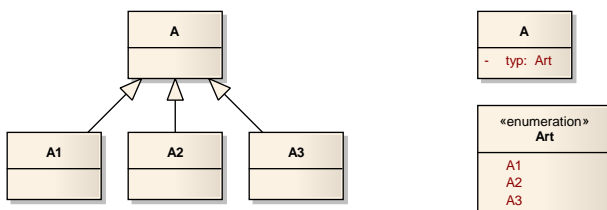
e)



f)



g)



Aufgabe 3: Reverse Engineering

Gegeben sei der unten angeführte Java ähnliche Code. Führen Sie ein Reverse Engineering des Codes in ein UML Klassendiagramm durch. Das heißt, Sie müssen ein UML Klassendiagramm entwerfen, das semantisch dem Java Code entspricht. Bilden Sie Referenzen möglichst durch Assoziationen ab.

```
abstract class Person {
    private String vorname;
    private String nachname;

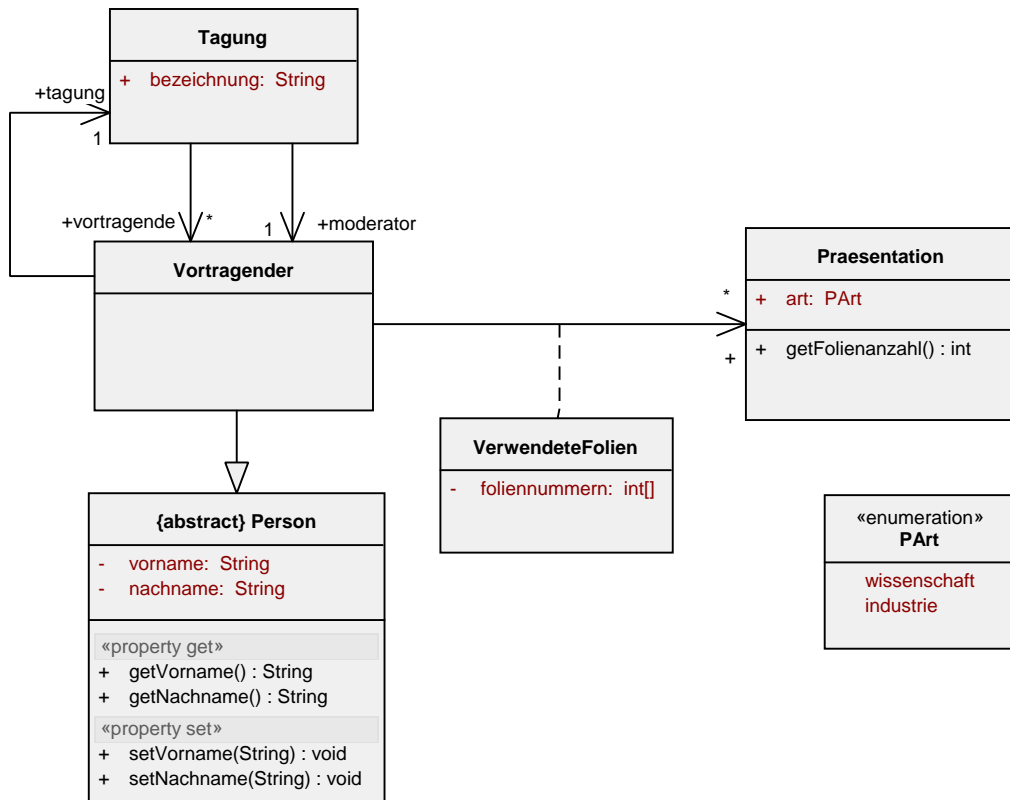
    public String getVorname() {
        return vorname;
    }
    public void setVorname(String vorname) {
        this.vorname = vorname;
    }
    public String getNachname() {
        return nachname;
    }
    public void setNachname (String nachname) {
        this.nachname = nachname;
    }
}

class Vortragender extends Person {
    public Tagung tagung;
    public Hashtable verwendeteFolien;
        // Key: Praesentation (Typ: Praesentation)
        // Value: foliennummern (Typ: int[])
}

class Praesentation {
    public PArt art;
    public int getFolienanzahl() {...}
}

class Tagung {
    public String bezeichnung;
    public Vortragender moderator;
    public Vortragender [] vortragende;
}

Enumeration PArt {
    wissenschaft;
    industrie;
}
```



Aufgabe 4: Fitnessclub

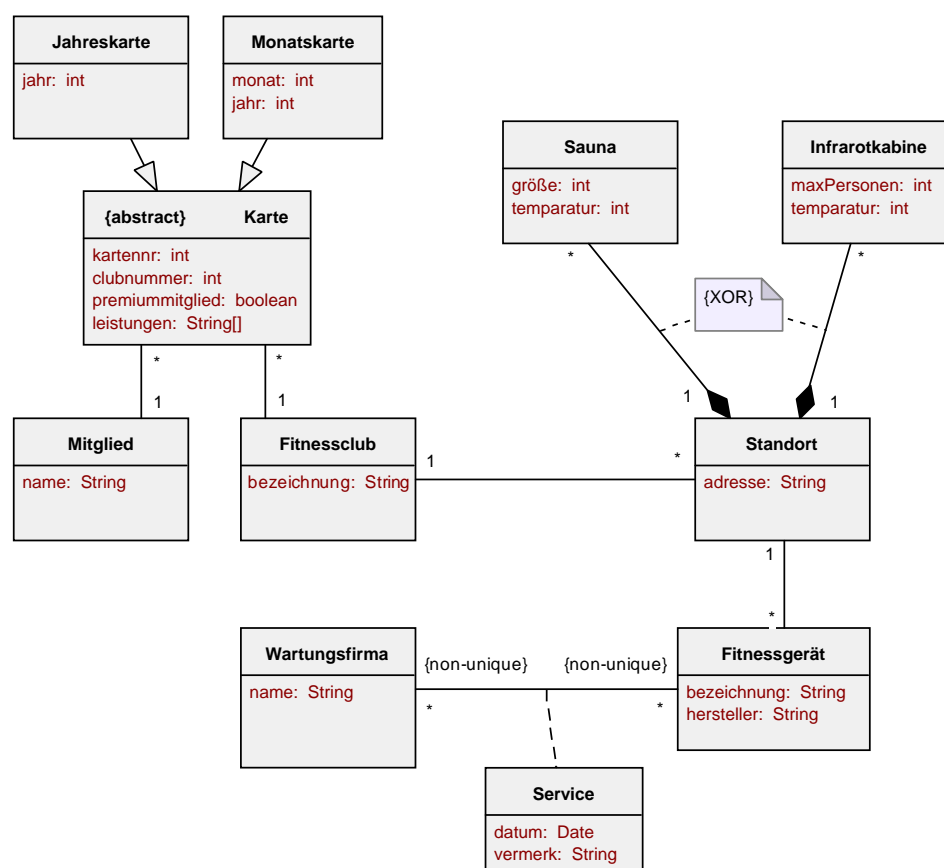
Modellieren Sie folgenden Sachverhalt mittels UML Klassendiagramm gemäß folgenden Informationen:

Jeder Fitnessclub hat eine Bezeichnung. Ein Fitnessclub verfügt über mehrere Standorte, ein Standort gehört zu genau einem Fitnessclub. Jeder Standort hat eine Adresse. Ein Standort verfügt über mehrere Fitnessgeräte, jedes Fitnessgerät gehört einem Standort. Fitnessgeräte haben eine Bezeichnung und einen Hersteller. Ein Fitnessgerät kann im Laufe der Zeit von mehreren Wartungsfirmen überprüft werden, eine Wartungsfirma ist für die Überprüfung von mehreren Fitnessgeräten zuständig. Das Datum und ein Vermerk zum durchgeführten Service wird für jedes bestimmte Fitnessgerät das von einer bestimmten Wartungsfirma überprüft wurde, gespeichert.

Weiters kann ein Standort eines Fitnesscenters entweder über mehrere fix eingebaute Saunen oder über mehrere fix eingebaute Infrarotkabinen verfügen. Von einer Sauna kennt man die Größe und die Temperatur, von einer Infrarotkabine sind die maximale Personenanzahl und die Temperatur bekannt.

Ein Mitglied eines Fitnessclubs hat einen Namen und eine Clubnummer. Es gibt auch Premiummitglieder, die spezielle Leistungen in Anspruch nehmen können. Jedes Mitglied besitzt eine Karte. Da man bei mehreren Fitnessclubs Mitglied sein kann, kann man im Besitz von mehreren Karten sein. Eine Karte gehört immer nur zu einem Mitglied und zu einem Fitnessclub. Ein Fitnessclub hat mehrere Mitglieder und ist daher im Besitz von mehreren Karten. Jede Karte hat eine eindeutige Kartenummer.

Bei Karten unterscheidet man zwischen genau zwei Arten: Jahreskarten und Monatskarten. Auf einer Jahreskarte ist das Jahr vermerkt, in dem die Karte gültig ist, bei einer Monatskarte sind Monat und Jahr vermerkt.



Aufgabe 5: Eventmanagement I

Folgendes Beispiel ist Ihnen bereits in einer ähnlichen Form aus der LVA Datenmodellierung bekannt:

Sie werden von einer Eventmanagement-Firma mit dem Entwurf eines Verwaltungssystems beauftragt. Bilden Sie die folgenden Sachverhalte in einem Klassendiagramm ab:

Eine Veranstaltung wird über eine ID (vid) identifiziert. Sie besitzt eine Bezeichnung (bez), eine Beschreibung (beschreibung) sowie einen Anfangs- (von) und einen Endzeitpunkt (bis). Die Dauer (dauer) der Veranstaltung berechnet sich aus der Differenz von End- und Anfangszeitpunkt.

Von dem System sollen auch spezielle Veranstaltungen erfasst werden können: Ein Film ist eine Veranstaltung und spielt in einem Genre – Komödie, Thriller oder Horror, andere Genres gibt es nicht. Außerdem werden auch noch der Regisseur (regisseur) und die Altersgrenze (fsk) vermerkt. Zu einem Konzert wird vermerkt, ob dieses ausverkauft ist oder nicht. An einem Konzert können maximal drei Bands teilnehmen, jede Band kann bei mehreren Konzerten auftreten. Eine Band hat einen Namen (name) und ein bestimmtes Genre – Pop, Rock oder Indie. Von einem Theaterstück weiß man den Namen der Theatergruppe (theatergruppe).

Eine Veranstaltung kann in genau einem Saal stattfinden, wobei in einem Saal mehrere Veranstaltungen stattfinden können. Ein Saal wird über eine ID (sid) identifiziert. Zusätzlich wird auch noch eine Bezeichnung (bez) zu dem Saal vermerkt. Ein Saal hat beliebig viele Plätze. Ein Platz kann innerhalb eines Saals über eine Platznummer (pnr) eindeutig identifiziert werden. Außerdem wird der Platzpreis (preis) gespeichert.

Modellieren Sie diesen Sachverhalt mittels UML Klassendiagramm.

Aufgabe 6: Eventmanagement II

Erweitern Sie das Klassendiagramm aus Aufgabe 5 wie folgt:

Zu einer Veranstaltung kann optional ein Poster vorhanden sein. Jedes Poster besitzt eine bestimmte Länge und eine bestimmte Breite.

Von dem Eventmanagement-System sollen verschiedene Veranstaltungsorte erfasst werden können. Ein Veranstaltungsort wird über seinen Namen (name) eindeutig identifiziert und umfasst mehrere Säle. Ein Saal ist Teil von genau einem Veranstaltungsort.

An einem Veranstaltungsort arbeiten mehrere Mitarbeiter, ein Mitarbeiter wird für mehrere Veranstaltungsorte eingesetzt. Ein Mitarbeiter wird über seine Sozialversicherungsnummer (syvr) eindeutig identifiziert. Außerdem werden noch Vorname (vname) und Nachname (nname) vermerkt. Für jeden bestimmten Mitarbeiter der an einem bestimmten Veranstaltungsort arbeitet, wird die Bezeichnung (bezeichnung) der ausgeführten Funktion gespeichert.

Lösung für Beispiel 5 und 6:

