# **🌍 Air Pollution & Health Analytics Using SQLite and Python**

**Google Colab Link:** [Open Project Notebook](https://colab.research.google.com/drive/11PSL0qJv6Snq4LiU4qAVSfvCpa89Z5pl?usp=sharing)

## **🔎 Project Overview**

This project explores the environmental and public health impact of air pollution across urban regions by integrating structured data processing with insightful visualizations. Using SQLite and Python in a Google Colab environment, the project transforms raw CSV data into a normalized relational database, performs in-depth analysis through SQL, and visualizes the most polluted cities and pollution trends over time.

## **🎯 Objectives**

* Build a clean and scalable **SQL database** from air quality datasets.
* Conduct **data-driven analysis** on air pollution trends and city-wise pollution levels.
* Visualize **top polluted cities** and year-wise patterns using Matplotlib.
* Demonstrate data pipeline and analytics skills in a **real-world environmental context**.

## **🛠️ Tech Stack**

* **Languages & Tools**: Python, SQL (SQLite), Google Colab
* **Libraries Used**: Pandas, Matplotlib, PrettyTable, ipython-sql
* **Database**: SQLite (via ipython-sql in Colab)

## **🧱 Database Schema Design**

* **cities**: Stores unique city names with auto-incremented IDs.
* **air\_quality**: Contains daily pollution data (CO, NO₂, PM10) along with temperature and humidity, linked to cities by city\_id.
* **health\_indicators** *(optional/extendable)*: Includes year-wise health metrics such as mortality rate and life expectancy.

CREATE TABLE IF NOT EXISTS cities (

city\_id INTEGER PRIMARY KEY AUTOINCREMENT,

city\_name TEXT NOT NULL

);

CREATE TABLE IF NOT EXISTS air\_quality (

id INTEGER PRIMARY KEY AUTOINCREMENT,

city\_id INTEGER,

date TEXT,

co REAL,

no2 REAL,

pm10 REAL,

temperature REAL,

humidity REAL,

FOREIGN KEY(city\_id) REFERENCES cities(city\_id)

);

CREATE TABLE IF NOT EXISTS health\_indicators (

id INTEGER PRIMARY KEY AUTOINCREMENT,

city\_id INTEGER,

year INTEGER,

mortality\_rate REAL,

life\_expectancy REAL,

FOREIGN KEY(city\_id) REFERENCES cities(city\_id)

);

## **🔄 ETL Workflow (Extract, Transform, Load)**

1. **Data Upload & Initial Inspection**
   * Imported a CSV file (AirQuality 2.csv) with pollutant and weather data.
   * Explored structure and identified column inconsistencies and encoding issues.
2. **Data Cleaning & Transformation**
   * Standardized column names: CO(GT) → co, PM10 (GT) → pm10, etc.
   * Added a consistent city\_name column for relational mapping.
   * Handled delimiter issues (;), null values, and unnamed columns.
3. **Database Population**
   * Created SQLite database (air\_quality\_health.db) in Colab.
   * Inserted unique cities and linked pollution data using city\_id.
   * Ensured schema integrity with foreign keys and primary key constraints.

## **📈 Data Analysis Using SQL**

### **🔝 Top 10 Most Polluted Cities by PM10**

SELECT c.city\_name, AVG(a.pm10) AS avg\_pm10

FROM air\_quality a

JOIN cities c ON a.city\_id = c.city\_id

GROUP BY c.city\_name

ORDER BY avg\_pm10 DESC

LIMIT 10;

* **Insight**: Highlights the cities with the highest concentration of PM10 particles on average.

### **📆 Yearly Trend in PM10 for Delhi**

SELECT SUBSTR(date, 1, 4) AS year, AVG(pm10) AS avg\_pm10

FROM air\_quality a

JOIN cities c ON a.city\_id = c.city\_id

WHERE c.city\_name = 'Delhi'

GROUP BY year

ORDER BY year;

* **Insight**: Observes PM10 fluctuations over time for a major metropolitan area.

### **🌡️ Overall Environmental Summary**

SELECT ROUND(AVG(temperature), 2) AS avg\_temp,

ROUND(AVG(humidity), 2) AS avg\_humidity,

ROUND(AVG(pm10), 2) AS avg\_pm10

FROM air\_quality;

* **Insight**: Provides a snapshot of environmental averages across all records.

## **📊 Data Visualization**

* **Top Polluted Cities (Bar Chart):**
  + Created using Matplotlib, this chart visually represents cities ranked by average PM10 concentration.
  + Enhanced with axis labels, rotated city names, and responsive layout.

plt.bar(top\_cities\_df['city\_name'], top\_cities\_df['avg\_pm10'], color='tomato')

plt.title('Top 10 Most Polluted Cities by PM10')

## **📌 Key Skills Demonstrated**

| **Area** | **Skills** |
| --- | --- |
| **Data Handling** | Data cleaning, transformation, Pandas |
| **SQL Mastery** | Joins, aggregation, filtering, subqueries |
| **Database Design** | Schema normalization, primary/foreign keys |
| **Visualization** | Matplotlib plots, trend analysis |
| **Integration** | SQL and Python in a unified Colab pipeline |
| **Real-World Relevance** | Pollution monitoring and public health |

## **📁 Dataset Details**

* **File Used**: AirQuality 2.csv (user-uploaded)
* **Size**: < 3MB (optimized for sharing with recruiters)
* **Attributes**: Date, CO, NO₂, PM10, Temperature, Humidity, City

## **🚀 Recruiter Value Proposition**

This project reflects my ability to:

* Design and manage relational databases from scratch
* Tackle real-world data challenges with SQL and Python
* Deliver actionable insights with visual storytelling
* Work in collaborative, cloud-native environments like Google Colab