14. OOP in Python Klassen, Methoden, Attribute

14.1. Definitionen

- Objekt
  - ein Objekt ist eine softwaretechnische Repräsentation eines realen oder gedachten, klar abgegrenzten Gegenstandes oder Begriffs.
  - das Objekt erfasst alle Aspekte des Gegenstandes durch Attribute (Eigenschaften) und Methoden.

- Attribute und Methoden
  - Attribute sind Eigenschaften des Objektes, sie beschreiben den Gegenstand vollständig.
  - sind Attribute gegen Manipulationen von außen geschützt, wird das als Kapselung bezeichnet
  - Methoden beschreiben Operationen, die mit dem Objekt (bzw. seinen Attributen) durchgeführt werden können.
  - von außen erfolgt der Zugriff auf die geschützten Attribute durch die Methoden.

14.2. Aufbau einer Klasse

- eine Klasse wird definiert mit dem Schlüsselwort **class** *Klassenname()*
  Klassennamen sollten mit Großbuchstaben beginnen

- innerhalb einer Klasse gibt es Attribute und Methoden, die durch Sichtbarkeitsmodifizierer (public, protected oder private) gekennzeichnet werden (siehe auch 12.3 Definitionen)

  - private Definition mit zwei vorangestellten Unterstrichen **__name**
    - auf diese Attribute und Methoden nur innerhalb der Klasse zugegriffen werden
    - sie sind außerhalb der Klasse nicht zugreifbar

  - protected Definition mit einem vorangestellten Unterstrich **_name**
    - ist bei der Vererbung relevant
    - diese Attribute sind nach außen geschützt, bleiben in der Vererbungshierarchie aber zugreifbar - dient in Python als Hinweis

  - public Definition **name**
    - alle Attribute und Methoden sind automatisch öffentlich (public)
    - sie bilden die Schnittstelle der Klasse nach außen
    - die Kommunikation mit der Klasse findet darüber statt

- mit diesen Sicherheitsmodifizieren
    - lassen sich Attribute kapseln und vor dem direkten Zugriff von außen schützen
    - private Methoden werden ausschließlich für den internen Gebrauch verwendet
    - öffentliche Methoden einer Klasse bilden ihre Schnittstelle zur Kommunikation nach außen

14.3. Instanziierung einer Klasse

- ein Objekt wird instanziiert (gebildet), in dem einem Variablennamen der Klassename mit leeren Klammern zugewiesen wird *name* = **class** *Klassenname***()**
    - in den Klammern können später Parameter eingefügt werden

14.4. Methoden einer Klasse anlegen

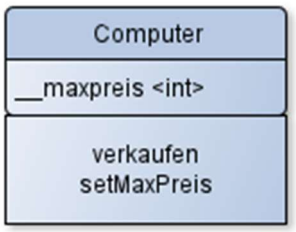
- eingerückt innerhalb der Klassendefinition (*class Klasenname()*) erfolgt die Definition der Methoden jeweils mit **def** *Name der Methode(…)*
    - mit dem Schlüsselwort **def**
    - dem Namen der Methode
    - und in runden Klammern umschlossen die Parameter

- jede Methode hat mindestens einen Parameter, eine Referenz auf das Objekt selbst - sie wird formuliert mit **self**

- über die Konstruktor-Methode **__init__** werden den Attributen des Objektes beim Erstellen (Instanziieren) Initialwerte zugewiesen **def __init__(self,** …**)**

- über eine Set-Methode lassen sich private Attribute verändern
    - in der Set-Methode werden die Parameter überprüft **def** set_attr**(self,** …**)**
    - nur geprüfte Werte werden den Attributen zugewiesen

14.5. Beispiel Klasse Computer



- Objekte der Klasse Computer sollen die folgenden Attribute (Eigenschaften) besitzen bzw. Methoden beherrschen:
    - Attribute
        - __maxpreis — Verkaufspreis
privates Attribut – von „außen“ nicht „sichtbar“ nur
    - Methoden
        - setMaxPreis — Methode, um den Verkaufspreis zu verändern
        - verkaufen — Methode zum Ausgeben des Verkauspreises

- Beispiel:

```
# Definition der Klasse Computer
class Computer:

    # Definition der Konstruktor-Methode
    def __init__(self):
        self.__maxpreis = 900

    # Definition der Methode verkaufen
    def verkaufen(self):
        print(f"Verkaufspreis: {self.__maxpreis}")

    # Definition der Set-Methode setMaxPreis
    def setmaxPreis(self, preis):
        self.__maxpreis = preis

# Instanziieren des Objektes c und Ausgeben des Verkaufspreises
c = Computer()
c.verkaufen()

# so KEINE Änderung des Verkaufspreises, da __maxpreis ein privates
Attribut ist
c.__maxpreis = 1000
c.verkaufen()


# über die SET-Methode setMaxPreis wird der Verkaufspreis geändert
# und anschließend ausgegeben
c.setMaxPreis(1000)
c.verkaufen()
```

14.6. Konzept der Vererbung

- eine Klasse (Eltern-Klasse wird mit Attributen und Methoden definiert
- eine weitere Klasse (Kind-Klasse) wird davon abgeleitet und über nimmt die Attribute und Methoden der Eltern-Klasse
- in der Kind-Klasse können die übernommenen (geerbten) Attribute und Methoden angepasst und erweitert werden

- Beispiel
    - ein Stecker ist noch sehr allgemein
    - spezielle Stecker sind Kaltgerätestecker, Monitor-Stecker, USB-Stecker
    - alles, was Stecker prinzipiell auszeichnet, gilt für jeden der speziellen Stecker
    - jeder der speziellen Stecker erbt die allgemeinen Eigenschaften vom allgemeinen Stecker und hat zusätzlich noch ein paar weitere spezielle Eigenschaften

14.7. Konzept der Kapselung

- alles, was zu einer Klasse gehört, soll nur von Objekten dieser Klasse angesprochen werden
- von außen sind die Eigenschaften der Klasse „versteckt“, und damit gerät man weniger in Gefahr, mit Attributen irgendeinen Unsinn zu machen

14.8. Konzept der Polymorphie

- Polymorphie heißt „Vielgestaltigkeit“
- damit ist gemeint, dass sich eine Eigenschaft in verschiedenen Klassen verschieden zeigen kann
- das Zeichen + kann für Zahlen, aber auch für Zeichenketten verwendet werden
    - in Verbindung mit Zahlen bedeutet es die Addition
    - in Verbindung mit Zeichenketten bedeutet es das Verketten der Zeichenfolgen

Aufgabe

1. Klasse Student
   Diese Klasse hat die geschützten Eigenschaften:
   - Geschlecht
   - Alter in Jahren
   - Fachrichtung

   und es gibt die Methoden
   - hat Geburtstag (Alter steigt i.d.R. um 1 Jahr), dann soll eine Ausgabe auf dem Bildschirm erscheinen mit dem entsprechenden Alter
   - wechselt die Fachrichtung, dann soll eine Ausgabe auf dem Bildschirm erscheinen mit dem alten Fach und dem neuen Fach

   Es sind 2 Objekte zu instanziieren:
   - Susi, Alter 22, Fachrichtung Technik
   - Benno, Alter 24, Fachrichtung Management

   - Benno soll 2-mal Geburtstag haben und Susi das Fach wechseln nach Organisation