

Prof. Dr. Claudia Müller-Birn, Barry Linnert

# Objektorientierte Programmierung, SoSe 17

## Übung 02

TutorIn: Thierry Meurers  
Tutorium 10

Stefaan Hessmann, Jaap Pedersen, Mark Niehues

9. Mai 2017

---

## 1 Datentypen in Python

*< input > : < value >, < type >*

1. *complex(0) : 0 + 0j, complex*
2. *complex(3) : 3 + 0j, complex*
3. *(1 + 2j) \* (3 + 0j) : 3 + 6j, complex*
4. *(2 + 3j)/5j : 0.6 - 0.4j, complex*
5. *() : (), tuple*
6. *(10) : (10), tuple*
7. *[] : [], list*
8. *(0, 3) + (1, 0) : (0, 3, 1, 0), tuple*
9. *2 \* [0, 1] \* 2 : [0, 1, 0, 1, 0, 1], list*
10. *[1, 2, 3] + [5, 4] : [1, 2, 3, 5, 4], list*
11. *2 in (1, 3, 3) : False, bool*
12. *2/3 : 0.666..., float*
13. *3<sup>16</sup> : 19, int*
14. *5|6 : 7, int*
15. *9%7 : 2, int*
16. *-3 : -3, int*
17. *2 << 4 : 32, int*
18. *2 >> 2 : 0, int*
19. *-2 << 4 : -32, int*
20. *-2 >> 2 : -1, int*

- 21.  $1//4 + 3//4 : 0, int$
- 22.  $3 * 3 : 27, int$
- 23.  $0.3 + 0.1 - 0.3 : 0.1, float$
- 24.  $0.1 - 0.3 : -0.2, float$

## 2 Anwendung von Datentypen in Python

Listing 1: Output des Programms

```
1 [2, 3, 8]
3 [[2, 3, 8], [2, 3, 8], 100]
5 [[2, 3, 8], 100, [[2, 3, 8], [2, 3, 8], 100]]
7 [2, 3, 8]
```

## 3 Dynamische Typsysteme

Bei der dynamischen Typisierung wird einer Variablen erst zur Laufzeit ein Datentyp zugeordnet und nicht explizit vorher im Programmcode angegeben. Das ermöglicht ggf. mehr Flexibilität und das Programm bleibt in manchen Fällen lauffähig obwohl ein nicht vorgesehener Datentyp verwendet wurde (und der Code ist meist besser lesbar). Andererseits fallen auch Zuweisungsfehler erst zur Laufzeit auf.

Quelle: [https://de.wikipedia.org/wiki/Typisierung\\_\(Informatik\)](https://de.wikipedia.org/wiki/Typisierung_(Informatik))