

Programmierung mit Python

Übung 1: Einstieg

0. Aufgabe: Code Verständnis

Was geben die folgenden Programm-Zeilen auf die Console aus?

<pre>a = 4.6 b = 2 c = 1/10 print(round(a + b - c, 1)) print(round(a - 2*b + c, 1)) print(round(a / c, 1)) print(round(8 / 3, 3)) print(8 % 3) print(pow(2,3))</pre>	<pre>if a < b: print(a) print(a == b) print(a <= b) z = a > b and b > c print(z)</pre>	<pre>text = "abcxyz" print(len(text)) print(text[1]) print(text[-1]) print(text[1:4])</pre>
---	---	--

1. Aufgabe: Summe zweier Zahlen

Schreiben Sie ein Programm, welches mittels der input-Funktion zwei Zahlen einliest und dann in ganze Zahlen (int) umwandelt. Das Programm soll die Summe der beiden Zahlen berechnen und mit print() ausgeben.

2. Aufgabe: Mittelwert

Schreiben Sie ein Programm, welches vier Zahlen einliest, in Zahlen (float) umwandelt und deren Mittelwert berechnet und ausgibt. Runden Sie das Resultat auf 2 Stellen genau.

3. Aufgabe: Netto-Brutto-Rechner

Schreiben Sie ein Programm, welches von einer Ware den Preis ohne MWST einliest. Das Programm soll den Preis inklusive 7.7% Mehrwertsteuer ausgeben. Runden Sie das Resultat auf 2 Stellen genau.

4. Aufgabe: Rabattrechner

Schreiben Sie ein Programm, welches den Normalpreis einer Ware (float) und den Rabatt (int) einliest und den Endpreis ausgibt. Runden Sie das Resultat auf 2 Stellen genau. (Zusatz: Fehlermeldung bei negativem Preis oder Rabatt >= 100%)

5. Aufgabe: Rückgeld

Schreiben Sie ein Programm, welches den Preis und den bezahlten Betrag einliest und daraus das Rückgeld berechnet. Falls zu wenig bezahlt wurde, soll das Programm eine Warnung ausgeben. Runden Sie das Resultat auf 2 Stellen genau.

6. Aufgabe: Kasse

Schreiben Sie ein Programm, welches den Preis von 3 Artikeln und den bezahlten Betrag einliest und daraus das Rückgeld berechnet. Falls zu wenig bezahlt wurde oder falls einer der Artikel einen negativen Preis hat, soll das Programm eine Warnung ausgeben. Runden Sie das Resultat auf 2 Stellen genau.

7. Aufgabe: Zinsberechnung

Schreiben Sie ein Programm, welches einen Geldbetrag (float) und einen Zinssatz (int) einliest. Die Ausgabe des Programms ist der Betrag plus Zinsen nach 3 Jahren. Runden Sie das Resultat auf 2 Stellen genau. (Zusatz: Fehlermeldung bei negativem Geldbetrag oder Zins $\geq 20\%$)

8. Aufgabe: Quader

Schreiben Sie ein Programm, bei dem die Breite, Länge und Höhe eines Quaders eingelesen wird. Das Programm soll dann die folgenden Werte ausgeben:

- das Volumen (Breite * Länge * Höhe)
- die Oberfläche $2 * (\text{Breite} * \text{Länge} + \text{Breite} * \text{Höhe} + \text{Länge} * \text{Höhe})$
- die Raumdiagonale = Quadrat-Wurzel von $(\text{Breite}^2 + \text{Länge}^2 + \text{Höhe}^2)$

(Zusatz: Fehlermeldung bei negativen Eingaben)

9. Aufgabe: Grundstückspreis

Schreiben Sie ein Programm, welches die Länge und Breite eines Grundstücks, den Quadratmeterpreis und die Maklerprovision in %, einliest und daraus den Grundstückspreis berechnet und ausgibt. (Zusatz: Fehlermeldung bei negativen Eingaben oder Maklerprovision $> 15\%$.)

10. Aufgabe: Wort umdrehen

Schreiben Sie ein Programm, welches ein Wort mit vier Buchstaben einliest, und dann das Wort in umgedrehter Richtung wieder ausgibt. (Zusatz: Fehlermeldung, falls das Wort zu lang oder zu kurz ist.)

11. Aufgabe: Wort halbieren

Schreiben Sie ein Programm, welches ein Wort in der Mitte zerteilt, z.B. Hochzeit \rightarrow Hoch – zeit. (Falls die Wortlänge ungerade ist, soll das erste Wort das längere sein, z.B. Firlifanz \rightarrow Firl – fanz.)

12. Aufgabe: Wort verlängern

Schreiben Sie ein Programm, welches ein Wort einliest und dann die ersten vier Buchstaben hinten an das Wort anhängt. Das so erzeugte Wort ist dann etwa um die Hälfte länger, z.B. *werkzeug* \rightarrow *werkzeugwerk*

13. Aufgabe: Wort zerhacken 1

Schreiben Sie ein Programm, welches ein Wort einliest und dann jeden zweiten Buchstaben löscht. Das so erzeugte Wort ist dann nur noch etwa halb so lang, z.B. *werkzeuge* \rightarrow *wrzue*.

14. Aufgabe: Wort zerhacken2

Schreiben Sie ein Programm, welches ein Wort einliest, jeden zweiten Buchstaben löscht und dann die gelöschten Buchstaben wieder hinten anhängt. Das so erzeugte Wort ist dann wieder gleich lang, z.B. *werkzeuge* \rightarrow *wrzueekeg*.