

# Hausaufgaben Lösung

Daniel Schicker

April 14, 2024

## 1. Interpreter vs Compiler

### Interpretierte Sprachen:

- Der Code wird zur Laufzeit Zeile für Zeile von einem Interpreter gelesen und ausgeführt.
- Änderungen am Code können sofort gesehen werden, ohne dass der Code neu kompiliert werden muss.

### Kompilierte Sprachen:

- Der Code wird vor der Ausführung vollständig in Maschinencode übersetzt.
- Kompilierte Programme laufen in der Regel schneller, da sie direkt in Maschinencode übersetzt werden.
- Änderungen am Code erfordern eine erneute Kompilierung.

## 2. Ergebnisse von den 3 Dateien

Figure 1: Ausführung von Script 1

```
daniel@LaptopDaniel:~/Scriptsprachen_Python/VL1$ python3 01_skript.py
Hallo! Ich bin (wahrscheinlich) das erste Pythonskript welches du direkt auf der Kommandozeile ausgeführt hast!
daniel@LaptopDaniel:~/Scriptsprachen_Python/VL1$
```

Figure 2: Ausführung von Script 2

```
01_skript.py docx: keine Identifizierung
daniel@LaptopDaniel:~/Scriptsprachen_Python/VL1$ python3 02_skript.docx
/usr/bin/python3: can't find '__main__' module in '/home/daniel/Scriptsprachen_Python/VL1/02_skript.docx'
daniel@LaptopDaniel:~/Scriptsprachen_Python/VL1$
```

Figure 3: Ausführung von Script 3

```
daniel@LaptopDaniel:~/Scriptsprachen_Python/VL1$ python3 03_skript.txt
Da ich eine einfache Textdatei bin, erkennt der Pythoninterpreter auch ohne die Dateiendung ".py", dass ich ein Pythonskript bin.
daniel@LaptopDaniel:~/Scriptsprachen_Python/VL1$
```

⇒ Man erkennt, dass Skript 1 und Skript 3 ausgeführt werden, aber nicht Skript 2, da der Interpreter nicht erkennt, dass es Python Code ist.

### 3. Tabellenauswertung

| Anweisung        | x     | y     | z     |
|------------------|-------|-------|-------|
| $x = y = 1$      | 1     | 1     | -     |
| $x = 2$          | 2     | 1     | -     |
| $z = 2 \times 3$ | 2     | 1     | 6     |
| $x, y = 1.6$     | error | error | error |
| $sy = 6/2$       | 2     | 1     | 6     |
| $z, x = 3.0, 6$  | 6     | 1     | 3.0   |
| $x = "y"$        | y     | 2     | 3.0   |

```
z = "_"
x = y = 1
x=2
print('x =', x, 'y =', y, 'z =', z)
z = 2*3
print('x =', x, 'y =', y, 'z =', z)
try:
    x,y = 1.6
except:
    print('fehler bei x,y =1.6')
sy = 6/2
print('x =', x, 'y =', y, 'z =', z)
z,x = 3.0,6
print('x =', x, 'y =', y, 'z =', z)
x = "y"
print('x =', x, 'y =', y, 'z =', z)
```

### 4. Weitere Datentypen

- **Float:** Eine Gleitkommazahl, die eine Dezimalzahl darstellt.
- **Bool:** Ein boolescher Wert, der entweder wahr oder falsch ist.
- **char:** Ein einzelnes Zeichen, das in einfachen Anführungszeichen steht.
- **Liste:** Eine Liste ist eine geordnete Sammlung von Elementen, die durch ein Komma getrennt sind und in eckigen Klammern stehen.
- **Tupel:** Ein Tupel ist eine geordnete Sammlung von Elementen, die durch ein Komma getrennt sind und in runden Klammern stehen.
- **Dictionary:** Ein Dictionary ist eine ungeordnete Sammlung von Elementen, die durch ein Komma getrennt sind und in geschweiften Klammern stehen.