2025/11/21 05:56 1/3 LU12b - Klassen definieren

# LU12b - Klassen definieren



Eine Klasse ist ein Bauplan, der den Aufbau von Objekten definiert.

#### 1 Klasse - 1 Datei

Es bietet sich an, für jede Klasse eine eigene Python-Datei zu erstellen. Für die Klasse Member sollte unsere Datei member.py heissen. Unsere Datei member.py sieht so aus:

#### member.py

```
from dataclasses import dataclass
@dataclass
class Member:
    a club member
    firstname: str
    lastname: str
    address: str
    place: str
    zip_code: str
    entry year: int
    birth year: int
    honorary_member: bool
if name == ' main ':
    test_person = Person()
    test person.firstname = 'Hans'
    print(test person)
```



Diese Definition einer Klasse ist wie ein Bauplan für Clubmitglieder. Der Bauplan definiert, aus welchen Bestandteilen ein Clubmitglied besteht.

Schauen wir uns den Code Stück für Stück an:

## dataclass

```
from dataclasses import dataclass

@dataclass
class Member:
    """
    a club member
    """
```



**dataclass** ist ein Pythonmodul für die Definition von Klassen, die hauptsächlich Daten verwalten sollen.

Der **Decorator** @dataclass kennzeichnet unsere Klasse als *Datenklasse*. Diese Annotation muss direkt vor dem class-Befehl stehen.

Um eine Klasse zu definieren, verwenden wir den Befehl class Member. Beachten Sie die Codingstandards: Im Gegensatz zu Variablen, werden Klassennamen mit einem Grossbuchstaben geschrieben.

Ein kurzer Kommentarblock zwischen dreifachen Anführungszeichen (") erklärt den Sinn und Zweck dieser Klasse.

### **Attribute**

```
class Member:

firstname: str
lastname: str
address: str
place: str
zip_code: str
entry_year: int
birth_year: int
honorary_member: bool

...
```

https://wiki.bzz.ch/ Printed on 2025/11/21 05:56

2025/11/21 05:56 3/3 LU12b - Klassen definieren



Attribute sind Variablen die im Kontext einer Klasse definiert werden. Diese Attribute sind in allen Funktionen innerhalb der Klasse verfügbar.

Nach dem class-Befehl definieren wir die Attribute und deren Datentyp.

## \_\_main\_\_ ?

```
if __name__ == '__main__':
    test_person = Person()
    test_person.firstname = 'Hans'
    print(test_person)
```



Eine Datenklasse ist nicht dafür gemacht, direkt ausgeführt zu werden. Zu Testzwecken können wir ein paar Objekte erstellen und ausgeben.

Klassen sind keine eigenständigen Pythonskripts, die selbständig ausführbar sind. Wir können aber zu Testzwecken eine Klasse direkt aufrufen. In diesem Codeblock erstellen wir ein paar Objekte, setzen die Attribute und geben die Objekte aus.

m319-C1F, m319-C1E, m319-G1G, m319-G1E



© Marcel Suter

From

https://wiki.bzz.ch/ - BZZ - Modulwiki

Permanent link:

https://wiki.bzz.ch/modul/archiv/m319python/learningunits/lu12/klasse

Last update: 2024/03/28 14:07

