

## Fragen, Antworten, Kommentare

Erinnerung: Melden Sie sich für alle Prüfungsleistungen an, die Sie im Winter ablegen wollen. Denken Sie daran, dass ein Schieben einer Klausur zwar theoretisch bedeutet, dass man einen Versuch aufsparen kann, praktisch es aber keinen Effekt hat. Studierende Personen die ein oder zweimal schieben und dann zwei oder einmal durchfallen erkennen oft, dass das gewählte Studium nicht zu ihnen passt. Denken Sie daran, dass jedwede Prüfungsleistungen des dritten Semesters nur möglich sind, wenn Sie bis dahin 40 Leistungspunkte haben.

Fragen: Nach nochmaligem schauen der Videos sind mir zwei Unstimmigkeiten bei den Objektspeicherdiagrammen aufgefallen.

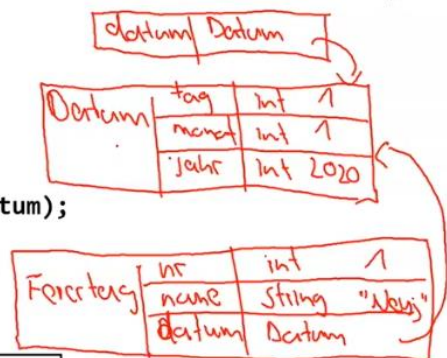
1. In diesem Video (<https://youtu.be/oC1pipelBc0> [1], Zeitstempel 5:13) wird der Wert des Strings direkt reingeschrieben während bei diesem Video (<https://youtu.be/osQV3CPRQGs> [2], Zeitstempel 15:31) der Wert des Strings noch einmal extra auf ein String-Objekt referenziert wird.

2. In diesem Video (<https://youtu.be/oC1pipelBc0> [3], Zeitstempel 7:52) werden direkt die Werte hingeschrieben, ohne den Typen int, wobei in anderen Beispielen stets auch der Typ mitgeschrieben wird.

Nun ist meine Frage wieso es unterschiedlich aufgeschrieben werden kann, bzw. ob es sich einfach nur um einen Fehler handelt. Und wenn es ein Fehler ist, welche Schreibweise ist die Richtige?

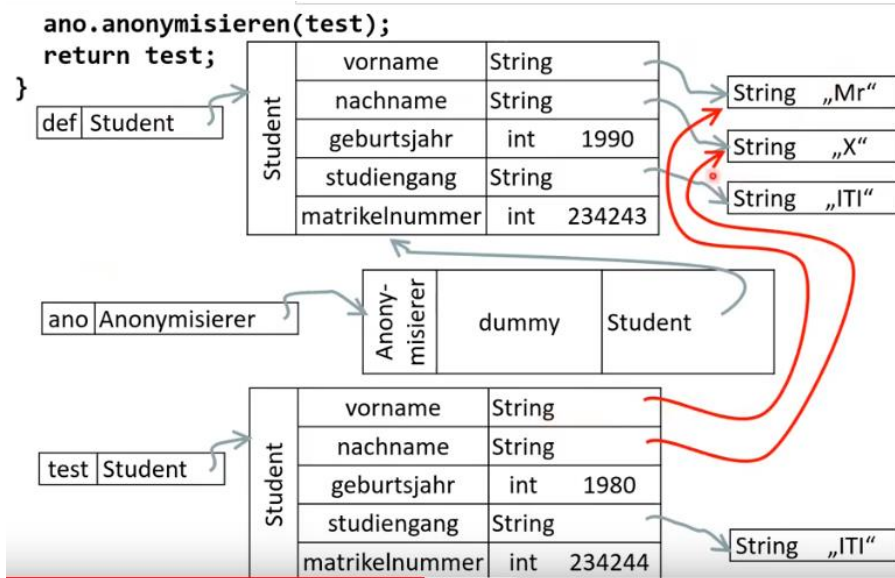
Antwort: Zunächst sehr präzise beobachtet.

```
class Analyse {
    EinUndAusgabe io = new EinUndAusgabe();
    void ersteSchritte(){
        Datum datum = new Datum(1, 1, 2020);
        this.io.ausgeben(datum.asText() + "\n");
        Feiertag f1 = new Feiertag(1, "Neujahr", datum);
        this.io.ausgeben(f1.asText() + "\n");
        datum.setJahr(2019);
        this.io.ausgeben(f1.asText() + "\n");
    }
}
```



```
1.1.2020
1: 1.1.2020 Neujahr
1: 1.1.2019 Neujahr
```

Die vorherige Abbildung zeigt deutlich, dass bei dem Objekt vom Typ Feiertag der String nicht als eigenständiges Objekt eingetragen ist, es keine Objektreferenz gibt. Dies ist falsch, sauber wäre ein eigenständig eingezeichnetes Objekt, wie es die nachfolgende Abbildung zu [2] zeigt. Nur so ist es möglich, dass mehrere Variablen den gleichen String referenzieren. Es gilt, sollte als Typ eine Klasse auftreten, muss das zugehörige Objekt immer als Referenz (Pfeil) eingezeichnet werden. Keine Pfeile stehen bei den elementaren Typen, die keine Klassen sind, also int, long, short, byte, double, float, boolean, char.

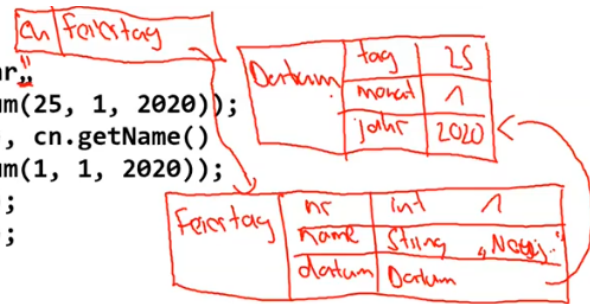


Als „Ausrede“ sei angemerkt, dass der Fehler für Strings minimal ist, da die Klasse die Eigenschaft hat, dass ihre Objekte nicht verändert werden können. Jede Methode also auch solche wie `replace()`, die das vom Namen her suggerierte Verhalten haben, liefern immer ein neues Objekt zurück. Sollten Sie in den Lösungen zur Praktikumsaufgabe die vereinfachte Darstellung bei Strings wählen, wäre das ok.

```

void verzahnteObjekte(){
    Feiertag cn = new Feiertag(1, "Neujahr",
        , new Datum(25, 1, 2020));
    Feiertag de = new Feiertag(cn.getNr(), cn.getName()
        , new Datum(1, 1, 2020));
    this.io.ausgeben(de.alsText() + "\n");
    this.io.ausgeben(cn.alsText() + "\n");
    de.setNr(0);
    cn.setName("Chinesisches Neujahr");
    this.io.ausgeben(de.alsText() + "\n");
    this.io.ausgeben(cn.alsText() + "\n");
}

```



```

1: 1.1.2020 Neujahr
1: 25.1.2020 Neujahr
0: 1.1.2020 Neujahr
1: 25.1.2020 Chinesisches Neujahr

```

Die vorherige Abbildung zeigt die Situation in [3] etwas später im Video. Die Vermutung ist korrekt, dass beim Datum schlichtweg die Typen vergessen wurden.

Die Darstellung im Skript ist nach meinem Erkenntnisstand konsistent und korrekt.