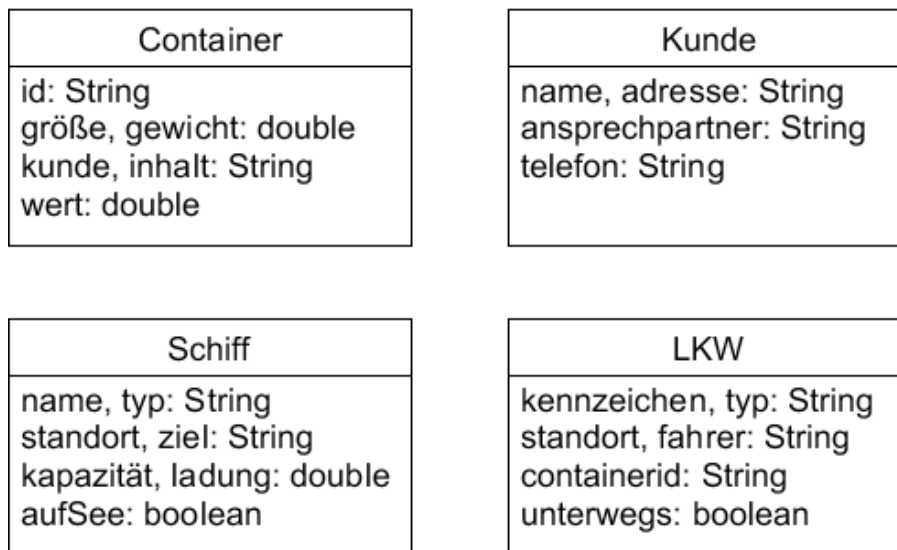


Aufgabe 1

a) **Klassendiagramm** mit Java-Datentypen:



b) **Implementierung**

```
class Container
{
    String id;
    String kunde, inhalt;
    double groesse, gewicht;
    double wert;
}
```

```
class Schiff
{
    String name, typ;
    String standort, ziel;
    double kapazität, ladung;
    boolean aufSee;
}
```

```
class Kunde
{
    String name, adresse;
    String ansprechpartner;
    String telefon;
}
```

```
class LKW
{
    String kennzeichen, typ;
    String standort, fahrer;
    String containerId;
    boolean unterwegs;
}
```

Beachte:

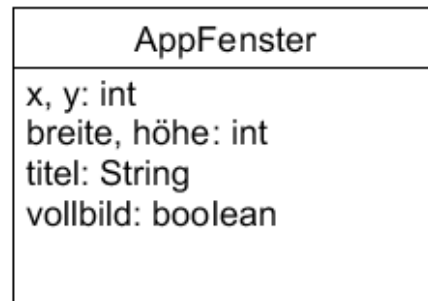
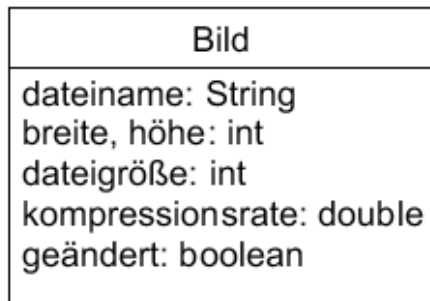
Im Klassendiagramm die Datentypen HINTER den Attributnamen

In der Implementierung die Datentypen VOR den Attributen, am Ende Semikolon.

Klassennamen mit Großbuchstaben, Attributnamen und einfache Datentypen mit Kleinbuchstaben.

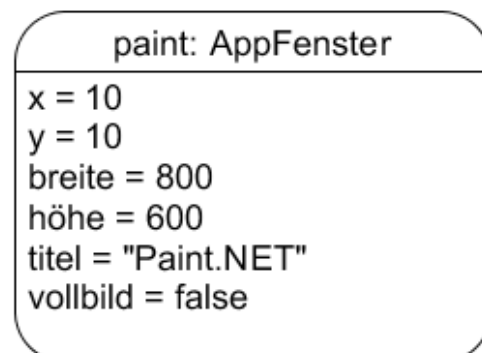
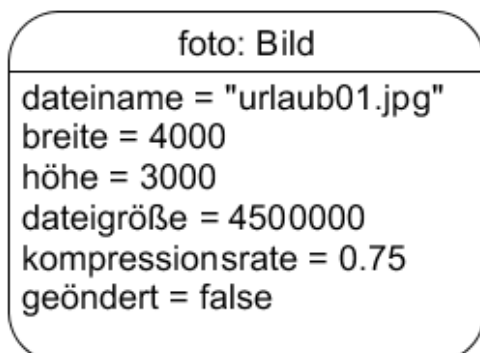
Aufgabe 2

- a) **Klassendiagramm** (mit Java-Datentypen)
- b) Zu jeder Klasse ein Beispielobjekt in Form eines **Objektdiagramms**



Klassendiagramm

Objektdiagramm



Beachte:

Klassen und Objekte nicht vermischen, sind zwei unterschiedliche Diagramm-Arten.

Klassennamen mit Großbuchstaben, alles andere mit Kleinbuchstaben

Objekte haben Namen, dahinter steht ein Doppelpunkt und der Klassenname.

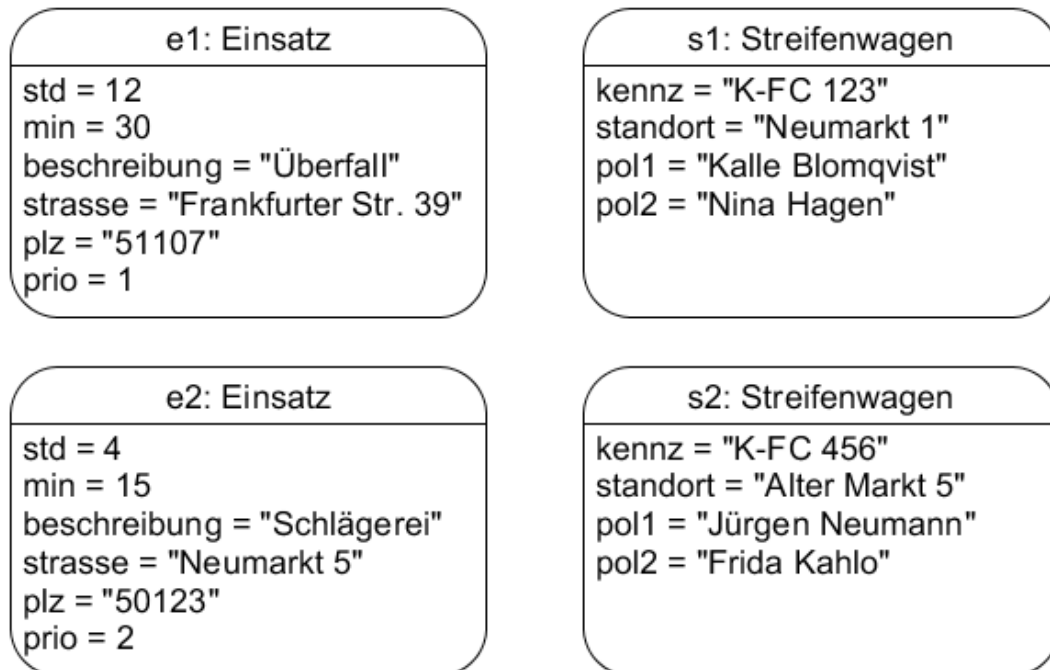
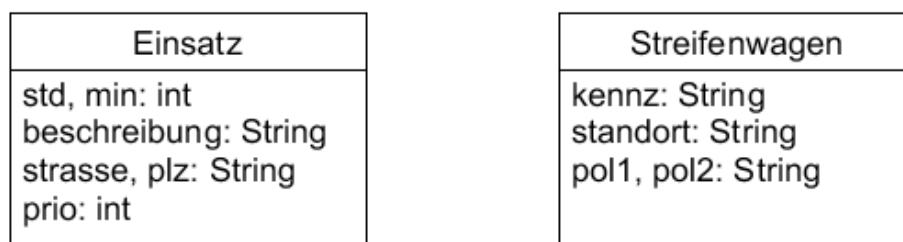
Texte als Attributwerte, z.B. "urlaub01.jpg" müssen in Gänsefüßchen stehen.

Im Klassendiagramm können Attribute mit Komma aufgelistet werden.

Im Objektdiagramm muss jedes Attribut einzeln mit Wert stehen.

Aufgabe 3 (Anwendungssituation: Abiturklausur 2015)

- a) Ein Zeitpunkt im Ablauf der Software als **Objektdiagramm**
 b) Zu den Objekten je eine Klasse in Form eines **Klassendiagramms**

*Objektdiagramm**Klassendiagramm***c) Implementierung**

```

class Einsatz
{
    int std, min;
    String beschreibung;
    String strasse, plz;
    int prio;
}
  
```

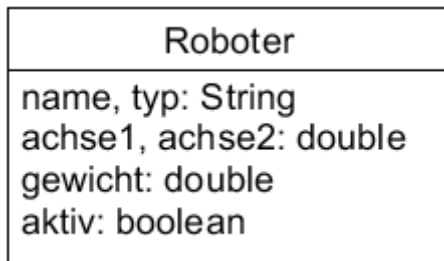
```

class Streifenwagen
{
    String kennz;
    String standort;
    String pol1, pol2;
}
  
```

Aufgabe 4

Korrigierte Version

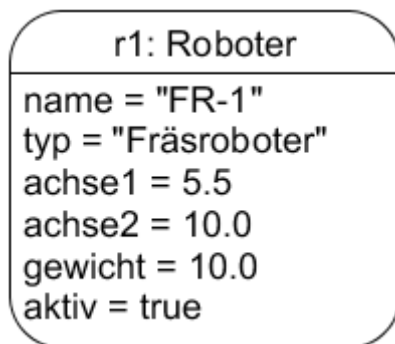
Klasse



Implementierung

```
class Roboter
{
    String name, typ;
    double achse1, achse2;
    double gewicht;
    boolean aktiv;
}
```

*Objekt
(Beispiel)*



Beachte:

Im Klassendiagramm kein Semikolon, Klassenname ohne class, nur Java-Datentypen

Im Objektdiagramm String-Werte in Gänsefüßchen, keine Einheiten.

Das Objekt muss für alle Attribute, die die Klasse vorgibt, einen Wert angeben.