**Destination Partner Ranking- Remittance Data Analysis**

Overview: -This project is a two-part initiative that focuses on both the source-partner side and the destination-partner side of the remittance process. The entire remittance flow involves several key entities: the sender customer, source-partner, TerraPay, destination-partner, financial institution, and the receiver customer. The project outlines the cross-border transfer process, where funds are securely transferred from one country to another. It aims to streamline and enhance the efficiency of cross-country payments by improving collaboration between financial institutions and partners across different regions. Through this process, we ensure smooth, secure, and prompt transactions that meet regulatory standards and customer expectations. Additionally, the project will find any potential gaps or challenges in the workflow, providing solutions to perfect the overall remittance experience.

A diagram of a payment method

Description automatically generated

Problem Statement:- **Partner Sustenance - Destination Partner Side**

The objective of this project is to evaluate and rank destination countries based on the performance of their respective destination partners in the remittance process. In the context of remittance operations, each destination country typically has multiple destination partners, and the goal is to assess and rank these countries using a set of key performance indicators (KPIs).

The ranking will be based on several critical criteria:

* Profitability per Transaction (profitability\_per\_txn): The profit generated by each transaction processed by the destination country’s partners.
* Cost per Transaction (cost\_per\_txn): The cost incurred by TerraPay for processing each transaction.
* Reversals(reversal\_per\_txn): Reversals count of the destination partner.
* Success Percentage (success\_percentage): The percentage of successful transactions processed by the destination country’s partners.
* Failure Percentage (failure\_percentage): The percentage of transactions that fail, lower failure percentage being preferable.
* Less than 15 Minutes Transactions (less\_15m): The percentage of transactions completed within 15 minutes, indicating faster transaction processing.
* Greater than 15 Minutes Transactions (greater\_15m): The percentage of transactions taking longer than 15 minutes, with a lower value indicating better performance.

By evaluating destination countries based on these metrics, the project aims to identify high-performing Destination Partners of a destination Country , understand their strengths, and provide insights into areas for improvement. This ranking system will support better decision-making in terms of optimizing partner relationships, improving operational efficiency, and driving the overall growth of remittance services in destination countries.

Receiver Customer

Financial Institution

Destination Partner

Destination Country

APPROACH:-

Data Collection:-

The data for this ranking system is collected in three parts, each sourced from different tables in the TerraPay database. These data parts contain critical transaction, financial, and performance metrics that are essential for evaluating the performance of destination countries and their partners in the remittance process. The data for each part is grouped by **destination\_country** and **destination\_partner**  and adjusted to ensure that each dataset contains the same number of rows for each destination country. Any mismatching or extra rows are removed to ensure consistency across the sheets.

**Part 1** of the data is extracted from multiple tables, including database **table**, and contains detailed transaction and financial metrics for each **destination\_country** and **destination\_partner**. The following fields are collected:

* + **destination\_country**: The country receiving the remittance.
  + **destination\_partner**: The specific partner operating in the destination country.
  + **gsv (Gross Settlement Value)**: The total value of remittance transactions processed.
  + **net\_recv\_fees**: The net fees received from the transactions.
  + **recv\_payout\_suc**: The number of successful payouts.
  + **recv\_payout\_rev**: The number of reversed payouts.
  + **txn\_count**: The total number of transactions processed.
  + **success\_count**: The total number of successful transactions.
  + **reversed\_count**: The total number of reversed transactions.
  + **gp (Gross Profit)**: The gross profit earned from the transactions.
  + **reversal\_per\_txn**: Reversal count by total transactions count (derived from the data).
  + **adjusted\_rev\_payout**: The adjusted value of reversed payouts.
  + **adjusted\_txn\_count**: The adjusted transaction count, accounting for numerous factors.
  + **cost\_per\_txn**: The cost incurred per transaction (**adjusted\_rev\_payout** /**adjusted\_txn\_count**).
  + **adjusted\_gross\_profit**: The adjusted gross profit considering various adjustments.
  + **profitability\_per\_txn**: The profitability per transaction (**adjusted\_gross\_profit/ adjusted\_txn\_count**).

These data points are collected for each destination country and partner, and derived metrics such as **profitability\_per\_txn**, **cost\_per\_txn**, and **reversal\_per\_txn** are computed from the raw data. This part of the data is saved as **"PART 2 sec 1"**, which forms the first section of the dataset used for further analysis and ranking. The data is grouped by **destination\_country** and **destination\_partner**, and any mismatching or extra rows are removed to ensure alignment across all records.

QUERY:

Select destination\_country,

destination\_partner,

Sum(gsv) as GSV,

Sum(recv\_payout)as Net\_recv\_fees,

sum(case when status='SUCCESS' then recv\_payout else -recv\_payout end) Recv\_payout\_suc,

(sum(case when status='SUCCESS' then recv\_payout else -recv\_payout end)-Sum(recv\_payout)) As recv\_payout\_rev,

Sum(txn\_count)as txn\_count,

SUM(case when status='SUCCESS'then txn\_count else 0 END) As Success\_count,

SUM(case when status='REVERSED'then txn\_count else 0 END) As reversed\_count,

Sum(gross\_profit) As GP,

ROUND(SUM(CASE

when status='REVERSED' then txn\_count ELSE 0 END)/

NULLIF(SUM(txn\_count),0),3) AS Reversal\_per\_txn,

ROUND(SUM(CASE

when status='SUCCESS' then recv\_payout

when status='REVERSED' then -recv\_payout

ELSE 0 END),3) as adjusted\_rev\_payout,

ROUND(Sum(CASE

when status='SUCCESS' then txn\_count

when status='REVERSED' then -txn\_count

ELSE NULL END),3) as adjusted\_txn\_count,

ROUND(CASE when

COUNT(case

when status='SUCCESS' then 1

when status='REVERSED' then -1

ELSE NULL END)=0 then 0

ELSE SUM(CASE

when status='SUCCESS' then recv\_payout

when status='REVERSED' then -recv\_payout

ELSE 0 END)/SUM(CASE

when status='SUCCESS' then txn\_count

when status='REVERSED' then -txn\_count

ELSE NULL END)END,3) as Cost\_per\_txn,

ROUND(SUM(CASE

when status='SUCCESS' then Gross\_profit

when status='REVERSED' then -Gross\_profit

ELSE 0 END),3) as adjusted\_Gross\_profit,

ROUND(CASE when

COUNT(case

when status='SUCCESS' then 1

when status='REVERSED' then -1

ELSE NULL END)=0 then 0

ELSE SUM(CASE

when status='SUCCESS' then gross\_profit

when status='REVERSED' then -gross\_profit

ELSE 0 END)/SUM(CASE

when status='SUCCESS' then txn\_count

when status='REVERSED' then -txn\_count

ELSE NULL END) END,3) as profitibality\_per\_txn

from <database table>

group by destination\_country,destination\_partner

order by destination\_country asc;

**Part 2** of the data is sourced from the <**database table 2>** table, which provides key transaction success and failure metrics for each **destination\_country** and **destination\_partner**. The following fields are extracted:

* + **destination\_country**
  + **destination\_partner**
  + **txn\_count**: The total number of transactions processed.
  + **success\_count**: The number of successful transactions.
  + **failure\_count**: The number of failed transactions.
  + **success\_percentage**: The percentage of successful transactions.
  + **failure\_percentage**: The percentage of failed transactions.

This data helps in evaluating the **success\_percentage** and **failure\_percentage**, which are essential for the performance evaluation and ranking process. This part of the data is saved as **"PART 2 sec 2"**, which forms the second section of the dataset. Like Part 1, destination\_country, and destination\_partner group the data, with any mismatching or extra rows removed for consistency across the records.

QUERY:

SELECT Destination\_country,

destination\_partner,

sum(txn\_count) as Txn\_count,

SUM(case when status='SUCCESS'then txn\_count else 0 End) as Success\_count,

SUM(case when status!='SUCCESS'then txn\_count else 0 End) as Failure\_count,

ROUND(SUM(case when status='SUCCESS' then txn\_count else 0 End)

\*100.00/NULLIF(SUM(txn\_count),0),3) as success\_percentage,

ROUND(SUM(case when status!='SUCCESS' then txn\_count else 0 End)

\*100.00/NULLIF(SUM(txn\_count),0),3) as failure\_percentage

From <database table 2>

Where destination\_country is Not NUll and

destination\_partner is NOT NULL

Group by destination\_country,destination\_partner

having SUM(case when status='SUCCESS'then txn\_count else 0 End)>0

order by destination\_country asc;

**Part 3** of the data is extracted from the **table 3** table, focusing on transaction completion times. The following fields are included:

* + **destination\_country**: The country receiving the remittance.
  + **destination\_partner**: The specific partner in the destination country.
  + **txn\_count**: The total number of transactions processed.
  + **less\_15m**: The number of transactions completed within 15 minutes.
  + **less\_15m\_percentage**: The percentage of transactions completed within 15 minutes.
  + **greater\_15m**: The number of transactions taking longer than 15 minutes.
  + **greater\_15m\_percentage**: The percentage of transactions taking longer than 15 minutes.

This data is crucial for evaluating **transaction processing speed**, focusing on the of transactions completed within 15 minutes (**less\_15m**) and those taking longer than 15 minutes (**greater\_15m**). This part of the data is saved as **"PART 2 sec 3"**, which forms the third section of the dataset. As with the previous parts, the data is grouped by **destination\_country** and **destination\_partner**, and mismatching or extra rows are deleted to ensure consistency in the data structure.

QUERY:

SELECT (select Country\_name

from country\_region\_info

where iso2= destination\_country\_code) as Destination\_country,

partner,sum(txn\_count) as txn\_count,

SUM(less\_10s\_count+less\_30s\_count+less\_01m\_count+less\_03m\_count+less\_05m\_count+less\_15m\_count) As less\_15m,

ROUND(SUM(less\_10s\_count+less\_30s\_count+less\_01m\_count+less\_03m\_count+less\_05m\_count+less\_15m\_count)\*100.00/Sum(txn\_count) ,3)as less\_15m\_percentage,

SUM(less\_30m\_count+less\_01h\_count+less\_02h\_count+less\_03h\_count+less\_24h\_count+less\_48h\_count+greater\_48h\_count) As greater\_15m,

ROUND(SUM(less\_30m\_count+less\_01h\_count+less\_02h\_count+less\_03h\_count+less\_24h\_count+less\_48h\_count+greater\_48h\_count)\*100.00/Sum(txn\_count),3)as greater\_15m\_percentage

from database\_table\_3

where Destination\_country\_code is not null and

partner is not null

group by destination\_country\_code,partner

order by destination\_country;

Together, these three data parts form a comprehensive and consistent dataset necessary for ranking the destination countries and their partners based on multiple performance metrics. These metrics include transaction volume, success and failure rates, cost, profitability, reversals, and transaction completion times. The structured and aligned data from **"PART 2 sec 1.csv"**, **"PART 2 sec 2.csv "**, and **"PART 2 sec 3.csv "** lays the foundation for calculating the ranking criteria and evaluating the overall performance of destination countries in the remittance process.

**Method Used:-**

To rank destination countries and partners effectively, data from three parts will be filtered, grouped by destination\_country, processed, and ranked. The following steps summarize the methods used:

1. Filtering and Grouping by Destination Country

Destination\_country and destination\_partner will group data. Any mismatched or extra rows will be removed for consistency across all datasets. The data will then be filtered based on key performance metrics for each part.

2. Filtering and Ranking Criteria

* Part 1 (Table 1):
  + Metrics: profitability\_per\_txn, cost\_per\_txn, reversal\_per\_txn
  + Ranking: High profitability\_per\_txn ranks higher. Low cost\_per\_txn and reversal\_per\_txn are preferred. The top values will receive rank in column RANK 1.
* Part 2 (Table 2):
  + Metrics: success\_percentage, failure\_percentage
  + Ranking: High success\_percentage ranks higher. Low failure\_percentage ranks higher. The top values will receive rank in column RANK 2.
* Part 3 (Table 3):
  + Metrics: less\_15m, greater\_15m
  + Ranking: High less\_15m ranks higher. Low greater\_15m ranks higher. The top values will receive rank in column RANK 3.

3. Assigning Ranks

After applying the filters:

* RANK 1 for Part 1.
* RANK 2 for Part 2.
* RANK 3 for Part 3.

Each rank column is applied within each destination\_country group for consistency.

4. Merging the Data into One Sheet

A new sheet will consolidate the filtered and ranked data, including:

* destination\_country, destination\_partner
* Metrics: profitability\_per\_txn, cost\_per\_txn, reversal\_per\_txn, success\_percentage, failure\_percentage, less\_15m, greater\_15m
* RANK 1, RANK 2, RANK 3 columns

Destination\_country and destination\_partner will sort the data.

5. Calculating Total Rank and Average Rank

* Total\_Rank: Sum of RANK 1, RANK 2, and RANK 3.
* Avg\_Rank: The average of the Total\_Rank (i.e., Total\_Rank / 3).
* The Avg\_Rank will be sorted in ascending order within each destination\_country. The lowest average rank indicates the best overall performance.

6. Final Output

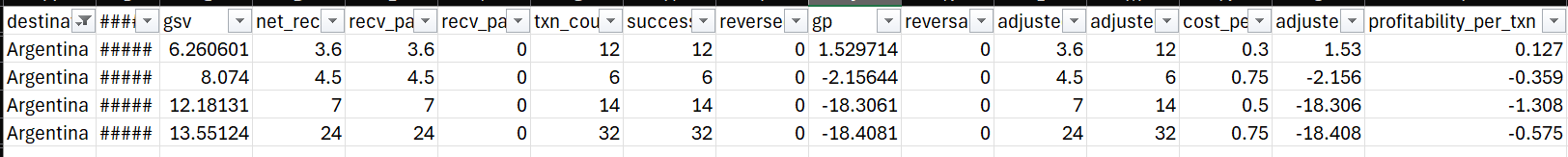
The final sheet will include the destination\_country, destination\_partner, ranked performance metrics, Total\_Rank, and Avg\_Rank. This provides a comprehensive ranking of partners across multiple metrics, helping TerraPay optimize remittance corridors and strengthen partner relationships.

**DRY Run - Methods Used**

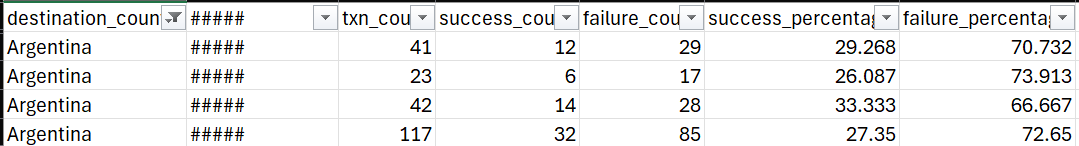
In this dry run, we will demonstrate the methods for performing the ranking and analysis process, using a single **destination\_country i.e.(**Argentina**)**. The steps below outline how the data is processed, filtered, ranked, and merged.

**Step 1: Data Collection and Grouping by Destination Country**

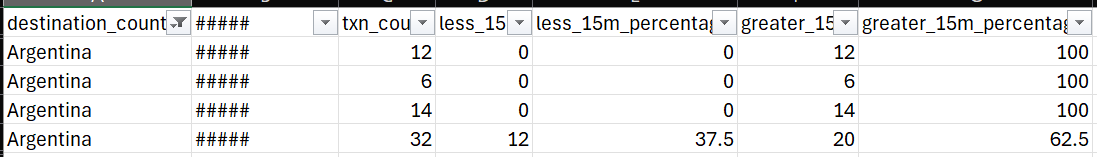
**PART 2 sec 1**

****

**PART 2 sec 2**



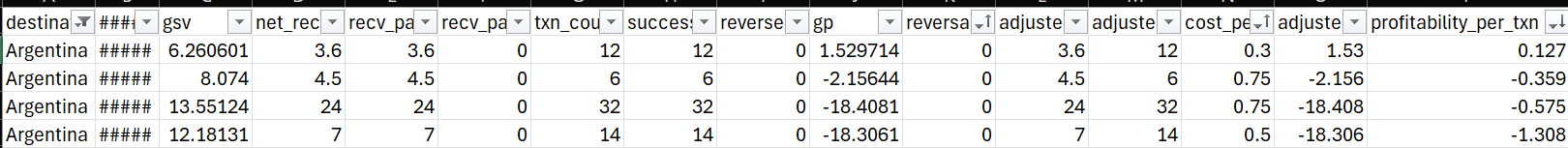
**PART 2 sec 3**



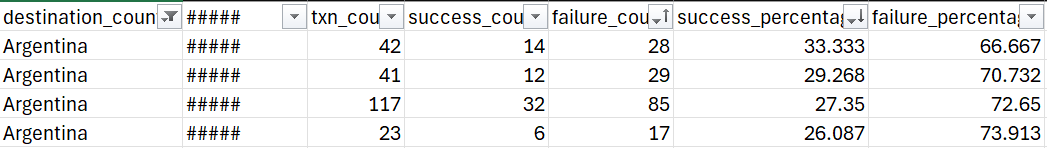
**Step 2: Apply Filters for Key Metrics**

P**art 1** (table 1):

* **Profitability per Transaction (profitability\_per\_txn)**: High values ranked higher.
* **Cost per Transaction (cost\_per\_txn)**: Low values ranked higher.
* **Reversal per Transaction (reversal\_per\_txn)**: Low values ranked higher.



**Part 2** (table\_2):

* **Success Percentage (success\_percentage)**: High values ranked higher.
* **Failure Percentage (failure\_percentage)**: Low values ranked higher.

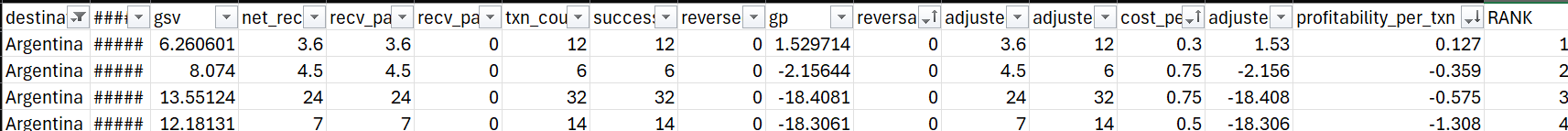
**Part 3** (table 3):

* **Transactions completed within 15 minutes (less\_15m)**: High percentages ranked higher.
* **Transactions taking longer than 15 minutes (greater\_15m)**: Low percentages ranked higher.

A screenshot of a computer

Description automatically generated

**Step 3 & 4: Rank Data Within Destination Country and assign them**

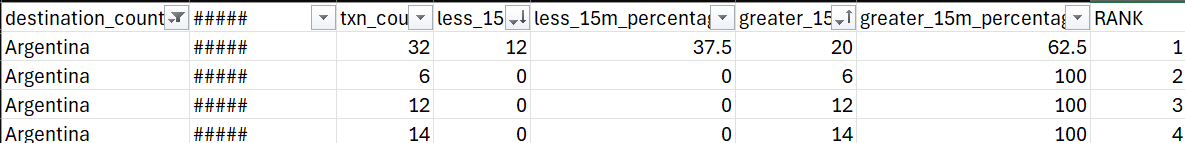
**Part 1:**

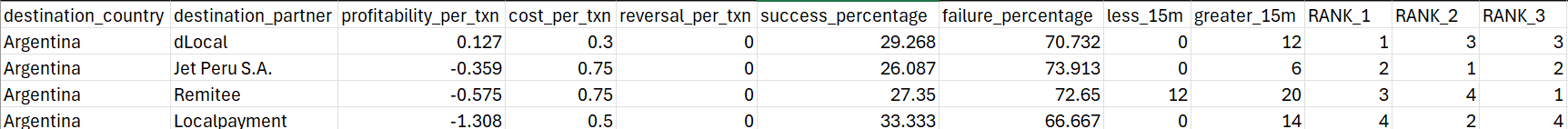
**Part 2:**

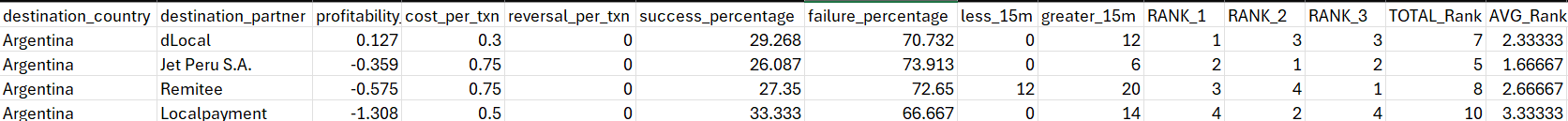
A screenshot of a computer

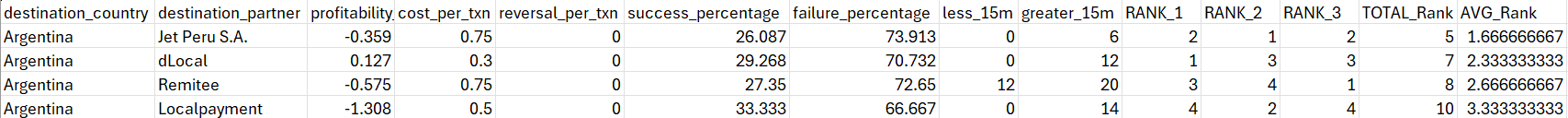
Description automatically generated

**Part 3:**

****

**Step 5: Merge Data into One Sheet**

**Step 6: Calculate Total Rank and Average Rank**

**Step 7: Sort and Group by Average Rank**

The above table shows the best destination partner which has the least rank . Hence for the above table i.e., Jet Peru S.A. is on the 1st rank.

**Final Approach for Documentation**

**1. Data Loading and Cleaning**

* **Objective**: Load CSV files for different sections (e.g., sec1, sec2, sec3), clean the data by removing any leading or trailing whitespace from column names to ensure consistency across files.
* **Method**: The process\_data\_and\_save() function loads each file, applies column name cleaning using str.strip(), and prepares the data for further processing.

**process\_data\_and\_save(input\_files, output\_files, group\_column)**

This function is responsible for processing multiple CSV files. It reads the data, sorts it based on specified criteria, and adds rank columns to facilitate further analysis. The processed DataFrames are then saved to new CSV files for easy access.

**Parameters:**

* **input\_files**: A list of strings representing the paths to the input CSV files that contain transaction data.
* **output\_files**: A list of strings representing the paths where the processed CSV files will be saved.
* **group\_column**: A string indicating the column name used for grouping the data during sorting.

**Returns:**

* None

**Example Code:**

**A computer code on a black background

Description automatically generated**This function ensures that the data is organized and ranked according to the specified grouping, making it easier to analyse performance metrics across different segments

**2. Sorting and Ranking the Data**

* **Objective**: Sort and rank the data within each group (defined by destination\_country in this case) based on specific columns relevant to each section.
* **Method**: For each input file (sec1, sec2, sec3), the data is sorted by multiple columns, such as profitability\_per\_txn, cost\_per\_txn, success\_percentage, and others, in a specified ascending or descending order. A rank column (e.g., Rank\_1, Rank\_2, Rank\_3) is then added to the DataFrame using cumcount(), which generates sequential rankings within each group.

**3. Merging and Sorting Multiple DataFrames**

* **Objective**: Merge the processed CSV files (e.g., ranked\_data1.11.csv, ranked\_data1.22.csv, ranked\_data1.33.csv) based on common keys (destination\_country and destination\_partner), calculate total and average ranks, and sort the merged data.
* **Method**: The merge\_and\_sort\_csv\_files() function merges three files on destination\_country and destination\_partner. After merging, duplicate columns are dropped, and columns related to ranks (Rank\_1, Rank\_2, Rank\_3) are handled. The total rank (Total\_Rank) is calculated by summing the individual rank columns, and the average rank (Average\_Rank) is derived by averaging the total rank.

**merge\_and\_sort\_csv\_files(file1\_path, file2\_path, file3\_path, file1\_cols, file2\_cols, file3\_cols)**

This function merges three CSV files based on common columns, specifically destination\_country and destination\_partner. It calculates total and average ranks from the ranking columns and sorts the resulting DataFrame. This is particularly useful for generating a comprehensive view of transaction performance across different partners.

**Parameters:**

* **file1\_path:** The file path to the first CSV file containing relevant transaction data.
* **file2\_path:** The file path to the second CSV file with additional metrics.
* **file3\_path:** The file path to the third CSV file containing further data points.
* **file1\_cols, file2\_cols, file3\_cols:** Lists of strings that specify which columns to include from each file.

**Returns:**

* A DataFrame that contains the merged and sorted results.

**Example Code:**

A screen shot of a computer screen

Description automatically generated

This function is essential for consolidating data from multiple sources, allowing for a more holistic analysis of transaction performance metrics.

**4. Swapping Rows Based on Comparison**

* **Objective**: Swap rows within each group (defined by destination\_country and Average\_Rank) based on a comparison of specified columns (e.g., profitability\_per\_txn, success\_percentage, etc.).
* **Method**: The swap\_rows\_based\_on\_columns() function groups the DataFrame by destination\_country and Average\_Rank, then compares the specified columns within each group. If any column in one row is greater than its counterpart in another row, the rows are swapped. This step ensures that the rows are sorted based on the values of the selected columns.

**5. Final Data Output**

* **Objective**: Combine the processed and modified data from the various steps into a single DataFrame and save the result as a final CSV file.
* **Method**: After merging, sorting, and row swapping, the final DataFrame is saved as Destination\_partner\_ranking.csv using the to\_csv() method. This file contains the top-ranked data for each source\_partner, along with relevant ranking information and sorted based on the provided columns.

**swap\_rows\_based\_on\_columns(df, columns\_to\_compare, columns\_to\_drop=None)**

This function modifies the DataFrame by swapping rows based on specified columns. It allows for a dynamic rearrangement of data, which can be crucial for comparative analysis. Users can also specify columns to drop after the operation to clean up the final DataFrame.

**Parameters:**

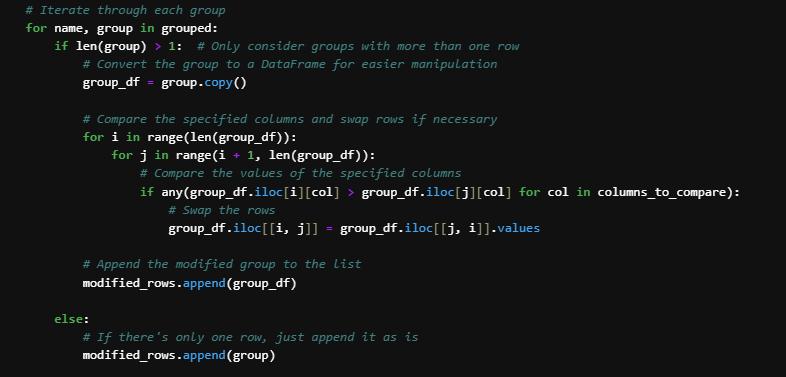
* **df**: The DataFrame that needs to be modified.
* **columns\_to\_compare**: A list of column names that will be used for comparison to determine which rows to swap.
* **columns\_to\_drop**: An optional list of column names that should be dropped from the final DataFrame.

**Returns:**

* A modified DataFrame with swapped rows and any specified columns removed.

**Example Code:**





A black background with white text

Description automatically generated

By utilizing this function, users can ensure that their data is organized in a way that highlights important performance metrics and comparisons.

**Output**

* The final output is a CSV file (Destination\_partner\_ranking.csv) containing the sorted and ranked data for each source\_partner, with added regional director information, and free from duplicates and irrelevant data. This file provides a comprehensive view of the top GSV data for each destination partner, with clear rankings based on various metrics.

Final result : The result is stored in the CSV file named Destination\_partner\_ranking.csv that contains the average ranking of all destination\_partner grouped by destination\_country the one with the lowest average value it the best for each country.

