

Bibliotheksmodule, z.B. `math`

Jede moderne Python-Installation umfasst eine Reihe von Programm-Modulen, mit denen man eine bestimmte Klasse von Aufgaben bewältigen kann. Nicht selten muss man eine Kombination von Modulen verwenden, um eine Aufgabe zu lösen. Man findet diese Module im Unterordner `Lib` der Python-Installation. Um ein Modul zu verwenden, gibt es 2 grundsätzliche Methoden:

- 1) `import module`, z.B. `import random`: Dadurch importiert man das gesamte Modul, muss aber beim Zugriff auf die einzelnen Bibliotheksfunktionen die Schreibweise `module.function(...)` verwenden:

```
import random          # importiere das ganze random-Modul
random.randint(1, 6)   # rufe randint() aus obigem Modul aus
```

- 2) `from module import blablabla, ...` : Dadurch importiert man nur die Bibliotheksfunktion `blablabla` und kann diese ohne die Angabe des Moduls verwenden: `blablabla(...)`

```
from random import randint          # importiere nur randint
randint(1, 6)                       # Wirf einen fairen Würfel
x = [randint(1, 6) for i in range(20)] # Würfle 20x
```

```
from statistics import mean, stdev    # importiere 2 Funktionen
mean(x)    # Mittelwert von x
stdev(x)   # Standardabweichung von x
```

Man kann hier zusätzlich einen Aliasnamen für die importierte Funktion angeben:

```
from statistics import mean as mittelwert [, ...]
mittelwert(x)
```

`math` enthält die meisten mathematischen Funktionen wie Winkelfunktionen, Logarithmen, Exponentialfunktionen etc. Wir verwenden diese Funktionen so:

```
import math          # importiere alles
x = math.sin(1.0)    # die Modulangabe ist hier nötig
```

oder:

```
from math import sin, cos, exp, log, log10 # importiere nur das
x = sin(1.0)                               # keine Modulangabe
```