Airbnb data pipeline using Apache Airflow

By Roger Yu 10906675

# Introduction

# Data summary

# Data pipeline

## Project set up

### Create schemas

In DBeaver, create schemas

* raw
* star
* data\_mart

## Requirements

## Star Schema

## DAGs

### Functions

* drop nulls (pandas)
* set column as primary key (postgres)
* PostgresOperator to add a primary key  
  ALTER TABLE table\_name ADD PRIMARY KEY (column\_1, column\_2);

### Control

SQL queries are called through [pandas.read\_sql](https://pandas.pydata.org/docs/reference/api/pandas.read_sql.html) and [pandas.DataFrame.to\_sql](https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to_sql.html). which means that while the intermediate data is passed into the Airflow server the actual processing is done in the Postgres server.

Another advantage of this approach the affordability to do post-transformation data validation through [Great Expectations](https://greatexpectations.io/) for example.

# Business questions

# Challenges

Uploading dataframe by first transforming them into dictionary. So used [pandas.DataFrame.to\_sql](https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to_sql.html) and [geopandas.GeoDataFrame.to\_postgis](https://geopandas.readthedocs.io/en/latest/docs/reference/api/geopandas.GeoDataFrame.to_postgis.html) instead.

The pandas method doesn’t have a parameter to specify the keys. So specifying keys is done after uploading the table.

---------------------------------------------------------------------------

InFailedSqlTransaction Traceback (most recent call last)

<ipython-input-136-8d87cf2c9e47> in <module>

----> 1 execute\_values(cursor, insert\_query, values, page\_size=len(insert\_df))

/opt/conda/lib/python3.8/site-packages/psycopg2/extras.py in execute\_values(cur, sql, argslist, template, page\_size, fetch)

**1290** parts.append(b',')

**1291** parts[-1:] = post

-> 1292 cur.execute(b''.join(parts))

**1293** if fetch:

**1294** result.extend(cur.fetchall())

InFailedSqlTransaction: current transaction is aborted, commands ignored until end of transaction block

Figure 1

# Improvements

* bigint for all int columns is not necessary
* text for all character columns is not necessary
* separate schemas, see [conversation](https://teams.microsoft.com/l/message/19:849f80caef04456f8cbdb673c32165e3@thread.tacv2/1620943350077?tenantId=e8911c26-cf9f-4a9c-878e-527807be8791&groupId=576a621b-4d1e-498b-8b03-e51297557e0c&parentMessageId=1620931136845&teamName=Big%20Data%20Engineering&channelName=Assignment%202&createdTime=1620943350077)
* data validation, e.g. [Great Expectations](https://greatexpectations.io/).