Python Algorithmen für Fahrzeugtechnik

To Do

* Logic
* Python commandline tool
* Conda
* Pycharm
* Keyword self
* Do:
  + Aufgaben aus Kapitel 2
  + Python CmdLine Tool testen
  + Erstellen eigener vollständiger Zusammenfassung
  + Auswendig Learn Session

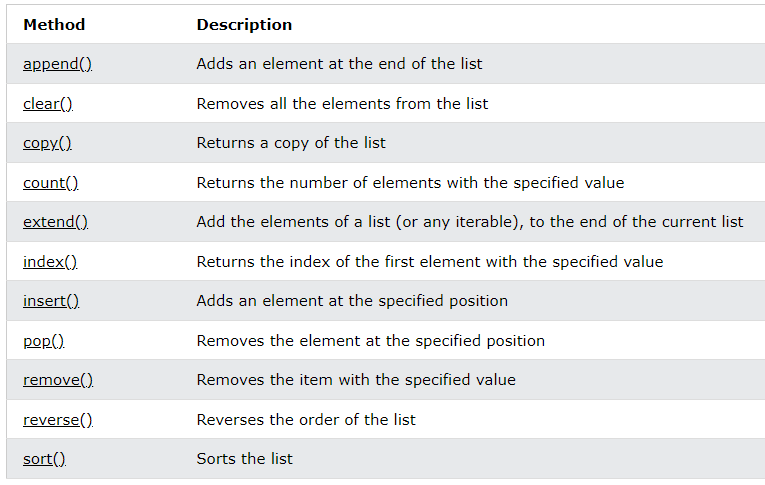
Topics to Study

* Script
* Vorlesung Videos (externe Quellen besprochen)
* Aufgaben aus Script (alle Themen)
* Online Test (Prüfer Link)
* Zusammenfassung aus Studydrive (eigene?)
* W3 Schools, eigenes Projekt

Zusammenfassung

1) Python Code General Stuff

* Method & Function
  + Function: def function(): …
  + Method: same, but in classes:
    - Used for object for which it is called
    - Access to class data
* Try Except
  + try: code that’s excecuted
  + except: triggered when error occurs (except Exception)
  + else: triggered after if no exception
  + finally: always triggerd after
  + –
  + If
    - Raise Exception (own Exception defined)
* Lists [ ]
  + Mutable, ordered, different entry types



* Tuples ( )
  + Immutable, ordered
* For loops
  + [x for x in range(0,11) if 1<3]
  + Continue
  + else
* Global
* Multi return statements are tuples (without brackets)
* Pass
  + Keyword for empty codeblock like for-loops to avoid error

2.) Methods Functions

* Sub()
* Choice()
* Random()
* Randrange()
* Find()
* Join()
* Encode(), decode()
* ~
* Print(object(s), sep=’seperator’, file=file, flush=flush)

3.) Clean Code

* Quote one
  + "Writing clean code is what you must do in order to call yourself a professional. There is no reasonable excuse for doing anything less than your best" [[Martin2008]](https://nbviewer.org/github/StephanRhode/py-algorithms-4-automotive-engineering/blob/master/references.bib).
* Quote two
  + "Programming is the art of telling another human being what one wants the computer to do." [Donald Knuth]
* Quote three
  + "Computers are good at following instructions, but not at reading your mind." [Donald Knuth]
* PEP8 = Style Guide for Python