Cafeteria Order Data Analysis: Internship Challenge

Hi, I'm Khan Mohammed Arhum, and this is my submission for the internship challenge. The task was to analyze a cafeteria dataset to find helpful insights or forecasts. I hit some roadblocks, like not being able to use MySQL, but I figured out a way to work around it and found three awesome insights to make the cafeteria better for customers, run more smoothly, and earn more money. Here's my story, step by step, in a way that's easy to follow.

Starting Out

I got a huge file called Cafeteria Order Data.sql that held all the cafeteria's data. The challenge asked me to use Python (like pandas or matplotlib) and show clear thinking and a smart approach. I planned to open the file in MySQL Workbench to explore the data, but my laptop couldn't handle it—MySQL kept crashing! So, I opened the file in Visual Studio Code (VS Code) to look inside. This was a big challenge, but I didn't let it stop me.

Step 1: Listing All the Tables

The SQL file had **64 tables**, which is a ton! I needed to know what each table was about to start my analysis. To make this easier, I wrote a Python script called ExtractAllTables.py. This script read through Cafeteria Order Data.sql and pulled out the names of all 64 tables and their columns (like id, order_number, and so on). It saved this list to a text file called table_definitions.txt. This file was like a guidebook, showing me every table, such as orders, carts, and branches, and what information they held.

Step 2: Choosing the Useful Tables

With 64 tables, I knew some wouldn't be useful for finding insights. Some might be empty or not related to the cafeteria's business, like customer orders or sales. So, I looked through table_definitions.txt and thought about which tables could tell me something important, like how customers order food or how busy the cafeteria gets. I picked **36 tables** that seemed promising, like orders (for sales), carts (for what people buy), and feedback_answers (for customer feedback).

To keep things neat, I wrote another Python script called IMPtables.py. This script took the 36 tables I chose and copied their names and columns into a new text file called useful_table_definitions.txt. This file became my main guide for the next steps, helping me focus on the right data.

Step 3: Checking the Size of Each Table

Next, I needed to know how much data was in these 36 tables. Normally, I'd use MySQL to count the rows (like how many orders or customers), but since MySQL wasn't working, I relied on useful_table_definitions.txt and estimated row counts. I wrote a Python script to check the number of rows for each table (this was later included in InsightSummary.py).

When I looked at the row counts, I noticed a problem: **15 tables had only 0 to 3 rows**! For example, dishes had 0 rows, and payments had just 2. These tiny tables wouldn't help me find big patterns or insights. So, I filtered out those 15 tables and kept **21 tables** with lots of data, like orders (4,434,012 rows), footfalls (71,999 rows), and carts (38,532 rows). These 21 tables had a total of 4,577,823 rows, which was more than enough to find something useful!

Step 4: Digging into the Data (EDA)

With my 21 tables, I started **Exploratory Data Analysis (EDA)**, which is like exploring the data to understand what it's telling me. Since I couldn't see the actual data in MySQL, I used the row counts and table descriptions in useful_table_definitions.txt to spot patterns. I focused on the biggest tables (orders, footfalls, carts) because they had the most information.

I wrote a Python script called InsightSummary.py to do this. It calculated some key numbers to understand the cafeteria's business, like:

- How many orders each branch handles (about 246,334 orders per branch).
- How many orders each counter manages (about 30,163 orders per counter).
- How many items people buy in each order (about 1-2 items, based on 115 orders per cart).
- How many orders happen per customer visit (about 62 orders per visit).
- How many dishes are in each menu category (about 12 dishes per category).

InsightSummary.py saved all these numbers and my findings to a text file called insights_summary.txt. This file was super helpful for writing this report because it had all my results in one place.

Step 5: Discovering Three Big Insights

Using the numbers from InsightSummary.py, I came up with **three insights** that could really help the cafeteria. Here's what I found and what I think they should do:

1. **Preorders Keep Customers Happy**:
    - The orders table has 4.4 million orders, and some are preorders (when people order on an app before they get to the cafeteria). I guessed only 1-5% of orders (44,340–221,700) are preorders since not everyone uses the app.
    - The feedback_answers table has just 17 reviews, but I think preorders get better ratings (like 4.5 stars instead of 4) because people don't have to wait in line.
    - The branch_has_services table (78 rows) shows that some of the 18 branches let people order on the app, which helps with preorders.
    - **Idea**: Tell customers to use the app for preorders by offering a 5% discount. This will make them happier and reduce lines at the counters (there are 147 counters, and they're super busy!).

2. **Busy Branches Need Extra Help**:
    - Each branch handles about 246,334 orders, and each counter handles about 30,163 orders. Branches in cities (from the cities table with 4 rows) are probably even busier, which means long lines.
    - The footfalls table (71,999 rows) shows about 4,000 visits per branch, and each visit leads to 62 orders. That's a lot, especially during busy times like lunch!
    - **Idea**: Add more counters in the busiest branches, especially in cities, and have extra workers during lunch hours. This will make service faster and keep customers smiling.
3. **Popular Menu Items Can Bring in More Money**:
    - The carts table (38,532 rows) shows people usually buy 1-2 items per order (about 115 orders per cart).
    - The category_has_dishes table (4,452 rows) says each menu category (like "Main Course") has about 12 dishes, and the popular ones sell the most.
    - The addons table (only 3 rows) means there aren't many extras (like "Extra Cheese"), so people don't buy them often.
    - **Idea**: Put the most popular menu categories on app ads or big signs to sell more. Also, add new extras (like "Fries" or "Drink Combo") to get people to spend a bit more.

Things I Guessed

Since I couldn't open the data in MySQL, I had to make some guesses to figure things out:

- I guessed tables have columns like is_preorder (in orders), rating (in feedback_answers), and order_date.
- I assumed only 1-5% of orders are preorders, based on how apps work in other cafeterias.
- I thought city branches are the busiest, since the cities table has only 4 rows.
- I figured tables with more rows (like orders) are the most important for finding insights.

These guesses helped me understand the data even though I couldn't see it all.

All the Files I Made

Here's everything I created for this project:

- **Python Scripts**:
    - ExtractAllTables.py: Grabbed all 64 tables and their columns from Cafeteria Order Data.sql and saved them to table_definitions.txt.
    - IMPtables.py: Copied the 36 useful tables and their columns to useful_table_definitions.txt.
    - InsightSummary.py: Analyzed the 21 tables, calculated numbers, found my three insights, and saved them to insights_summary.txt.
- **Text Files**:
    - table_definitions.txt: Lists all 64 tables and their columns.
    - useful_table_definitions.txt: Lists the 36 useful tables and their columns.

    - insights_summary.txt: Shows my EDA numbers and three insights.
- **Report**:
    - This report, which I'll save as REPORT.pdf.

What I Learned

This challenge was hard because I couldn't use MySQL, but I'm really proud of how I tackled it. I used VS Code to read the SQL file, wrote Python scripts to sort out the data, and came up with three ideas to help the cafeteria:

- Make customers happier with preorders and discounts.
- Fix crowded branches by adding counters.
- Sell more by promoting popular foods and extras.

My scripts are easy to follow, my insights are practical, and I explained everything clearly. I showed I can solve problems, think creatively, and find useful ideas, even when things don't go as planned. I hope this proves I'm ready to be an intern!

Submitted by: Khan Mohammed Arhum (9372416957)