Skills test for Junior Data Engineer

1.

The scenario:

Each day I get CSV files and I never know how clean or dirty the data really is, and if the column order is always the same. The only thing I know is that each CSV file has columns blogId, views, clicks (I don't know if this will be always the order of the columns).

The way I have approach this task is following: at the end of each day I put all the CSV files I got that day and I run the following Python Script. The script takes all CSV files, it puts them in right column order and adds the current date time and saves it in new CSV file. I also uploaded test CSV files so you can run the script in order to test it. Also I have written the comments if order to explain what each code line means.

```
import csv
import datetime
import os
from tabulate import tabulate


#define paths
csv_path = 'C:/Users/SOFIJA/PycharmProjects/SkillTest/jDataEngineer'
output_file = './out.csv'

#if you just want date and not time, put "%B %d, %Y"
current_date = datetime.datetime.now().strftime("%B %d %Y, %I:%M.%S %p")


#Load the output csv file or create an empty dataframe to store the new data
fields = ['blogid','views','clicks','current_date','csv']
old_csvs=set()
if os.path.isfile(output_file):
    out_csv = csv.DictReader(open(output_file))
    for row in out_csv:
        old_csvs.add(dict(row)['csv'])
else:
     with open(output_file, 'a+', encoding='utf-8') as output:
        writer = csv.DictWriter(output, fieldnames=fields)
        writer.writeheader()

#iterate through csv files in the directory
for f in os.listdir(csv_path):
    #process file only if it's not already proccessed in a previous date and indexed
in output_file
    if f.endswith('.csv') and f not in old_csvs:
        csv_file = os.path.join(csv_path,f)
        csv_data = csv.DictReader(open(csv_file))
        with open(output_file, 'a+', encoding='utf-8') as output:
            writer = csv.DictWriter(output, fieldnames=fields)
            for row in csv_data:
                row_clean = dict((k.lower().replace(' ', ''), v) for k,v in
dict(row).items())
                row_clean['current_date'] = current_date
                row_clean['csv'] = f
                #write row for output file
                writer.writerow(row_clean)


#print data
rows = []
```

```
out_csv = csv.DictReader(open(output_file))
for row in out_csv:
    rows.append(row)
print (tabulate(rows,headers=dict(zip(fields,fields)), tablefmt='orgtbl'))
```

The output will be following:

views	csv	clicks	blogid	current_date
10	1.csv	3	1	December 01 2017, 10:57.00 AM
14	2.csv	15	1	December 01 2017, 10:57.00 AM
9	3.csv	8	6	December 01 2017, 10:57.00 AM

The end result is a dataset that contains a date dimension that would allow presentation via a BI/Analytics tool to show blog traffic by day.

There are tools and software that does all work you need like:

1. We can use **Alteryx** - leader in the self-service data analytics movement with a platform that can prep, blend, and analyze all of your data, then deploy and share analytics at scale for deeper insights in hours. (www.alteryx.com)
2. We can use **Cloudera Impala** with **Apache Hadoop** (**www.cloudera.com**)

2.

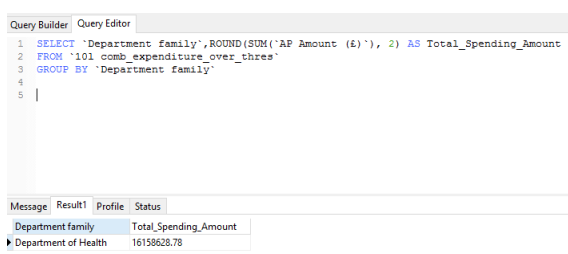
I have inserted the dataset is in CSV format into Navicat database and I did the queries using MySQL.

2a. Total spending (amount) by Department family.

SELECT `Department family`, ROUND(SUM(`AP Amount (£)`), 2) AS Total_Spending_Amount

FROM `10l comb_expenditure_over_thres`

GROUP BY `Department family`



2b. Total spending (amount) by Department family + Expense type.

SELECT `Department family`, `Expense Type`, ROUND(SUM(`AP Amount (£)`), 2) AS Total_Spending_Amount

FROM `10l comb_expenditure_over_thres`

GROUP BY `Department family`, `Expense Type`

Query Builder | Query Editor

```
SELECT `Department family`, `Expense Type`, ROUND(SUM(`AP Amount (£)`), 2) AS Total_Spending_Amount
FROM `10l comb_expenditure_over_thres`
GROUP BY `Department family`, `Expense Type`
```

Message | Result1 | Profile | Status

Department family	Expense Type	Total_Spending_Amount
Department of Health	C&M-GMS Cost of Drugs -Dispensing	72115.97
Department of Health	C&M-GMS DES Learn Dsblty Hlth Chk	420.00
Department of Health	C&M-GMS DES Minor Surgery	18833.34
Department of Health	C&M-GMS DES TPP QRISK	175.50
Department of Health	C&M-GMS DES Violent Patients	333.00
Department of Health	C&M-GMS Global Sum	930042.13
Department of Health	C&M-GMS MPIG Correction Factor	6365.34
Department of Health	C&M-GMS PCO Doctors Ret Scheme	3846.12
Department of Health	C&M-GMS PCO Other	870.34
Department of Health	C&M-GMS PCO Seniority	40529.41
Department of Health	C&M-GMS Prem Actual Rent	135150.50
Department of Health	C&M-GMS Prem Cost Rent	4589.58
Department of Health	C&M-GMS Prem Notional Rent	34980.33

3.

SELECT `Expense area`, `Expense Type`, ROUND(SUM(`AP Amount (£)`), 2) AS Total_Spending_Amount

FROM `10l comb_expenditure_over_thres`

GROUP BY `Expense area`, `Expense Type`

ORDER BY 1

Query Builder | Query Editor

```
SELECT `Expense area`,`Expense Type`,ROUND(SUM(`AP Amount (£)`), 2) AS Total_Spending_Amount
FROM `10l comb_expenditure_over_thres`
GROUP BY `Expense area`, `Expense Type`
ORDER BY 1
```

Message | Result1 | Profile | Status

Expense area	Expense Type	Total_Spending_Amount
ACUTE COMMISSIONING	Hcare Srv Rec Fdtn Trust-Contract Baseline	528914.10
ACUTE COMMISSIONING	Hcare Srv Rec Fdtn Trust-CQUIN	5610.90
ACUTE COMMISSIONING	Hcare Srv Rec NHS Trust-Contract Baseline	330476.83
ADMINISTRATION & BUSINESS SUPPORT	Charges from CSU	97721.70
ADMINISTRATION & BUSINESS SUPPORT	Clinical&Medical-Othe Public Sector	14675.01
ADMINISTRATION & BUSINESS SUPPORT	Rent	14675.01
ADMINISTRATION & BUSINESS SUPPORT	Service Charge	1467.50
ADULT JOINT FUNDED CONTINUING CARE	Cont Care-Learning Disab(<65)	41598.99
BALANCE SHEET	Income tax <1Yr	39207.45
BALANCE SHEET	Income tax <1Yr-Student Loans	629.00
BALANCE SHEET	National Insurance < 1 yr-NI- ERS	28625.08
BALANCE SHEET	National Insurance < 1 yr-NI-EES	20420.63
BALANCE SHEET	NHS Payables <1Yr-Recharges In	10787309.00
BALANCE SHEET	Oth Pybls <1Yr-Other	60729.00

I used Tableau Public in order to get the table you have requested. Please go to the link public.tableau.com/profile/sofija.milunov