

Aufgabenstellung: Kennenlernen und Einüben der Python Datentypen

Sie haben in der Vorlesung die verschiedenen Datentypen und ihre Eigenschaften kennengelernt. Ziel dieser Übung ist es nun, diese anzuwenden und die Erkenntnisse zu festigen.

Aufgabe 1: Volumen- und Flächenberechnung von geometrischen Körpern

Schreiben Sie ein Python Programm, das sowohl das Volumen als auch die Oberfläche von folgenden dreidimensionalen Körpern berechnet:

- Würfel
- Quader
- Quadratische Pyramide

Um dem Programm etwas mehr Flexibilität zu verleihen können Sie die geforderten Werte (Seitenlängen) über eine Eingabeaufforderung einlesen. Nutzen Sie dazu die Funktion `input()` wie folgt:

```
eingabe = input("Aufforderungstext: ")
```

Der zurückgegebene Wert ist vom Typ `str` und muss für Berechnungen in einen `int` oder `float` konvertiert werden. Dazu nutzen Sie die Funktion:

```
int(...) oder float(...)
```

Sollten mathematische Funktionen erforderlich sein, so muss das `math` Paket durch die folgende Zeile zu Beginn des Python Programms eingebunden werden:

```
import math
```

Anschließend können Funktion wie bspw. `sqrt` wie folgt genutzt werden:

```
math.sqrt(x)
```

Hinweis: Führen Sie die Datenerfassung und Berechnung für jeden geometrischen Körper einzeln durch.

Aufgabe 1a: Verwendung von ganzzahligen Seitenlängen.

Verwenden Sie Variablen, die Ganzzahlen aufnehmen und berechnen Sie anschließend die gesuchten Größen. Geben Sie die Ergebnisse jeweils auf der Konsole aus.

Aufgabe 1b: Verwendung von beliebigen (Gleitkommazahl) Seitenlängen.

Verwenden Sie Variablen, die Gleitkommazahlen aufnehmen und berechnen Sie anschließend die gesuchten Größen. Geben Sie die Ergebnisse jeweils auf der Konsole aus.

Aufgabe 2: Nummerierung von Monatsnamen

Die Monate unseres Kalenders gehen von Januar bis Dezember. In der Softwareentwicklung wird häufig eine numerische Repräsentation eines Monats benötigt. Dabei kann der Januar als 0 oder auch als 1 codiert werden.

Schreiben Sie ein Python Programm und legen Sie die Monatsnamen in einem geeigneten Datentyp ab. Ergänzen Sie anschließend das Problem um einen weiteren Datentyp, der die Nummerierung von bspw. 0 bis 11 beinhaltet.

Erzeugen Sie nun eine Zuordnung von Monatsnamen zu Monatszahl und geben Sie diese auf der Konsole aus.

Die Ausgabe sollte in etwa wie folgt aussehen:

```
(( 'Januar', 0), ('Februar', 1), ('März', 2), ('April', 3), ('Mai', 4), ('Juni', 5), ('Juli', 6), ('August', 7), ('September', 8), ('Oktober', 9), ('November', 10), ('Dezember', 11))
```

oder

```
[('Januar', 0), ('Februar', 1), ('März', 2), ('April', 3), ('Mai', 4), ('Juni', 5), ('Juli', 6), ('August', 7), ('September', 8), ('Oktober', 9), ('November', 10), ('Dezember', 11)]
```

Konzeptskizze:

Machen Sie sich vor der Implementierung Gedanken zu Ihrem Konzept, wie Sie die Aufgabe lösen möchten!

Aufgabe 3: Zuordnung Bundesland und Hauptstadt

Schreiben Sie ein Python Programm, das eine Zuordnung von Bundesland und Hauptstadt (bspw. Bayern – München) übernimmt. Füllen Sie einen Datentyp mit den 16 Bundesländern aus Deutschland. Durch die Eingabe eines Bundeslandes in der Konsole soll das Programm die zu dem Bundesland gehörende Hauptstadt auf der Konsole ausgeben.

Konzeptskizze:

Machen Sie sich vor der Implementierung Gedanken zu Ihrem Konzept, wie Sie die Aufgabe lösen möchten! Überlegen Sie sich einen geeigneten Datentyp für diese Aufgabenstellung.

Zusatzfrage: Was passiert, wenn nach einem nicht vorhandenen Bundesland gefragt wird?