4 Sequenzdiagramm

Das Sequenzdiagramm zeigt die Interaktion von Objekten im zeitlichen Ablauf auf. Es sind somit mindesten 2 Objekte beteiligt

.

UML-Notation

(1) Instanziierung eines Objektes

```
class A:
  # Klasse A erzeugt selber ein Objekt der Klasse B.

  def __init__(self):
    # Erzeugung des Objektes vom Typ B im Konstruktor von A
    self._anObject = B()
    pass
```

(2) Selbstaufruf einer Methode mit Wertrückgabe

```
class A:
  # Klasse A ruft eine eigene Methode – hier mit einem Rückgabewert – auf

  @property
  def value(self):
    return self._value

  def doSomething(self):
    .....
    v = self.value
    pass
```

(3) Aufruf einer Methode in einer anderen Klasse

```
class A:                                          class B:
  # A ruft eine Methode der Klasse B auf.           # B stellt eine Methode
bereit

  def doSomething(self):                            def doIt(self):
    .....                                             pass
```

```
    self._anObject.doIt()
```

(4) Aufruf einer Methode mit Wertrückgabe

```
class A:                                          class B:
 # A ruft Methode von B auf,                      # B stellt eine Methode bereit
 # die  einen Wert liefert.e

                                                 @property
 def doSomething(self):                             def text(self):
    .....                                             return aText
    txt = self._anObject.text
    pass
```

Objektzerstörung

Nicht mehr benutzte Objekte müssen aus dem Speicher entfernt werden, damit der belegte Speicherplatz wieder freigegeben werden kann.
Python führt eine `garbage collection`. Diese ist besorgt, dass nicht mehr benutze Objekte aus dem Speicher entfernt werden.

```
class A:                                          class B:
  ....                                              # Destruktor definieren.
Hier kann die Klasse

                                                    # ordentlich beendet werden.
  del(self._anObject)                               def __del__(self):
    pass                                               pass
```

© René Probst

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

Permanent link:
https://wiki.bzz.ch/modul/archiv/m320/merkblaetter/merkblatt_4?rev=1788091603

Last update: **2026/08/30 14:06**