docker run –name assignment2 -e POSTGRES\_PASSWORD=masters123 -e POSTGRES\_USER=risshikaa -e POSTGRES\_DB=shelter postgres

sampletest\_1

docker exec -it sampletest\_1 bash

docker run --name sampletest1 -e POSTGRES\_PASSWORD=masters123 -e POSTGRES\_USER=risshikaa -e POSTGRES\_DB=shelter -d postgres

import pandas as pd

import numpy as np

import psycopg2

import argparse

from sqlalchemy import create\_engine

def extract\_data(source):

return pd.read\_csv(source)

def transform\_data(data):

new\_data = data.copy()

new\_data.dropna(inplace = True, axis=0)

new\_data.columns = new\_data.columns.str.lower().str.replace(' ', '\_')

new\_data['datetime'] = pd.to\_datetime(new\_data['datetime'], format='%m/%d/%Y %I:%M:%S %p')

new\_data[['month','year']] = new\_data.monthyear.str.split(' ', expand = True)

new\_data['sex'] = new\_data['sex\_upon\_outcome'].replace('Unkown', np.nan)

new\_data.drop(columns = ['monthyear', 'sex\_upon\_outcome'], inplace = True)

print(new\_data.columns)

return new\_data

def load\_data(data):

# Dimension Table 1 (dim\_animal)

dim\_animal = new\_data[['animal\_id', 'name', 'date\_of\_birth', 'animal\_type', 'breed', 'color', 'sex']]

# Dimension Table 2 (dim\_time)

dim\_time = new\_data[['datetime', 'month', 'year']]

dim\_time['date\_id'] = dim\_time.index + 1

# Dimension Table 3 (dim\_outcome\_type)

dim\_outcome\_type = new\_data[['outcome\_type', 'outcome\_subtype']]

dim\_outcome\_type['outcome\_id'] = dim\_outcome\_type.index + 1

# Merge the dim\_animal, dim\_time, and dim\_outcome\_type DataFrames to create the fact table

fact\_animal = pd.merge(dim\_animal, dim\_time, left\_index=True, right\_index=True)

fact\_animal = pd.merge(fact\_animal, dim\_outcome\_type, left\_index=True, right\_index=True)

# Define the primary key for the fact table (animal\_outcome\_id)

fact\_animal['animal\_outcome\_id'] = fact\_animal.index + 1

# Reorder the columns to match the table structure

fact\_animal = fact\_animal[['animal\_outcome\_id', 'animal\_id', 'date\_id', 'outcome\_id', 'month', 'year', 'outcome\_type']]

dim\_animal.to\_sql("dim\_animal", conn, if\_exists="append", index=False)

dim\_time.to\_sql("dim\_time", conn, if\_exists="append", index=False)

dim\_outcome\_type.to\_sql("dim\_outcome\_type", conn, if\_exists="append", index=False)

fact\_animal.to\_sql("fact\_animal\_outcome", conn, if\_exists="append", index=False)

# dim\_animal = data[['animal\_id','name','date\_of\_birth','animal\_type','breed','color']]

# dim\_time = data[['datetime','month','year']]

# dim\_time['date\_id'] = dim\_time.index + 1

# dim\_outcometype = data[['outcome\_type','outcome\_subtype']]

# dim\_outcometype['outcome\_id'] = dim\_outcometype.index + 1

# # Merge the dim\_animal, dim\_time, and dim\_outcometype DataFrames to create the fact table

# fct\_animal = pd.merge(dim\_animal, dim\_time, left\_index=True, right\_index=True)

# fct\_animal = pd.merge(fct\_animal, dim\_outcometype, left\_index=True, right\_index=True)

# # Define the primary key for the fact table (breed\_id)

# fct\_animal['breed\_id'] = fct\_animal.index + 1

# # Reorder the columns to match the table structure

# fct\_animal = fct\_animal[['breed\_id', 'animal\_id', 'date\_id', 'outcome\_id', 'month', 'year', 'outcome\_type']]

db\_url = "postgresql+psycopg2://risshikaa:masters123@database:5432/shelter"

conn = create\_engine(db\_url)

data.to\_sql("outcomes", conn, if\_exists = "append", index = False)

# dim\_animal.to\_sql("dim\_animal", conn, if\_exists = "append", index = False)

# dim\_time.to\_sql("dim\_time", conn, if\_exists = "append", index = False)

# dim\_outcometype.to\_sql("dim\_outcometype", conn, if\_exists = "append", index = False)

# fct\_animal.to\_sql("fct\_animal", conn, if\_exists = "append", index = False)

if \_name\_ == "\_main\_":

parser = argparse.ArgumentParser()

parser.add\_argument('source', help = 'source csv')

# parser.add\_argument('target', help = 'target csv')

args = parser.parse\_args()

print("Starting...")

df = extract\_data(args.source)

new\_df = transform\_data(df)

load\_data(new\_df)

print("complete")

db:5432/animalshelter

@database:5432/shelter