

Projektarbeit im 1. Sem. Informatik

Studiengang Wirtschaftsingenieurwesen, DHBW Stuttgart (Prof. Heuser)

Projekt 2 („Umsatzdaten“) – mittelschwer

Als Grundlage zu dieser Projektarbeit dienen folgende Dateien:

- 101_parse_umsatzbericht.py (zu ergänzende Python-Datei)
- 101_Umsatzbericht.xlsx (Grunddaten als Excel-Datei)

Die Dateien 101_Umsatzbericht.xlsx bestehen jeweils aus fiktiven *Umsatzdaten* eines Unternehmens, welche Sie mit Hilfe von 101_parse_umsatzbericht.py weiter analysieren sollen.

	A	B	C	D	E	F
1	Produkt	Kunde	Qrtl 1	Qrtl 2	Qrtl 3	Qrtl 4
2	Alice Mutton	ANTON	- €	702,00 €	- €	- €
3	Alice Mutton	BERGS	312,00 €	- €	- €	- €
4	Alice Mutton	BOLID	- €	- €	- €	1.170,00 €
5	Alice Mutton	BOTTM	1.170,00 €	- €	- €	- €
6	Alice Mutton	ERNSH	1.123,20 €	- €	- €	2.607,15 €
7	Alice Mutton	GODOS	- €	280,80 €	- €	- €
8	Alice Mutton	HUNGC	62,40 €	- €	- €	- €
9	Alice Mutton	PICCO	- €	1.560,00 €	936,00 €	- €
10	Alice Mutton	RATTC	- €	592,80 €	- €	- €
11	Alice Mutton	REGGC	- €	- €	- €	741,00 €
12	Alice Mutton	SAVEA	- €	- €	3.900,00 €	789,75 €

Abb. 1: Ausschnitt aus der Datei 101_Umsatzbericht.xlsx

Die Python-Datei soll dabei zunächst die Summe der Umsätze über ein vom Benutzer vorgegebenes Quartal (z.B. 'Qrtl 3') für sämtliche Produkte („Produkt“) oder Kunden („Kunde“) ausgeben. (s. Abb. 2).

```
Summe der Umsätze über das Quartal 'Qrtl 3' für 'Produkt' ...
{"Sir Rodney's Marmalade": 3061.8, "Sir Rodney's Scones": 1733.0, 'Teatime Chocolate Biscuits': 841.8, 'Mozzarella di Giova
nni': 3027.6, 'Longlife Tofu': 0.0, "Chef Anton's Gumbo Mix": 288.22, 'Boston Crab Meat': 4412.32, 'Camembert Pierrot': 957
9.5, 'Laughing Lumberjack Lager': 350.0, 'Gorgonzola Telino': 515.0, "Grandma's Boysenberry Spread": 1750.0, "Uncle Bob's O
rganic Dried Pears": 2700.0, "Jack's New England Clam Chowder": 1582.6, 'Northwoods Cranberry Sauce': 0.0, 'Sasquatch Ale':
0.0, 'Veggie-spread': 842.88, 'Aniseed Syrup': 140.0, 'Steeleye Stout': 1323.0, 'Ipoh Coffee': 1196.0, 'Ravioli Angelo': 3
90.0, 'Louisiana Fiery Hot Pepper Sauce': 1375.62, 'Louisiana Hot Spiced Okra': 68.0, 'Alice Mutton': 4836.0, 'Filo Mix': 2
89.8, "Chef Anton's Cajun Seasoning": 1337.6}
```

Abb. 2: Start des Programms 101_parse_umsatzbericht inkl. Ausgabe der Umsatzsummen für ein vom Benutzer vorher festgelegtes Quartal (hier: „Qrtl 3“) und sämtliche Produkte („Produkt“)

Im zweiten Schritt soll das Ergebnis als Graph visualisiert werden (s. Abb. 3).

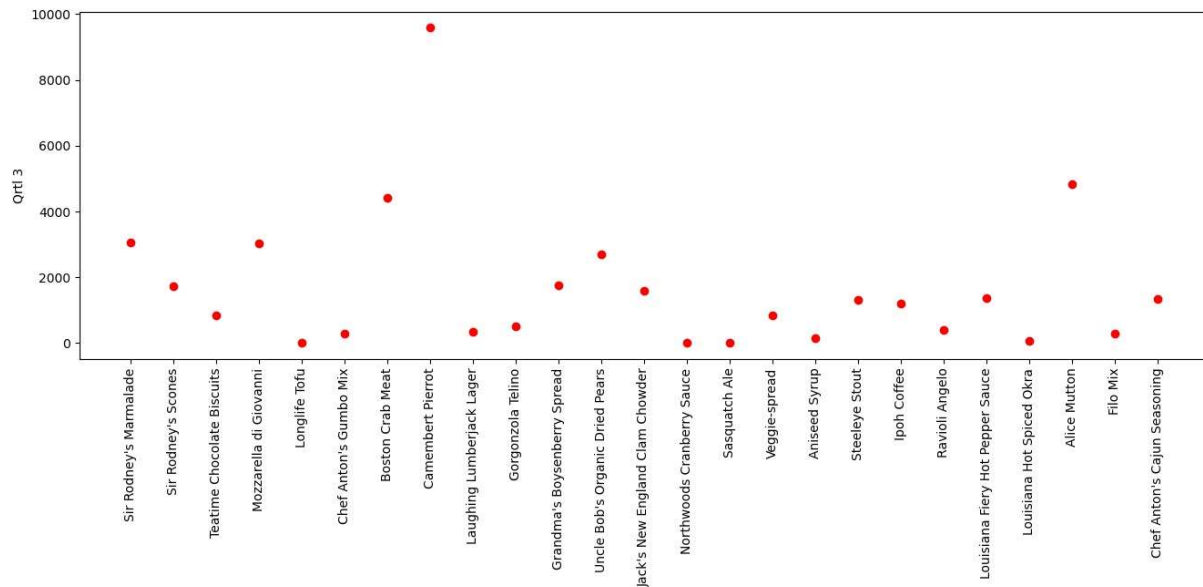


Abb. 3: Graphische Visualisierung der Ergebnisdaten zu Abb. 2

In einem dritten Schritt sollen Sie einerseits die graphische Ausgabe durch ein aussagekräftigeres Säulendiagramm ersetzen. Zum anderen sollen Sie optional den prozentualen Anteil der Umsätze jedes Kunden sowie jedes Produkts am Gesamtumsatz berechnen und auf der Konsole (vgl. Abb. 2) ausgeben.

Die Python-Datei 101_parse_umsatzbericht.py startet dabei nach der Kommentarzeile # __main__ (Zeile 122) und ruft nacheinander verschiedene Funktionen innerhalb der gleichen Python-Datei auf.

Ihre Aufgabe im Team

Halten Sie sich bei der Bearbeitung der Aufgabe zunächst immer an die allgemeinen Projektvorgaben aus der Datei „Projektarbeit“!

Spezielle Projektaufgaben:

1. Ergänzen Sie die Funktionsrümpfe `xfile_read()` sowie `plot_client_data()`.
2. Ersetzen Sie `plt.plot()` innerhalb `plot_client_data()`, so dass statt eines Punkt- ein Säulendiagramm ausgegeben wird.
3. Geben Sie die Summe aller Umsätze über ein spezifisches vom Kunden vorgegebenes Quartal und Produkt oder Kunde folgendermaßen aus (s. Abb. 2):

```
print(f"\nSumme der Umsätze über das Quartal '{quartal}' für '{kunde_produkt}' ... \n{y_axis}")
```

Optionale Projektaufgaben:

4. (Optional 1) Zusätzlich zu der unter Punkt 3 vorbereiteten Ausgabe soll folgende Ausgabe auf die Konsole erfolgen:

```
print(f"\nProzentualer Anteil der Umsätze am Gesamtumsatz über das Quartal '{quartal}' für '{kunde_produkt}' ... \n{???}")
```

5. (Optional 2) Es soll zudem der Mittelwert aller Umsätze eines beliebigen Quartals und Produkt oder Kunde ausgerechnet und anschließend als Gerade im Graph ausgegeben werden.

Das Team, welches sich *besonders programmiersicher* fühlt und es sich zutraut kann sich beispielsweise an einem der folgenden Probleme versuchen:

6. (Optional 3) Gesucht ist die weitestgehende Integration in Excel. Wir wollen also Daten aus der gleichen Exceldatei 101_Umsatzbericht.xlsx lesen als auch Ergebnisse in diese wieder zurückschreiben. Dies wird ermöglicht durch die Python-Library ‚xlwings‘ (<https://www.xlwings.org/>), welche es erlaubt, beliebige Aufgaben innerhalb unserer Excel-Tabellenblätter in Analogie zu Excel-Makros zu automatisieren. Python in Kombination mit xlwings kann dabei das inzwischen veraltete VBA (Visual Basic for Applications; https://de.wikipedia.org/wiki/Visual_Basic_for_Applications#Nachteile) ersetzen. In dieser Aufgabe sollen Sie `xfile_read()` sowie `plot_client_data()` oder eine beliebige andere Funktion mit Hilfe von xlwings abbilden, so dass die Funktionsergebnisse (und optionale Ergebnisgraphen) wieder in die gleiche Excel-Datei zurückgeschrieben werden können. Tipp: Sichern Sie vorher Ihre Grunddaten, z.B. als 101_Umsatzbericht.xlsx!

Zusatzfragen

Mögliche Fragen zu Ihrem Projekt könnten sein:

1. Erklären Sie bitte auch die Hilfsfunktionen `plt.plot()` bzw. `matplotlib.pyplot.plot()`
2. Erklären Sie das Dictionary, welches in den Z. 77-79 initialisiert wird.
3. Erklären Sie das Pandas DataFrame, welches wir bereits in `xfile_read()` definieren.

Hilfe

Für sämtliche graphische Funktionen der Libraries Matplotlib, Pandas und darüber hinaus stehen Ihnen viele nützliche Websites als Hilfe zur Verfügung. Sollten Sie also Funktionsdetails der Bibliotheken oder Beispiel-Anwendungen mit Hilfe der Bibliotheksfunktionen nachschlagen wollen, so sei Ihnen folgende URL empfohlen:

- <https://docs.python.org/3.9/> (sämtliche in Python integrierte Funktionen inkl. Library Reference und Tutorial; Python Version 3.9 oder neuere)
- <https://matplotlib.org/stable/tutorials/introductory/pyplot.html> (Pyplot)
- https://pandas.pydata.org/pandas-docs/stable/user_guide/10min.html (Pandas)
- <https://realpython.com/beginners-guide-python-turtle/> (Turtle)
- ...

Häufig ist es auch ratsam, mit Hilfe von Google nach Hilfen im Web zu suchen.