**Scripts for converting the data into a format suitable for  
PostgreSQL**

**In the IDLE Shell**

Import - import pandas as pd  
  
**Reading the dataset**

df = pd.read\_csv('/Path-to-dataset.csv')

Example

df=pd.read\_csv('/Users/meghanab/Downloads/Improved\_Cleaned\_meghana\_Dataset (1).csv')  
  
**Identify columns and rows that are majority empty (e.g., more than 50% missing values)**

**threshold = 0.5**

columns\_to\_drop = df.columns[df.isnull().mean() > threshold]

rows\_to\_drop = df.index[df.isnull().mean(axis=1) > threshold]  
  
**Drop identified columns and rows**

df\_cleaned = df.drop(columns=columns\_to\_drop).drop(index=rows\_to\_drop)

**Replace remaining null or missing values with calculated values (mean for numeric columns)**

for col in df\_cleaned.select\_dtypes(include=['float64', 'int64']).columns:

if df\_cleaned[col].isnull().sum() > 0:

df\_cleaned[col].fillna(df\_cleaned[col].mean(), inplace=True)

**Replace missing values in categorical columns with the most frequent value**

for col in df\_cleaned.select\_dtypes(include=['object']).columns:

if df\_cleaned[col].isnull().sum() > 0:

most\_frequent\_value = df\_cleaned[col].mode()[0]

df\_cleaned[col].fillna(most\_frequent\_value, inplace=True)

**Display the cleaned dataset for review**

import ace\_tools as tools;

tools.display\_dataframe\_to\_user(name="Cleaned Metro Market Dataset Without Altering Values", dataframe=df\_cleaned)

**Unpivot the dataset**  
# Python Script to unpivot Zillow home value index

pd.melt(df, id\_vars=['RegionID', 'SizeRank', 'RegionName', 'RegionType', 'StateName'], var\_name='Date', value\_name='ZHVI')

# Python Script to unpivot market heat index

pd.melt(df, id\_vars=['RegionID', 'SizeRank', 'RegionName', 'RegionType', 'StateName'], var\_name='Date', value\_name='Market\_heat\_Index')

# Python Script to unpivot Sales data

pd.melt(df, id\_vars=['RegionID', 'SizeRank', 'RegionName', 'RegionType', 'StateName'], var\_name='Date', value\_name='Sale\_count')  
  
  
**Extract the data where year is >= 2018**

Step 1- Format the date field to DateTime if already not done

pd.to\_datetime(df['Date'], format='%d-%m-%Y')

Step 2 – Extract the rows having year >= 2018

Command - df[df['Date'].dt.year >= 2018]

**Saving the dataset**

df.to\_csv('/path-to-save.csv', index=False)  
  
For eg to save it in Documents use below command.

df.to\_csv('/Users/meghanab/Documents/cleaned\_zillow\_data\_meghana.csv', index=False)