Generische Klassen:

Im Kapitel 7 hat man das Konzept von verkette Liste kennegelernt

Diagram

Description automatically generated

So eine Liste ist praktisch, wie kommen wir zu einer Variante, in der wir auch z.B. Objekte vom Typ Schaf ablegen können?

Frage: Wie können wir eine Liste schaffen, in der wir beliebige Objekte (z.B. Tiere, Artikel, Personen, …) ablegen können?

🡪 Idee: Wir verallgemeinern die Liste, so dass diese all Objekte vom Typ Objekt annimmt. Auf Grund der Vererbung sind alle Objekte Zuweisungskompatibel und können abgelegt werden.

**Die Liste kann jedes Objekt aufnehmen:**

► List liste = new List();

liste.append(new Schaf("Schaf 1"));

liste.append(new Schaf("Schaf 2"));

**Problem**: Wenn man ein Objekt aus der Liste holt, ist unklar, welche Art von Objekt es ist (dynamischer Typ!)

Die getValueAt(int index)-Methode liefert ein Objekt vom (statischen) Typ Object. → Typ-Casts sind notwendig.

Bsp.

// irgendein anderer Programmierer (irgendwo im Quellcode)

liste.append(new Kuh("Kuh 1"));

…

// irgendwann später im Programm:

Schaf s = (Schaf)list.getValueAt(2);

// → ClassCastException zur Laufzeit!

**Lösungsidee:**

Anstatt überall Objekt zuschreiben nutzen wir eine Typvariable für den Datentyp.

Beim Anlegen eines Objektes bestimmen wir den konkreten Datentyp für die Typvariable.

Bei anderen Sprachen gibt es ähnliche Konzepte, z.B. Templates in C++.