

Lösungsvorschläge Python mit JupyterLite

Gruppenarbeiten 1 bis 5



Übersicht

A	Gruppenarbeit 1: Kursverwaltung	2
B	Gruppenarbeit 2: Notenauswertung	3
C	Gruppenarbeit 3: Mini-Adressbuch	4
D	Gruppenarbeit 4: Umsatzanalyse	5
E	Gruppenarbeit 5: Quiz-Generator	7



A Gruppenarbeit 1: Kursverwaltung

In diesem Notebook wird eine kleine Kursverwaltung mit Python umgesetzt. Es werden Kursnamen in einer Liste gespeichert, mit einer Schleife ausgegeben, Teilnehmerzahlen ergänzt und der Durchschnitt berechnet.

Aufbau des Notebooks

Markdown-Zelle

```
# Gruppenarbeit 1: Kursverwaltung
```

In diesem Notebook wird eine kleine Kursverwaltung mit Python umgesetzt. Es werden Kursnamen `in` einer Liste gespeichert, mit einer Schleife ausgegeben, Teilnehmerzahlen ergänzt und der Durchschnitt berechnet.

Codezelle

```
kurse = [  
    "Excel Grundlagen",  
    "Word für Einsteiger",  
    "PowerPoint kompakt",  
    "Outlook im Berufsalltag",  
    "Teams erfolgreich nutzen",  
    "Einführung in Python",  
    "Digitale Zusammenarbeit",  
    "Datenschutz im Büroalltag"  
]  
  
teilnehmerzahlen = [12, 15, 10, 18, 14, 11, 16, 9]  
  
print("Kursliste:")  
for kurs in kurse:  
    print("-", kurs)  
  
durchschnitt = sum(teilnehmerzahlen) / len(teilnehmerzahlen)  
print()  
print(f"Durchschnittliche Teilnehmerzahl: {durchschnitt}")
```

Dokumentation

Markdown-Zelle

```
## Zusammenfassung
```

Es wurden acht Kurse `in` einer Liste erfasst und mit einer Schleife ausgegeben.
Zu jedem Kurs wurde eine Teilnehmerzahl gespeichert.
Aus allen Teilnehmerzahlen wurde anschliessend der Durchschnitt berechnet.



B Gruppenarbeit 2: Notenauswertung

In diesem Notebook werden Noten mit Python ausgewertet. Es werden Durchschnitt, beste Note und tiefste Note berechnet. Zusätzlich wird jede Note als genügend oder ungenügend beurteilt.

Aufbau des Notebooks

Markdown-Zelle

```
# Gruppenarbeit 2: Notenauswertung
```

Markdown-Zelle

```
In diesem Notebook werden Noten mit Python ausgewertet.  
Es wird der Durchschnitt berechnet, die beste und die tiefste Note  
ermittelt  
und jede Note als genügend oder ungenügend beurteilt.
```

Codezelle

```
noten = [5.5, 4.0, 3.5, 6.0, 4.5, 5.0, 2.5, 4.0, 5.5, 3.0]  
  
durchschnitt = sum(noten) / len(noten)  
beste_note = max(noten)  
tiefste_note = min(noten)  
  
print(f"Durchschnitt: {durchschnitt}")  
print(f"Beste Note: {beste_note}")  
print(f"Tiefste Note: {tiefste_note}")  
print()  
for note in noten:  
    if note >= 4.0:  
        print(note, "- genügend")  
    else:  
        print(note, "- ungenügend")
```

Dokumentation

Markdown-Zelle

```
## Beobachtungen
```

```
Die Notenliste enthält genügend und ungenügende Leistungen.  
Der Durchschnitt zeigt, wie die gesamte Gruppe abgeschnitten hat.  
Mit der Schleife lässt sich jede einzelne Note einfach und übersichtlich  
beurteilen.
```



C Gruppenarbeit 3: Mini-Adressbuch

In diesem Notebook wird ein kleines Adressbuch mit Python erstellt. Drei Listen werden mit einer Schleife zusammengeführt und mit einem f-String formatiert ausgegeben.

Aufbau des Notebooks

Markdown-Zelle

```
# Gruppenarbeit 3: Mini-Adressbuch
```

Markdown-Zelle

In diesem Notebook wird ein kleines Adressbuch mit Python erstellt. Dafür werden drei Listen verwendet: Vornamen, Nachnamen und Orte. Die Daten werden mit einer Schleife formatiert ausgegeben und anschliessend dokumentiert.

Codezelle

```
vornamen = ["Mila", "Luca", "Nina", "Jan", "Sofia"]
nachnamen = ["Keller", "Meier", "Schmid", "Brunner", "Ammann"]
orte = ["Bern", "Zürich", "Luzern", "Basel", "Winterthur"]

print("Mini-Adressbuch:")
print()
for i in range(len(vornamen)):
    print(f"{vornamen[i]} {nachnamen[i]} - {orte[i]}")
```

Dokumentation

Die Datensätze können am Schluss zusätzlich als kleine Übersicht festgehalten werden.

Vorname	Nachname	Ort
Mila	Keller	Bern
Luca	Meier	Zürich
Nina	Schmid	Luzern
Jan	Brunner	Basel
Sofia	Ammann	Winterthur



D Gruppenarbeit 4: Umsatzanalyse

In diesem Notebook werden zwölf Monatsumsätze mit Python ausgewertet. Berechnet werden Summe, Durchschnitt, höchster und tiefster Wert. Zusätzlich werden alle Monate über dem Durchschnitt ausgegeben.

Aufbau des Notebooks

Markdown-Zelle

```
# Gruppenarbeit 4: Umsatzanalyse
```

In diesem Notebook werden zwölf Monatsumsätze mit Python ausgewertet. Berechnet werden die Summe, der Durchschnitt, der höchste und der tiefste Umsatz.

Falls matplotlib verfügbar ist, kann zusätzlich ein einfaches Liniendiagramm erstellt werden.

Codezelle

```
monate = [
    "Januar", "Februar", "März", "April", "Mai", "Juni",
    "Juli", "August", "September", "Oktober", "November", "Dezember"
]

umsaetze = [12500, 13800, 14200, 11900, 15100, 16000, 15800,
            14900, 15500, 16800, 17200, 18100
]

summe = sum(umsaetze)
durchschnitt = summe / len(umsaetze)
hoechster_wert = max(umsaetze)
tiefster_wert = min(umsaetze)

print(f"Summe: {summe} CHF")
print(f"Durchschnitt: {durchschnitt:.2f} CHF")
print(f"Höchster Wert: {hoechster_wert} CHF")
print(f"Tiefster Wert: {tiefster_wert} CHF")
print()
for i in range(len(monate)):
    if umsaetze[i] > durchschnitt:
        print(f"{monate[i]}: {umsaetze[i]} CHF")
```



Zusätzliche Codezelle für Diagramm (Plot)

```
def zeige liniendiagramm(monatsnamen, werte):  
    try:  
        import matplotlib.pyplot as plt  
  
        plt.figure(figsize=(10, 5))  
        plt.plot(monatsnamen, werte, marker="o")  
        plt.title("Monatsumsätze")  
        plt.xlabel("Monate")  
        plt.ylabel("Umsatz in CHF")  
        plt.xticks(rotation=45)  
        plt.tight_layout()  
        plt.show()  
  
    except ModuleNotFoundError:  
        print("Matplotlib ist in dieser Umgebung nicht verfügbar.")  
  
zeige_liniendiagramm(monate, umsaetze)
```

Dokumentation

Markdown-Zelle

Beobachtungen

Die Umsätze schwanken im Verlauf des Jahres, steigen gegen Jahresende aber deutlich an.
Mehrere Monate liegen über dem Durchschnitt, vor allem im zweiten Halbjahr.
Mit Python lassen sich solche Listen einfach auswerten und bei Bedarf auch grafisch darstellen.



E Gruppenarbeit 5: Quiz-Generator

In diesem Notebook wird eine Funktion erstellt, die zu einer Punktzahl eine Bewertung zurückgibt. Die Funktion wird getestet und anschliessend auf eine kleine Punkteliste angewendet.

Aufbau des Notebooks

Markdown-Zelle

```
# Gruppenarbeit 5: Quiz-Generator
```

In diesem Notebook wird eine Funktion erstellt, die zu einer Punktzahl eine Bewertung ausgibt.
Die Funktion wird mit mehreren Werten getestet. Anschliessend wird eine Liste mit Punktzahlen mit einer Schleife ausgewertet.

Codezelle

```
def bewertung_ausgeben(punkte):  
    if punkte >= 9:  
        return "sehr gut"  
    elif punkte >= 7:  
        return "gut"  
    elif punkte >= 5:  
        return "genügend"  
    else:  
        return "ungenügend"  
  
print("Tests der Funktion:")  
print("10 Punkte:", bewertung_ausgeben(10))  
print("8 Punkte:", bewertung_ausgeben(8))  
print("6 Punkte:", bewertung_ausgeben(6))  
print("3 Punkte:", bewertung_ausgeben(3))
```

Codezelle

```
punkte_liste = [10, 7, 4, 9, 6, 5, 8, 3]  
  
print("Auswertung der Punkteliste:")  
print()  
for punkte in punkte_liste:  
    print(f"{punkte} Punkte -> {bewertung_ausgeben(punkte)}")
```


Dokumentation

Markdown-Zelle

Beobachtungen

Die Funktion weist jeder Punktzahl eine passende Bewertung zu. Mit der Schleife kann eine ganze Liste von Ergebnissen schnell ausgewertet werden. Die Lösung ist übersichtlich aufgebaut, weil Funktion, Tests und Auswertung klar getrennt sind.

