



Für eine Lagerverwaltung ist eine Klasse `DVD` zu erstellen. Eine Instanz dieser Klasse beinhaltet folgende Daten:

- `private String titel;` (speichert den Titel)
- `private double preis;` (speichert den Verkaufspreis)
- `private int anzahl;` (speichert die Stückzahl im Lager)

Eine Klassenvariable `gesamtAnzahl` speichert die insgesamt im Lager vorhandenen DVD's. Erstelle die Klasse `DVD` so, dass die gegebene Startmethode fehlerfrei arbeitet:

```
public static void main(String []args) {
    DVD dvd1 = new DVD("Alles geht!", 10.9, 5);
    DVD dvd2 = new DVD("Country Ballads", 9.5); // 1 Stueck
    DVD dvd3 = new DVD("Best of Johnny Cash", 9.5, 2);
    DVD []lager = {dvd1, dvd2, dvd3};
    System.out.format("Gesamzahl der DVDs: %d\n",
                      DVD.getGesamtAnzahl());
    System.out.format("Gesamtwert des Lagers: %5.2f\n",
                      DVD.gesamtWert(lager));
    dvd1.verkaufe(2);
    dvd2.verkaufe(3); // wirft eine StueckZahlException
    System.out.format("Gesamzahl der DVDs: %d\n",
                      DVD.getGesamtAnzahl());
    System.out.format("Gesamtwert des Lagers: %5.2f\n",
                      DVD.gesamtWert(lager));
}
```

Schreibe die Klassen `DVD` und `StueckZahlException` (unchecked Exception). Ergänze die Methode `main()` auch um entsprechende `try - catch` - Blöcke.

Beachte folgende Vorgaben:

- Der dreiparametrische Konstruktor übernimmt den Titel, den Verkaufspreis und den Anfangsbestand im Lager. Sollte dieser Wert kleiner als 1 sein, so ist eine `StueckZahlException` zu werfen. Der zweiparametrische Konstruktor setzt den Anfangsbestand auf 1 Stück.
- Die Methode `verkaufe(int n)` vermindert den Lagerbestand um `n` Stück. Wenn `n` kleiner als 1 ist bzw. wenn nicht genügend Stück dieser DVD auf Lager sind, so ist wieder eine `StueckZahlException` zu werfen und der Lagerbestand ist nicht zu verändern.
- Die Methode `getGesamtAnzahl()` liefert die Anzahl der insgesamt im Lager vorhandenen DVD's.

- Die Methode `gesamtWert()` berechnet und liefert den Wert aller im Lager vorhandenen DVD's.

Die oben gegebene Anwendung sollte also mit eingefügter Fehlerbehandlung z.B. die folgende Ausgabe liefern:

```
Gesamzahl der DVDs: 8
Gesamtwert des Lagers: 83.00 €
Fehler: Zu wenig Stück auf Lager
Gesamzahl der DVDs: 6
Gesamtwert des Lagers: 61.20 €
```

Schreibe eine Klasse `Tools` mit 2 Klassenmethoden:

- a) Eine Methode, die ein Integerarray `a` und einen boolean `even` erhält. Steht `even` auf `true`, so retourniert die Methode die Summe der Werte der geraden Zahlen im Array `a`, sonst die Anzahl der ungeraden Werte.
- b) Eine Methode, die ein Wort gemäß dem folgenden Schema in Zahlen verwandelt. Zunächst werden alle Buchstaben in Großbuchstaben umgewandelt und dann wird jeder Buchstabe durch seine Position im Alphabet ersetzt (A = 1, B = 2, ... ,Z = 26). Sonderzeichen und Umlaute brauchen nicht behandelt zu werden. Die Summe dieser Zahlen wird retourniert.

Beispiele:

```
oddOrEven(new int[]{3, 5, 8, 4, 7, 1}, true)    --> 12
oddOrEven(new int[]{3, 5, 8, 4, 7, 1}, false)   --> 16

summe("ein")          --> 28 (5+9+14)
summe("Informatik")   --> 116 (9+14+6+15+18+13+1+20+9+11)
```