

In [85]: `import pandas as pd
from sqlalchemy import create_engine`

In [86]: `#read the csv

df_price = pd.read_csv("../Resources/electricity_price.csv")
df_price.head(10)`

Out[86]:

	Series Key	ELEC.PRICE.US-ALL.M	Series Key.1	ELEC.PRICE.US-RES.M	Series Key.2	ELEC.PRICE.US-COM.M	Series Key.3	ELEC.PRICE.US-IND.M
0	Series Name	Average retail price of electricity : United S...	Series Name	Average retail price of electricity : United S...	Series Name	Average retail price of electricity : United S...	Series Name	Average retail price of electricity : United S...
1	Units	cents per kilowatthour	Units	cents per kilowatthour	Units	cents per kilowatthour	Units	cents per kilowatthour
2	Frequency	M	Frequency	M	Frequency	M	Frequency	M
3	Start Date	200101	Start Date	200101	Start Date	200101	Start Date	200101
4	End Date	202009	End Date	202009	End Date	202009	End Date	202009
5	Source	EIA, U.S. Energy Information Administration	Source	EIA, U.S. Energy Information Administration	Source	EIA, U.S. Energy Information Administration	Source	EIA, U.S. Energy Information Administration
6	202009	11.07	202009	13.55	202009	11.07	202009	7.01
7	202008	11.11	202008	13.31	202008	10.95	202008	7.09
8	202007	11.14	202007	13.26	202007	10.9	202007	7.17
9	202006	10.96	202006	13.28	202006	10.95	202006	6.94

In [87]: `df_price.describe()`

Out[87]:

	Series Key	ELEC.PRICE.US-ALL.M	Series Key.1	ELEC.PRICE.US-RES.M	Series Key.2	ELEC.PRICE.US-COM.M	Series Key.3	ELEC.PRICE.US-IND.M
count	243	243	243	243	243	243	243	243
unique	243	174	243	197	243	172	243	152
top	201811	10.28	201811	12.09	201811	10.46	201811	6.53
freq	1	6	1	4	1	6	1	7

In [88]: `df_price = df_price.drop(["Series Key.1", "Series Key.2", "Series Key.3", "ELEC.PRICE.US-ALL.M"], axis=1)
df_price.head(10)`

Out[88]:

	Series Key	ELEC.PRICE.US-RES.M	ELEC.PRICE.US-COM.M	ELEC.PRICE.US-IND.M
0	Series Name	Average retail price of electricity : United S...	Average retail price of electricity : United S...	Average retail price of electricity : United S...
1	Units	cents per kilowatthour	cents per kilowatthour	cents per kilowatthour
2	Frequency	M	M	M
3	Start Date	200101	200101	200101
4	End Date	202009	202009	202009
5	Source	EIA, U.S. Energy Information Administration	EIA, U.S. Energy Information Administration	EIA, U.S. Energy Information Administration
6	202009	13.55	11.07	7.01
7	202008	13.31	10.95	7.09
8	202007	13.26	10.9	7.17
9	202006	13.28	10.95	6.94

In [89]: `df_price=df_price.drop([0,1,2,3,4,5])`

In [90]: `df_price.head(10)`

Out[90]:

	Series Key	ELEC.PRICE.US-RES.M	ELEC.PRICE.US-COM.M	ELEC.PRICE.US-IND.M
6	202009	13.55	11.07	7.01
7	202008	13.31	10.95	7.09
8	202007	13.26	10.9	7.17
9	202006	13.28	10.95	6.94
10	202005	13.15	10.46	6.53
11	202004	13.28	10.42	6.4
12	202003	13.09	10.41	6.38
13	202002	12.85	10.36	6.41
14	202001	12.79	10.24	6.33
15	201912	12.68	10.32	6.38

In [91]: `df_price_final = df_price.sort_values(by=['Series Key'], ascending = True).reset_index(drop=True)
df_price_final.head(10)`

Out[91]:

	Series Key	ELEC.PRICE.US-RES.M	ELEC.PRICE.US-COM.M	ELEC.PRICE.US-IND.M
0	200101	7.73	7.25	4.73
1	200102	8.04	7.51	4.8
2	200103	8.32	7.7	4.86
3	200104	8.46	7.73	4.87
4	200105	8.83	7.77	5
5	200106	9.07	8.13	5.23
6	200107	9.03	8.41	5.57
7	200108	9.01	8.35	5.5
8	200109	8.92	8.22	5.31
9	200110	8.84	8.27	5.07

In [92]: `df_price_final.drop(["Series Key"], axis=1, inplace = True)
df_price_final`

Out[92]:

	ELEC.PRICE.US-RES.M	ELEC.PRICE.US-COM.M	ELEC.PRICE.US-IND.M
0	7.73	7.25	4.73
1	8.04	7.51	4.8
2	8.32	7.7	4.86
3	8.46	7.73	4.87
4	8.83	7.77	5
...	...	...	...
232	13.15	10.46	6.53
233	13.28	10.95	6.94
234	13.26	10.9	7.17
235	13.31	10.95	7.09
236	13.55	11.07	7.01

237 rows x 3 columns

In [93]: `df_price_final=df_price_final.rename(columns={ "ELEC.PRICE.US-RES.M": "residential_price", "ELEC.PRICE.US-COM.M": "commercial_price", "ELEC.PRICE.US-IND.M": "industrial_price"})
df_price_final.head()`

Out[93]:

	residential_price	commercial_price	industrial_price
0	7.73	7.25	4.73
1	8.04	7.51	4.8
2	8.32	7.7	4.86
3	8.46	7.73	4.87
4	8.83	7.77	5

In [94]: `df_price_final.index.name = "date_id"
df_price_final`

Out[94]:

	residential_price	commercial_price	industrial_price
date_id			
0	7.73	7.25	4.73
1	8.04	7.51	4.8
2	8.32	7.7	4.86
3	8.46	7.73	4.87
4	8.83	7.77	5
...	...	...	...
232	13.15	10.46	6.53
233	13.28	10.95	6.94
234	13.26	10.9	7.17
235	13.31	10.95	7.09
236	13.55	11.07	7.01

237 rows x 3 columns

## Demand

In [95]: `#read the csv

df_demand = pd.read_csv("../Resources/electricity_demand_bysector.csv")
df_demand.head(10)`

Out[95]:

	Series Key	ELEC.SALES.US-RES.M	Series Key.1	ELEC.SALES.US-COM.M	Series Key.2	ELEC.SALES.US-IND.M
0	Series Name	Retail sales of electricity : United States : ...	Series Name	Retail sales of electricity : United States : ...	Series Name	Retail sales of electricity : United States : ...
1	Units	million kilowatthours	Units	million kilowatthours	Units	million kilowatthours
2	Frequency	M	Frequency	M	Frequency	M
3	Start Date	200101	Start Date	200101	Start Date	200101
4	End Date	202009	End Date	202009	End Date	202009
5	Source	EIA, U.S. Energy Information Administration	Source	EIA, U.S. Energy Information Administration	Source	EIA, U.S. Energy Information Administration
6	202009	127583.66965	202009	112213.76361	202009	77504.78094
7	202008	158821.41757	202008	122024.58524	202008	82582.31949
8	202007	166890.82666	202007	125994.21203	202007	81276.25286
9	202006	131242.46216	202006	108695.0422	202006	75210.98956

In [96]: `df_demand = df_demand.drop(["Series Key.1", "Series Key.2"], axis=1)
df_demand.head(10)`

Out[96]:

	Series Key	ELEC.SALES.US-RES.M	ELEC.SALES.US-COM.M	ELEC.SALES.US-IND.M
0	Series Name	Retail sales of electricity : United States : ...	Retail sales of electricity : United States : ...	Retail sales of electricity : United States : ...
1	Units	million kilowatthours	million kilowatthours	million kilowatthours
2	Frequency	M	M	M
3	Start Date	200101	200101	200101
4	End Date	202009	202009	202009
5	Source	EIA, U.S. Energy Information Administration	EIA, U.S. Energy Information Administration	EIA, U.S. Energy Information Administration
6	202009	127583.66965	112213.76361	77504.78094
7	202008	158821.41757	122024.58524	82582.31949
8	202007	166890.82666	125994.21203	81276.25286
9	202006	131242.46216	108695.0422	75210.98956

In [97]: `df_demand=df_demand.drop([0,1,2,3,4,5])
df_demand.head(10)`

Out[97]:

	Series Key	ELEC.SALES.US-RES.M	ELEC.SALES.US-COM.M	ELEC.SALES.US-IND.M
6	202009	127583.66965	112213.76361	77504.78094
7	202008	158821.41757	122024.58524	82582.31949
8	202007	166890.82666	125994.21203	81276.25286
9	202006	131242.46216	108695.0422	75210.98956
10	202005	105411.88446	93405.71294	71219.94475
11	202004	97464.57724	90631.15901	69590.77096
12	202003	104006.3208	102936.85644	77564.65112
13	202002	111931.38861	101904.75773	75650.08455
14	202001	124414.14153	108856.99417	77425.60211
15	201912	121078.07664	108001.47569	80094.7451

In [98]: `df_demand_final = df_demand.sort_values(by=['Series Key'], ascending = True).reset_index(drop=True)
df_demand_final.head(10)`

Out[98]:

	Series Key	ELEC.SALES.US-RES.M	ELEC.SALES.US-COM.M	ELEC.SALES.US-IND.M
0	200101	127065.78354	90825.27235	83477.36044
1	200102	99877.67304	81466.85778	82248.23193
2	200103	92804.58576	84064.24963	83521.51752
3	200104	82453.7826	81060.53414	82253.07658
4	200105	81731.15295	87470.74429	86096.5808
5	200106	99407.29496	95658.29115	85268.86123
6	200107	120707.42753	102156.27207	83175.56383
7	200108	129205.36178	105621.93345	86797.43767
8	200109	105943.19324	96704.45191	82631.49485
9	200110	85419.58912	89377.68751	83144.74635

In [99]: `df_demand_final.drop(["Series Key"], axis=1, inplace = True)
df_demand_final`

Out[99]:

	ELEC.SALES.US-RES.M	ELEC.SALES.US-COM.M	ELEC.SALES.US-IND.M
0	127065.78354	90825.27235	83477.36044
1	99877.67304	81466.85778	82248.23193
2	92804.58576	84064.24963	83521.51752
3	82453.7826	81060.53414	82253.07658
4	81731.15295	87470.74429	86096.5808
...	...	...	...
232	105411.88446	93405.71294	71219.94475
233	131242.46216	108695.0422	75210.98956
234	166890.82666	125994.21203	81276.25286
235	158821.41757	122024.58524	82582.31949
236	127583.66965	112213.76361	77504.78094

237 rows x 3 columns

In [100]: `df_demand_final.rename(columns={"ELEC.SALES.US-RES.M": "residential_demand", "ELEC.SALES.US-COM.M": "commercial_demand", "ELEC.SALES.US-IND.M": "industrial_demand"}, inplace=True)`

In [101]: `df_demand_final.index.name = "date_id"
df_demand_final`

Out[101]:

	residential_demand	commercial_demand	industrial_demand
date_id			
0	127065.78354	90825.27235	83477.36044
1	99877.67304	81466.85778	82248.23193
2	92804.58576	84064.24963	83521.51752
3	82453.7826	81060.53414	82253.07658
4	81731.15295	87470.74429	86096.5808
...	...	...	...
232	105411.88446	93405.71294	71219.94475
233	131242.46216	108695.0422	75210.98956
234	166890.82666	125994.21203	81276.25286
235	158821.41757	122024.58524	82582.31949
236	127583.66965	112213.76361	77504.78094

237 rows x 3 columns

## Load Data into Postgres

In [112]: `# Create database connection
connection_string = "postgres:postgres@localhost:5432/ETL_Project"
engine = create_engine(f'postgres://{connection_string}')`

In [113]: `# Confirm tables
engine.table_names()`

Out[113]: `['Date', 'Supply_ThousandKWH', 'Demand_MillionKWH', 'Price_CentsPerKWH']`

In [114]: `# Load DataFrames into database
df_price_final.to_sql(name='Price_CentsPerKWH', con=engine, if_exists='append', index=True)`

In [115]: `df_demand_final.to_sql(name='Demand_MillionKWH', con=engine, if_exists='append', index=True)`

In [ ]:

In [ ]: