

**Aufgabe 2**

(a)

muss Interface Serializable implementieren.

statische und als transient gekennzeichnete Attribute werden nicht serialisiert.

(d)

SimpleStringProperty ist eine Klasse, aber sie implementiert kein Interface Serializable. Um ein Objekt einer Klasse zu Serialisieren, muss die Klasse Interface Serializable implementieren.

(f)

Nach der Veränderung der Klassen kann man nicht mehr alte Objekte deserialisieren.

Um einen Fehler zu provozieren, zieht man ein Attribut in der Klasse aus und generiert eine neue serialVersionUID. Danach tritt ein Fehler InvalidClassException bei der Deserialisierung auf.

Um den obigen erwähnten Fehler zu verhindern, muss man serialVersionUID der Klasse nicht ändern.

(g)

serialVersionUID ist ein Merkmal einer Klasse, die auch in einem serialisierten Objekt gespeichert wird. Dadurch kann man erkennen, zu welcher Klasse ein serialisiertes Objekt gehört. Jede Veränderung der Klasse hat eine Veränderung der serialVersionUID zur Folge, falls sie neu erzeugt wird.

(h)

überschreibe clone() API zur flachen Kopie, d.h. primitive Felder werden kopiert, aber für Objekte werden nur Referenzen kopiert.

Bei der tiefen Kopie muss man selbst alle Attribute in dem Objekt in ein mit new Schlüsselwort erstelltes Objekt zuweisen.

**Aufgabe 3**

(c)

Binär

Speichert die Information eines Objektes in ByteCode.

XML

getter, setter und Standardkonstruktor. (JavaBeans)

Speichert die Information in Character, in Java besitzt ein Character zwei Bytes.

(d)

Binäre Serialisierung Vorteile:

verbraucht weniger Speicher.

schneller bei der Verarbeitung.

kann nicht durch dritte verändert werden.

einfach zu bedienen.

Binäre Serialisierung Nachteile:

nicht lesbar, man braucht spezielle Editoren.

kann nicht verändert werden.

XML Vorteile:

man kann die Datei einfach mit einem Browser oder Text-Editor lesen und den Wert eines Attributes verändern.

Keine Änderung an den Klassen nötig. (XStream)

Fast analog zur binären Serialisierung.

Den Beizeicher in Spitzeklammern kann man umbenennen. (XStream)

XML Nachteile:

Standardkonstruktor, getter und setter nötig. (Java Beans)

verbraucht viel Speicher wegen dem ausgeschlossenen Tags.