IFCO Data Engineering Challenge

Introduction and configuration.

In the repository, you can find 6 documents, the executable scripts script1, script2, and script3, as well as two documents that present the source information, the documents orders.txt and invoices.json. Finally, you will find this document with the execution instructions.

For the correct execution of the codes, please make sure that your Python compiler has the libraries csv, json, and matplotlib.pyplot installed.

You should place the files orders.txt and invoices.json from the repository in the reading path of your Python compiler. Warning: the document orders.txt has been modified from the original document orders.csv due to formatting inconsistencies in 3 records. These records had the wrong format because of forgetting to add '[' or ']' in the 'contact_data' field. The provided file 'orders.txt' contains the correct information and is free of formatting errors

Script 1

Please open and run script1 in your Python compiler. You will see the following text:

```
The column number 1 has unique records
The column number 2 has unique records
The column number 3 does not have unique records
The column number 4 does not have unique records
The column number 5 does not have unique records
The column number 6 does not have unique records
The column number 7 does not have unique records
```

This text corresponds to a preliminary study of the data, in which you can see that the source information, stored in the ORDERS matrix, contains unique records in columns 1 and 2. For the purpose of studying the data, it is important to check for unique records, as these define the structure of the table. Since the table contains information about the orders, and the records in column 1 are unique—column 1 corresponds to the order identifier—we can conclude that each record in the table contains the information of one and only one order.

Task 1

Please enter the variable 'crateTypeDistribution' in the console. This contains the following information:

```
['company_name',n1,n2,n3]
```

Each of these lists of 4 elements corresponds to a unique company, so the list does not have duplicate company names. The field 'company_name' corresponds to the name of the company, and the values 'n1', 'n2', and 'n3' correspond to the number of boxes of 'Plastic', 'Wood', and 'Metal', respectively. In this list, you can see how, for each company, the orders are distributed based on whether they have requested boxes of one type or another.

The calculation of this information is performed in 'PART 2'. Please note that the code works for an arbitrary number *k* of boxes.

Task 2 and 3

Please enter the variable 'datFrame1' in the console. This variable contains the following information:

```
['order_id','contact_full_name']
```

The field 'order_id' contains the unique identifier for the order. The field 'contact_full_name' contains the contact person's name. If the information is not found, as specified in the task, the name John Doe was used.

Now please enter the variable 'datFrame2' in the console. This variable contains the following information:

```
['order_id','contact_address']
```

The field 'order_id' contains the unique identifier for the order. The field 'contact address' contains the name of the city and the postal code in the format 'city name, zip code'.

The calculation of these two data frames is performed in 'PART 3' with the help of JSON.

Script 2

Please open and run script2 in your Python compiler

Task 4

Please enter the variable 'comissionsSalesowners' in the console. This variable contains the following information:

```
['Salesowner','commission']
```

The field 'Salesowner' corresponds to a salesperson; this field is unique. The field 'commission' corresponds to the value in euros with two decimal places of precision for the commissions earned by that salesperson on all their orders.

The calculation of this list is performed in 'PART 5'. Since the information in 'invoices.json' only contains payment information for a limited number of 'order_id's, the calculation of the commissions has been done only considering the 'order_id's that did have an associated commission.

Task 5

Please enter the variable 'datFrame3' in the console. This variable contains the following information:

```
['company_id','company_name','list_salesowners']
```

The field 'company_id' contains the unique identifier for the company. In the source information, a single identifier corresponds to two different names. The two cases where this occurs are:

1e2b47e6-499e-41c6-91d3-09d12dddfbbd --> 'Fresh Fruits Co' and 'Fresh Fruits c.o'
20dfef10-8f4e-45a1-82fc-123f4ab2a4a5 --> 'healthy snacks c.o.' and 'Healthy Snacks Co'

Since the information has been grouped by the 'company_id' field, the differences in the actual names of the companies are ignored. The field 'company_name' corresponds to the name of the company. In the case of the two identifiers with different names, one of the two names is chosen based on the arbitrary order of the source data, as it is understood that both are equally valid. The field 'list_salesowners' contains an alphabetically sorted list of all the salesowners who have participated in at least one order for the company.

The calculation has been performed in 'PART 6'.

Script 3

Please open and run script3 in your Python compiler. Make sure you have matplotlib.pyplot installed for generating the graphs.

You will see that 3 graphs appear.

Task 6

Distribution of orders by type of box. For this part, a pie chart has been chosen to display the proportion of orders by type of box.

The calculation for this part is performed in 'PART 8'.

Sales of plastic boxes in the last 12 months. To assess which salespeople have the worst performance in plastic box sales, a bar chart has been created that displays the number of plastic box sales per salesperson in order. Through this chart, you can see who has participated in fewer plastic box sales. Since there is data up to 15.06.25, it is assumed that we are on this date.

The calculation for this part is performed in 'PART 9'.

Top 5 performers in the last 3 months segmented by month. For this part, a graph has been created that breaks down sales by salesperson for each month. It can be seen that the overall winner is 'Chris Pratt'; however, the winner in the month of April was 'Leonard Cohen.' Through this graph, the performance differences among various salesowners can be appreciated, with some showing more consistency across different months while others have peaks of activity. Note that this information refers only to plastic boxes; salespeople may perform well in selling other types of boxes.

The calculation for this part is performed in 'PART 10'.