

Ich sehe, Sie kennen sich nun auch mit **Zugriffen auf Listen** aus. Zuvor hatten Sie ja bereits Funktionen kennengelernt, um **Elemente** an Listen **anzuhängen** bzw. zu **löschen**.



Mit den folgenden **Vertiefungsaufgaben** können Sie nun all Ihre **Kompetenz** im Umgang mit **Listen** unter Beweis stellen.



Ralf Meier -
Abteilungsleiter
Software Code
Solutions

Aufgabe: Buchstabensalat entschlüsseln

- Finden Sie mit geeigneten **Slicing-Operationen** heraus, welche zwei Sätze sich hinter dem folgenden Buchstabensalat verstecken:

"DIenr diesmt Sdienrn eb eisstte HLielhfree
rz,u rd eSre lsbiscthh inlafceh iumnnde rn
aucnhs eürbee rofbleürsstieg Mmaaxcihmte.."

- Geben Sie beide Sätze auf der Konsole aus.
- **HINWEIS:** Laden Sie sich hierfür die **Codevorlage** herunter.




 10 Minuten

Aufgabe: Palindrome erkennen

- Ein **Palindrom** ist eine **Zeichenkette**, die sich vorwärts genauso liest wie rückwärts, z.B. das Wort „Radar“ (Groß- und Kleinschreibung wird dabei nicht beachtet).
- Erstellen Sie ein **Python-Skript**, das mithilfe von **String-Slicing** bestimmt, ob ein vorgegebenes Wort ein Palindrom ist. Gehen Sie davon aus, dass alle Wörter klein geschrieben werden.
→ z.B. `wort = "Madam"`
- Testen Sie Ihr Programm für folgende Wörter:
 - *Palindrome: radar , madam, rentner, ...*
 - *Keine Palindrome: messer, tisch, sonne,*



Zusatzaufgabe: Passen Sie Ihr Programm an, sodass Groß- und Kleinschreibung nicht beachtet wird → z.B. soll „Rentner“ ebenfalls ein Palindrom sein.

 **15 Minuten**