Global Air Network Socio-Economic Analysis

Philip Spaier and Emile Klerner

Github Repository: https://github.com/Penguinbeanie/database_assigment_airtraffic.git

Disclaimer: For an ideal viewing experience, open the root/README.html in your browser.

Project Overview

This project aims to analyze the structure of the **Global Air Transportation Network** by integrating it with **country-level economic indicators**. The goal is to move beyond simple network connectivity to explore complex analytical questions regarding how national wealth, stability, and air travel infrastructure correlate.

Setup and Data Ingestion

This project utilizes `compose.yml` to set up the necessary database environment. Data is then ingested into the database using the `ingestion.sql` script, which populates the tables with cleaned and processed data.

Data Sources

This project combines data from two main sources: one for aviation and one for country-level economics. After cleaning, the final datasets are stored in the `clean_data` folder, ready to be loaded into the database.

Source	Description	Files & Row Counts	URL
Global Air Transportation Network (Kaggle)	Core aviation data, valid for 2022.	`routes.csv` (67,661), `airports.csv` (7,697), `airlines.csv` (6,162), `airplanes.csv` (246)	`https://www.kaggle.com/datasets/thedevastatc air-transportation-network-mapping-the-wo/dat`
World Bank DataBank	Economic and social data, valid for 2023.	`country_gdp.csv` (265 rows), covering GDP, political stability, and population size.	`https://databank.worldbank.org/source/world-development-indicators`

Database Schema

The database will consist of six primary tables. The linkages are defined by aviation codes (IATA/ICAO) and geographical names.

Core Aviation Tables (4)

Table Name	Description	Key Columns
`airlines`	Carrier details and operational status.	`Airline_ID` (PK), `Name`, `Alias`, `IATA`, `ICAO`, `Callsign`, `Active`
`airplanes`	Aircraft model and identification codes.	`IATA` (PK), `Name`, `ICAO`
`airports`	Geographic and infrastructure data for every airport.	`Airport_ID` (PK), `Name`, `City`, `Country`, `IATA`, `ICAO`, `Latitude`, `Longitude`, `Altitude`, `Timezone`, `DST`, `Tz_database_time_zone`, `Type`, `Source` (FK to `countries.Country_Name`)
`routes`	Defined flight segments between two airports.	`Routes_ID` (PK), `Airline` (Name), `Airline_ID` (FK to `airlines.Airline_ID`), `Source_airport` (IATA/ICAO), `Source_airport_ID` (FK to `airports.Airport_ID`), `Destination_airport` (IATA/ICAO), `Destination_airport_ID` (FK to `airports.Airport_ID`), `Codeshare`, `Stops`, `Equipment` (FK to airplanes.IATA)

Enrichment Tables (1)

This table introduces the socio-economic context for analysis.

Table Name	Source	Key Columns	Linkage to `airports`
`countries`	World Bank	`Country_Name` (PK), `Time`, `Time_Code`, `Country_Code`, `GDP_current_US`, `GDP_per_capita_current_US`, `Political_Stability`, `Population`	`airports.Country` ↔ `countries.Country_Name`

Data Cleaning

To get the data ready for analysis, we performed the following cleaning steps:

1. Standardizing Values

- **Fixing Null Values**: The source files used different ways to show null values (like `\N` or `<null>`). We standardized these to be consistent.
- **Standardizing Booleans**: We changed fields like `Codeshare` to use a simple `1` (for true) or `0` (for false) instead of text like 'Y'.
- **Cleaning Up Text**: We removed extra spaces and empty strings from text fields.

2. Ensuring Data Integrity

- **Handling Primary Keys**: We made sure every row in a table had a primary key. If a row was missing its key, we removed it.
- **Removing Duplicates**: We deleted any rows that were exact duplicates of each other.
- **Removing Conflicting Rows**: In some cases, the same primary key was used for different rows. We removed these conflicting entries, as they likely represented outdated data.
- **Removing Incomplete Rows**: We got rid of rows that were mostly empty because they weren't useful for analysis.

3. Combining and Validating Datasets

- **Matching Country Names**: The country names in the aviation and economic datasets didn't always match (e.g., "Bahamas" vs "Bahamas, The").
  For example, some country name mismatches we found:

World Bank Name	Kaggle Name
"Bahamas, The"	"Bahamas"
"North Macedonia"	"Macedonia"
"Myanmar"	"Burma"

  - We used a fuzzy-matching script (`rapidfuzz`) to automatically match names that were over 90% similar.
  - We manually fixed the rest of the names that didn't get matched.
  - Finally, we only kept economic data for countries that were present in our airport data.
- **Validating Routes**: We checked that every flight route correctly linked to an existing airline and airport. If a route had an invalid ID, we removed it.

Documentation of Question Design, Adjustments, and Results

Database Connection Setup

All analytical SQL queries were executed from a Jupyter Notebook, using a PostgreSQL database running inside Docker. After starting the environment via:

```
docker compose up -d
```

the PostgreSQL server becomes available at:

- **Host:** `localhost`
- **Port:** `5432`
- **User:** `postgres`
- **Password:** `postgres`
- **Database:** `postgres`

We connect to this database using SQLAlchemy:

```
from sqlalchemy import create_engine
import pandas as pd

engine = create_engine("postgresql://postgres:postgres@localhost:5432/postgres")
```

All queries are executed using:

```
pd.read_sql(query, engine)
```

Question 1 (Basic RA: Select + Projection)

Which airlines are inactive? Provide ID and name.

```
query_question01 = """
SELECT
    "Airline_ID",
    "Name"
FROM airlines
WHERE "Active" = 'N';
"""
q1_df = pd.read_sql(query_question01, engine)
q1_df
```

Query Result:

	Airline_ID	Name
0	2	135 Airways
1	4	2 Sqn No 1 Elementary Flying Training School
2	5	213 Flight Unit
3	6	223 Flight Unit State Airline

	Airline_ID	Name
4	7	224th Flight Unit
...	...	...
4901	20963	Atlantic Air Cargo
4902	21056	Dummy
4903	21181	Air Andaman (2Y)
4904	21240	TDA Toa Domestic Airlines
4905	21251	Lynx Aviation (L3/SSX)

4906 rows × 2 columns

Relational Algebra Question 01

$$\pi_{Airline\_ID,\ Name}\left(\sigma_{Active='N'}(airlines)\right)$$

Question 2 (Basic RA: Union)

Show all airprts located in either Germany or Austria. Include name, city, and country.

```
query_question02 = """
SELECT
    "Name",
    "City",
    "Country"
FROM airports
WHERE "Country" = 'Germany'
UNION
SELECT
    "Name",
    "City",
    "Country"
FROM airports
WHERE "Country" = 'Austria'
"""

q2_df = pd.read_sql(query_question02, engine)
q2_df
```

Query Result:

	Name	City	Country
0	Hamburg Airport	Hamburg	Germany
1	Geilenkirchen Air Base	Geilenkirchen	Germany
2	Vilshofen Airport	Vilshofen	Germany
3	St. Johann In Tirol Airport	St. Johann in Tirol	Austria
4	Wipperfürth-Neye Airport	Wipperfuerth	Germany
...	...	...	...
264	Hohn Air Base	Hohn	Germany
265	Rügen Airport	Ruegen	Germany
266	Torgau-Beilrode Airport	Gransee	Germany
267	Aalen-Heidenheim/Elchingen Airport	Aalen-heidenheim	Germany
268	Rothenburg/Görlitz Airport	Rothenburg/ol	Germany

269 rows × 3 columns

Relational Algebra Question 02

$$\pi_{Name, City, Country}(\sigma_{Country = \text{"Germany"}}(airports))$$
$$\cup$$
$$\pi_{Name, City, Country}(\sigma_{Country = \text{"Austria"}}(airports))$$

Question 3 (Extended RA: Join)

Show all routes with one stop with the name of the airline, the departure and arrival airports (IATA codes).

```
query_question03 = """
SELECT
    a."Name",
    r."Source_airport",
    r."Destination_airport"
FROM routes r
JOIN airlines a ON r."Airline_ID" = a."Airline_ID"
```

```
WHERE r."Stops" = 1
"""

q3_df = pd.read_sql(query_question03, engine)
q3_df
```

Query Result:

	Name	Source_airport	Destination_airport
0	Canadian North	YRT	YEK
1	Air Canada	ABJ	BRU
2	Air Canada	YVR	YBL
3	Cubana de Aviación	FCO	HAV
4	AirTran Airways	HOU	SAT
5	AirTran Airways	MCO	ORF
6	Scandinavian Airlines System	ARN	GEV

Relational Algebra Question 03

$$\pi_{a.Name,\, r.Source\_airport,\, r.Destination\_airport}\Big(\rho_r\big(\sigma_{Stops=1}(routes)\big) \bowtie_{r.Airline\_ID = a.Airline\_ID} \rho_a(airlines)\Big)$$

$$\pi_{airlines.Name,\, routes.Source\_airport,\, routes.Destination\_airport}\Big(\big(\sigma_{Stop=1}(routes)\big) \bowtie_{routes.Airline\_ID = airlines.Airlines\_ID} airlines\Big)$$

Question 4 (Extended RA: Aggregate Functions)

How many routes does each airline have?

```
query_question04 = """
SELECT
    a."Name",
    COUNT(*) AS Route_Count
FROM routes r
JOIN airlines a ON r."Airline_ID" = a."Airline_ID"
GROUP BY a."Name"
HAVING COUNT(*) > 80;
"""
q4_df = pd.read_sql(query_question04, engine)
q4_df
```

Query Result:

	Name	route_count
0	Air Bourbon	210
1	TransAsia Airways	92
2	Air India Limited	364
3	Meridiana	140
4	EVA Air	114
...	...	...
136	LOT Polish Airlines	114
137	Sriwijaya Air	106
138	LAN Airlines	285
139	Iberia Airlines	797
140	Philippine Airlines	144

141 rows × 2 columns

$$\pi_{Name, Route\_Count}(\sigma\ Route\_count > 80(\gamma\ Name; COUNT(*) \to Route\_Count(routes \bowtie_{routes.Airline\_ID = airlines.Airline\_ID} airlines)))$$

Question 5 (Extended RA: Aggregate Functions)

Which countries have the highest percentage of domestic routes?

```
query_question05 = """
WITH route_with_countries AS (
    SELECT
        r."Routes_ID",
        src_c."Country_Name"  AS src_country,
        dest_c."Country_Name" AS dest_country
    FROM routes r
    JOIN airports src_a
        ON r."Source_airport" = src_a."IATA"
    JOIN countries src_c
        ON src_a."Country" = src_c."Country_Name"
    JOIN airports dest_a
        ON r."Destination_airport" = dest_a."IATA"
    JOIN countries dest_c
        ON dest_a."Country" = dest_c."Country_Name"
),
route_stats AS (
    SELECT
        src_country AS country,
        COUNT(*) AS total_routes,
        COUNT(*) FILTER (WHERE src_country = dest_country) AS domestic_routes
    FROM route_with_countries
    GROUP BY src_country
)
SELECT
    country,
    total_routes,
    domestic_routes,
    domestic_routes::FLOAT / total_routes AS domestic_share
FROM route_stats
WHERE total_routes >= 100
ORDER BY domestic_share DESC
LIMIT 15;
"""

q5_df = pd.read_sql(query_question05, engine)
q5_df
```

Query Result:

	country	total_routes	domestic_routes	domestic_share
0	China	7894	6743	0.854193
1	Brazil	1325	1129	0.852075
2	French Polynesia	108	91	0.842593
3	Indonesia	817	603	0.738066
4	Iran, Islamic Rep.	356	257	0.721910
5	United States	7508	5382	0.716835
6	India	1145	773	0.675109

	country	total_routes	domestic_routes	domestic_share
7	Australia	836	557	0.666268
8	Argentina	253	166	0.656126
9	Colombia	337	219	0.649852
10	Chile	155	97	0.625806
11	Philippines	393	240	0.610687
12	New Zealand	188	114	0.606383
13	Venezuela, RB	145	81	0.558621
14	Norway	410	222	0.541463

$$\text{SrcC} \leftarrow \rho_{\text{src\_c}}(\text{countries})$$

$$\text{DestC} \leftarrow \rho_{\text{dest\_c}}(\text{countries})$$

$$\text{SrcA} \leftarrow \rho_{\text{src\_a}}(\text{airports})$$

$$\text{DestA} \leftarrow \rho_{\text{dest\_a}}(\text{airports})$$

$$\text{routes\_with\_countries} \leftarrow \pi_{\text{RoutesID, src\_c.CountryName} \to \text{src\_country, dest\_c.CountryName} \to \text{dest\_country}}$$
$$\Big(\Big(\Big(\big(\text{routes} \bowtie_{\text{Source\_airport} = \text{src\_a.IATA}} \text{SrcA}\big)$$
$$\bowtie_{\text{src\_a.Country} = \text{src\_c.Country}} \text{SrcC}\Big)$$
$$\bowtie_{\text{Destination\_airport} = \text{dest\_a.IATA}} \text{DestA}\Big)$$
$$\bowtie_{\text{dest\_a.Country} = \text{dest\_c.Country\_Name}} \text{DestC}\Big)$$

$$\text{route\_stats} \leftarrow \gamma_{\text{src\_country} \to \text{country};}$$
$$\text{total\_routes} := \text{COUNT}(*),$$
$$\text{domestic\_routes} := \text{SUM}(I(\text{src\_country} = \text{dest\_country}))\ (\text{routes\_with\_countries})$$

$$\text{result} \leftarrow \pi_{\text{country, total\_routes, domestic\_routes, domestic\_share} := \text{domestic\_routes}/\text{total\_routes}}$$
$$(\sigma_{\text{total\_routes} >= 100}(\text{route\_stats}))$$

Question 6 (Extended RA: Aggregate Functions)

Which aircraft types (Equipment codes) appear on the largest number of distinct routes

```
query_question06 = """
WITH expanded_equipment AS (
    SELECT
        r."Routes_ID",
```

```
        trim(equip) AS equipment
    FROM routes r,
         regexp_split_to_table(r."Equipment", ' ') AS equip
    WHERE trim(equip) <> ''
)

SELECT
    equipment,
    COUNT(DISTINCT "Routes_ID") AS route_count
FROM expanded_equipment
GROUP BY equipment
ORDER BY route_count DESC
LIMIT 10;
"""
q6_df = pd.read_sql(query_question06, engine)
q6_df
```

Query Result:

	equipment	route_count
0	320	14844
1	738	9745
2	319	7338
3	321	3379
4	737	2628
5	E90	1816
6	AT7	1529
7	333	1517
8	73G	1406
9	332	1386

$$EO \leftarrow \mu_{equip \in SPLIT\_SPACE(routes.Equipment)} \left(\pi_{Routes\_ID,\ Equipment}(routes)\right)$$

$$expanded\_experiment \leftarrow \rho_{equipment := TRIM(equip)} \left(\sigma_{TRIM(equip) \neq ''}(EO)\right)$$

$$U \leftarrow \delta\left(\pi_{equipment,\ Routes\_ID}(expanded\_equipment)\right)$$

$$R \leftarrow \gamma_{equipment;\ routes\_count := COUNT(*)} (U)$$

Question 7 (Extended RA: Aggregate Functions)

What is the average number of unique destination countries reachable from each country

```
query_question7 = """
SELECT
    origin_c."Country_Name" AS origin_country,
    COUNT(DISTINCT dest_c."Country_Name") AS reachable_countries
FROM routes r
JOIN airports origin_a ON r."Source_airport" = origin_a."IATA"
JOIN countries origin_c ON origin_a."Country" = origin_c."Country_Name"
JOIN airports dest_a ON r."Destination_airport" = dest_a."IATA"
JOIN countries dest_c ON dest_a."Country" = dest_c."Country_Name"
GROUP BY origin_c."Country_Name"
ORDER BY reachable_countries DESC;
"""
q7_df = pd.read_sql(query_question7, engine)
q7_df
```

Query Result:

	origin_country	reachable_countries
0	France	112

	origin_country	reachable_countries
1	United Kingdom	98
2	Germany	96
3	United States	90
4	Turkiye	90
...	...	...
217	Falkland Islands	1
218	Tuvalu	1
219	Saint Pierre and Miquelon	1
220	Niue	1
221	Samoa	1

222 rows × 2 columns

$$\text{OriginA} \leftarrow \rho_{\text{origin\_a}}(\text{airports})$$
$$\text{OriginC} \leftarrow \rho_{\text{origin\_c}}(\text{countries})$$
$$\text{DestA} \leftarrow \rho_{\text{dest\_a}}(\text{airports})$$
$$\text{DestC} \leftarrow \rho_{\text{dest\_c}}(\text{countries})$$

$$P \leftarrow \pi_{\text{origin\_c.Country\_Name} \rightarrow \text{origin\_country},\ \text{dest\_c.Country\_Name} \rightarrow \text{dest\_country}} \Big(\big(\big((\text{routes} \bowtie_{\text{Source\_airport} = \text{origin\_a.IATA}} \text{OrgA}) \bowtie_{\text{origin\_a.Country} = \text{origin\_c.Country\_Name}} \text{OrgC}\big) \bowtie_{\text{Destination\_airport} = \text{dest\_a.IATA}} \text{DestA}\big) \bowtie_{\text{dest\_a.Country} = \text{dest\_c.Country Name}} \text{DestC}\Big)$$

$$U \leftarrow \delta(P) \quad // \text{DISTINCT (origin\_country, dest\_country)}$$

$$R \leftarrow \gamma_{\text{origin\_c};\ \text{reachable\_countries} := \text{COUNT}(*)}(U)$$

Question 8 (Extended RA: Aggregate Functions)

What are the top 5 countries where the ratio of total outgoing routes (from all airports in the country) to the country's GDP per Capita is the highest?

```
query_question08 = """
WITH outgoing_per_country AS (
    SELECT
        a."Country",
        COUNT(*) AS outgoing_routes
    FROM routes r
    JOIN airports a
        ON r."Source_airport_ID" = a."Airport_ID"
    GROUP BY a."Country"
),
country_with_gdp AS (
    SELECT
        c."Country_Name",
        c."GDP_per_capita_current_US"
    FROM countries c
    WHERE c."GDP_per_capita_current_US" IS NOT NULL
      AND c."GDP_per_capita_current_US" > 0
)
SELECT
    og."Country",
    og.outgoing_routes,
    cg."GDP_per_capita_current_US" AS gdp_per_capita,
    (og.outgoing_routes / cg."GDP_per_capita_current_US") AS ratio
FROM outgoing_per_country og
JOIN country_with_gdp cg
    ON og."Country" = cg."Country_Name"
ORDER BY ratio DESC
LIMIT 5;
"""

q8_df = pd.read_sql(query_question08, engine)
q8_df
```

Query Result:

	Country	outgoing_routes	gdp_per_capita	ratio
0	China	8013	12951.178240	0.618708
1	India	1145	2530.120313	0.452548
2	Pakistan	249	1365.169274	0.182395
3	Indonesia	817	4876.307745	0.167545
4	Madagascar	65	508.718428	0.127772

$$
\begin{aligned}
\text{Outgoing\_per\_country} \leftarrow\ & \gamma_{\substack{a.\text{Country};\\ \text{outgoing\_routes} := \text{COUNT}(*)}} \\
& (\text{routes } r \bowtie_{r.\text{Source\_airport\_ID} = a.\text{Airport\_ID}} \text{airports } a)
\end{aligned}
$$

$$
\begin{aligned}
\text{country\_with\_gpg} \leftarrow\ & \pi_{c.\text{CountryName},\ c.\text{GDP\_per\_capita\_current\_us}} \\
& (\sigma_{\substack{c.\text{GDP\_per\_capita\_current\_us} \neq \text{NULL} \\ \land \\ c.\text{GDP\_per\_capita\_current\_us} > 0}} (\text{countries } c))
\end{aligned}
$$

$$
\begin{aligned}
\text{Result} \leftarrow\ & \pi_{\substack{\text{Country} := \text{Outgoing\_per\_Country.Country}, \\ \text{outgoing\_routes}, \\ \text{gdp\_per\_capita} := \text{country\_with\_gpg.GDP\_per\_capita\_current\_us}, \\ \text{ratio} := \text{outgoing\_routes} / \text{country\_with\_gpg.GDP\_per\_capita\_current\_us},}} \\
& (\text{Outgoing\_per\_Country} \bowtie_{\text{Outgoing\_per\_Country.Country} = \text{country\_with\_gpg.GDP\_per\_capita\_current\_us}} \text{country\_with\_gpg})
\end{aligned}
$$

Question 9 (Extended RA: Aggregate Functions)

Which airports have the largest disparity between the number of outgoing and incoming routes?

```
query_question9 = """
WITH outgoing AS (
    SELECT
        r."Source_airport_ID" AS airport_id,
        COUNT(*) AS outgoing_count
    FROM routes r
    GROUP BY r."Source_airport_ID"
),
incoming AS (
    SELECT
        r."Destination_airport_ID" AS airport_id,
        COUNT(*) AS incoming_count
    FROM routes r
    GROUP BY r."Destination_airport_ID"
),
combined AS (
    SELECT
        a."Airport_ID",
        a."Name",
        a."City",
        a."Country",
        COALESCE(o.outgoing_count, 0) AS outgoing_count,
        COALESCE(i.incoming_count, 0) AS incoming_count,
        ABS(COALESCE(o.outgoing_count, 0) - COALESCE(i.incoming_count, 0)) AS disparity
    FROM airports a
    LEFT JOIN outgoing o ON a."Airport_ID" = o.airport_id
    LEFT JOIN incoming i ON a."Airport_ID" = i.airport_id
)
SELECT *
FROM combined
ORDER BY disparity DESC
LIMIT 10;
"""
q9_df = pd.read_sql(query_question9, engine)
```

```
q9_df
```

Query Result:

	Airport_ID	Name	City	Country	outgoing_count	incoming_count	dispar
0	2006	Auckland International Airport	Auckland	New Zealand	79	99	20
1	2074	Prince Mohammad Bin Abdulaziz Airport	Madinah	Saudi Arabia	34	50	16
2	73	Halifax / Stanfield International Airport	Halifax	Canada	19	30	11
3	4059	Jomo Kenyatta International Airport	Nairobi	Kenya	97	107	10
4	3941	Eleftherios Venizelos International Airport	Athens	Greece	166	176	10
5	1638	Humberto Delgado Airport (Lisbon Portela Airport)	Lisbon	Portugal	194	202	8
6	3862	Portland International Jetport Airport	Portland	United States	1	9	8
7	346	Munich Airport	Munich	Germany	326	318	8
8	1382	Charles de Gaulle	Paris	France	476	469	7

	Airport_ID	Name	City	Country	outgoing_count	incoming_count	dispar
		International Airport					
9	3861	Louis Armstrong New Orleans International Airport	New Orleans	United States	70	77	7

$$\text{outgoing} \leftarrow \gamma_{\text{outgoing\_count} := \text{COUNT}(*),\ \text{Source\_airport} \rightarrow \text{airport\_id}}(\text{routes})$$

$$\text{incoming} \leftarrow \gamma_{\text{Destination\_airport} \rightarrow \text{airport\_id},\ \text{incoming\_count} := \text{COUNT}(*)}(\text{routes})$$

$$AO \leftarrow \text{airports} \bowtie_{\text{Airport\_ID} = \text{airport\_id}} \text{outgoing}$$

$$AOI \leftarrow AO \bowtie_{\text{Airport\_ID} = \text{airport\_id}} \text{incoming}$$

$$\text{combined} \leftarrow \pi_{\text{Airport\_ID}, \text{City}, \text{Country}, \text{Name}, \text{outgoing\_count} := \text{coalesce}(\text{outgoing\_count}, 0), \text{incoming\_count} := \text{coalesce}(\text{incoming\_count}, 0), \text{disparity} := \text{abs}(\text{coalesce}(\text{outgoing\_count}, 0) - \text{coalesce}(\text{incoming\_count}, 0))}(AOI)$$

Question 10 (Extended RA: Aggregate Functions)

What are the top 10 cities globally, based on the total number of airports?

```
query_question10 = """
SELECT
    a."City",
    a."Country",
    COUNT(*) AS airport_count,
    STRING_AGG(a."Name", ', ' ORDER BY a."Name") AS airport_names
FROM airports a
WHERE a."City" IS NOT NULL
  AND a."City" <> ''
GROUP BY a."City", a."Country"
```

```
ORDER BY airport_count DESC
LIMIT 10;
"""

q10_df = pd.read_sql(query_question10, engine)
q10_df
```

Query Result:

	City	Country	airport_count	airport_names
0	Columbus	United States	8	Bolton Field, Columbus Metropolitan Airport, C...
1	New York	United States	6	Downtown-Manhattan/Wall St Heliport, Indianola...
2	Moscow	Russian Federation	6	Bykovo Airport, Domodedovo International Airpo...
3	London	United Kingdom	6	London City Airport, London Gatwick Airport, L...
4	Jacksonville	United States	6	Cecil Airport, Jacksonville Executive at Craig...
5	Houston	United States	6	Andrau Airpark, David Wayne Hooks Memorial Air...
6	Atlanta	United States	5	Cobb County-Mc Collum Field, DeKalb Peachtree ...
7	Greenville	United States	5	Donaldson Field Airport, Greenville Downtown A...
8	Izmir	Turkiye	5	Adnan Menderes International Airport, Çiğli Ai...
9	Vancouver	Canada	5	Coal Harbour Seaplane Base, Harbour (Public) H...

$$A \leftarrow \sigma_{City \neq NULL \wedge City \neq ''}(airports)$$

$$G \leftarrow \gamma_{City, Country;\ airport\_count := COUNT(*)}(A)$$

$$R \leftarrow \gamma_{City, Country;\ airport\_count := COUNT(*),\ airport\_names := STRING\_AGG\_SORTED(Name, ', ')}(A)$$