## Luca Savarese

CIS 4400

Homework 1

**Metro Bikeshare Data Analysis Project**

## Datasets Source: [**https://bikeshare.metro.net/wp-content/uploads/2024/10/metro-trips-2024-q3.zip**](https://bikeshare.metro.net/wp-content/uploads/2024/10/metro-trips-2024-q3.zip) [**https://bikeshare.metro.net/wp-content/uploads/2024/10/metro-bike-share-stations-2024-10-01.csv**](https://bikeshare.metro.net/wp-content/uploads/2024/10/metro-bike-share-stations-2024-10-01.csv)

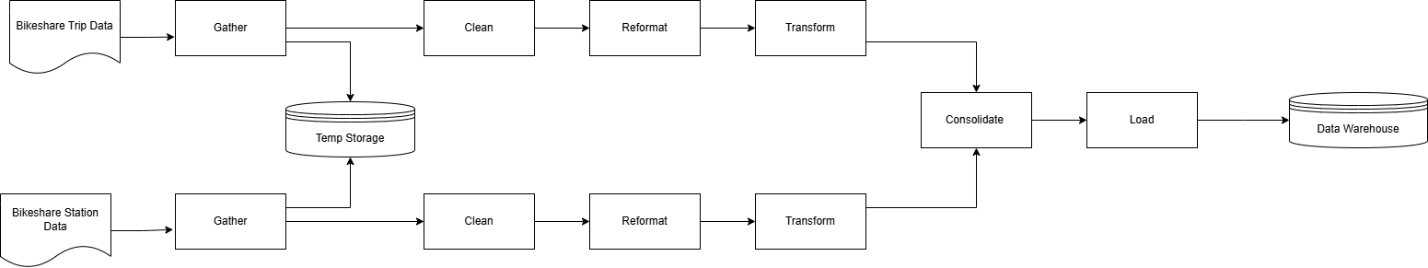
## Business Requirements:

* **Peak Times Analysis:** Understand when bike usage is highest.
* **Station Demand:** Identify the busiest and least-used stations.
* **Passholder Trends:** Explore how different passholder types contribute to bike usage.

## Functional Requirements:

* **Data Loading**: Import data into the analysis tool.
* **Data Cleaning**: Remove missing or inconsistent values.
* **Time-Based Analysis**: Group trips by hour/day to find peak usage times.
* **Station Analysis**: Count trips starting and ending at each station.
* **Trip Duration Analysis**: Calculate and summarize trip durations.

## Information Architecture:



This information architecture provides the general data flow of the Metro Bikeshare System. The whole data cycle begins with collecting trip and station data, then cleaning, reformatting, and transforming to ensure consistency and quality. Data is staged into a temporary space before ultimately being consolidated. Then, the consolidated data is loaded into the data warehouse which will serve as the primary resource for analysis. The data warehouse will power visualizations to allow us to explore different types of analytics such as peak usage times, station demand, and passholder trends.

## Data Architecture:

A diagram of a data storage

Description automatically generated

The data architecture for the Metro Bikeshare project employs an ETL (Extract, Transform, Load) mechanism for data management and processing. Raw bikeshare trip and station data are consolidated, cleaned, and formatted using Python before being transformed into a dimensional model. The processed data is stored temporarily in a staging area and then uploaded to a centralized data mart optimized for analytical queries and seamless integration with visualization tools. This architecture supports analyses related to peak usage hours, station demand, and trends in passholder numbers.

## Dimensional Modeling:

A screenshot of a computer

Description automatically generated

**Fact table:**

**Fact\_Trips:**

CREATE TABLE Fact\_Trips (

trip\_id INT PRIMARY KEY,   
duration INT,   
start\_time DATETIME,   
end\_time DATETIME,   
start\_station\_id INT,   
end\_station\_id INT,   
passholder\_type\_id INT,   
bike\_type\_id INT,   
route\_id INT,   
FOREIGN KEY (start\_station\_id) REFERENCES Dim\_Start\_Station(start\_station\_id), FOREIGN KEY (end\_station\_id) REFERENCES Dim\_End\_Station(end\_station\_id),   
FOREIGN KEY (passholder\_type\_id) REFERENCES Dim\_Passholder(passholder\_type\_id),   
FOREIGN KEY (bike\_type\_id) REFERENCES Dim\_Bike\_Type(bike\_type\_id),   
FOREIGN KEY (route\_id) REFERENCES Dim\_Trip\_Route(route\_id)

);  
  
**Dimension Tables:**

**Dim\_Start\_Station:**

CREATE TABLE Dim\_Start\_Station (

start\_station\_id INT PRIMARY KEY,  
start\_kiosk\_name VARCHAR(255),  
start\_region VARCHAR(255),  
start\_lat DOUBLE,  
start\_lon DOUBLE

);

**Dim\_End\_Station:**

CREATE TABLE Dim\_End\_Station (

end\_station\_id INT PRIMARY KEY,  
end\_kiosk\_name VARCHAR(255),  
end\_region VARCHAR(255),  
end\_lat DOUBLE,  
end\_lon DOUBLE

);

**Dim\_Passholder:**

CREATE TABLE Dim\_Passholder (

passholder\_type\_id INT AUTO\_INCREMENT PRIMARY KEY,  
passholder\_type VARCHAR(255)

);

**Dim\_Bike\_Type:**

CREATE TABLE Dim\_Bike\_Type (

bike\_type\_id INT AUTO\_INCREMENT PRIMARY KEY,

bike\_type VARCHAR(255)

);

**Dim\_Trip\_Route:**

CREATE TABLE Dim\_Trip\_Route (

route\_id INT AUTO\_INCREMENT PRIMARY KEY,  
trip\_route\_category VARCHAR(255)

);