

1a "best practice"

IDEEN

it works ✓

$f^u X = \{x\}$

✓ Typecheck

✓ Naming Skript

✓ Comments Ziel/Zweck

✓ Output-Infos

✓ README.mel mit Bsp.

↙ Correctness

✓ ↙ Error Handling z.B. Eingabe vorhanden?

1b UX "Usability":

1 DEBN ✓

urls.txt



start.py

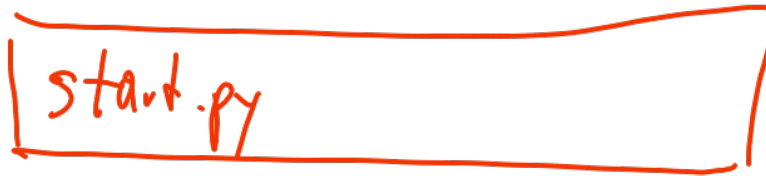


images/

img1.py
img2.py
img3.py

✓ Feedback

✓ order?
→ exit
→ create



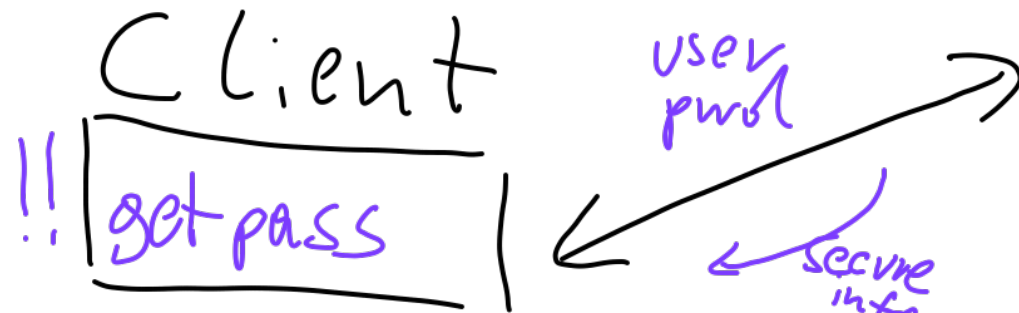
✓ defaults
→ opt. param

✓ input checks
✓ destructive "overwrite"?

✓ output
✓ err url? → crash x
→ except ✓

1c "Hard coded"
NEVER!

Server



python -m http.server

-d demo data

TODO: <URL>

(System)
(DNS)

Default

TOKEN

Env

Config

cmd-param

User
(PIPE)

8080

8080

1d input validation:

⚡ exec ⚡

? eval ?

Was fun?

- o) parsing, check
- a) blacklisting ⚡
- a) whitelisting ✓

Don't

UX ✓

Hard coded X

exec / using ✓

Try Mo

git

tool

ven ✓

... Mi

Data

mgmt

visualization

✓

evtl.
excel/files