OOP: Vererbung (Python)	Klasse:
Musterlösung	Datum:

Aufgabe 1: Gartencenter-Produktverwaltung in Python

SAE

L. Bung

```
1
   class Produkt():
2
       def __init__(self, produktnummer, bezeichnung, preis):
           self.produktnummer = produktnummer
3
4
           self.bezeichnung = bezeichnung
           self.preis = preis
5
       def verkaufen(self):
6
7
           print(f"Das Produkt mit der Nummer {self.produktnummer} wurde fü
               r {self.preis} verkauft.")
8
9
   class Pflanze(Produkt):
       def __init__(self, produktnummer, bezeichnung, preis, gattung, hoehe
10
          ):
           super().__init__(produktnummer, bezeichnung, preis)
11
12
           self.gattung = gattung
13
           self.hoehe = hoehe
       def einpflanzen(self):
14
15
           print(f"Die Pflanze mit der Produktnummer {self.produktnummer}
               wurde eingepflanzt.")
16
17
   class Blumentopf(Produkt):
18
       def __init__(self, produktnummer, bezeichnung, preis, material,
19
           super().__init__(produktnummer, bezeichnung, preis)
           self.material = material
20
           self.volumen = volumen
21
22
       def befuellen(self):
23
           print(f"Der Topf mit der Nummer {self.produktnummer} wurde mit {
               self.volumen} Liter befüllt.")
24
25
   class Werkzeug(Produkt):
26
       def __init__(self, produktnummer, bezeichnung, preis, istElektrisch)
27
           super().__init__(produktnummer, bezeichnung, preis)
           self.istElektrisch = istElektrisch
28
```

Aufgabe 2: Richtung der Vererbung

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

```
class GamingLaptop:
def leistung(self):
return "Sehr leistungsstark"
class Laptop(GamingLaptop):
def leistung(self):
return "Alltagstauglich"
```

b) Diskutieren Sie, welche Variante mehr Sinn ergibt.

Die Variante, in der GamingLaptop von Laptop erbt, ergibt mehr Sinn: Ein Gaming-Laptop ist eine spezialisierte Variante eines "normalen" Laptops und nicht umgekehrt. Es könnte beispielsweise auch noch eine zweite Subklasse ArbeitsLaptop geben.

Aus logischer Sicht ergibt das Codebeispiel aus der Aufgabe so wie es dasteht daher keinen Sinn.