

0.1 Klassendiagramme

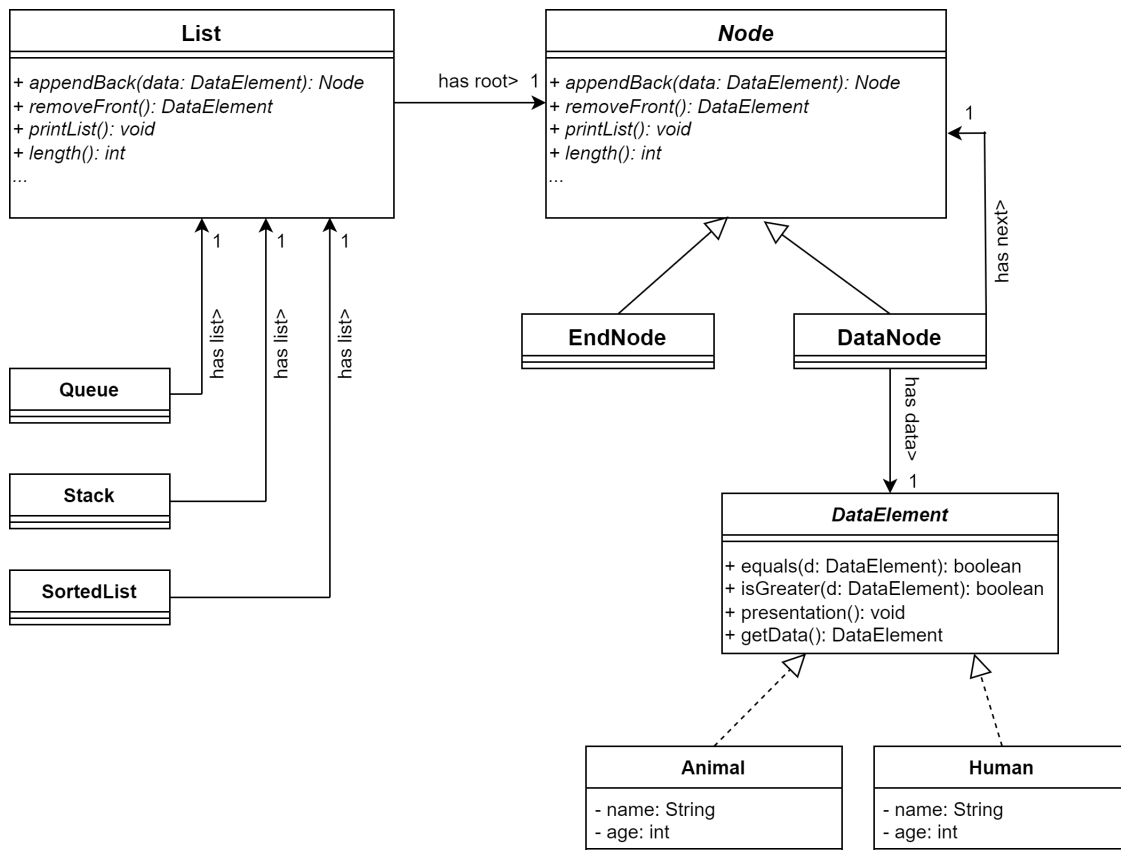
Ein Klassendiagramm enthält eine Klassenkarte für alle beteiligten Klassen, sowie Pfeile zwischen ihnen, um die Beziehungen zu veranschaulichen. Für das Abitur wird noch zwischen erweiterten Klassendiagrammen und „normalen“ Klassendiagrammen unterschieden. Das reguläre Diagramm enthält:

- Attribute (nur Bezeichner)
- Methoden mit den Übergabeparametern (Bezeichner)
- Beziehungen zwischen Klassen, Referenzen werden auf Verbindungen modelliert, inklusive Mengenangaben.
- i.d.R. ohne Geben- und Setzen- Methoden

Für das erweiterte Klassendiagramm zusätzlich:

- Datentypen der Attribute
- ggf. Referenzattribute
- Rückgabewerte der Parameter
- Konstrukturen

Vererbungen werden als Pfeile mit nicht-ausgemalter Spitze dargestellt. Abstrakte Klassen und Interfaces werden kursiv dargestellt oder (einfacher zu zeichnen) mit <<abstract>> bzw. <<interface>> gekennzeichnet. Werden Interfaces implementiert, so sieht der Pfeil aus wie bei einer Vererbung, allerdings gestrichelt. Folgendes Beispiel aus dem Skript enthält die meisten relevanten Elemente als Beispiel.

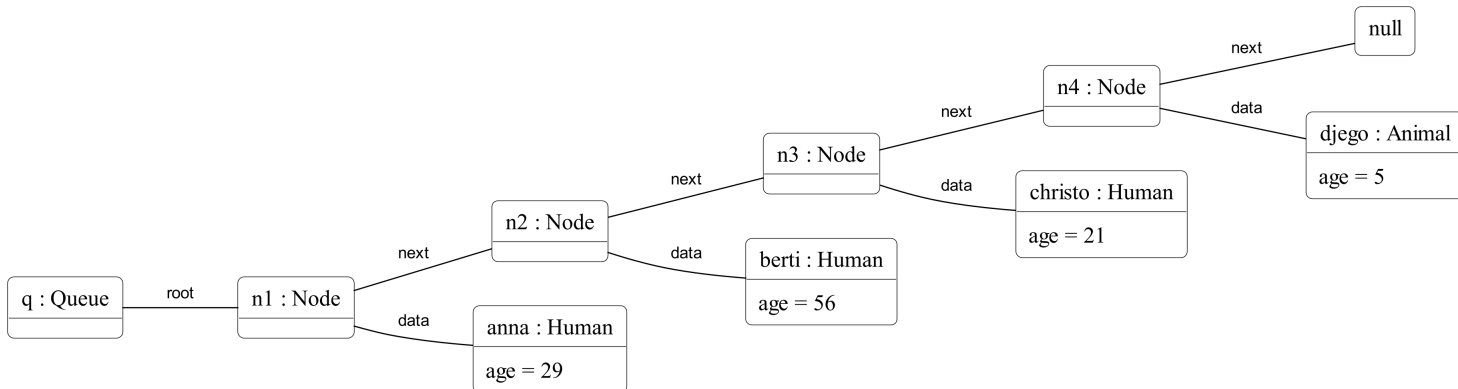


Hinweis: Im Zweifelsfall schadet mehr Information im Diagramm nicht. Das erweiterte Klassendiagramm ist zu bevorzugen.

0.2 Objektdiagramme

Im Gegensatz zum Klassendiagramm, das die allgemeine Struktur der Implementierung darstellen soll, stellt das Objektdiagramm den Zustand des Systems zu einem bestimmten Zeitpunkt dar.

Das folgende Beispiel zeigt den Zustand einer Liste zu einem bestimmten Zeitpunkt. Jedes Objekt wird mit einem Kasten mit abgerundeten Ecken dargestellt. Es wird mit <Bezeichner>:<Klassenname> beschriftet. Die Beziehungen werden wiederum auf den Verbindungslinien dargestellt, die jeweiligen Attribute mit ihren aktuellen Werten in einem separaten Abschnitt.



0.3 Sequenzdiagramme

Die Aufgabe des Sequenzdiagramms ist es, Methodenaufrufe - insbesondere über mehrere Objekte hinweg - darzustellen. Jedes Objekt erhält dabei eine sogenannte Lebenslinie. Diese Linie stellt dar, wie lange das Objekt „lebt“, da es z.B. auch nur kurzzeitig erzeugt werden könnten.

Methodenaufrufe werden durch Pfeile dargestellt und der Methodenname wird über den Pfeil geschrieben. Die Rückgabe wird ebenfalls mit einem Pfeil dargestellt, dieses Mal mit dem Rückgabewert über ihm dargestellt. Der Rückgabepfeil wird häufig gestrichelt dargestellt.

Sobald ein Objekt aktiv ist, wird ein Aktivitätsbalken gezeichnet. (I.d.R. gestartet durch einen Methodenaufruf und beendet durch eine Rückgabe).

Relevante Attribute können im abgerundeten Kasten am Beginn der Lebenslinie eingetragen werden. Beispiel aus dem Skript: Längenbestimmung einer verketteten Liste.

