

AUFGABEN SESSION 2:

AUFGABE 1: VARIABLENAUSGABE ÜBER DEN EINGABESTROM

1. Überlege dir beliebige Variablen mit passenden Namen, befülle sie aber nicht selber mit Werten, sondern realisiere sie mithilfe des Eingabestroms (siehe Skript: `input()`). Befülle die Variablen also im laufenden Programm mit Werten.
2. Gebe die Variablen aus, um zu prüfen, ob die Variablen mit den Benutzereingaben befüllt wurden.
3. Versuche den Steckbrief aus Aufgabe 4 der letzten Session nun mithilfe des Eingabestroms zu realisieren! Zuerst sollen die persönlichen Informationen abgefragt und danach ausgegeben werden.

AUFGABE 2: ERSTER EIGENER TASCHENRECHNER

1. Deklariere zwei Variablen und weise ihnen beliebige Werte zu.
2. Addiere (du kannst auch subtrahieren, multiplizieren, dividieren) die beiden Variablen und speichere das Ergebnis in einer dritten Variablen ab!
3. Gib das Ergebnis aus!
4. Wenn das alles funktioniert, nutze `input()` um die Variablen durch Benutzereingaben befüllen zu können (d.h. durch Eingabe in der Konsole) und nicht durch direkte Eingabe im Programmcode!
Hinweis: Beachte hierbei das **explizite type casting** (siehe Folien und Skript)
5. Verbessere die Benutzerfreundlichkeit deines Programms und gebe Aufforderungen aus, wenn der Benutzer etwas tun soll oder wenn das Programm etwas gemacht hat! Die Rede ist von Phrasen wie „Bitte erste Zeile eingeben:“, „Bitte zweite Zahl eingeben:“, „Das Ergebnis ist:“ ...

AUFGABE 3: IF-ELIF-ELSE KONSTRUKT

Versuche das if-elif-else Konstrukt und seine Funktionsweise zu verinnerlichen, indem du das Zahlenvergleichsprogramm (Programmiernote) aus dem Kurs nachbaust ohne das Skript / die Folien zu verwenden. Arbeite dieses Mal allerdings mit der Hilfe von `input()` zur Befüllung der Variablen (`programmier_Note`), statt mit der direkten Zuweisung von Werten im Code.

AUFGABE 4: KOPFRECHNUNGSÜBUNGEN DURCH IF-ELSE

1. Erstelle mithilfe von Variablen und input-Funktionen ein Programm, welches den Benutzer auffordert, bestimmte Kopfrechnungen durchzuführen und das Ergebnis in das Programm einzutragen. (Evtl. 3 Rechnungen hintereinander, in der Art „Was ergibt $35/5+3*10$?“)

2. Mithilfe von if-Abfragen soll nun geprüft werden, ob die Eingaben korrekt waren und entsprechend ausgegeben werden, ob der Benutzer richtig gerechnet hat!

AUFGABE 5: ERWEITERUNG DER AUFGABE 4



Schreibe das gleiche Programm aus Aufgabe 4! Verwende zum Prüfen der Eingaben aber nur folgende Hilfsmittel:

- Verwende nur die Abfragen „if“ und „elif“
- Verwende nur die Operatoren „==“ und „!=“

AUFGABE 6: ALTERSPRÜFUNG



Erstelle ein Programm, welches nach dem Angeben des Alters eines Anwenders prüft, welche Arten von KFZ gefahren werden dürfen. Nach dem Eingeben des Alters sollen unterschiedliche Ausgaben gemacht werden, je nach dem was der Anwender fahren darf. Folgendes soll das Programm unterscheiden und dann jeweils ausgeben können:

- Jünger als 14 – „Du darfst noch nicht Mofa fahren“
- Ab 14 und jünger als 16 – „Du darfst Mofa fahren aber nicht Moped!“
- Ab 16 und jünger als 18 Jahre – „Du darfst Moped fahren aber nicht Auto!“
- 18 Jahre oder älter – „Du darfst endlich Auto fahren!“

Tip: Bedingungen in Abfragen können auch verknüpft werden! Recherchiere hierbei den Nutzen von „and“ und „or“ bei if-Abfragen und in Programmiersprachen allgemein!

HARDCORE AUFGABE 7: PRÜFUNG EINER GANZZAHL-DIVISION



1. Erstelle ein Programm, welches eine beliebige Ganzzahl durch eine weitere Ganzzahl teilen kann! Die beiden Zahlen soll der Benutzer durch eine Eingabe beliebig wählen dürfen!
2. Das Programm soll nun erkennen ob das Ergebnis dieser Rechnung einen Rest beinhaltet. (z.B. 15/7 ergibt ja 2 Rest 1) Existiert kein Rest, soll der Benutzer ebenfalls entsprechend informiert werden.

Wichtiger Tipp: Es gibt in Python den sog. „Modulo“-Operator, ohne den du hier nicht weiterkommst ;-)