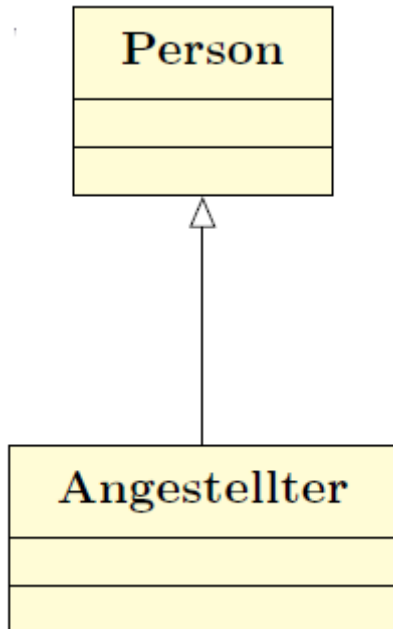


Programmieren 1 (PRG1)

Übung 9

Abbildung 1:

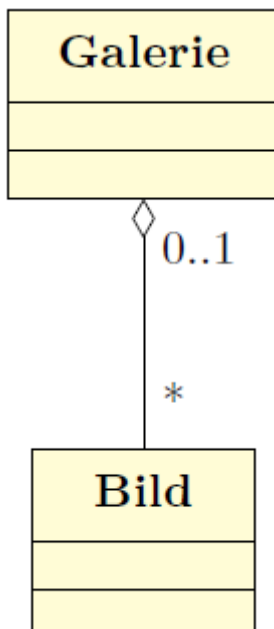


UML Diagramm I:

Das gezeigte Diagramm stellt eine Vererbung dar. Hierbei ist die Klasse „Person“ die Ober- und „Angestellter“ die Unterklasse.

Die Unterklasse erbt alle Attribute und Funktionen der Oberklasse.

Ein Beispiel für ein Python Programm wäre die Vererbung von Attributen und Funktionen von einer Klasse zu einer Subklasse.

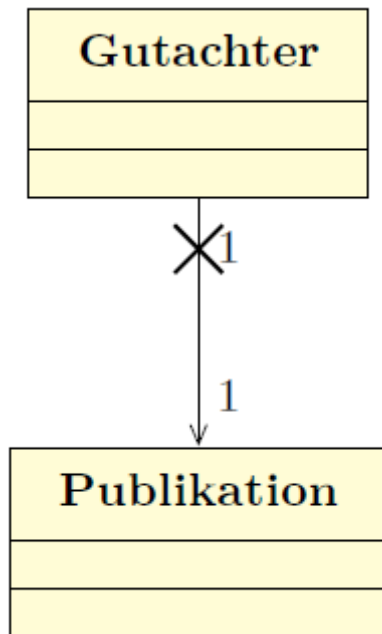


UML Diagramm II:

Das gezeigte Diagramm stellt eine Aggregation („Ganzes-Teile-Hierarchie“) dar. Hierbei besteht die Galerie aus Bildern. Die Bezeichnung 0..1 sagt aus, dass es entweder eine oder keine Galerie gibt.

Die Bezeichnung * sagt aus, dass es eine unbegrenzte Anzahl von Bildern gibt, die auch ohne eine Galerie existieren können.

Ein Beispiel in Python wären Funktionen innerhalb einer Klasse. Es Kann eine unbestimmte Anzahl an Funktionen innerhalb einer Klasse geben, aber keine Funktion kann Teil von zwei Klassen gleichzeitig sein. (Subklassen enthalten nur eine exakte Kopie der Oberklasse).



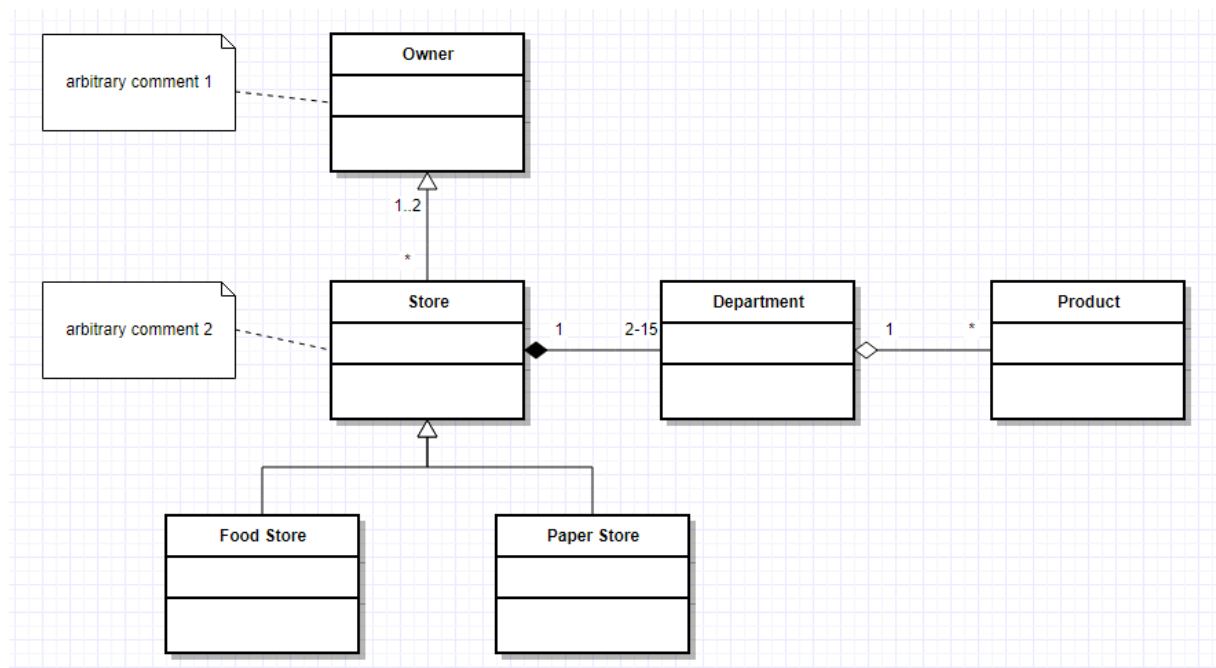
UML Diagramm III:

Dieses Diagramm stellt eine gerichtete Assoziation dar. Hierbei hat die Klasse „Gutachter“ Zugriff auf die Klasse „Publikation“, aber umgekehrt ist kein Zugriff möglich. Weiterhin sagt die jeweilige Bezeichnung 1 aus, dass genau ein Gutachter und eine Publikation existiert.

Ein Beispiel in Python wäre, dass die Klasse „Gutachter“ privat ist (Klasse `__Gutachter()`), während die Klasse „Publikation“ öffentlich ist (Klasse `Publikation()`).

Quelle: PRG1 Vorlesung; „V17-UML_in_Grundzügen“; Seite 33 – 40

9.2



Geschrieben mittels <https://go.gliffy.com/go/html5/launch>; Bild erstellt mittels Snipping Tool