**Swiggy Analysis**

1. **Mount the Azure Storage Gen2 to Azure DataBricks.**
   1. Created 2 containers Raw and Processed
   2. Uploaded the Raw Csv file to Raw Container



1. **Ingest the data from Azure Storage Gen 2 raw container**
   1. Create a Schema of the dataset using StructType
   2. Create a Spark DataFrame by reading the CSV
   3. Create the id column using the monotonically\_increasing\_id imported from pyspark.sql.functions
   4. Added the column id using Window function and row\_number() starting from 1 and increasing
   5. Converted to Parquet file format and written to the Azure Storage Gen 2 Container.



1. **Creation of Dimension Table.**
   1. Read the raw file from Azure Storage
   2. Created an array with required the column header
   3. Created a function that takes a parameter as column header and selects the distinct value and creates a table.
   4. Using Column header array used a for loop and called a function to create a table for each column field.

 

1. **Creation of Fact Table.**
   1. Read the raw file from Azure Storage
   2. Created a data frame for all the dimension table.
   3. Created a window specification for generating ID
   4. Used a left join to get all the data from dimension table and generated fact\_id and created a fact table and renamed the column.
   5. Created the fact table.

 

1. **Reports**
   1. Count of Restaurant City Wise
      1. Read the required fields using dimensional table and fact table.
      2. Used GroupBy() to group the City name and used Count() to count the restaurants.
      3. Converted the DataFrame to Parquet format and written to the Azure Storage Gen 2 Processed Container

 

* 1. Top 10 Restaurants based on average cost of 2
     1. Read the required fields using dimensional table and fact table.
     2. Used the Window function to create a Partition based on city with Average cost of 2 in descending Order
     3. Created a row\_number for each partition and filtered with <10 to get the top 10 restaurants
     4. Converted the DataFrame to Parquet format and written to the Azure Storage Gen 2 Processed Container

 

* 1. Top 10 Restaurants based on average votes
     1. Read the required fields by joining dimensional table and fact table.
     2. Used the Window function to create a Partition based on city with Votes in descending Order
     3. Created a row\_number for each partition and filtered with <10 to get the top 10 restaurants
     4. Converted the DataFrame to Parquet format and written to the Azure Storage Gen 2 Processed Container

 

* 1. Top 10 restaurant based on Rating City wise
     1. Read the required fields by joining dimensional table and fact table.
     2. Used the Window function to create a Partition based on city with Rating Stars out of 5 in descending Order and Restaurant Name in Ascending Order
     3. Created a row\_number for each partition and filtered with <10 to get the top 10 restaurants
     4. Converted the DataFrame to Parquet format and written to the Azure Storage Gen 2 Processed Container

 

* 1. Rating based on delivery availability City wise
     1. Read the required fields by joining dimensional table and fact table.
     2. Used the Window function to create a Partition based on city with Has Online Delivery in Ascending Order
     3. Grouped “City” and “Has online delivery” column and implemented a min function to give the minimum of “Yes” and “No”
     4. Converted the DataFrame to Parquet format and written to the Azure Storage Gen 2 Processed Container

 

* 1. Avg cost for 2 based on cuisine city wise
     1. Read the required fields by joining dimensional table and fact table.
     2. Used the Window function to create a Partition based on city with Cuisines in Ascending Order
     3. Grouped “City” and “Cuisines” column and implemented a avg function to give the average of each cuisines based on the city
     4. Converted the DataFrame to Parquet format and written to the Azure Storage Gen 2 Processed Container

 