

Support Team → Katalav izin haga hader.

Melden /

legen:

Selbsttest

... ; - - - . . ,

Frau Christina Meyer

KL 337 → Hilfe

Datei: Ignatius-Abrasion.docx ← Hilfe benötigen.

Ordnen → , , , Tugay,

Buch: "Einführung Python 3"

Nächste Woche : 20.09 } bis 17:10  
Ende : 03.10 ..

Bei Krankheit / Bewerbung /: → melden:

Nachmeldung → Unterguppen

↳ ~~Tutor~~

Abschlussarbeit → Projekt ← Vorlesung Woche 3

↳ mindest Anforderungen (Punkte)

↳ Woche 4

- Crawler

- Machine Learning / Deep Learning

→ Anaconda Installation

↳ has its own Package Management System

↳ Anaconda environment / Umgebung

Anaconda Navigator:

Version: PyCharm 2023.2.1

↳ Environment (Python) (Conda)

(Pipenv  
(venv))

:

Python Crash Course

at home download again

11/11

Dienstag, 12.09.2023

Python history.

Python Syntax

Class / Klasse ~ Klassen

↳ Bauplan für ein Objekt

Zusammengehörigkeit durch Einrückung

↳ Einrückung = indenting, indentation

Mehrfache Zuordnung

Typen in Python → data type

Umwandlung von Typen → type casting

Umwandlung

type(variable\_name)

~~anumber = int('5')~~ → anumber is 5 w/ int

\* int(x) ≠ is not rounding, but cutting.

anumber = int('5.973') → error

anumber = float('5.973') → works :)

Packages

Namespaces

Conda install -C anaconda pandas

Mittwoch, 13.09.2023

String

$\times [5 : 8]$   $\rightarrow \times [5], \times [6], \times [7]$   
↑      ↘  
Inklusiv    exklusiv

(\*)  $\text{update Day 2, Day 3 Vormittag, Day 3}$   
file

$b = \{ \text{bla bla} \{ 13 \text{ bla bla} \{ 03 \text{ bla bla} \}$   
↑                         ↑  
index                    index  
1                        0

Datum - (time & date)

- UTC?

- ISO 8601  $\rightarrow$  Tag, Monat, Jahr
- USA  $\rightarrow$  Monat, Tag, Jahr

package datetime

Zeit-jetzt = datetime.datetime.now()  
~~~~~      ~~~~  
package      class

(\*) File: - Kurs-einweisung (di folder - Kurseinweisung)

- Benutzung untersetzen (di folder Stellenrecherche\_BenutzPath)

- beitreten hängt di.folder.. > Job

(Übung (Solutions))

`dt.datetime.strptime(test_date, "%d-%m-%Y %H:%M:%S")`

~~~~~  
String

~~~~~  
Format als Text  
↓  
are there "\n"

Datei löschen:

.txt, .csv, .xlsx

`input()`

`open()`

`myData = open('bsp.txt', 'r')`

~~~~~  
Path as string → remember backslash double

`open(file, "xx", encoding="utf-8")`

↓  
w ← write  
a ← append  
r ← read  
r+ ← read & write

Output txt

`open("bsp2.txt", "w").write(bspString)`

`myData = open("bsp.txt", "r+")`

`myData.write(bspString)`

↑  
String !!!

```
print(f"    ", %(), )  
print({f" ... { } ... { }"})  
print("    ", "    ", "    ", "    ", "    ", "    ", Sep="...")  
                                ↑  
                                separator
```

## list, tuple, dict

mylist = [1, 5, 7, 3, 9]  
 -5 - - - -2 +  
 0 1 2 . 4 5  
 = index  
 = index

mylist.insert(1, 99)  
 ↑      ↑  
 index   value

dict

L keys

nd = {"key": value, "key2": value}

mydict = {"Zielhobby": "Maya", "Alter": 31, "Alter": 7}

↑  
 überschreiben.

len (...) ← Anwendung

len ← Verwendung

sorted(mylist, key=len)

↑  
 Verwurf

gallen

- - +  
 ↓

option  
 sorted(" " = " ")

dict("Name": " ", "Left": " ")

Mittwoch (1)

- File:
- Kursanmerkung (die folgen Kursanmerke) ✓
  - Bewertung Übungen (die folgen Sitzungsschreiber) ✓
  - bewerben hieran die folgen ✓
  - ~~Übung~~ Folientüte & Juhu (Week 1, Day 3 ... ) ✓

for i in range(0, len(myString))

↑              ↑  
inclusive    exclusive

Zip-Funktion

↳ combine 2 lists

for i, j in zip(mylist1, mylist2):  
....

Enumerate Funktion

if-statement

if ...:

...

else: ~~pass~~

...

{     if ...:  
      ...  
   elif ...:  
      ...  
   else:  
      ...

List comprehension

mylist = [x for x in range(0, 8)]

mylist2 = [x for x in range(0, 8) if x%2 == 0 and x < 5]

function & docstring.

Dienstag, 19.09.2023

function:

global variable

Eigener "package" erkennt

Limitierter Projekt = niceout.py  
└ nice()

Import niceout  
niceout.nice("...")

Default parameter      Default parameter

Non-keyword arguments : \* args

↳ jeder Argument wird als Element einer tuples.

↓  
tuple with the length 1 → \* args

~~Keyword Arguments:~~

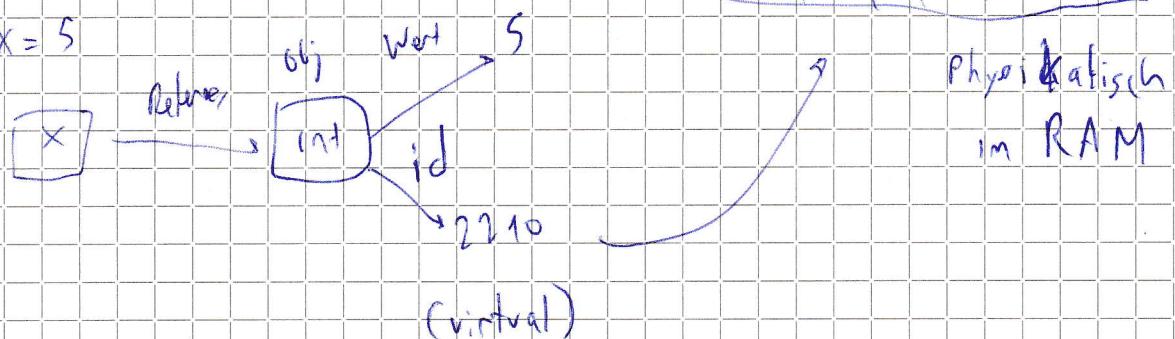
Keyword Arguments : \*\* kwargs

Package typing

```
def myfunction(x:int, text:str) → list:  
    out = [x, text]  
    return out,
```

function als paramet → function is first class objects.

## Reference

 $x = 5$ 

(25 ⚡) ↪ pick integer more than

fürth aus: Mutationen

Nach vollzogen Refenz innerhalb der für-th  
Referenz innerhalb führt → id

13 - 20

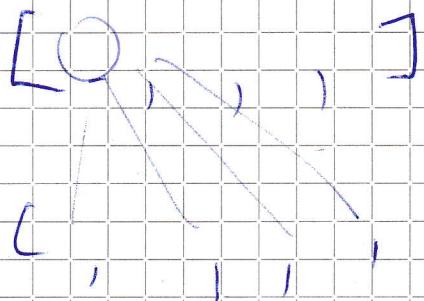
lambda fürt

map function → element wise

results = map [FUNCTION, mylist]

Freitag, 22.09.2023

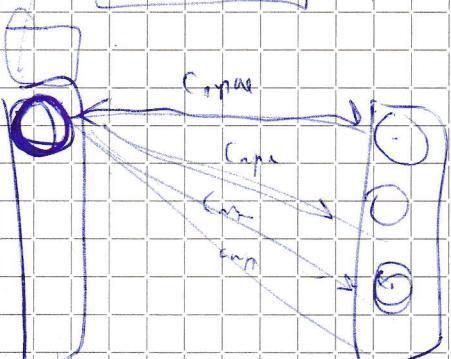
## Recursion



0, 1, 2, 3, 4



Dungeon Crawler



list

list

design:  
Same for next week

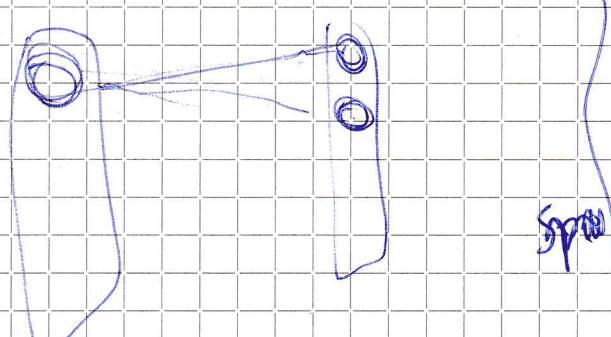
- What purpose...

text-based Adventure Game

text-based adventure Game

Console → interface

else



Montag, 25.09.2023

Imperative Programmierung → basis per basis & GO TO

Functional programming ≈ subset of declarative programming

Lazy Evaluation

↳ range

Objekte in Python

↳ Alles in Python ist ein Objekt

? Dungeon Crawler ?

Reflektive Programmierung ?

↳ know all scripts ...

Esoterische Programmiersprachen

filter function

for i in filter(...):

[ ]

try, except, finally }

try, except, else

Fehlerarten

↳ NameError, TypeError, MemoryError, OverflowError

~~REVIEW~~

Fehler type

except Type Error:

....

try:

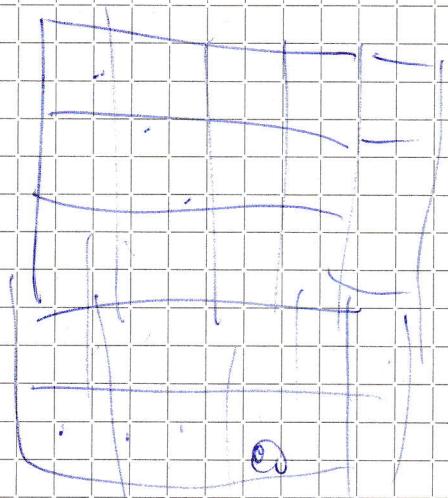
....

except Type Error:

except IndexError:

....

class Maze



Worm hole

L Locat coords  
(x, y)

class ~~walker~~  
walker



L locat Coords  
(x, y)

Dienstag, 26.09.2023

Raise Error

Class

Text-based adventure game

self & other

public	myfun()
protected	-myfun-
private	--myfun--()

Magic methods

Mittwoch, 27.09.2023

- function
- magic method
- method,      ↪ inside a class

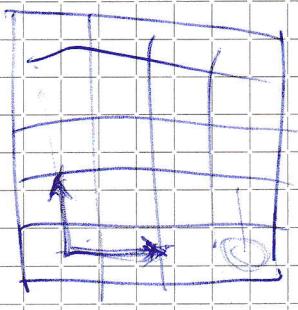
Magic methods,

↳ beschreiben grundlegende Eigenschaften der Klasse

mutable & immutable objects.

↳ ist der Speicherort fest oder variabel?

Speicherort  $\sim$  id



Donnerstag, 28.09.2023

immutable objects

Ist in tuple

Datentypen → immutabile class?

~~Inheritance~~

Inheritance ~ Vererbung

Korrigefragen @ hays.de  
0621 1788 1188 } Hays

## Inheritance

### Multilevel Inheritance

```
class Vehicle:
    def __init__(self, brand, model):
        self.brand = brand
        self.model = model
```

```
class Car(Vehicle):
    def __init__(self, brand, model, year):
        super().__init__(brand, model)
        self.year = year
```

```
class Opel(Car):
    def __init__(self, brand, model, year, engine_type):
        super().__init__(brand, model, year)
        self.engine_type = engine_type
```

### Multiple Inheritance

```
class motorcycle(MeanOfTransportation, Motor):
    pass
```

### Project Covid 19

↳ selecting a province

choose province near that prov

choose province far away from that prov

↳ stat descriptive: plotting...

↳ correlation

↳ optional: create prediction model... → plotting the picture

Access modifier :

- public
- Protected
- Private

Decorator

↳ wrapper

↳ static method

Dataclasses

GUI:

↳ tkinter is standard GUI in Python

↳ event-driven programming

import tkinter as tk

mywindow = tk.Tk()

alfateam-b@alfatraining.de

↳ until ketidakhadiran

Freitag, 06. 10. 2023

import re ← for regex (regular express.)

Yatlabo git  
↳ Covid ...

cph Dokumenten