

# LS 1 – Hallo Python

## Ausgangssituation

Die Entwicklung von Skripten spielt in der System-Administration eine immer größere Rolle. Deshalb sollen Sie sich in die Programmiersprache Python einarbeiten und erste kleine Programme erstellen.



## Aufgabe 1 – Einarbeitung in erste Entwicklungswerkzeuge

Machen Sie sich mit der Entwicklungsumgebung Python-IDLE (alternativ Visual Studio Code) und dem Zeichenprogramm Structorizer vertraut.



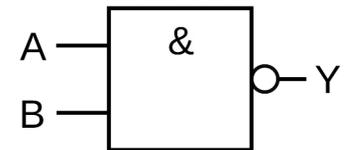
- Erstellen Sie ein **Taschenrechner** mit Python, welcher zwei Zahlen einliest. Anschließend wird Summe, Differenz, Produkt und Quotient berechnet und die Ergebnisse angezeigt. Speichern Sie das Programm unter dem Namen **aufgabe\_ls1\_01.py** ab.
- Erstellen Sie einen **Temperaturumrechner** mit Python. Das Programm soll von Grad Celsius zu Grad Fahrenheit umrechnen können. Speichern Sie das Programm unter dem Namen **aufgabe\_ls1\_02.py** ab.
- „**Parallelschaltung**“ - Erstelle ein Programm, welches die Widerstandswerte zweier parallel geschalteter Widerstände einliest und anschließend den Gesamtwiderstand ausgibt. Speichere das Programm unter dem Namen **aufgabe\_ls1\_03.py** ab.
- „**Kugel**“ - Erstelle ein Programm, welches den Radius einliest und anschließend die Oberfläche und das Volumen einer entsprechenden Kugel ausgibt. Speichere das Programm unter dem Namen **aufgabe\_ls1\_04.py** ab.
- Erweitern Sie Programme für die Aufgaben (a) und (d) mit einfachen Schleifen, so dass die Berechnungen wiederholt ablaufen. Erweitern Sie Aufgabe (a) indem Sie zuerst einen Operator einlesen und nur die jeweils gewünschte Rechnung durchführen. Erweitern Sie Aufgabe (b) indem der Benutzer wählen kann, in welche Richtung die Temperaturumrechnung erfolgt.
- Erstellen Sie je ein **Struktogramm** oder einen **PAP** für die Aufgaben (a) bis (e).

## Aufgabe 2 – Reflexion der benutzten Programmierkonstrukte und Programmierwerkzeuge

- Mit welcher Anweisung können Sie in Python eine Ausgabe auf dem Bildschirm realisieren? Geben Sie beispielhaft den Text „LF5 ist der didaktische Höhepunkt der Woche aus.“
- Mit welcher Anweisung können Sie in Python einen Text einlesen?
- Was ist beim Einlesen einer Zahl zu beachten? Lesen sie beispielhaft den Radius (Kommazahl) ein und speichern Sie ihn in einer Variable `radius`.
- Was versteht man unter einer Variablen?
- Nennen Sie beispielhaft einige häufig genutzte Datentypen für Variablen?
- Was versteht man unter einer Anweisung?

- g) Am Anfang vieler Python-Programme finden Sie `import`-Anweisungen. Wozu dienen diese?
- h) Wozu dienen Kommentare? Wie können Sie Kommentare in Python kennzeichnen?
- i) Welche Anweisungen gibt es in Python, um Schleifen aufzubauen?
- j) Zählen Sie beispielhaft in einer `WHILE`-Schleife rückwärts von 10 bis 1.
- k) Zählen Sie beispielhaft in einer `FOR`-Schleife von 5 bis 25.
- l) Welche Anweisungen gibt es in Python, um Fallunterscheidungen aufzubauen?
- m) Wozu benutzt man bei der Programmierung einen (Text-)Editor?
- n) Was ist der Unterschied zwischen einem Interpreter und einem Compiler?
- o) Wird bei Python mit einem Compiler oder mit einem Interpreter gearbeitet?
- p) Nennen Sie zwei Entwicklungsumgebungen, die Sie bei der Erstellung von Python-Programmen nutzen können?
- q) Welche Entwicklungsumgebung wird direkt mit Python ausgeliefert?
- r) Worauf müssen Sie achten, wenn Sie Visual-Studio-Code zur Python-Entwicklung nutzen wollen?
- s) Was ist der Unterschied zwischen einem Textverarbeitungsprogramm und einem Text-Editor?
- t) Neben Python gibt es noch weitere Programmiersprachen. Nennen Sie drei weitere.
- u) Lesen Sie sich den einführenden Text über Python durch (FISI-LF5-LS1-Text1). Welche Anwendungsbereiche für Python werden dort genannt?
- v) Wofür steht die Abkürzung IDE?

### Aufgabe 3 - Programm erweitern



Erweitern Sie das Hilfsprogramm zur Anzeige logischer Funktionen und führen Sie folgenden Arbeiten aus:

- Dokumentieren Sie das Original-Programm mit einem Struktogramm und einem Programmablaufplan (PAP).
- Erweitern Sie das Programm, so dass es auch Wertetabellen für die folgenden logischen Verknüpfungen darstellen kann:
  - NAND
  - NOR
  - ÄQUIVALENZ
  - ANTIVALENZ

#### Hinweise:

- Das Originalprogramm finden Sie in der Datei `aufgabe_ls1_05_orginal.py`.
- Speichere das Programm unter dem Namen `aufgabe_ls1_05.py` ab.

### Aufgabe 4 - Mit Text- und CSV-Dateien arbeiten



- Schreiben Sie ein Python-Programm, welches eine Text-Datei einliest und deren Inhalt auf dem Bildschirm ausgibt. Speichere das Programm unter dem Namen `aufgabe_ls1_06.py` ab.
- Schreiben Sie ein Python-Programm, welches alle 2 Sekunden einen Zeitstempel und einen beliebigen Text auf dem Bildschirm ausgibt und in eine Text-Datei schreibt. Der Name der Text-Datei soll vorher erfragt werden. Speichere das Programm

unter dem Namen **aufgabe\_ls1\_07.py** ab.

- c) In der CSV-Datei `widerstand.csv` stehen zeilenweise jeweils 3 Widerstandswerte in Ohm. Schreiben Sie ein Python-Programm, welches die drei Widerstände addiert und die 3 Widerstandswerte und die Summe wieder in eine CSV-Datei namens `gesamtwiderstand.csv` schreibt. Zur Kontrolle soll das Programm auf dem Bildschirm die Namen der Ein- und Ausgabedatei, die Anzahl der Berechnungen, das aktuelle Datum und die Uhrzeit ausgeben. Speichere das Programm unter dem Namen **aufgabe\_ls1\_08.py** ab.
- d) Erstellen Sie jeweils ein **Struktogramm** oder einen **PAP** für die Aufgaben (a) und (d).

### Aufgabe 5 – Reflexion Dateien



- a) Wofür steht die Abkürzung CSV?
- b) Wozu werden CSV-Dateien typischerweise genutzt?
- c) Welche Alternativen gibt es zu CSV-Dateien?
- d) Was ist der Unterschied zwischen einer Text- und einer Binär-Datei?
- e) Nennen Sie einige Beispiele für typische Binärdateien. Welche Endungen zu ihrer Kennung werden genutzt?
- f) Zeigen Sie, wie man eine Text-Datei mit Python einlesen kann?
- g) Welches Modul zum Umgang mit CSV-Dateien gibt es in Python?

### Aufgabe 6 – Visualisierung



- a) Wozu wird ein Kontext-Diagramm genutzt?
- b) Wozu werden Struktogramme und PAPs genutzt?
- c) Wofür steht die Abkürzung PAP?
- d) Wie werden Struktogramme auch genannt?
- e) Was versteht man unter strukturierter Programmierung?
- f) Was versteht man unter einfachen linearen Programmen? Wie wird die hierbei genutzte Struktur auch genannt?
- g) Was ist der Unterschied zwischen einer kopfgesteuerten und einer fußgesteuerten Schleife? Wie werden sie in Struktogrammen dargestellt?
- h) Welche Möglichkeiten zur Darstellung von Schleifen gibt es bei PAPs?
- i) Bei Programmieren mit Python werden häufig andere vorhandene Module genutzt. Welche Möglichkeiten zur Visualisierung gibt es?

**Aufgabe 7 - Aufgaben zu Struktogrammen und PAPs**

a)

Nach dem Einscannen des Aufklebers werden aus einer Datenbank Gewicht, Breite, Länge und Dicke des Buchs ermittelt. Sensoren im Rückgabefach erfassen die Maße des eingelegten Buchs. Nur wenn alle gemessenen Maße mit den zum Buch gespeicherten Maßen übereinstimmen, wird das Buch angenommen, sonst wird ein Bibliotheksmitarbeiter benachrichtigt.

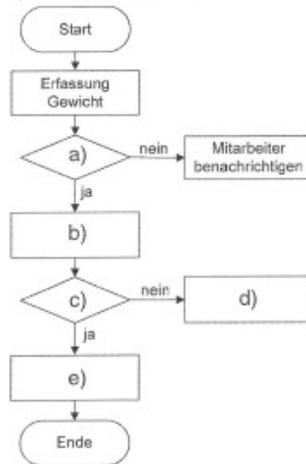
Ordnen Sie den mit a) bis e) gekennzeichneten Symbolen des Flussdiagramms die zutreffenden Beschriftungen für diesen Vorgang zu.

Tragen Sie die Ziffer vor der jeweils zutreffenden Beschriftung in das Kästchen ein.

Beschriftungen

- 1 Gewicht o.k.?
- 2 Breite, Länge, Dicke o. k.?
- 3 Mitarbeiter benachrichtigen
- 4 Breite, Länge, Dicke erfassen
- 5 Buch annehmen

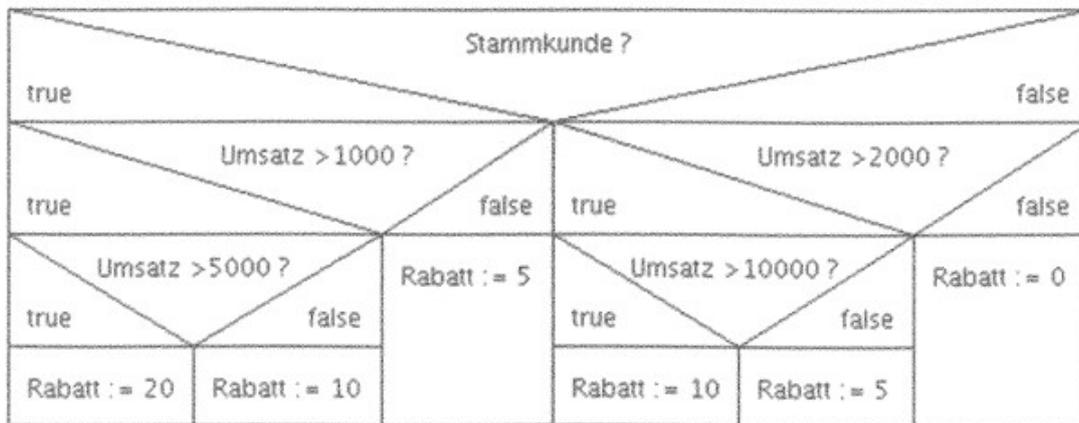
Symbole des Flussdiagramms



b)

Die IT-SoftServ GmbH soll eine Methode entwickeln, mit der die Werk AG für Stammkunden und übrige Kunden nach Umsatz gestaffelte Rabatte ermitteln kann. Folgendes Struktogramm wurde bereits entwickelt:

Rabatt ermitteln



Hinweis:

- Die Variable Rabatt enthält den Rabattsatz in Prozent.
- Die Variable Umsatz enthält den Umsatz des Kunden in EUR.
- Auf eine Deklaration der Variablen wurde verzichtet.

Welche der folgenden Aussagen trifft auf das Struktogramm zu?

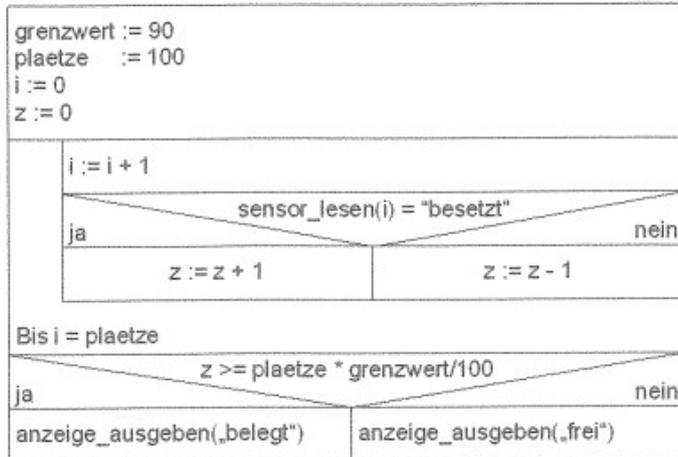
Tragen Sie die Ziffer vor der zutreffenden Aussage in das Kästchen ein.

- 1 Ab einem bestimmten Umsatz erhalten Stammkunden und übrige Kunden den gleichen Rabattsatz zugewiesen.
- 2 Ein Stammkunde mit 2.000 EUR Umsatz erhält einen Rabatt von 10 %.
- 3 Kunden, die nicht Stammkunden sind, erhalten maximal 5 % Rabatt.
- 4 Stammkunden erhalten mindestens 10 % Rabatt.
- 5 Ein Kunde mit 7.000 EUR Umsatz, der kein Stammkunde ist, erhält 10 % Rabatt.

c)

Das Parkhaus verfügt über 100 Stellplätze. Jeder Platz ist mit einem Sensor ausgestattet, der angibt, ob ein Fahrzeug auf diesem Platz steht. Die Sensoren werden von der Funktion „sensor\_lesen(Stellplatznummer)“ abgefragt. Diese gibt den Status „frei“ oder „besetzt“ zurück.

Zur regelmäßigen Abfrage der Parkhausbelegung soll ein Programm geschrieben werden, das bei einer Belegung von weniger als 90 % der Parkplätze „frei“ und sonst „belegt“ ausgibt. Folgendes fehlerhafte Struktogramm liegt für das Programm vor:



Welche der folgenden Auswirkungen hat der Fehler im Struktogramm?

Tragen Sie die Ziffer vor der zutreffenden Auswirkung in das Kästchen ein.

Die Anzeige ...

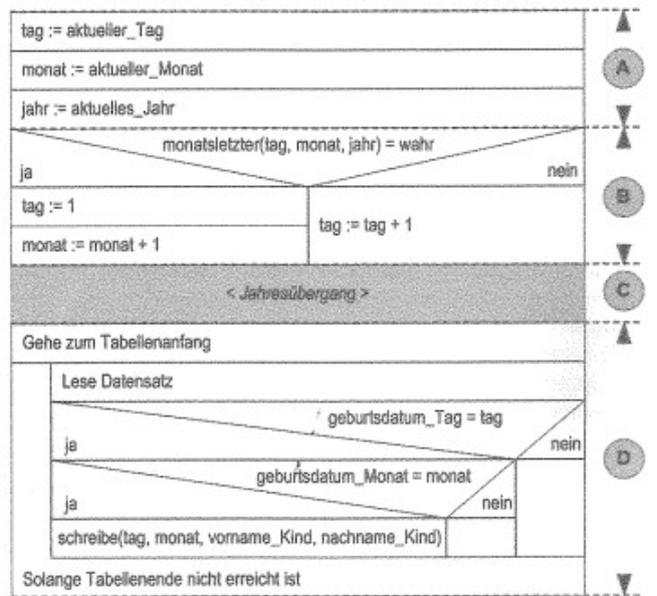
- 1) schaltet nie auf „belegt“.
- 2) schaltet nie auf „frei“.
- 3) schaltet schon bei 10 % Belegung auf „belegt“.
- 4) schaltet erst bei über 90 % Belegung auf „belegt“.
- 5) schaltet bei einem völlig leeren Parkhaus auf „belegt“.

d)

Die Leitung des Kindergartens möchte jeden Abend eine Übersicht der Kinder erhalten, die am nächsten Tag Geburtstag haben. Dazu wurde bereits nebenstehendes Struktogramm erstellt.

Hinweis: Die Programmabschnitte A, B und D wurden für die Aufgabe 3.3 markiert.

Die Funktion *monatsletzter(tag, monat, jahr)* wurde bereits erstellt und gibt „wahr“ aus, wenn tag.monat.jahr der letzte Tag des Monats ist, sonst „falsch“.

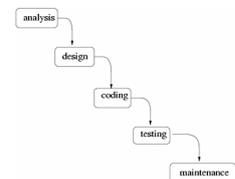


Das Struktogramm muss in dem grau markierten Bereich C noch ergänzt werden, um einen Übergang in das nächste Jahr zu gewährleisten.  
 Welcher der folgenden Struktogrammteile muss ergänzt werden?  
 Tragen Sie die Ziffer vor dem zutreffenden Struktogrammteil in das Kästchen ein.

<p>1</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">monat = 12</td> </tr> <tr> <td style="width: 50%;">ja</td> <td style="width: 50%;">nein</td> </tr> <tr> <td colspan="2" style="text-align: center;">monat := monat + 1 jahr := jahr + 1</td> </tr> </table>	monat = 12		ja	nein	monat := monat + 1 jahr := jahr + 1		<p>2</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">monat = 13</td> </tr> <tr> <td style="width: 50%;">ja</td> <td style="width: 50%;">nein</td> </tr> <tr> <td colspan="2" style="text-align: center;">jahr := jahr + 1</td> </tr> </table>	monat = 13		ja	nein	jahr := jahr + 1		<p>3</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">monat = 13</td> </tr> <tr> <td style="width: 50%;">ja</td> <td style="width: 50%;">nein</td> </tr> <tr> <td colspan="2" style="text-align: center;">monat := 1 jahr := jahr + 1</td> </tr> </table>	monat = 13		ja	nein	monat := 1 jahr := jahr + 1		
monat = 12																					
ja	nein																				
monat := monat + 1 jahr := jahr + 1																					
monat = 13																					
ja	nein																				
jahr := jahr + 1																					
monat = 13																					
ja	nein																				
monat := 1 jahr := jahr + 1																					
<p>4</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">jahr &gt; aktuelles_jahr</td> </tr> <tr> <td style="width: 50%;">ja</td> <td style="width: 50%;">nein</td> </tr> <tr> <td colspan="2" style="text-align: center;">jahr := aktuelles_jahr</td> </tr> </table>	jahr > aktuelles_jahr		ja	nein	jahr := aktuelles_jahr		<p>5</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">monat = 1</td> </tr> <tr> <td style="width: 50%;">ja</td> <td style="width: 50%;">nein</td> </tr> <tr> <td colspan="2" style="text-align: center;">jahr := jahr + 1</td> </tr> </table>	monat = 1		ja	nein	jahr := jahr + 1									
jahr > aktuelles_jahr																					
ja	nein																				
jahr := aktuelles_jahr																					
monat = 1																					
ja	nein																				
jahr := jahr + 1																					

**Aufgabe 8 - Aufgaben zu Vorgehensmodellen**

- a) Bei der Anwendungsentwicklung gibt es das klassische Wasserfallmodell als Vorgehensmodell. Skizzieren Sie dieses.
- b) Welche Kritikpunkte gegen die alten eher statischen Vorgehensmodelle wurden laut?
- c) Nennen Sie neben dem Wasserfallmodell noch andere eher klassische Vorgehensmodelle.
- d) Nennen Sie ein neues eher dynamisches Vorgehensmodell zur Anwendungsentwicklung.
- e) In den Vorgangsmodellen wird oft zwischen einer Analyse- und einer Design-Phase unterschieden. Was ist der Unterschied?



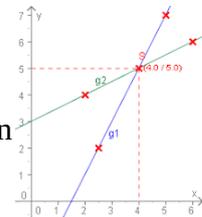
## Aufgabe 9 - Zusatzaufgaben

Lösen die folgenden Aufgaben mit Python. Erstellen Sie geeignete Diagramme zur Visualisierung.



- a) Verschiedene Datenlogger erzeugen Textdateien, in denen Dezimalzahlen stehen. Pro Datei kann es viele Zahlen geben, wobei jede Zahl in einer eigenen Zeile steht. Bei manchen Dateien wird ein Punkt als Dezimalkennzeichen genutzt (3.14), bei manchen Dateien wird ein Komma als Dezimalkennzeichen genutzt (3,14). Erstellen Sie mit Python einen Konvertierer, der in beide Richtungen konvertieren kann. Speichern Sie das Programm unter dem Namen **aufgabe\_ls1\_09.py** ab.

- b) „**Geradenschnittpunkt**“ - Erstellen Sie ein Programm zur Berechnung des Schnittpunkts zweier Geraden:  
Zuerst werden die Koeffizienten der beiden Geraden eingelesen.  
Anschließend berechnet das Programm den Schnittpunkt der beiden Geraden und gibt diesen aus. Speichere das Programm unter dem Namen **aufgabe\_ls1\_10.py** ab.



- c) „**Tankfüllung**“ - Gegeben ist ein Tank in Form eines liegenden geraden Zylinders mit dem Radius  $r$  und der Länge  $l$ , sowie die Füllhöhe  $h$ . Erstelle ein Programm, welches den Radius, die Länge und die Füllhöhe einliest und anschließend das Volumen (Tankinhalt) ausgibt. Speichere das Programm unter dem Namen **aufgabe\_ls1\_11.py** ab.

