## Part D: Data Retrieval and Interpretation

#### Task D1

**Interpretation**: This query totals Australia's COVID-19 vaccinations by month from the first Australian record in the dataset up to 30 June 2021. We filter to isoCode='AUS', add up each month's daily doses (or, if only cumulative data exist, use the day-to-day increase and treat any negative corrections as zero), then output one row per month (YYYY-MM) with the monthly total. Partial months are included with whatever data is available.

## Query:

SELECT strftime('%Y', v.date) AS Year,

strftime('%m', v.date) AS Month,

sum(v.peopleVaccinated + v.peopleFullyVaccinated + v.peopleBooster) AS

'Monthly Vaccinations'

FROM vaccinationCountry v

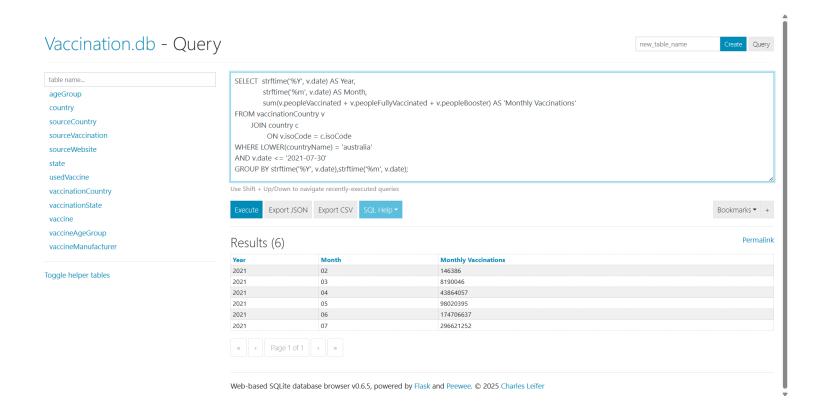
JOIN country c

ON v.isoCode = c.isoCode

WHERE LOWER(countryName) = 'australia'

AND v.date <= '2021-07-30'

GROUP BY strftime('%Y', v.date), strftime('%m', v.date);



#### Task D2

**Interpretation**: For each of the six countries (Australia, India, US, Argentina, Canada, Greece), list every month that has data and show how much the total number of people fully vaccinated increased compared to the previous month. For a country's first month in the dataset, set the change to 0 (no prior month). If a country has no data for a month, skip that month for that country. In short: compute, per country and per month, monthly change = VOM(current month) - VOM(previous month), with the first month's change = 0, and output one row per (country, month) that exists in the data.

# Query:

```
SELECT strftime('%Y', t.month start) AS "Year (OY)",
        strftime('%m', t.month start) AS "Month (OM)",
        COALESCE(SUM(CASE WHEN t.country = 'Australia' THEN
t.monthly change END), 0) AS "Australia",
        COALESCE(SUM(CASE WHEN t.country = 'India' THEN t.monthly change
END), 0) AS "India",
        COALESCE(SUM(CASE WHEN t.country = 'United States' THEN
t.monthly change END), 0) AS "United States",
        COALESCE(SUM(CASE WHEN t.country = 'Argentina' THEN
t.monthly change END), 0) AS "Argentina",
      COALESCE(SUM(CASE WHEN t.country = 'Canada' THEN t.monthly change
END), 0) AS "Canada",
      COALESCE(SUM(CASE WHEN t.country = 'Greece' THEN t.monthly change
END), 0) AS "Greece"
FROM (
 SELECT m.country,
      m.month start,
  CASE
   WHEN p.VOM IS NULL THEN 0
   ELSE (m.VOM - p.VOM)
  END AS monthly change
 FROM (
```

```
SELECT C.countryName AS country,
        date(strftime('%Y-%m-01', V.date)) AS month_start,
      MAX(V.peopleFullyVaccinated) AS VOM
  FROM VaccinationCountry V
   JOIN Country C USING (isoCode)
  WHERE C.countryName IN ('Australia','India','United
States', 'Argentina', 'Canada', 'Greece')
  GROUP BY country, month start
 ) AS m
 LEFT JOIN (
  SELECT C.countryName AS country,
        date(strftime('%Y-%m-01', V.date)) AS month start,
        MAX(V.peopleFullyVaccinated) AS VOM
  FROM VaccinationCountry V
   JOIN Country C USING (isoCode)
  WHERE C.countryName IN ('Australia','India','United
States','Argentina','Canada','Greece')
  GROUP BY country, month_start
 ) AS p
  ON p.country = m.country
 AND p.month start = date(m.month start, '-1 month')
) AS t
GROUP BY t.month start
HAVING COUNT(DISTINCT t.country) = 6
ORDER BY "Year (OY)", "Month (OM)";
```

# Vaccination.db - Query

table name...
ageGroup
country
sourceCountry
sourceVaccination
sourceWebsite
state
usedVaccine
vaccinationCountry
vaccinationCountry
vaccinationState

Toggle helper tables

vaccineManufacturer



## Results (unable to determine)

Permalink

new\_table\_name

Year (OY)	Month (OM)	Australia	India	United States	Argentina	Canada	Greece
2021	02	0	2456191	22724848	205481	422333	274759
2021	03	72712	6878504	34688799	401425	176334	293520
2021	04	185805	17286460	50221605	308993	421781	336956
2021	05	247725	16637655	32042852	1979850	1011199	1114164
2021	06	1127180	14489306	16703222	1318698	9644229	1730619
2021	07	2293622	43950050	7487141	3055689	10618652	1376760
2021	08	3442264	46098976	9850509	6431463	3140130	572440
2021	09	4017019	89679107	9957501	6849088	1501531	385280
2021	10	5143103	91612373	6338180	3326595	1263988	252357
2021	11	2279370	118912324	5079954	3877508	740278	238468
2021	12	947025	155223875	7826958	2922771	383097	426922
2022	01	455565	106701626	4854852	1916003	727023	294144
วกวว	0.2	271020	90620907	2115002	1060000	740425	201216

#### Vaccination.db - Query

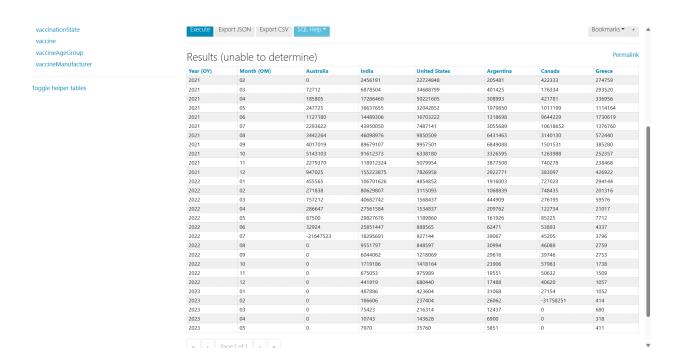
table rane...
aggidroup
country
soures Country
soures Country
soures Montanton
soures Wester
state
used Nacione
used and country
vaccination Country
vaccination Country
vaccinet sources
vaccine
vaccinet sources

loggle helper tables

ERECT strilliner(W)\*, tunorth, start) A5 "Neer (OV)\*,
start (NAME (NAME

Results (unable to determine)

Permalii



#### References

- W3Schools, 2024. SQL CASE Statement. [online] Available at: https://www.w3schools.com/sql/sql\_case.asp [Accessed 30 October 2025].
- W3Schools, 2024. SQL COALESCE Function. [online] Available at: https://www.w3schools.com/sql/func\_sqlserver\_coalesce.asp [Accessed 30 October 2025].
- GeeksforGeeks, 2024. *SQL* | *CASE Statement*. [online] Available at: https://www.geeksforgeeks.org/sql-case-statement/ [Accessed 30 October 2025].
- Oracle, 2024. *CASE Expressions*. [online] Available at: https://docs.oracle.com/en/database/ [Accessed 30 October 2025].
- Microsoft Learn, 2024. COALESCE (Transact-SQL). [online] Available at: https://learn.microsoft.com/en-us/sql/t-sql/functions/coalesce-transact-sql [Accessed 30 October 2025].