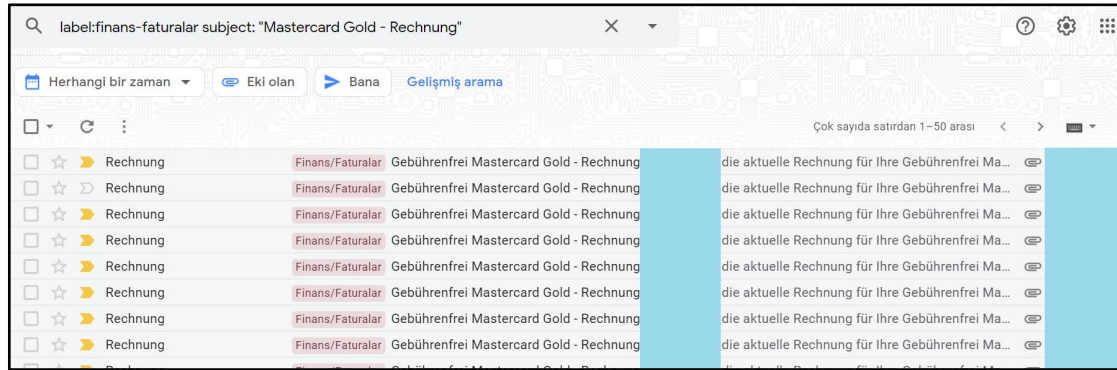


Credit Card Bill Analyzer Script:

The main idea is to be able to retrieve credit card bill data automatically and try to generate some valuable information out of it.

The source of the data is the email. Credit card bills are being sent to my personal email monthly. The bill comes in .pdf format as an attachment in the email (and I have many of them!).



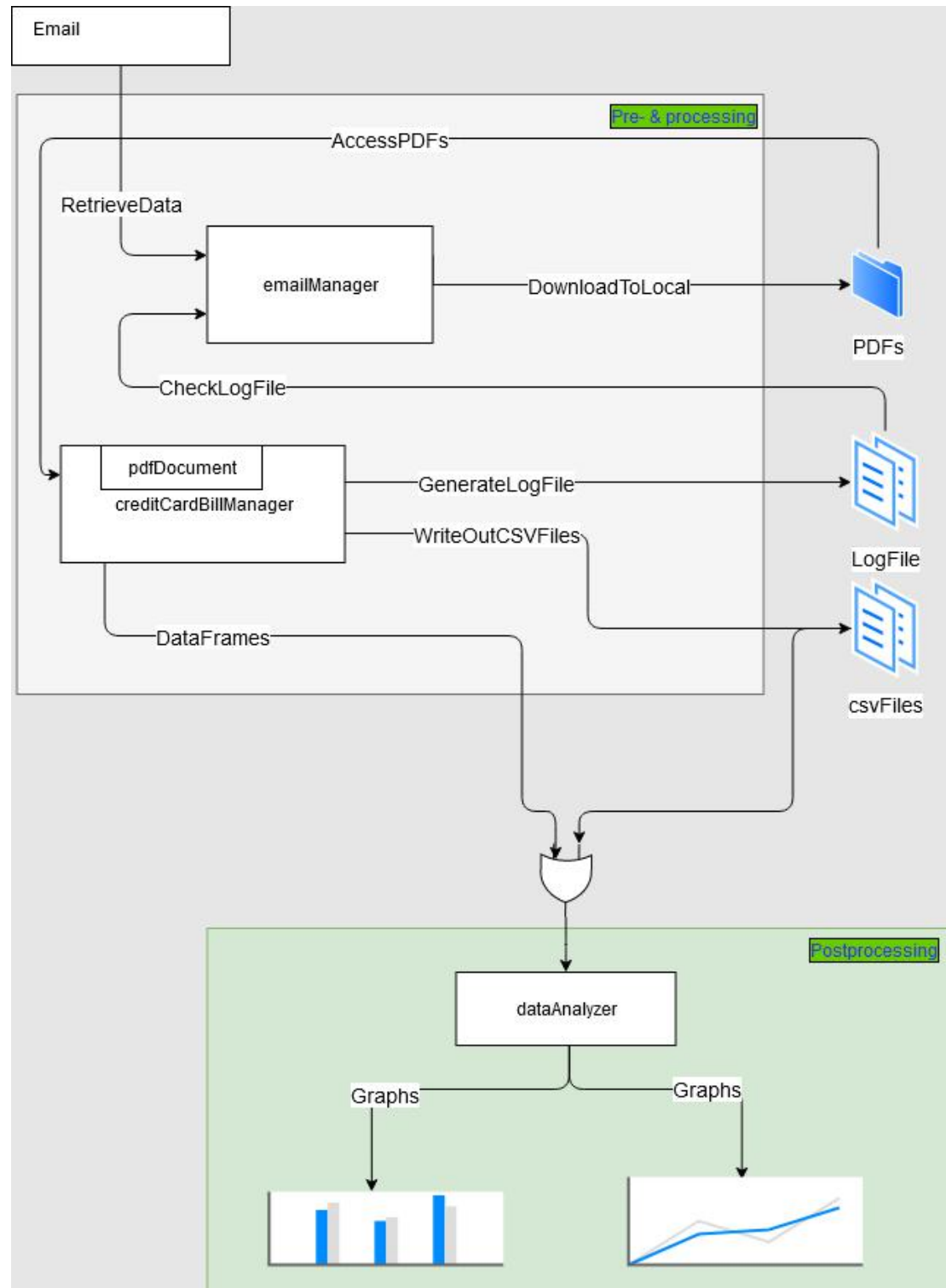
The .pdf document has a standard format as shown:

[Redacted]		Deutsche Postanschrift		Geschäftssitz	
M [Redacted] E [Redacted]		[Redacted]			
[Redacted]		Kontakt			
[Redacted]		[Redacted]			
Rechnung Gebührenfrei Mastercard Gold					
Kreditkartennummer: **** [Redacted]			Rechnung vom 03.09. [Redacted]		
Datum	Beschreibung	Ort	Betrag EUR		
04.08.	ALTER SALDO	[Redacted]	[Redacted]		
07.08.	[Redacted]	[Redacted]	[Redacted]		
07.08.	[Redacted]	[Redacted]	[Redacted]		
07.08.	[Redacted]	[Redacted]	[Redacted]		
10.08.	EINZAHLUNG	[Redacted]	[Redacted]		
13.08.	[Redacted]	[Redacted]	[Redacted]		
13.08.	Lieferando.de	[Redacted]	[Redacted]		
14.08.	[Redacted]	[Redacted]	[Redacted]		
15.08.	[Redacted]	[Redacted]	[Redacted]		
15.08.	[Redacted]	[Redacted]	[Redacted]		
15.08.	[Redacted]	[Redacted]	[Redacted]		
25.08.	[Redacted]	[Redacted]	[Redacted]		
29.08.	[Redacted]	[Redacted]	[Redacted]		
29.08.	[Redacted]	[Redacted]	[Redacted]		
02.09.	[Redacted]	[Redacted]	[Redacted]		
03.09.	[Redacted]	[Redacted]	[Redacted]		
03.09.	NEUER SALDO	[Redacted]	[Redacted]		
Mindestbetrag fällig am 21.09. [Redacted]					
Kreditlimit	Verfügbar	Saldo	Mindestbetrag	Fällig am	
[Redacted]	[Redacted]	[Redacted]	[Redacted]	[Redacted]	

Necessary files:

main.py
pdfDocument.py
creditCardBillManager.py
emailManager.py
dataAnalyzer.py

The following sketch shows the basics of script:



pdfDocument.py

Responsible for reading each bill (in .pdf) and convert it to a table (Pandas data frame). It takes out previous month's the paid amount.

creditCardBillManager.py

Responsible for retrieving tables (Pandas data frame) of each credit card bill, pdfDocument objects, and generating the master & monthly tables. Also, double checks if the bills that are processed are fine, sum of individual costs match to the months' end -checksum.

emailManager.py

Responsible for accessing email and looking for credit card bill emails under "Finans/Faturalar" folder. It cross checks the latest email date with the log file, i.e. if the bills from last months are already downloaded, it is not going to re-download those.

dataAnalyzer.py

Responsible for creating plots or showing credit card data. It can either give an overview of all the bills or year & month specific bill.

main.py

Calls the other modules, for now it is very basic for demonstration purposes. Although the script can run on shell with main.py, it is recommended to use it either on Spyder or in a Jupyter Notebook because of the graphs etc.

Examples:

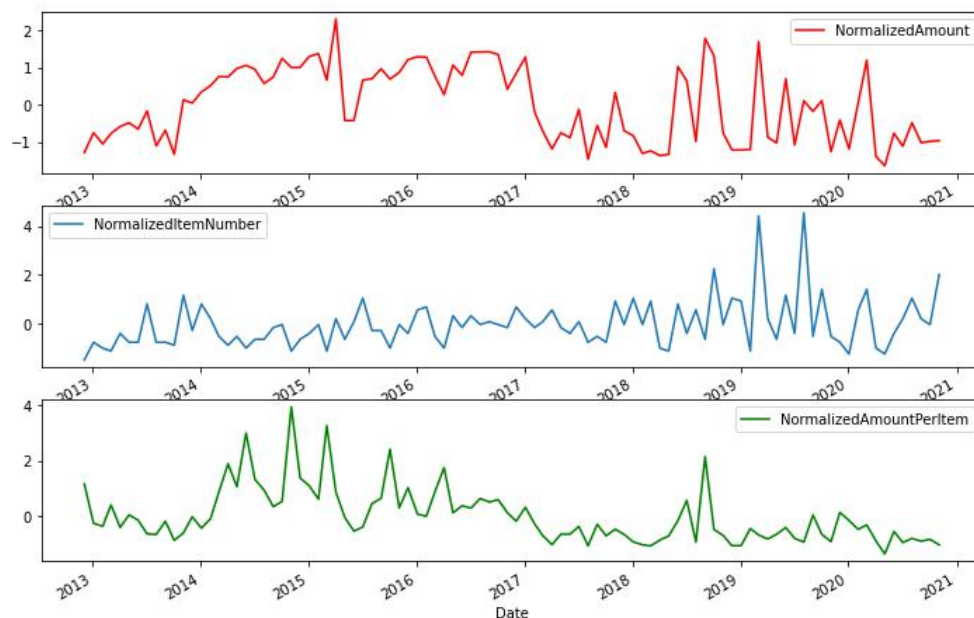
Plot1: Plot of bills per month over the years:

Amount*: That month's bill amount, i.e. x € to pay

ItemNumber*: That month's number of items purchased

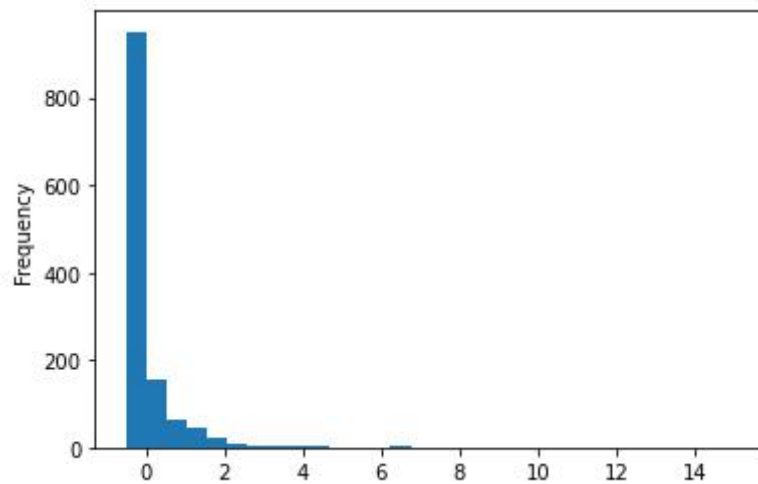
AmountPerItem*: Average cost of an item of that month

*All of these values in the graphs are normalized due to data privacy.



Plot1: Plot of bills per month over the years

Plot2: Frequency distribution of cost of individual items, i.e. more than 800 items purchased over the years cost around -x to 0€, again the values are normalized.



Plot2: Frequency distribution of cost of individual items

Next actions:

- Grouping or clustering items based on explanation & location.
- Create a database of monthly income from 2013 onwards, and correlate it to the amount of bills.

File Templates:

.csv files look like that, master data template and monthly data template respectively:

A	B	C	D
Date	Explanation	Location	Amount
dd/mm/yyyy	Some Fancy Restaurant	Istanbul	XX,XX
dd/mm/yyyy	EWER Supermarket	Munich	YY,YY

A	B	C	D	E
Date	OldSum	OldPaid	NewAmount	ItemNumber
dd/mm/yyyy	XXX,XX	-YYY,YY	ZZZ,ZZ	7

.log file looks like that:

```
*creditCardBillManager.log - Notepad
File Edit Format View Help
Auto-generated report for the creditCardBillManager.py:
Number of bills analyzed: XX
Last bill date: yyyy-mm-dd mm:ss:xx
```