**Project Objective** The project aims to extract and integrate data from multiple sources, transform, and load in a relational database (PostgreSQL) for a more effective data analysis.

**Project Team**

* Mary Jane Rafol
* Marie Sanon
* Yuanfeng Xu

Michelin Star restaurants are renowned for their distinct, consistent, and excellent quality cuisines. Quality and affordability are salient factors that influence consumer behavior. Having an integrating data set that enables stratification of these restaurants by affordability indicators ($-$$$$) becomes a useful tool for consumer decision-making.

**Project Methodology** The project team applied the data integration process, Extract, Transform and Load (ETL). Two data sources were identified—the list of Michelin Star restaurants, and the restaurant affordability ratings, which were all extracted and imported in Jupyter notebook using Pandas. The relevant data were reviewed for consistency and transformed through normalization, cleansing, formatting, and conversion. After the integrity of data is validated, the table schema was created, and the data were loaded into PostgreSQL.

ETL Methodology

Extract Transform Load

Tool: **Pandas**

* Filter relevant data
* Normalize and cleanse key field column data (“restaurant name")
* Convert data to correct format (“phone numbers”, “affordability”)
* Validate data quality and exclude invalid records

Source: **Kaggle**

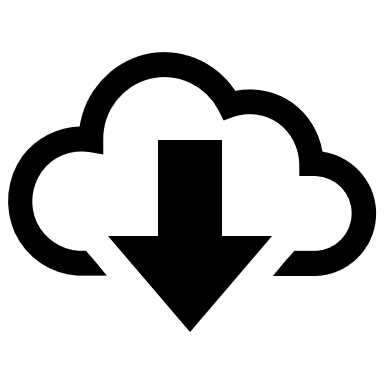
* Michelin Star Restaurants ([.csv](https://www.kaggle.com/datasets/ngshiheng/michelin-guide-restaurants-2021?select=michelin_my_maps.csv))
* Affordability Indicator ($-$$$$$) ([.csv](https://www.kaggle.com/datasets/jackywang529/michelin-restaurants))

Tool: **Pandas & PostgreSQL**

* Create the table schema
* Load the transformed data from the Pandas to PostgreSQL database
* Stratify records based on affordability

**Project Limitations** The project focused on Michelin Star restaurants located in the United States of America. Due to the limitation of data sets available in Kaggle, whereby only 2019 affordability rating data set is available, the recently awarded restaurants in 2021 do not have a corresponding affordability data. The ETL data integration process would have been more useful if the 2021 affordability rating data source is available and used for the project.

ETL Process Visualization & Narrative



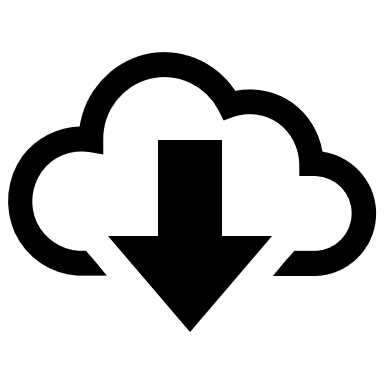
**Python ETL Script**

**Table 1**

Import restaurant list .csv file

Import and append multiple affordability rating .csv files

Michelin Star Restaurants (.csv)



Cleanse, convert and formatted relevant data for consistency

Normalize primary key field column data in both tables

**Table 2**

Affordability Rating (.csv)

This

Prepare data for loading, connect to DB and load data

Validate data quality and exclude invalid records

PostgreSQL

“Michelin-Star-DB”



One-star (.csv)

Two-star (.csv)

Three-star (.csv)

Three (3) Files

**Extract** Two data sources from Kaggle.com were identified—(1) the list of Michelin Star restaurants, and (2) the restaurant affordability ratings, divided in three separate files based on the number of stars awarded. The .csv files were downloaded and imported into Pandas data frame using Python script and reviewed for data integrity and records with missing information were identified using the .count() function.

**Transform** Some data issues were identified during the review of imported data sources. These issues had to be corrected through normalization, cleansing, formatting, and conversion in order to prepare the data prior to the load process. The major issues and how the team managed to resolve them are presented in the succeeding paragraphs:

* 1. *Phone numbers read in Pandas as float.* The data issue was resolved by specifying a datatype parameter (dtype={“PhoneNumber”: “str”}) to convert format to string upon import of the csv file. Note that numbers in csv are recognized in Pandas as float.
  2. *The primary key field name (column title) from two data sources has casing variation.* The data issue was resolved by renaming and aligning the column names of the primary key (PK) in the two data frames (PK: “RestaurantName”).
  3. *The affordability rating ($) uses dollar sign which is considered an illegal character in Python and therefore not read properly in Pandas.* The data issue was resolved by applying the lambda function to replace the dollar sign ($) with a plain letter (S). The data returned was then replaced them with a qualitative description for better user comprehension:

$$$$$ Extremely expensive

$$$$ Very expensive

$$$ Expensive

$$ Moderately expensive

$ Less expensive

* 1. *Some records have incomplete data (phone number, website URL).* The data issue was resolved by excluding records with incomplete details using the *.dropna()* function (see Limitations) .

**Load** After the data issues were resolved, the data was prepared for loading in PostgreSQL database. Prior to loading, a database, named “Michelin-Star-DB” was established and the schemas for two tables—restaurant\_list, and affordability\_rating, were created in in PostgreSQL. We used Python script to set-up the connection to the created database and to load our data frames in the two tables created. The two tables were joined to extract the stratified list of restaurants by price range using PostgreSQL query.