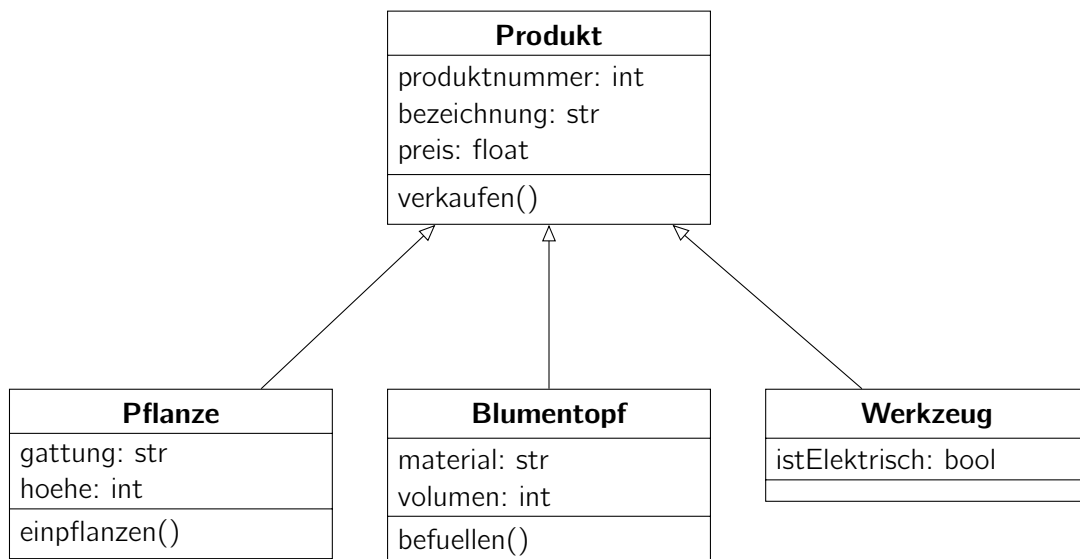


👤 Aufgabe 1: Gartencenter-Produktverwaltung in Python

Nachdem wir in der letzten Stunde gemeinsam ein Softwarekonzept für die Produktverwaltung des Gartencenters entworfen haben, soll dieses nun implementiert werden.



Setzen Sie das obenstehende UML-Diagramm in Python um. Bilden Sie insbesondere die Vererbungsstrukturen korrekt ab.

👤 Aufgabe 2: Richtung der Vererbung

Betrachten Sie den folgenden Code.

```
1 class Laptop:
2     def leistung(self):
3         return "Alltagstauglich"
4 class GamingLaptop(Laptop):
5     def leistung(self):
6         return "Sehr leistungsstark"
```

a) Verändern Sie den Code, sodass die Richtung der Vererbung umgedreht wird.

b) Diskutieren Sie, welche Variante mehr Sinn ergibt.

⊕ Bonusaufgabe 3: Mehrfachvererbung

Am Ende der letzten Stunde haben wir kurz über die Probleme gesprochen, die bei mehrfacher Vererbung entstehen können (Beispiel: `Wasserflugzeug` als Subklasse von `Flugzeug` und `Boot`).

a) Recherchieren Sie, wie in Python mit Mehrfachvererbung umgegangen wird.

b) Setzen Sie das oben beschriebene Beispiel in Python um und halten Sie Ihre Beobachtungen fest.
