

Kapitel 1 (Allgemeines)

- Du kannst die Endung von Python-Programmen angeben.
- Du kannst das Kommentarzeichen von Python-Programmen angeben.
- Du kannst angeben, mit welcher Art von Programm man einen Python-Quelltext schreibt.
- Du kannst angeben, welcher Art von Programm einen Python-Quelltext ausführt.
- Du kannst drei wesentliche Typen von Programmierfehlern beschreiben und in Beispielpogrammen erkennen.

Kapitel 2 (Einfache Datentypen)

- Datentypen: ganze Zahlen (integer), Gleitkommazahlen (float) und Zeichenketten (string) erkennen können.
- mathematische Operatoren: +, -, *, /, **, //, %
- mathematische Funktionen: derzeit nur abs(...)
- Wahrheitswerte: True, False und die Operatoren not, and, or
- Vergleichsoperatoren: ==, !=, <=, <, >=, >

Kapitel 3 (Variablen)

- Die Regel für die Bildung von Bezeichnern (Variablenamen): Das erste Zeichen muss ein Buchstabe oder ein Unterstrich sein. Danach dürfen weitere Buchstaben oder Unterstriche oder Ziffern folgen. Sonderzeichen wie +, -, *, #, ", @, ... sind nicht erlaubt.
- Zuweisungsoperator: =
- Mehrfachzuweisungen: z.B. x, y, z = 5, 7, -3
- Erweiterte Zuweisungsoperatoren: +=, *=, usw.

Kapitel 4 (Bedingte Anweisungen und Verzweigungen)

- bedingte Anweisungen:

```
if <bedingung>:
    <block>
```
- einfache Verzweigung:

```
if <bedingung>:
    <block1>
else:
    <block2>
```
- mehrfache Verzweigung:

```
if <bedingung1>:
    <block1>
elif <bedingung2>:
    <block2>
...
elif <bedingungN>:
    <blockN>
else:
    <block>
```
- Verschachtelte bedingte Anweisungen und Verzweigungen

Kapitel 5 (Schleifen)

- for-Schleife (Zählschleife): `for <var> in range(<start>,<stop>[,<step>]):`
`<block>`
- while-Schleife: `while <bedingung>:`
`<block>`
- Schleifenabbruch mit `break`
- Überspringen von Schleifendurchläufen mit `continue`

Kapitel 6 (Listen)

- Bildung von Listen mit `L = [...]`
- Listen mit `for` iterieren (durchlaufen): `for <var> in <liste>:`
`<block>`
- Listensyntax `L[0]`, `L[1]`, ...
- Bedeutung negativer Listenindizes wie `L[-1]`
- Slices: `L[a:b]`, `L[a:]`, `L[:b]`, `L[a:]`
- Du weißt, dass das Kopieren einer Listenvariablen nur das Kopieren einer Listenreferenz bedeutet und dass eine Änderungen an der Kopie eine Änderung am Original bewirkt.
- Anzahl Elemente: `len(L)`
- Listenmethoden: `L.append(...)`, `L.pop(...)`, `L.insert(...)`

Kapitel 7 (Funktionen)

- Definition von Funktionen: `def <funktionsname>(<parameterliste>):`
`<funktionsblock>`
- Ausführung von Funktionen mit der Ersetzung der formalen durch die aktuellen Parameter
- Wirkung der `return`-Anweisung (Funktionsabbruch mit Rückgabewert)
- Rekursion

Kapitel 8 (Zeichenketten)

- Syntax von von Zeichenketten mit `'...'` oder `"..."`
- String-Operatoren: `+` und `*`
- String-Funktion: vorläufig nur `len(<str>)`
- Die String-Methode `str.format(...)` mit `{...}`-Ersetzungen

Kapitel 9 (Ein- und Ausgabe)

- `print()`-Anweisung (zusammen mit der `format()`-Methode)
- Funktion `int(<str>)` zur Umwandlung einer Zeichenkette in eine ganze Zahl
- Funktion `float(<str>)` zur Umwandlung einer Zeichenkette in eine Gleitkommazahl
- Lesen von Dateien und Schreiben in Dateien mit der Funktion `open(...)` und den Methoden `fd.write(...)`, `fd.read(...)` und `fd.close()`.