

# Passwordmanager

Rania Hajjout (1360147)  
Issam Boutachdat (1352294)  
Luke Cefariello (1360053)

# Aufbau des Programms

Alle wichtigen Module importiert

Klassen erstellt

Methoden für Befehle erstellt (z.B generate)

Main-Part (mit Befehle und Methoden)

→ Generate

→ Store

→ Delete

→ Modify

→ List

→ Search

→ Change

→ Check

→ Exit

# Problem/ Lösung Befehl: „generate“

## Problem:

- Zeichenarten wurden hinzugefügt die NICHT angefordert wurden

## Lösung:

- Eine While eingebaut die solange die geforderten Zeichenkategorien hinzugefügt hat bis die Länge erreicht wurde

Problem/  
Lösung Befehl:  
„generate“

### Problem:

- Passwortlänge größer als gewünschte Länge

### Lösung:

- Vor jedem einfügen die Länge prüfen

# Befehl: „generate“

## Probleme – Lösungen

```
if laenge >= anzahl_anforderungen and anzahl_anforderungen >= 1:
    #Abfragen ob User gewisse Option haben möchte oder nicht
    while laenge > len(kombiniert):

        if soll_zahlen_enthalten:
            kombiniert += secrets.choice(zahlen)

        if soll_kleinbuchstaben_enthalten and laenge > len(kombiniert):
            kombiniert += secrets.choice(kleinbuchstaben)

        if soll_grossbuchstaben_enthalten and laenge > len(kombiniert):
            kombiniert += secrets.choice(grossbuchstaben)

        if soll_sonderzeichen_enthalten and laenge > len(kombiniert):
            kombiniert += secrets.choice(sonderzeichen)

    #Eingabe Passwortlänge mit for-schleife durchlaufen, um Optionen (siehe oben einzufügen und zu

    #hier wird das Passwort durchgemischt!
    pw = ''.join(random.sample(kombiniert,len(kombiniert)))

    print("Passwort: " + verdecken(pw) + " in Zwischenablage gespeichert.")

    zwischenablage_speichern(pw)
    time.sleep(0.2)
    passwortsicherheit(pw)

elif laenge == 0:
    print("Passortlänge muss mindestens 1 betragen.")
else:
    print("Sie können kein Passwort generieren mit " + str(anzahl_anforderungen) + " verschiedenen
```

# Passwortsicherheit

## Problem:

- Passwörter sollen nicht sichtbar sein

## Lösung:

- Einen Hard codierten Schlüssel für den „Ent-Verschlüssler“

# Passwortsicherheit

## Problem:

- Open-Source code

## Lösung:

- Masterpassword als Schlüssel für den „Ent-Verschlüssler“

# Passwortsicherheitsstufen

Erstmal ohne Variable “staerke” versucht, sondern mit print-Anweisungen

## Problem:

- es wurde nie die richtige Stäerke ausgegeben.

## Lösung:

- mit Variable “staerke” und einzelnen if-Abfragen, so wird jede Bedingung geprüft



# Passwortsicherheitsstufen

Erstmal alle Abfragen in der Methode passwortsicherheit eingefügt.

## Problem:

- zu unübersichtlich, es ist nicht in jeden zweig reingegangen → Fehler schleichen sich ein

## Lösung:

- Für jede Bedingung eigene Methode definieren und nur die Methode in passwortsicherheit() aufrufen

## Passwortsicherheitsstufen

Erstmal eigene Variablen deklariert,  
die die jeweiligen Zeichen enthalten.

### Problem:

- Nicht alle Sonderzeichen sind valide

### Lösung:

- Asciii-Tabelle, beinhaltet alle Zeichen, die man braucht.  
Sonderzeichen umgehen → nicht Zahlen, nicht Buchstaben

# Vielen Dank für Ihre Aufmerksamkeit

Rania Hajjout (1360147)  
Issam Boutachdat (1352294)  
Luke Cefariello (1360053)