66Degrees Assessment

Insights from exploratory data analysis and data pre-processing

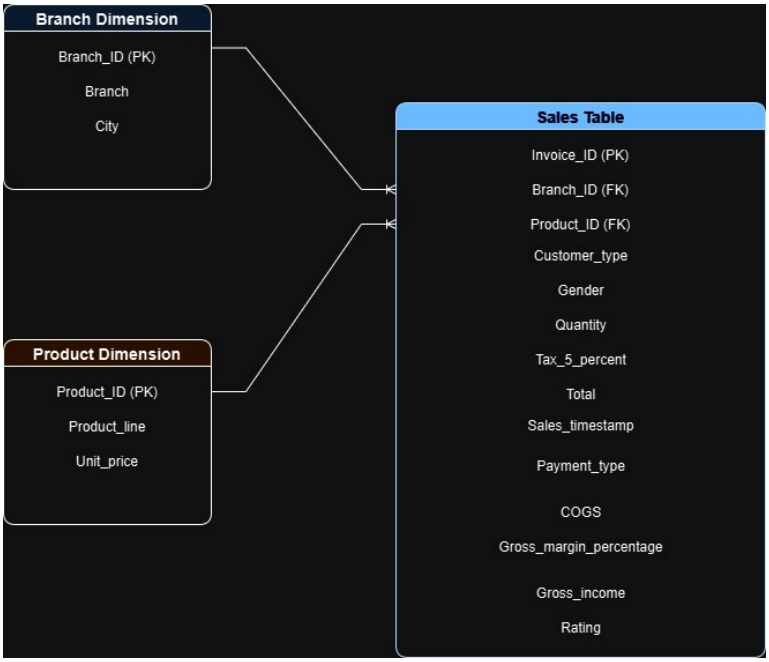
- This dataset show details of 3 branches(A,B,C) sales for a supermarket through the first quarter of 2019.
- All columns have no null values.
- The data has sales information about 6 different product_line.
- The male and female distribution is almost equal with 501 females and 499 males.
- The Product line Electronic Accessories has highest quantity of items sold while the maximum Profit is generated by Food and Beverages Product line.
- Yangon City has the highest number of sales transactions recorded.
- The most common payment method is Ewallet.
- The feature names were not standard hence removed any trailing spaces and replaced spaces with _.

Schema Design

- Star Schema
- Fact Table – Sales related information
- Dimension Table –

  Product – Product related information

  Branch – Branch related information
- Unique Identifier generation for the data to make sure we can map the tables.



SQL Queries for data analysis

1. Total Gross Income by City from Sales Data

```
SELECT
   City,
   SUM(Gross_income)
FROM
   Sales A
   LEFT JOIN Branch B ON A.Branch_ID = B.Branch_ID
GROUP BY
   City;
```

City	Total_Gross_Income
Mandalay	5057.03
Naypyitaw	5265.18
Yangon	5057.16

2. Aggregating Total Quantity of Products Sold by Gender and Product Line

```
SELECT
   Gender,
   Product_Line,
   sum(quantity) AS c1
FROM
   Sales A
   LEFT JOIN Product B ON A.Product_ID = b.Product_ID
GROUP BY
   Gender,
   Product_Line
Order BY
   c1 desc,
   Gender;
```

Gender	Product_line	Qty_purchased
Female	Fashion accessories	530
Female	Food and beverages	514
Male	Health and beauty	511
Female	Home and lifestyle	498
Female	Sports and travel	496
Female	Electronic accessories	488
Male	Electronic accessories	483
Male	Food and beverages	438
Male	Sports and travel	424
Male	Home and lifestyle	413
Male	Fashion accessories	372
Female	Health and beauty	343

3. Ranking Gross Income by Product Line within Each City from Sales Data

```
select
   b.city,
   p.product_line,
   SUM(s.gross_income) as total_profit,
   RANK() OVER (
      PARTITION BY
         b.city
      ORDER BY
         SUM(s.gross_income) DESC
   ) AS Sales_Rank_By_Branch_Product
FROM
   Sales s
   LEFT JOIN Branch b ON s.branch_id = b.branch_id
   LEFT JOIN Product p ON s.product_id = p.product_id
GROUP BY
   city,
   product_line;
```

City	Product_line	Total_Profit	Rank
Mandalay	Sports and travel	951.819	1
Mandalay	Health and beauty	951.46	2
Mandalay	Home and lifestyle	835.6745	3
Mandalay	Electronic accessories	811.9735	4
Mandalay	Fashion accessories	781.5865	5
Mandalay	Food and beverages	724.5185	6
Naypyitaw	Food and beverages	1131.755	1
Naypyitaw	Fashion accessories	1026.67	2
Naypyitaw	Electronic accessories	903.2845	3
Naypyitaw	Health and beauty	791.206	4
Naypyitaw	Sports and travel	750.568	5
Naypyitaw	Home and lifestyle	661.693	6
Yangon	Home and lifestyle	1067.4855	1
Yangon	Sports and travel	922.5095	2
Yangon	Electronic accessories	872.2435	3
Yangon	Food and beverages	817.2905	4
Yangon	Fashion accessories	777.7385	5
Yangon	Health and beauty	599.893	6

4. Cumulative Sales over month

```
SELECT
   strftime('%m', s.sales_timestamp) as month,
   SUM(s.Total) AS daily_sales,
   SUM(SUM(s.Total)) OVER (ORDER BY strftime('%m', s.sales_timestamp)) AS cumulative_sales
FROM
   Sales s
GROUP BY
   month
ORDER BY
   month;
SELECT
   strftime('%m', s.sales_timestamp) as month,
   SUM(s.Total) AS daily_sales,
```

```
    SUM(SUM(s.Total)) OVER (ORDER BY strftime('%m', s.sales_timestamp)) AS cumulative_sales
FROM
    Sales s
GROUP BY
    month
ORDER BY
    month;
```

Month	Sales	Cumulative Sales
1	116292	116292
2	97219	213511
3	109456	322967

Cloud Architecture Diagram