

Budget-Rechner

Programmierprojekt Grundlagen Programmierung mit Python

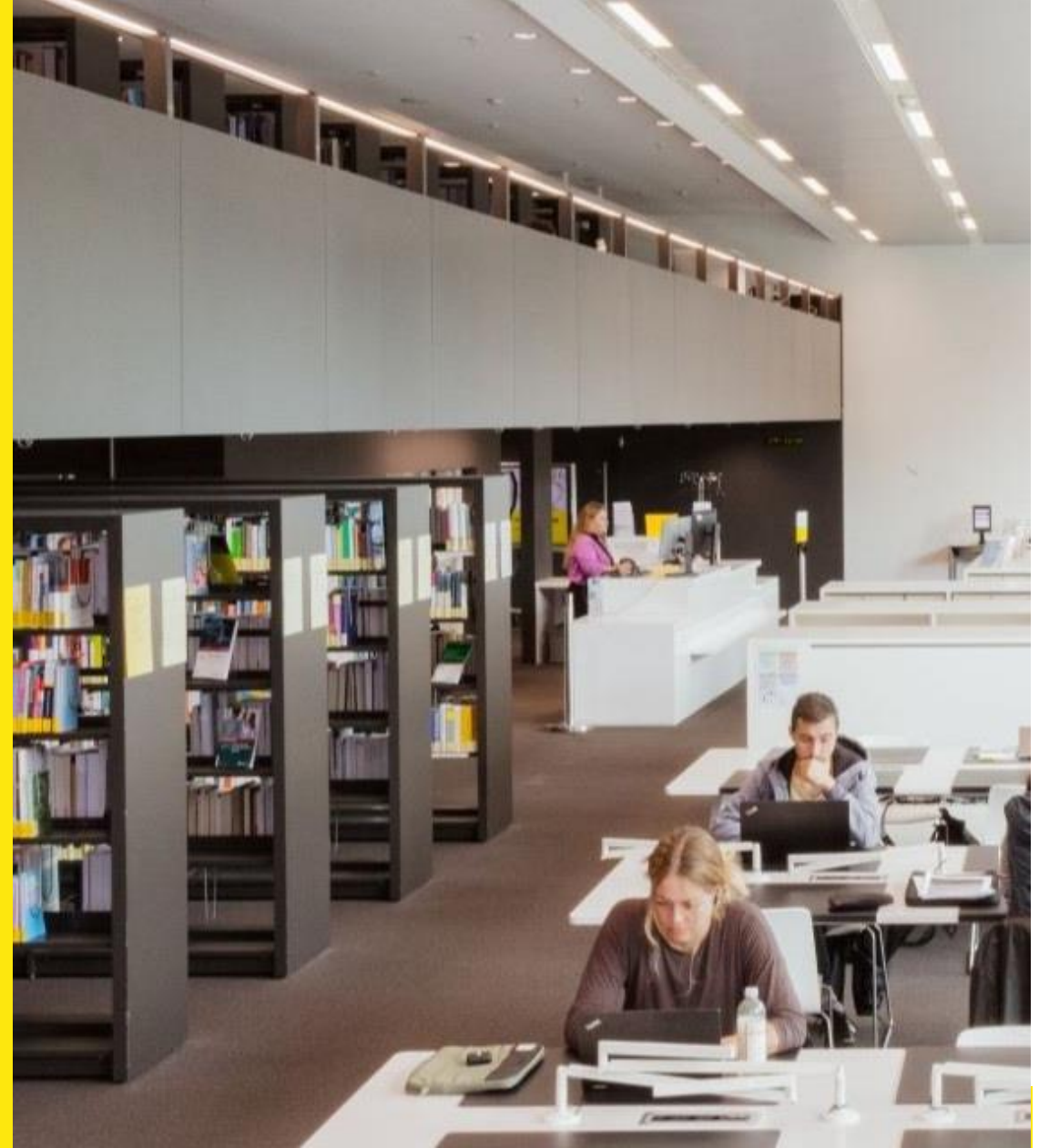
WI-TZT-25

Alessio Cusentino

Jana Kräuchi

Rouven Mallach

Dezember 2025



Ablauf unserer Präsentation

- Motivation
- Ziele und -funktionen
- Projektmanagement
- Live-Demo
- Fragerunde



Motivation

- Interesse an Finanzen und Geldvorgängen
- Vereinfachte Darstellung von Budgetverwaltung
- Möglichkeit, persönliche Sparziele einfacher zu planen und einzuhalten
- Tool mit Fokus auf die Hauptaufgabe



Ziele und Funktionen

Ziele

- Einfache Verwaltung von Einnahmen und Ausgaben.
- Übersichtliche Darstellung der finanziellen Situation einer Person.
- Möglichkeit, Budgets zu setzen und zu überwachen oder Sparziele einzuhalten
- Benutzerfreundliche Bedienung für alle Altersgruppen.

Funktionen

- Einnahmen und Ausgaben erfassen
- Übersicht über Verzehr- oder Sparquote
- Bearbeitung und Löschung sämtlicher Angaben jederzeit möglich
- Festlegung von Sparzielen
- Berechnung von Sparzielen
- Anzeige von Vermögen und Budget
- Export von Daten (CSV-Datei)

Projektmanagement

- Jede:r hat ein eigenes Programm in dessen Grundstruktur gemäss des Projektbeschriebs erstellt
- grösserer Lerneffekt und bessere Einarbeitung da keiner von uns Vorkenntnisse mit Python hatte
- Flexibilität bei der Gestaltung
- Entscheidung im Team, welches Projekt schliesslich gewählt wird
- gemeinsame Ausarbeitung des gewählten Programms (Optimierung)



Projektmanagement

Verlauf:

- Harmonie im Team
- Unterstützung wo nötig
- regelmässige Check-Ins

Ergebnis:

- Zielerreichung und Lerneffekt gross
- Zusammenarbeit physisch vor Ort und Austausch zielführend via Teams



Live Demo

```
def bearbeite_kosten(benutzerdaten, kosten_art):
    """Hilfsfunktion zum Bearbeiten von Fix- oder variablen Kosten."""
    kosten_dict = benutzerdaten.get(kosten_art, {})

    if kosten_dict == {}:
        benutzerdaten[kosten_art] = {}
        kosten_dict = benutzerdaten[kosten_art]

    while True:
        print(f"\n--- {kosten_art} bearbeiten ---")
        if not kosten_dict:
            print("Aktuell sind keine Kostenpunkte erfasst.")
        else:
            print("Aktuelle Kostenpunkte:")
            for idx, (posten, wert) in enumerate(kosten_dict.items()):
                print(f"{idx+1}. {posten}: {format_waehrung(wert)}")

            print("\nOptionen:")
            print("A. Neuen Posten hinzufügen")
            print("B. Bestehenden Posten ändern")
            print("C. Bestehenden Posten löschen")
            print("D. Zurück zur Datenanpassung")

            wahl = input("Ihre Wahl (A/B/C/D): ").upper().strip()

            if wahl == 'A':
                posten_name = input("Name des neuen Postens: ").strip()
                if posten_name:
                    wert = eingabe_pruefung(f"Monatliche Kosten für {posten_name} in CHF: ")
                    kosten_dict[posten_name] = wert
                    print(f"Posten '{posten_name}' hinzugefügt.")
            elif wahl == 'B':
                if not kosten_dict:
                    print("Keine Posten zum Ändern.")
                    continue
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
alessio.cusintino@MacBook Air von Alessio Cusintino ~ % /usr/bin/python3 /Users/alessio.cusintino/Downloads/Budget-Rechner.py
alessio.cusintino@MacBook Air von Alessio Cusintino ~ % /usr/bin/python3 /Users/alessio.cusintino/Downloads/Budget-Rechner.py

=====
WILKOMMEN ZUM BUDGET-PLANER
=====

Hauptmenü:
1. Neu registrieren und Daten erfassen
2. Bestehende Textdatei hochladen/einloggen
3. Programm beenden
Ihre Wahl (1/2/3):
```

Ln 377, Col 39 Spaces: 4 UTF-8 CRLF {} Python 3.9.6

Fragen?



**Vielen Dank für die
Aufmerksamkeit.**