## **Overview**

This repository contains three data engineering projects focused on extracting, transforming, and loading (ETL) data into **PostgreSQL** and **Google Cloud Platform (GCP)** using **Python, Pandas, PostgreSQL, Google Cloud Storage (GCS), BigQuery, and dbt**.

## **Project 1: Data to PostgreSQL**

Project 1 is designed to extract data from CSV and JSON sources, clean it, and insert it into a PostgreSQL database.

### *Workflow*

1. **Extract Data:**
   * Reads a CSV file (ev\_population\_data.csv).
   * Fetches JSON data from an API (https://data.cityofchicago.org/resource/ydr8-5enu.json).
2. **Transform Data:**
   * Basic cleaning and standardizing the data using Pandas.
3. **Load Data into PostgreSQL:**
   * Inserts cleaned data into specific PostgreSQL tables:
     + ev\_population\_data.ev\_data
     + permits\_data.permits

## **Project 2: PostgreSQL to Google Cloud (GCS & BigQuery)**

Project 2 extends the data pipeline by extracting data from CSV, JSON, and PostgreSQL, uploading it to Google Cloud Storage (GCS), and loading it into BigQuery.

### *Workflow*

1. **Extract Data:**
   * Reads CSV and JSON files similar to Project 1.
   * Fetches data from PostgreSQL
2. **Transform Data:**
   * Cleans and standardizes data.
3. **Upload to GCS:**
   * Converts data to Parquet format and uploads it to a GCS bucket.
4. **Load into BigQuery:**
   * Loads the data from GCS into BigQuery tables:
     + ev\_population\_data.ev\_data
     + permits\_data.permits
     + postgres\_data.permits\_postgres

**Project 3: Data Transformation & Versioning with dbt**

Project 3 focuses on **transforming and modeling data using dbt**, implementing **staging models, deduplication, and snapshots** to track historical changes.

### *Workflow*

1. **Staging (stg\_\* Models):**
   * Cleans and standardizes raw data from BigQuery sources.
   * Ensures proper data types and removes invalid records.
2. **Intermediate & Fact/Dimension Models:**
   * Deduplicates data and applies business transformations.
   * Creates fact and dimension tables for efficient querying.
3. **Snapshots:**
   * Tracks changes over time for specific columns (e.g., country in customers table).
4. **Testing & Documentation:**
   * Implements dbt tests (e.g., uniqueness, not null constraints).
   * Documents models and relationships.

## **Conclusion**

These projects demonstrate a complete ETL pipeline from **local and API-based data ingestion** to **cloud storage, data transformation, and analytics-ready datasets**. The use of dbt in Project 3 ensures **scalable and modular transformations**, making data more accessible for business intelligence and analytics in Google Cloud.