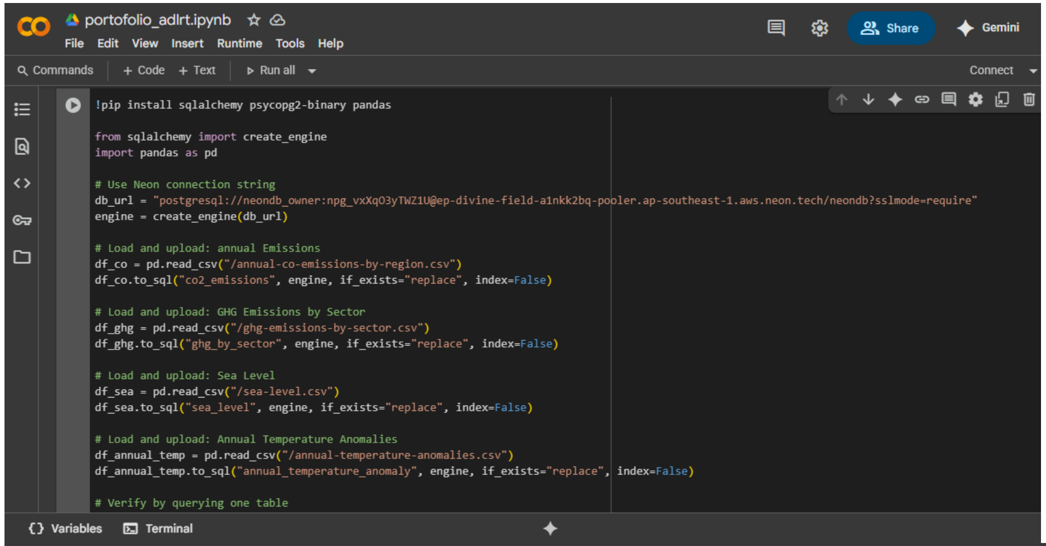
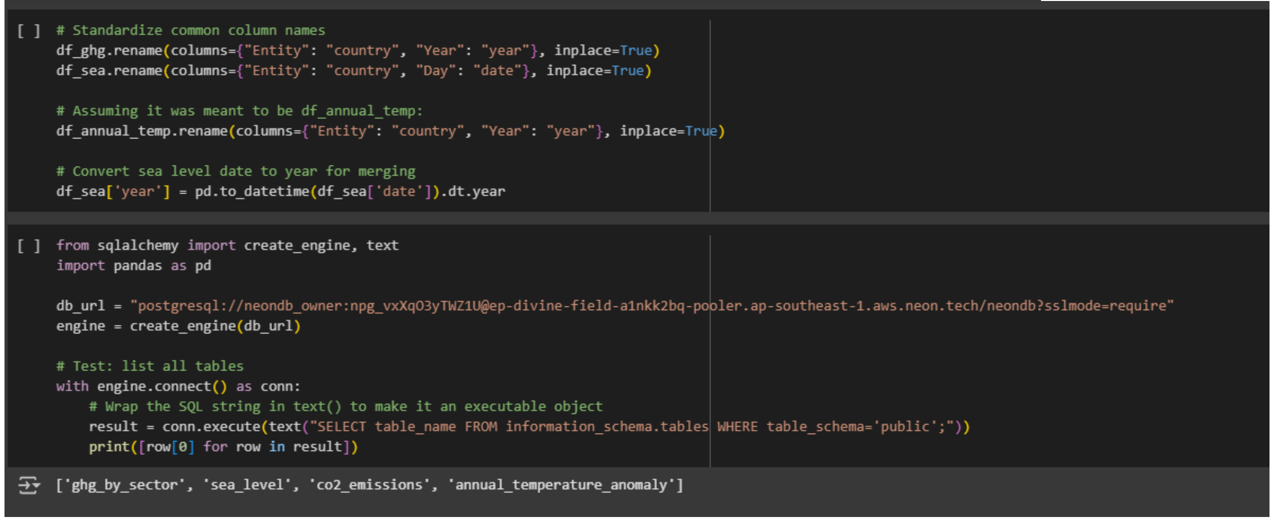
1. connect local database to cloud database using google colab (phyton), add data to database, clean data. – DTL & DDL


portofolio_adlrt.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text ▷ Run all
Share
Gemini
Connect
Variables
Terminal

```
!pip install sqlalchemy psycopg2-binary pandas

from sqlalchemy import create_engine
import pandas as pd

# Use Neon connection string
db_url = "postgresql://neondb_owner:npg_vxXqO3yTWZ1U@ep-divine-field-a1nkk2bq-pooler.ap-southeast-1.aws.neon.tech/neondb?sslmode=require"
engine = create_engine(db_url)

# Load and upload: annual Emissions
df_co = pd.read_csv("/annual-co-emissions-by-region.csv")
df_co.to_sql("co2_emissions", engine, if_exists="replace", index=False)

# Load and upload: GHG Emissions by Sector
df_ghg = pd.read_csv("/ghg-emissions-by-sector.csv")
df_ghg.to_sql("ghg_by_sector", engine, if_exists="replace", index=False)

# Load and upload: Sea Level
df_sea = pd.read_csv("/sea-level.csv")
df_sea.to_sql("sea_level", engine, if_exists="replace", index=False)

# Load and upload: Annual Temperature Anomalies
df_annual_temp = pd.read_csv("/annual-temperature-anomalies.csv")
df_annual_temp.to_sql("annual_temperature_anomaly", engine, if_exists="replace", index=False)

# Verify by querying one table
```

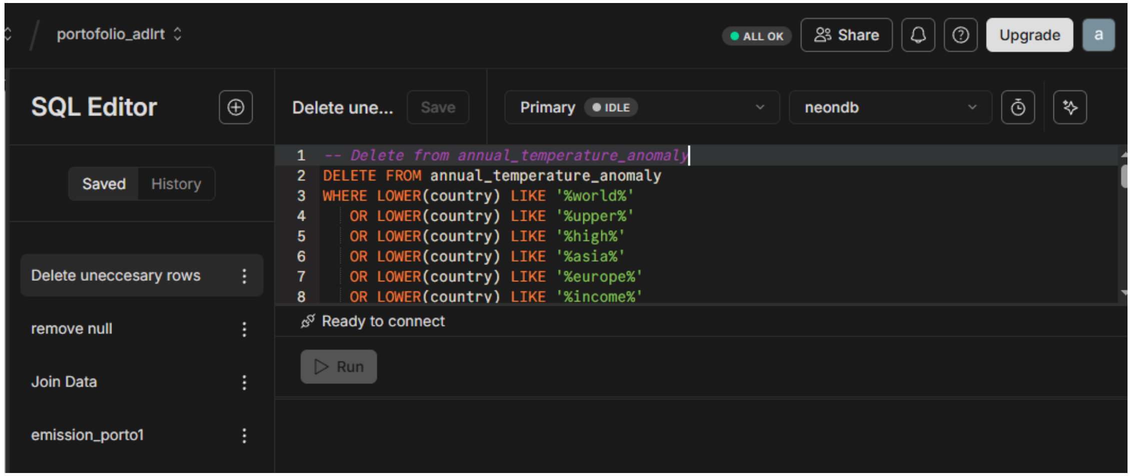


```
[ ] # Standardize common column names
    df_ghg.rename(columns={"Entity": "country", "Year": "year"}, inplace=True)
    df_sea.rename(columns={"Entity": "country", "Day": "date"}, inplace=True)

    # Assuming it was meant to be df_annual_temp:
    df_annual_temp.rename(columns={"Entity": "country", "Year": "year"}, inplace=True)

    # Convert sea level date to year for merging
    df_sea['year'] = pd.to_datetime(df_sea['date']).dt.year
```

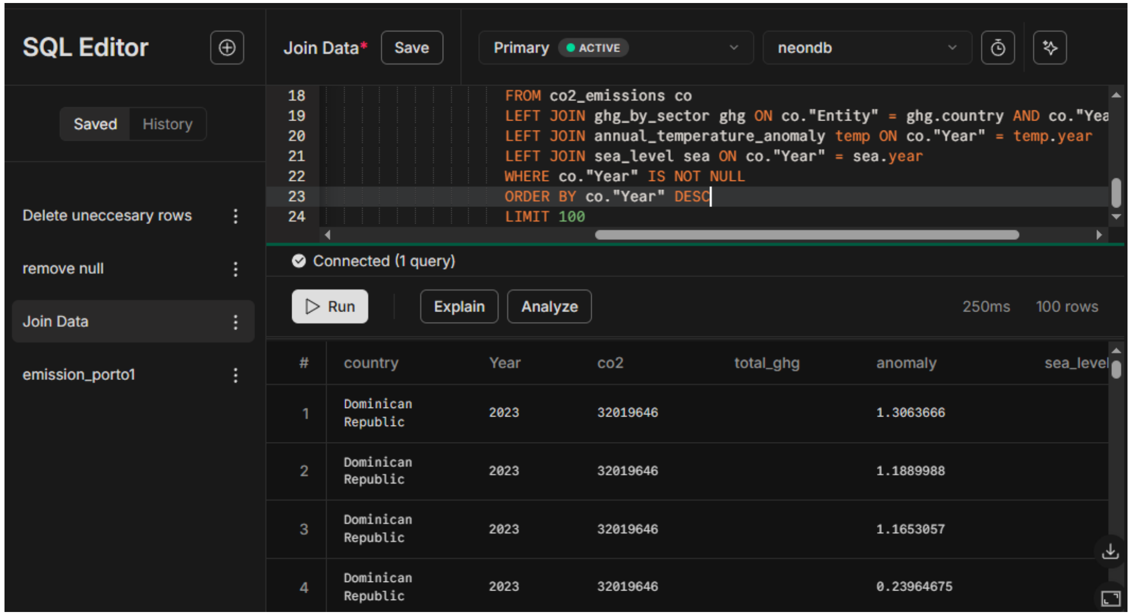
```
[ ] from sqlalchemy import create_engine, text
    import pandas as pd

    db_url = "postgresql://neondb_owner:npg_vxXqO3yTWZ1U@ep-divine-field-a1nkk2bq-pooler.ap-southeast-1.aws.neon.tech/neondb?sslmode=require"
    engine = create_engine(db_url)

    # Test: list all tables
    with engine.connect() as conn:
        # Wrap the SQL string in text() to make it an executable object
        result = conn.execute(text("SELECT table_name FROM information_schema.tables WHERE table_schema='public';"))
        print([row[0] for row in result])
```

```
['ghg_by_sector', 'sea_level', 'co2_emissions', 'annual_temperature_anomaly']
```

2. Data cleaning using sql in neondb – DML



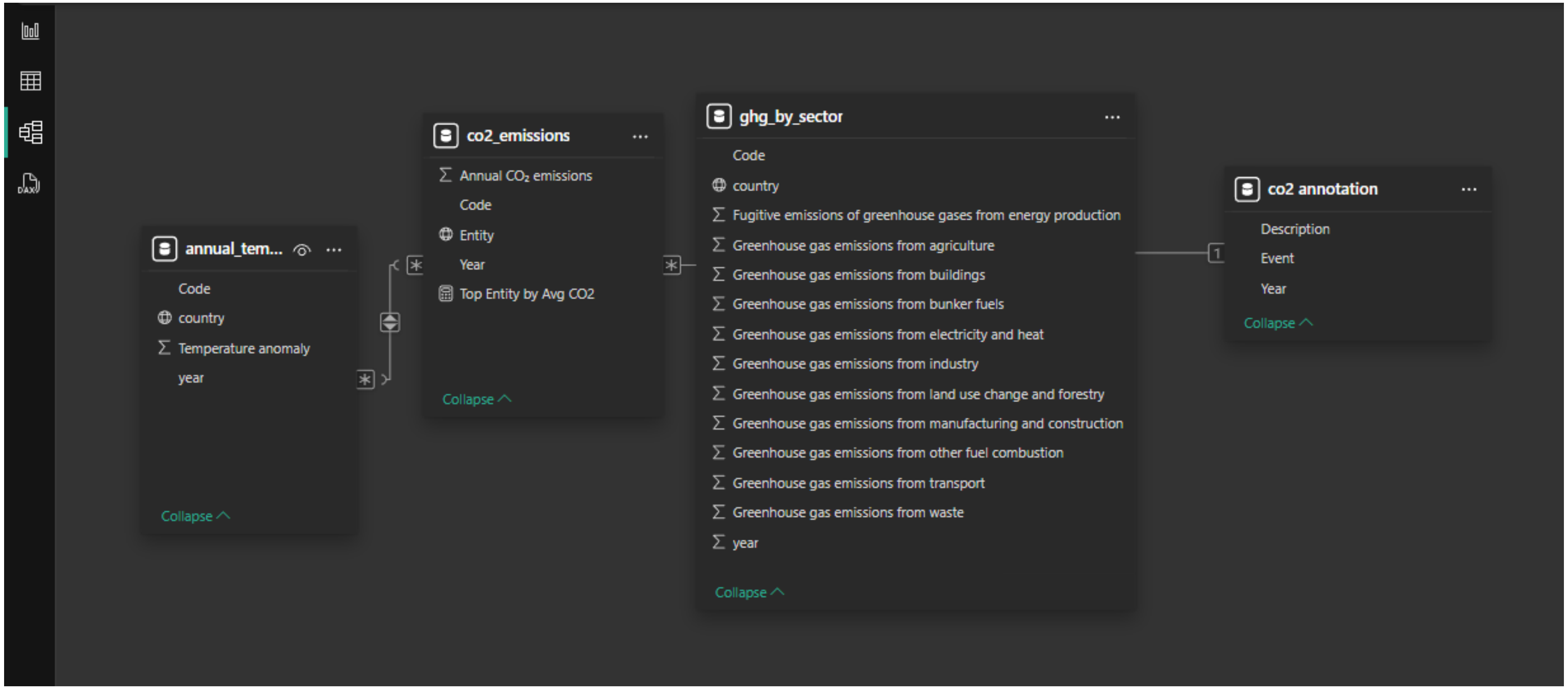
```sql
-- Delete from annual_temperature_anomaly
DELETE FROM annual_temperature_anomaly
WHERE LOWER(country) LIKE '%world%'
   OR LOWER(country) LIKE '%upper%'
   OR LOWER(country) LIKE '%high%'
   OR LOWER(country) LIKE '%asia%'
   OR LOWER(country) LIKE '%europe%'
   OR LOWER(country) LIKE '%income%'
```



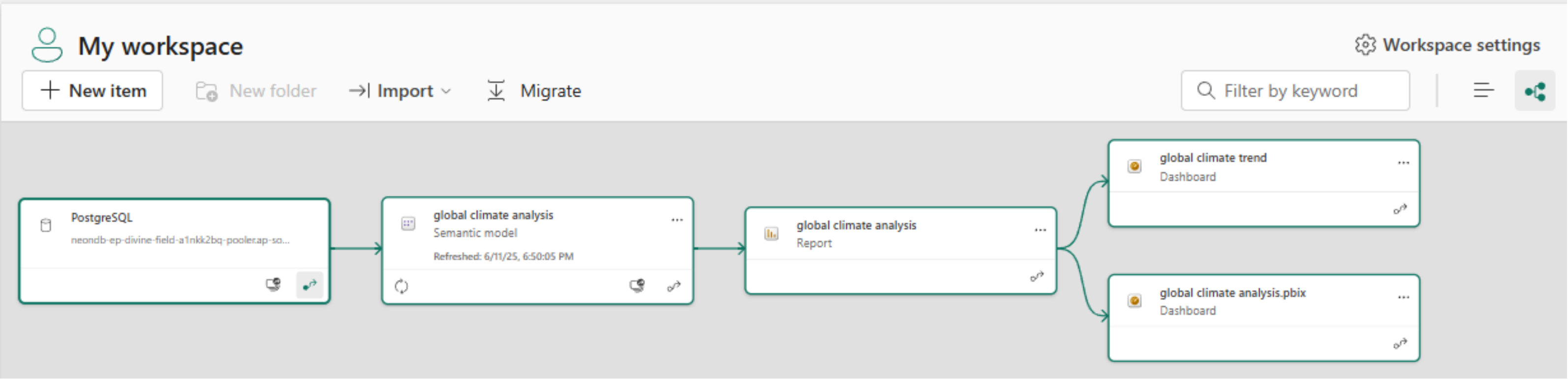
```sql
    FROM co2_emissions co
    LEFT JOIN ghg_by_sector ghg ON co."Entity" = ghg.country AND co."Yea
    LEFT JOIN annual_temperature_anomaly temp ON co."Year" = temp.year
    LEFT JOIN sea_level sea ON co."Year" = sea.year
    WHERE co."Year" IS NOT NULL
    ORDER BY co."Year" DESC
    LIMIT 100
```

#	country	Year	co2	total_ghg	anomaly	sea_level
1	Dominican Republic	2023	32019646		1.3063666	
2	Dominican Republic	2023	32019646		1.1889988	
3	Dominican Republic	2023	32019646		1.1653057	
4	Dominican Republic	2023	32019646		0.23964675	

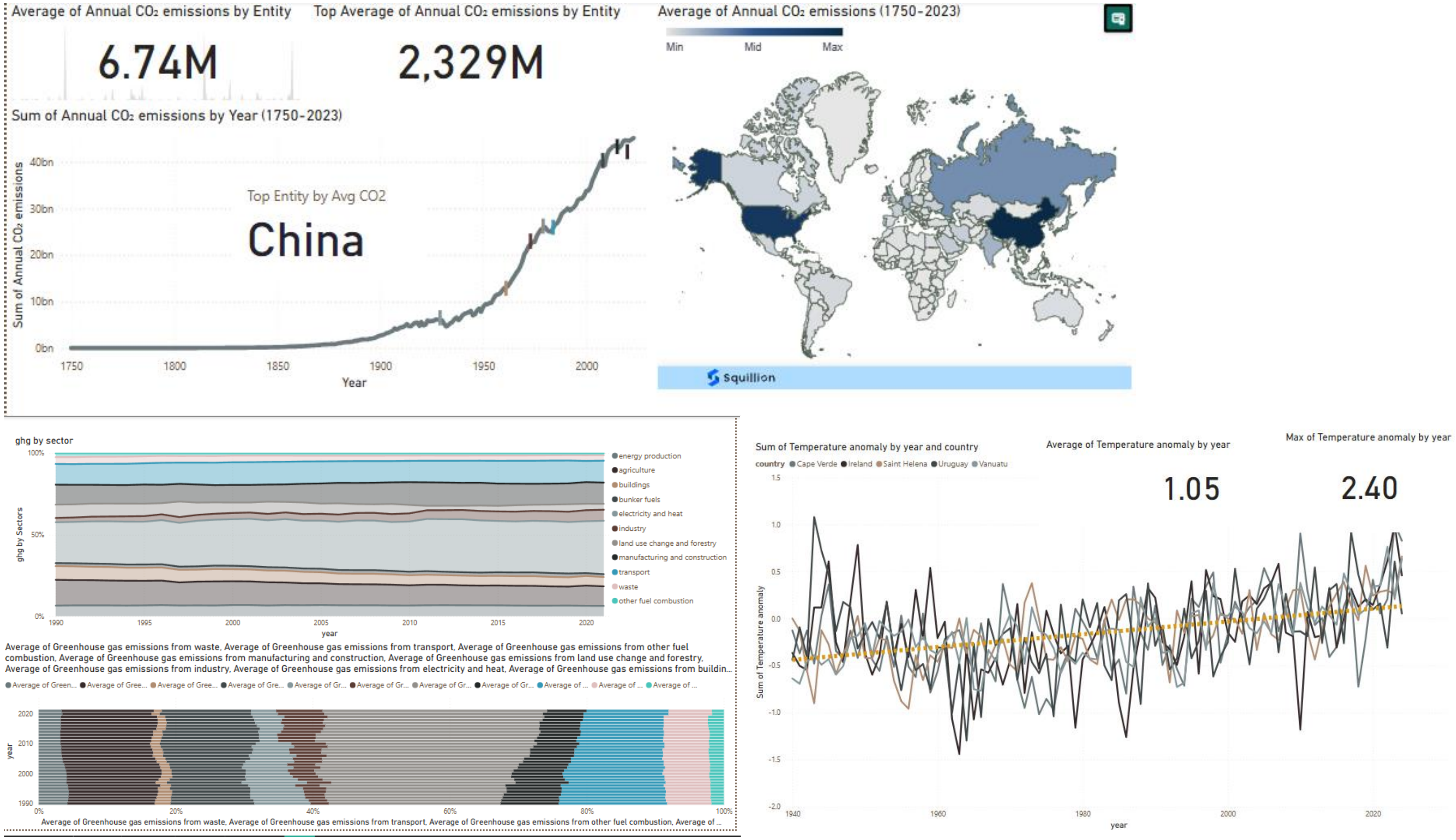
3. Table relationship


annual_tem...
Code
country
Temperature anomaly
year
Collapse
co2_emissions
Annual CO₂ emissions
Code
Entity
Year
Top Entity by Avg CO2
Collapse
ghg_by_sector
Code
country
Fugitive emissions of greenhouse gases from energy production
Greenhouse gas emissions from agriculture
Greenhouse gas emissions from buildings
Greenhouse gas emissions from bunker fuels
Greenhouse gas emissions from electricity and heat
Greenhouse gas emissions from industry
Greenhouse gas emissions from land use change and forestry
Greenhouse gas emissions from manufacturing and construction
Greenhouse gas emissions from other fuel combustion
Greenhouse gas emissions from transport
Greenhouse gas emissions from waste
year
Collapse
co2 annotation
Description
Event
Year
Collapse
*
*
*
1

4. Semantic data models



5. Dashboard Power BI



Average of Annual CO₂ emissions by Entity
6.74M
Top Average of Annual CO₂ emissions by Entity
2,329M
Average of Annual CO₂ emissions (1750-2023)
Min
Mid
Max
Squillion
Sum of Annual CO₂ emissions by Year (1750-2023)
Top Entity by Avg CO2
China
Sum of Annual CO₂ emissions
40bn
30bn
20bn
10bn
0bn
1750
1800
1850
1900
1950
2000
Year
ghg by sector
100%
50%
0%
ghg by Sectors
1990
1995
2000
2005
2010
2015
2020
year
energy production
agriculture
buildings
bunker fuels
electricity and heat
industry
land use change and forestry
manufacturing and construction
transport
waste
other fuel combustion
Average of Greenhouse gas emissions from waste, Average of Greenhouse gas emissions from transport, Average of Greenhouse gas emissions from other fuel combustion, Average of Greenhouse gas emissions from manufacturing and construction, Average of Greenhouse gas emissions from land use change and forestry, Average of Greenhouse gas emissions from industry, Average of Greenhouse gas emissions from electricity and heat, Average of Greenhouse gas emissions from buildin...
Average of Green... Average of Gree... Average of Gree... Average of Gre... Average of Gr... Average of Gr... Average of Gr... Average of Gr... Average of ... Average of ... Average of ...
2020
2010
2000
1990
year
0%
20%
40%
60%
80%
100%
Average of Greenhouse gas emissions from waste, Average of Greenhouse gas emissions from transport, Average of Greenhouse gas emissions from other fuel combustion, Average of ...
Sum of Temperature anomaly by year and country
country Cape Verde Ireland Saint Helena Uruguay Vanuatu
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
Sum of Temperature anomaly
1940
1960
1980
2000
2020
year
Average of Temperature anomaly by year
1.05
Max of Temperature anomaly by year
2.40