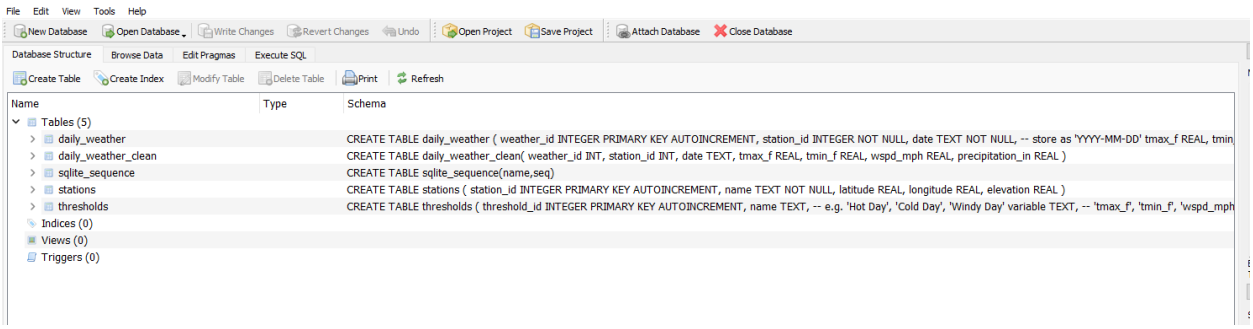
About the Data & Description

This project is part of a **practice data analytics series** designed to help strengthen SQL and data storytelling skills with real datasets. The focus of this project is **Extreme Weather Data Analysis**, where the goal is to practice querying a weather database, extracting meaningful insights.

The dataset is provided as a SQLite database (weather.db), containing three key tables:

- **daily_weather** – daily weather observations, including maximum/minimum temperatures, wind speed, and precipitation.
- **stations** – station metadata (in this case, only one station is included).
- **thresholds** – definitions of different "extreme" conditions (e.g., hot days, cold days, windy days).



17

	weather_id	station_id	date	tmax_f	tmin_f	wspd_mph	precipitation_in
1	1	1	2023-07-01	95.2	75.1	12.0	0.0
2	187	1	2023-07-01	24.0	67.0	7.0	-0.35
3	2	1	2023-07-02	88.5	70.3	8.5	0.1
4	188	1	2023-07-02	68.0	3.0	18.0	0.55
5	3	1	2023-07-03	92.0	71.0	21.3	0.0
6	189	1	2023-07-03	-19.0	51.0	16.0	-0.05
7	4	1	2023-07-04	85.6	65.2	15.8	0.0
8	190	1	2023-07-04	-6.0	-13.0	26.0	0.6
9	5	1	2023-07-05	97.1	77.0	10.2	0.05
10	191	1	2023-07-05	58.0	6.0	-6.0	0.95

By completing this project, I was able to:

- Write and execute SQL queries for exploratory and analytical questions.

- Work with **time-based analysis** (yearly and monthly trends).
- Detect and count **extreme weather days** based on thresholds.
- Translate query results into a **simple Power BI visualization**.

To complete this project, I used the following tools:

- **DB Browser for SQLite** → to explore the weather.db file and run SQL queries.
- **SQLite** → as the SQL engine for querying and aggregating the dataset.
- **Power BI** → for creating simple data visualizations.
- **Excel/CSV export** → query results were exported as .csv files for use in Power BI.

Steps Taken

1. Downloaded the weather.db file and opened it in DB Browser for SQLite.
2. Explored the schema to understand available tables and columns.
    - daily_weather (date, tmax_f, tmin_f, wind speed, precipitation).
    - stations (station metadata).
    - thresholds (pre-defined conditions).
3. Wrote SQL queries in the **Execute SQL** tab of DB Browser to answer specific practice questions.
4. Exported query results to .csv format.
5. Imported the CSV into **Power BI** to build visualizations.

2) SQL Analysis

I explored the dataset using SQL queries to answer a set of practice questions. Below are the key questions, the queries I used, and a short description of the answers.

Q1. How many days of data are recorded in total?

```
1
2
3   /* Check for total rows */
4   SELECT COUNT(*) as total_days FROM daily_weather;
5
6   /*Showing unique rows */
7   SELECT COUNT(DISTINCT date) as unique_days FROM daily_weather;
8
```

	total_days
1	370

```sql

/* Check for total rows */
SELECT COUNT(*) as total_days FROM daily_weather;

/*Showing unique rows */
SELECT COUNT(DISTINCT date) as unique_days FROM daily_weather;

```

	unique_days
1	365

```sql
/* because unique rows and total rows are not the same, there are duplicate dates */
/*Checking the duplicates rows */

SELECT * FROM daily_weather
WHERE date IN (
    SELECT date FROM daily_weather
    GROUP BY date
    HAVING COUNT(*) > 1
)
ORDER BY date;
```

	weather_id	station_id	date	tmax_f	tmin_f	wspd_mph	precipitation_in	year
1	1	1	2023-07-01	95.2	75.1	12.0	0.0	2023
2	187	1	2023-07-01	24.0	67.0	7.0	-0.35	2023
3	2	1	2023-07-02	88.5	70.3	8.5	0.1	2023
4	188	1	2023-07-02	68.0	3.0	18.0	0.55	2023
5	3	1	2023-07-03	92.0	71.0	21.3	0.0	2023
6	189	1	2023-07-03	-19.0	51.0	16.0	-0.05	2023

```
Execution finished without errors.
```

Q2. What was the highest temperature recorded, and on what date?

```sql
SELECT dw.date, dw.tmax_f
FROM daily_weather dw
ORDER BY tmax_f DESC
LIMIT 10;
```

	date	tmax_f
1	2023-01-22	109.0
2	2023-03-23	109.0
3	2023-05-20	109.0
4	2023-05-23	109.0
5	2023-10-16	109.0
6	2023-03-31	108.0

Q3. What was the lowest temperature recorded, and on what date?

```
35  SELECT dw.date, dw.tmin_f
36  FROM daily_weather dw
37  ORDER BY tmin_f ASC
38  LIMIT 8;
39
40  /*Alternative Query */
41  SELECT date, tmin_f
42  FROM daily_weather WHERE tmin_f = (SELECT MIN(tmin_f) FROM daily_weather);
43
44
```

	date	tmin_f
1	2023-05-09	-88.0
2	2023-09-04	-88.0
3	2023-06-22	-83.0
4	2023-05-05	-79.0
5	2023-06-01	-77.0
6	2023-10-24	-76.0

Q4. How many days each year had extreme heat (temperature > 90°F)?

```
44
45  /* Question 4: How many days each year had "extreme heat" (temperature > 90°F)? */
46  SELECT strftime('%Y', date) AS year, COUNT(*) AS N_Hot_days
47  FROM daily_weather
48  WHERE tmax_f > 90
49  GROUP BY year
50  ORDER BY CAST(year AS INT)
51
52
```

	year	N_Hot_days
1	2023	28

Q5. How many days each year had extreme cold (temperature < 32°F)?

```
55  SELECT year, COUNT(*) AS COLD_DAYS
56  FROM daily_weather
57  WHERE tmin_f < 32
58  GROUP BY year;
59
```

	year	COLD_DAYS
1	2023	211

Q6. What is the average wind speed for each year?

```
62  SELECT year, round(avg(wspd_mph), 0) AS AV_Wind_SPD
63  FROM daily_weather
64  GROUP BY year;
65
```

	year	AV_Wind_SPD
1	2023	5.0

Q7. On how many days did wind speeds exceed 20 mph?

```
66  /* Question 7: On how many days did wind speeds exceed 20 mph? */
67
68  SELECT COUNT(*) AS N_Ex20
69  FROM daily_weather
70  WHERE wspd_mph > 20
71
```

	N_Ex20
1	105

```
68  SELECT COUNT(*) AS N_Ex20
69  FROM daily_weather
70  WHERE wspd_mph > 20
71
72  /* What's if we want to find them per year?! */
73  SELECT year, COUNT(*) AS N_W_Ex20
74  FROM daily_weather
75  WHERE wspd_mph > 20
76  GROUP BY year
77  ORDER BY CAST( year AS INT);
78
```

	year	N_W_Ex20
1	2023	105

Q8. How many extreme weather days (heat > 90°F, cold < 32°F, or wind > 20 mph) occurred each year?

```sql
SELECT year, COUNT(*) AS Ex_Weather_days
FROM daily_weather
WHERE tmax_f > 90 OR tmin_f < 32 OR wspd_mph > 20
GROUP BY year
ORDER BY CAST(year AS INT);
```

	year	Ex_Weather_days
1	2023	265

Q9. Which month had the highest number of extreme weather days?

```sql
/* 9- Which month had the highest number of extreme weather days across the dataset? */
SELECT strftime('%m', date) AS month, COUNT(*) AS EX_Weather
FROM daily_weather
WHERE (tmax_f > 90 OR tmin_f < 32 OR wspd_mph > 20)
GROUP BY month
ORDER BY Ex_Weather DESC
LIMIT 5;
```

	month	EX_Weather
1	10	28
2	07	26
3	05	25
4	12	23
5	09	23

Q10. What percentage of all recorded days were extreme weather days?

```sql
SELECT COUNT(*) FROM daily_weather AS Total_Records;

SELECT COUNT(*) FROM daily_weather AS Ext_Weather
WHERE (tmax_f >90 OR tmin_f <32 OR wspd_mph > 20);

SELECT ROUND(100.0 * 265.0/370.0, 1) || '%' AS pct_extreme_days;
```

	pct_extreme_days
1	71.6%

2) Visualization Insight

The line chart illustrates **average monthly high temperatures (tmax_f), low temperatures (tmin_f), and wind speeds (wspd_mph)** across the year.

- **Temperature trends:**
  - Average daily highs (tmax_f) peaked in **July (~30°F)**, before tapering off into the fall.
  - Average daily lows (tmin_f) were coldest in **January (~41°F)** and **February (~36°F)**, then climbed through the summer to around **28°F in July** before dropping again in the fall.
  - This highlights the expected seasonal cycle, with the largest gap between highs and lows occurring in mid-summer.
- **Wind speed trends:**
  - Wind speeds were generally low throughout the year, averaging between **0–10 mph**.
  - Slight increases were observed in **March (~6 mph)** and **July (~11 mph)**, suggesting seasonal gustiness in early spring and midsummer.
- **Combined story:**
  - July stands out as both the **hottest month** (highest tmax_f) and a period of **stronger winds** compared to most months.
  - Winter months (January–February) show the **coldest lows**, but wind activity remains moderate.

This visualization confirms the SQL findings: **July and August are peak months for extreme heat**, while **January and February drive extreme cold conditions**.