Aufgabe: Klassen in Python – Verwaltung von Lagerbeständen

Ihr seid beauftragt, ein einfaches System zur Verwaltung von Lagerbeständen für ein kleines Geschäft zu entwickeln. Jedes Produkt im Lager soll als Objekt repräsentiert werden, das bestimmte Eigenschaften und Verhaltensweisen besitzt.

Teil 1: Die Produkt-Klasse (Grundlagen)

1. **Definiere eine Klasse namens Produkt.**
    - Diese Klasse soll einen Konstruktor (__init__) haben, der beim Erstellen eines Produkts die folgenden Attribute entgegennimmt und setzt:
        - name (String): Der Name des Produkts (z.B. "Laptop", "Maus", "Tastatur").
        - produkt_id (String): Eine eindeutige ID für das Produkt (z.B. "LP001", "MS002").
        - preis (Float): Der Preis des Produkts (z.B. 1200.50, 25.00).
        - bestand (Integer): Die aktuelle Anzahl dieses Produkts im Lager (z.B. 10, 50).
2. **Füge eine Methode anzeigen_details hinzu.**
    - Diese Methode soll keine Parameter außer self entgegennehmen.
    - Sie soll alle Details des Produkts (Name, ID, Preis, Bestand) in einem gut lesbaren Format auf der Konsole ausgeben. Beispiel:

      ```
      Produkt: Laptop (ID: LP001)
      Preis: 1200.50 EUR
      Verfügbar: 10 Stück
      ```

3. **Füge eine Methode produkt_verkaufen hinzu.**
    - Diese Methode soll einen Parameter menge (Integer) entgegennehmen, der angibt, wie viele Einheiten des Produkts verkauft wurden.
    - Die Methode soll den bestand des Produkts um die verkaufte Menge reduzieren.
    - **Wichtig:** Überprüfe, ob genügend Produkte auf Lager sind, bevor du den Verkauf durchführst. Wenn nicht genügend Produkte vorhanden sind, gib eine entsprechende Meldung aus (z.B. "Nicht genügend Laptop auf Lager, verfügbar: 5, versucht zu verkaufen: 10").
4. **Füge eine Methode bestand_auffuellen hinzu.**
    - Diese Methode soll einen Parameter menge (Integer) entgegennehmen, der angibt, wie viele Einheiten des Produkts dem Lager hinzugefügt wurden.
    - Die Methode soll den bestand des Produkts um die angegebene Menge erhöhen.

Teil 2: Anwendung der Produkt-Klasse (Interaktion)

1. **Erstelle mehrere Instanzen der Produkt-Klasse.**
    - Erstelle mindestens 3 verschiedene Produkte mit sinnvollen Daten (z.B. einen Laptop, eine Maus, eine Tastatur).
2. **Führe verschiedene Operationen mit deinen Produkten durch:**
    - Zeige die Details jedes Produkts an, nachdem du sie erstellt hast.
    - Verkaufe eine bestimmte Menge von einem Produkt, bei dem genügend Bestand vorhanden ist.
    - Versuche, eine zu große Menge von einem anderen Produkt zu verkaufen, sodass die Warnmeldung erscheint.
    - Fülle den Bestand eines Produkts auf.
    - Zeige die Details der Produkte erneut an, um die Änderungen im Bestand zu

überprüfen.

Bonus-Aufgabe: Die Lager-Klasse

1. **Definiere eine weitere Klasse namens Lager.**
    - Diese Klasse soll einen Konstruktor (__init__) haben, der eine leere Liste namens produkte als Attribut initialisiert. Diese Liste wird alle Produkt-Objekte speichern, die sich im Lager befinden.
2. **Füge eine Methode produkt_hinzufuegen zur Lager-Klasse hinzu.**
    - Diese Methode soll ein Produkt-Objekt als Parameter entgegennehmen.
    - Sie soll das übergebene Produkt-Objekt zur produkte-Liste des Lagers hinzufügen.
3. **Füge eine Methode alle_produkte_anzeigen zur Lager-Klasse hinzu.**
    - Diese Methode soll über alle Produkte in der produkte-Liste iterieren und für jedes Produkt die Methode anzeigen_details aufrufen.
4. **Modifiziere dein Hauptprogramm:**
    - Erstelle eine Instanz der Lager-Klasse.
    - Füge die von dir erstellten Produkt-Instanzen dem Lager-Objekt hinzu.
    - Rufe die Methode alle_produkte_anzeigen auf dem Lager-Objekt auf, um zu sehen, ob alle Produkte korrekt angezeigt werden.