

Aufgabe Reimerei:

Lösungsidee: Für jede Bedingung gibt es eine Hauptfunktion. Wenn die Bedingung erfüllt ist, wird ein boolescher Wert zurückgegeben. Zum Schluss wird überprüft, ob alle Wahr sind. Wenn das der Fall ist, wird das Wort auf der Konsole ausgegeben.

Umsetzung: Die Umsetzung erfolgt über Python. Es gibt zwei Grundfunktionen: Die erste sucht die Vokalgruppen in einem Wort und gibt sie anschließend als Dictionary zurück. Das sieht dann bei dem Wort "Schiene" so aus: {'ie25283': [3, 4], 'e19900': [6, 6]} (Vokalgruppe, ID, startIndex, endIndex). Die zweite wichtige Funktion gibt die Buchstaben ab der maßgeblichen Vokalgruppe zurück: Schiene -> iene. Aus diesen zwei Funktionen setzt fast das ganze Programm zusammen. In den Bedienungsfunktionen wird das Grundwort mit dem möglicherweise passenden Wort verglichen. Dazu werden die eben erläuterten Grundfunktionen benutzt.

```
def finde_vokale(wort):
    "Sucht die Vokalgruppen in einem Wort, return {'ie25283': [3, 4], 'e19900': [6, 6]}"
    vokale_index = dict()
    index = 0
    while index != len(wort):
        buchstabe = wort[index]
        vokal_gruppe = ""
        vokal_s_e = []
        # Es wird eine Id gebraucht, weil gleiche Vokale mehrmals vorkommen können und sie somit überschrieben werden
        # können
        id = (str(uuid.uuid4()).fields[-1])[:5])

        if buchstabe in vokale:
            vokal_gruppe += buchstabe
            vokal_s_e.append(index)

        # zb Biene -> e am ende -> sucht nach vokalen hinter dem ersten Vokal aber Wort ist zu ende -> IndexError
        try:
            index += 1
            buchstabe = wort[index]

            if buchstabe in vokale:
                vokal_gruppe += buchstabe
                vokal_s_e.append(index)
                vokale_index[vokal_gruppe + id] = vokal_s_e
                index += 1
            else:
                vokal_s_e.append(vokal_s_e[0])
                vokale_index[vokal_gruppe + id] = vokal_s_e
        except IndexError:
            vokal_s_e.append(vokal_s_e[0])
            vokale_index[vokal_gruppe + id] = vokal_s_e

    else:
        index += 1
    #print(vokale_index)
    return vokale_index
```

Main Funktion:

```
woerter = []

# liest die Textdatei ein und speichert die Wörter in einer Liste
with open(r"BWINF_WETTBEWERB_2022\Reimerei\reimerei0.txt", "r", encoding="utf-8") as f:
    zeile = f.readlines()
    for word in zeile:
        woerter.append(word.strip())

# für jedes Wort in der Liste werden dann alle Wörter überprüft
for grundwort in woerter:
    # da es mehrere passende Wörter geben kann, werden sie hier zwischengespeichert
    all_pass = ""
    for moeglichesWort in woerter:
        moeglichesWortc = str(moeglichesWort).lower()
        grundwortc = str(grundwort).lower()

        #print(grundwort, moeglichesWort)

        be1 = bedingung1(grundwortc, moeglichesWortc)
        be2 = bedingung2(grundwortc, moeglichesWortc)
        be3 = bedingung3(grundwortc, moeglichesWortc)

        if be1 == True and be2 == True and be3 == False:
            all_pass += moeglichesWort + "; "

    print(f"Wort: {grundwort} / Reime: {all_pass}")
```

Ergebnis:

Wort: bemühen / Reime: glühen;
Wort: Biene / Reime: Hygiene; Schiene;
Wort: breitschlagen / Reime:
Wort: glühen / Reime: bemühen;
Wort: hersagen / Reime:
Wort: Hygiene / Reime: Biene; Schiene;
Wort: Knecht / Reime: Recht;
Wort: Recht / Reime: Knecht;
Wort: Schiene / Reime: Biene; Hygiene;
Wort: schlank / Reime:
Wort: Schwank / Reime: