

## „Grundlagen der Programmierung 1“, Wintersemester 2017/18

Das Projekt soll als Teil der Prüfungsvorleistung in den nächsten Wochen schrittweise durch jeweils zwei Studierende gemeinsam bearbeitet werden. Die Abgabe erfolgt als ZIP-Datei über den OPAL-Kurs. Die genauen Zeiten zur Abgabe entnehmen Sie dem Terminplan im OPAL-Kurs.

### Aufgabenstellung

Entwickeln Sie ein einfaches Informationssystem für ein Warenlager. Als Ausgangspunkt erhalten Sie ein Klassendiagramm, das im Rahmen eines Eclipse-Projektes in Java-Quellcode umzusetzen ist. Alle wesentlichen Funktionen sind im Folgenden näher beschrieben. Fragen Sie bei Unklarheiten rechtzeitig bei den Lehrenden nach.

### Grundlegendes

Legen Sie zusätzlich eine Klasse `Start` an, in der Sie die `main`-Methode wie bereits früher erläutert anlegen. In diese Methode schreiben Sie Quellcode, mit dem Sie ihre entwickelten Klassen ausprobieren, d.h. Sie legen Instanzen der Klassen an, rufen deren Methoden auf und prüfen so, ob die entwickelten Klassen korrekt funktionieren.

### Funktionsbeschreibung der Lagerverwaltung

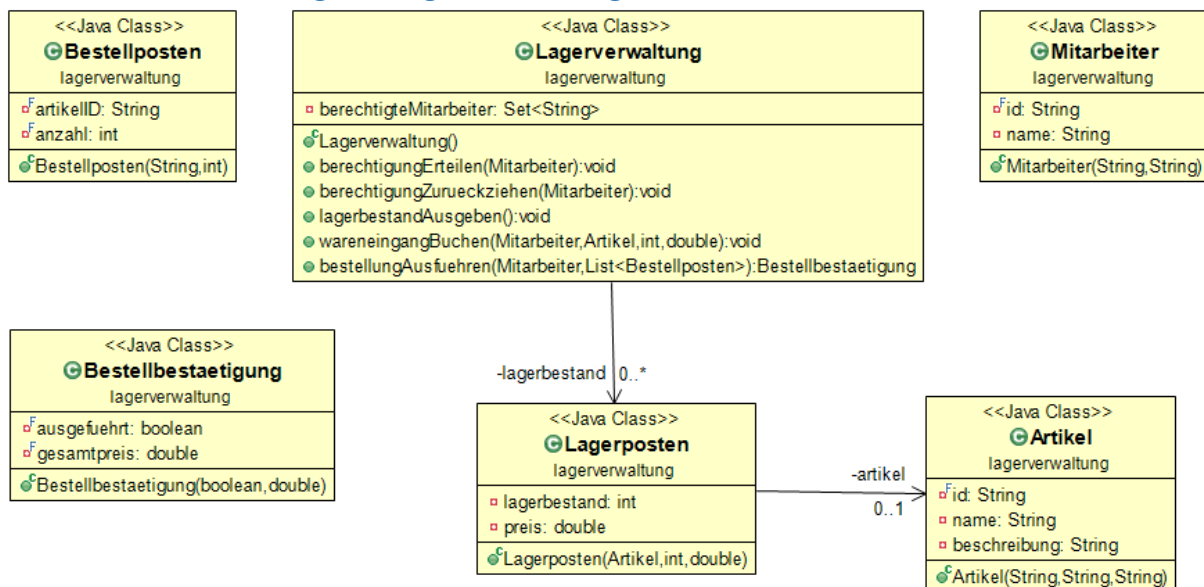


Abbildung 1: Klassenübersicht zur Lagerverwaltung

### Erläuterungen zum Klassendiagramm

- Getter und Setter sind im Diagramm nicht dargestellt und sollten nach Bedarf ergänzt werden.
- Die Konstruktoren übernehmen (außer bei der Klasse `Lagerverwaltung`) jeweils Parameter für die Instanzvariablen der anzulegenden Objekte (initialisierende Konstruktoren).
- Die mit einem kleinen blauen „f“ gekennzeichneten Instanzvariablen sind mit „final“ zu deklarieren, d.h. ein einmal (bei der Objektkonstruktion/-initialisierung) vergebener Wert kann nachträglich nicht mehr verändert werden. Überlegen Sie, welche Auswirkung das auf Setter hat und setzen Sie dies entsprechend im Quellcode um.

## Funktionale Anforderungen an die Lagerverwaltung

Die Software soll folgende Funktionen bieten:

- Die Software soll dazu geeignet sein, einen Lagerbestand zu verwalten (u.a. Menge, Preise und Namen von Artikeln).
- Es sollen sowohl Wareneingänge gebucht als auch Bestellungen ausgeführt werden können. Bei beiden Vorgängen wird jeweils mit verbucht, welcher Mitarbeiter dafür zuständig war und es wird zunächst geprüft, ob der Mitarbeiter berechtigt ist, den Warenbestand zu aktualisieren (Objektvariable `berechtigteMitarbeiter` in Objekt `Lagerverwaltung`). Bei der Buchung eines Wareneingangs wird auch jeweils der aktuelle Preis mit übergeben, der dann einheitlich auch für die betreffenden bereits eingelagerten Artikel/Lagerposten gelten soll.
- Bei der Bearbeitung einer Bestellung soll jeweils geprüft werden, ob der Mitarbeiter berechtigt ist, die Bestellung auszulösen (Objektvariable `berechtigteMitarbeiter` in Objekt `Lagerverwaltung`), und ob überhaupt genügend Waren vorhanden sind. Die bestellten Waren sind dann natürlich entsprechend dem Lager zu entnehmen. Bestellungen können (um die Implementierung dieses Projektes einfach zu halten) immer nur vollständig oder gar nicht ausgeführt werden.

## Dateiarbeit

Alle relevanten Buchungen (Wareneingang, Bestellausführung, Erteilen oder Entziehen von Berechtigungen an Mitarbeiter) sollen jeweils mit dem aktuellen Datum versehen in eine Datei geschrieben werden

Informieren Sie sich dazu im Internet, wie mit Java in Dateien geschrieben werden kann und setzen Sie dies in Ihrer Software um. Verwenden Sie insbesondere die Klasse `PrintWriter` aus der Java-Standardbibliothek: <https://docs.oracle.com/javase/9/docs/api/java/io/PrintWriter.html>.

## Javadoc

Alle Klassen sowie Ihre Methoden sollen mit vollständigen Javadoc-Kommentaren inkl. der Nutzung von `@param` und `@return` beschrieben werden.