

# Blatt 6

Vincent Kümmerle und Elvis Gnaglo

28. November 2025

## 1 Listen

Für die Liste  $a = [2, "d", 5, 8, 233, "dx", 54, "we", "g", \dots, 72, 23, "g"]$  sind die Zugriff Befehle in Tabelle 1 aufgeführt.

Tabelle 1: Zugriff auf verschiedene Listenelemente in Python.

Element	Zugriff
viertes	$a[3]$
vorletztes	$a[-2]$
drittles bis drittletztes	$a[2:-2]$
jedes 2. ab dem 4.	$a[3::2]$
jedes 3. rückwärts ab dem vorletzten	$a[-2::-3]$
7. entfernen	$\text{del } a[6]$

## 2 Datentypen und Ausdrücke

Die erwarteten Ergebnisse sind mit Begründungen in Tabelle 2 aufgelistet.

Tabelle 2: Erwartete Ergebnisse für verschiedene Ausdrücke in Python.

Ausdruck	Ergebnis	Begründung
$3 + 5$	8	Ganzzahlsummation: int + int = int
$3 + 5.0$	8.0	Typkonvertierung: Addition mit float ergibt float
“3” + “5”	“35”	Zeichenketten werden aneinandergehängt (Konkatenation)
“3” * 5	“33333”	String wird fünfmal wiederholt
$3 // 2$	1	Ganzzahldivision; Ergebnis wird abgerundet
$3 / 2$	1.5	Normale Division ergibt float
$\text{int}(2.71828)$	2	$\text{int}()$ schneidet Nachkommastellen ab
$\text{round}(2.71828)$	3	Mathematische Rundung auf ganzzahliges Ergebnis
“hallo” + “Welt”	“halloWelt”	String-Konkatenation

### 3 Gerade / Ungerade

```
1 def is_even(number):
2     """Check, if a given number is even."""
3     assert isinstance(number, int) and number >= 1, "Die gegebene
4         Zahl ist keine positive, natürliche Zahl."
5     return number % 2 == 0
6
7 eingabe_text = input("Bitte gib eine Zahl ein: ")
8
9 try:
10     zahl = int(eingabe_text)
11
12     if is_even(zahl):
13         print(f"Die Zahl {zahl} ist gerade.")
14     else:
15         print(f"Die Zahl {zahl} ist ungerade.")
16
17 except ValueError:
18     print("Das war keine gültige ganze Zahl!")
19 except AssertionError as e:
20     print(f" Fehler: {e}")
```

### 4 Summation

[REDACTED]

### 5 Fakultät

[REDACTED]