

Assignment 4

Name: Arya Patil

Assignment name: Database Design Project

Part D:

Task D.1

a.

```
SELECT
    -- selecting the columns needed for the query results
    om1.month AS 'Observation Months 1 (OM1)',
    loc.location AS 'Country Name',
    om1.total_vaccinations AS 'Administered Vaccine on OM1 (VOM1)',
    om2.month AS 'Observation Months 2 (OM2)',
    om2.total_vaccinations AS 'Administered Vaccine on OM2(VOM2)',
    om2.total_vaccinations - om1.total_vaccinations AS 'Difference of totals'
FROM
    -- selecting the first column OM1
    (SELECT
        iso_code,
        strftime('%Y-%m', substr(date, 7, 4) || '-' || substr(date, 4, 2) || '-' ||
        substr(date, 1, 2)) AS month, -- using substr to convert the date format to
        strftime format
        SUM(total_vaccinations) AS total_vaccinations
        FROM Vaccinations
        WHERE strftime('%Y-%m', substr(date, 7, 4) || '-' || substr(date, 4, 2) || '-' ||
        substr(date, 1, 2)) = '2022-04' -- filtering the April data
        GROUP BY iso_code, strftime('%Y-%m', substr(date, 7, 4) || '-' ||
        substr(date, 4, 2) || '-' || substr(date, 1, 2))) AS om1 -- naming it as om1 for
        easy access
JOIN
    -- joining on iso_code
    -- selecting the fifth column OM2
    (SELECT
        iso_code,
        strftime('%Y-%m', substr(date, 7, 4) || '-' || substr(date, 4, 2) || '-' ||
        substr(date, 1, 2)) AS month,
        SUM(total_vaccinations) AS total_vaccinations
        FROM Vaccinations
        WHERE strftime('%Y-%m', substr(date, 7, 4) || '-' || substr(date, 4, 2) || '-' ||
        substr(date, 1, 2)) = '2022-05'
        GROUP BY iso_code, strftime('%Y-%m', substr(date, 7, 4) || '-' ||
        substr(date, 4, 2) || '-' || substr(date, 1, 2))) AS om2
ON
    om1.iso_code = om2.iso_code
```

```

JOIN
  Locations loc ON om1.iso_code = loc.iso_code
ORDER BY
  loc.location;

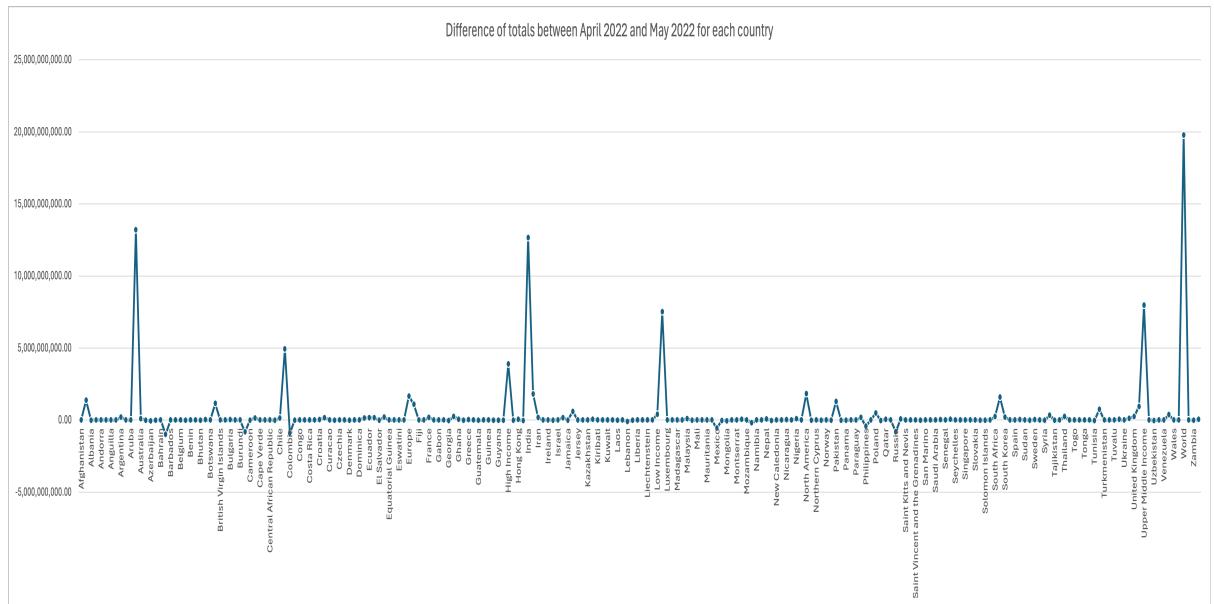
```

b. Query results

Total rows loaded: 226

		Observation Months 1 (OM1)	Country Name	Administered Vaccine on OM1 (VOM1)	Observation Months 2 (OM2)	Administered Vaccine on OM2(VOM2)	Difference of totals
1	2022-04		Afghanistan	29733868	2022-05		30259766 525898
2	2022-04		Africa	13613469631	2022-05		14982327419 1368857788
3	2022-04		Albania	13936029	2022-05		11406489 -2529540
4	2022-04		Algeria	13772044	2022-05		30411708 16639664
5	2022-04		Andorra	304764	2022-05		458757 153993
6	2022-04		Angola	17896626	2022-05		37388140 19491514
7	2022-04		Anguilla	23126	2022-05		70461 47335
8	2022-04		Antigua and ...	0	2022-05		125877 125877
9	2022-04		Argentina	2948195781	2022-05		3143874634 195678853
10	2022-04		Armenia	4291152	2022-05		4299345 8193

c.



Task D.2

a.

```
SELECT
    -- selecting the columns needed for the query results
    loc.location AS 'Country_Name',
    vacc.month AS 'Month',
    vacc.total_vaccinations AS 'Cumulative_Doses'
FROM
    -- to get Month and Cumulative doses
    (
        SELECT
            iso_code, -- selecting this to use it in group_by clause
            strftime('%Y-%m', substr(date, 7, 4) || '-' || substr(date, 4, 2) || '-' ||
            substr(date, 1, 2)) AS Month,
            SUM(total_vaccinations) AS total_vaccinations
        FROM Vaccinations
        GROUP BY iso_code, strftime('%Y-%m', substr(date, 7, 4) || '-' ||
        substr(date, 4, 2) || '-' || substr(date, 1, 2))
    ) AS vacc -- naming it as vacc for easy access
JOIN
    -- joining in iso_code and to get country name
    Locations loc ON vacc.iso_code = loc.iso_code

-- total vaccinations should be greater than the average of total_vaccinations
WHERE vacc.total_vaccinations > (
    SELECT
        AVG(vacc1.total_vaccinations)
    FROM
        (
            SELECT
                iso_code,
                strftime('%Y-%m', substr(date, 7, 4) || '-' || substr(date, 4, 2) ||
                '-' || substr(date, 1, 2)) AS Month,
                SUM(total_vaccinations) AS total_vaccinations
            FROM Vaccinations
            GROUP BY iso_code, strftime('%Y-%m', substr(date, 7, 4) || '-' ||
            substr(date, 4, 2) || '-' || substr(date, 1, 2))
        ) AS vacc1
    WHERE vacc1.month = vacc.month -- same column to match the
months
)
ORDER BY
    Month,
    Country_Name;
```

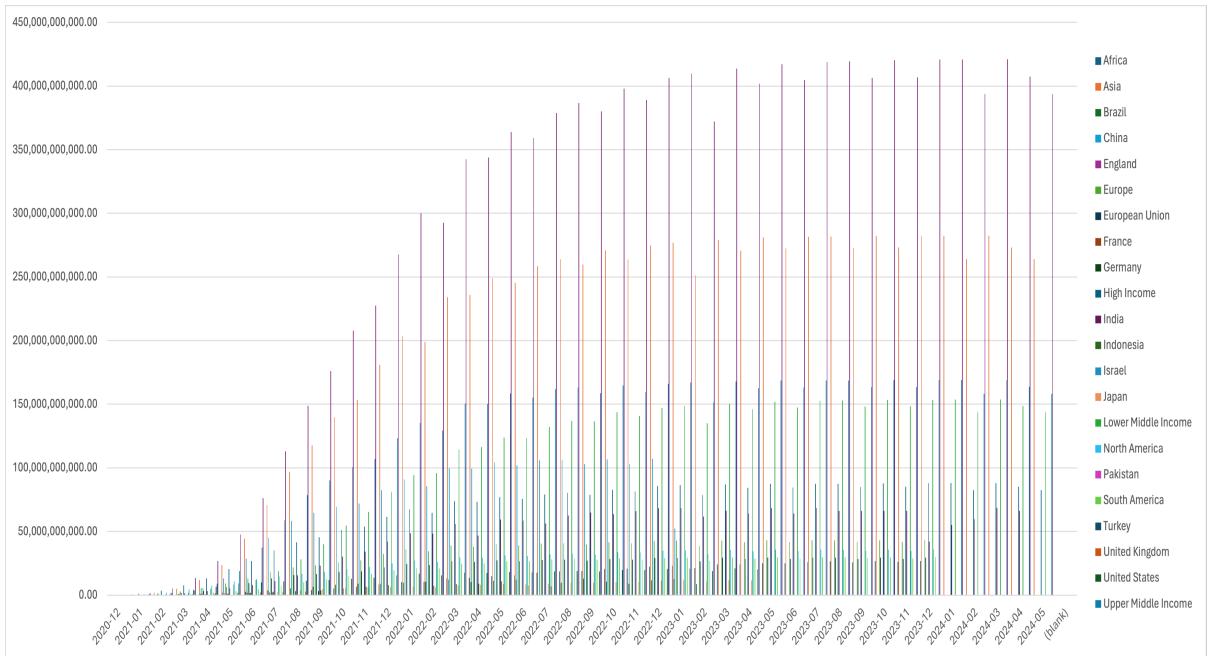
b.

Grid view Form view

Total rows loaded: 558

	Country_Name	Month	Cumulative_Doses
1	Asia	2020-12	33426341
2	High Income	2020-12	48278222
3	North America	2020-12	42460693
4	United States	2020-12	41731199
5	Upper Middle Income	2020-12	29429070
6	World	2020-12	77707292
7	Asia	2021-01	574808407
8	China	2021-01	80767000
9	England	2021-01	112616859
10	Europe	2021-01	348196034

c.

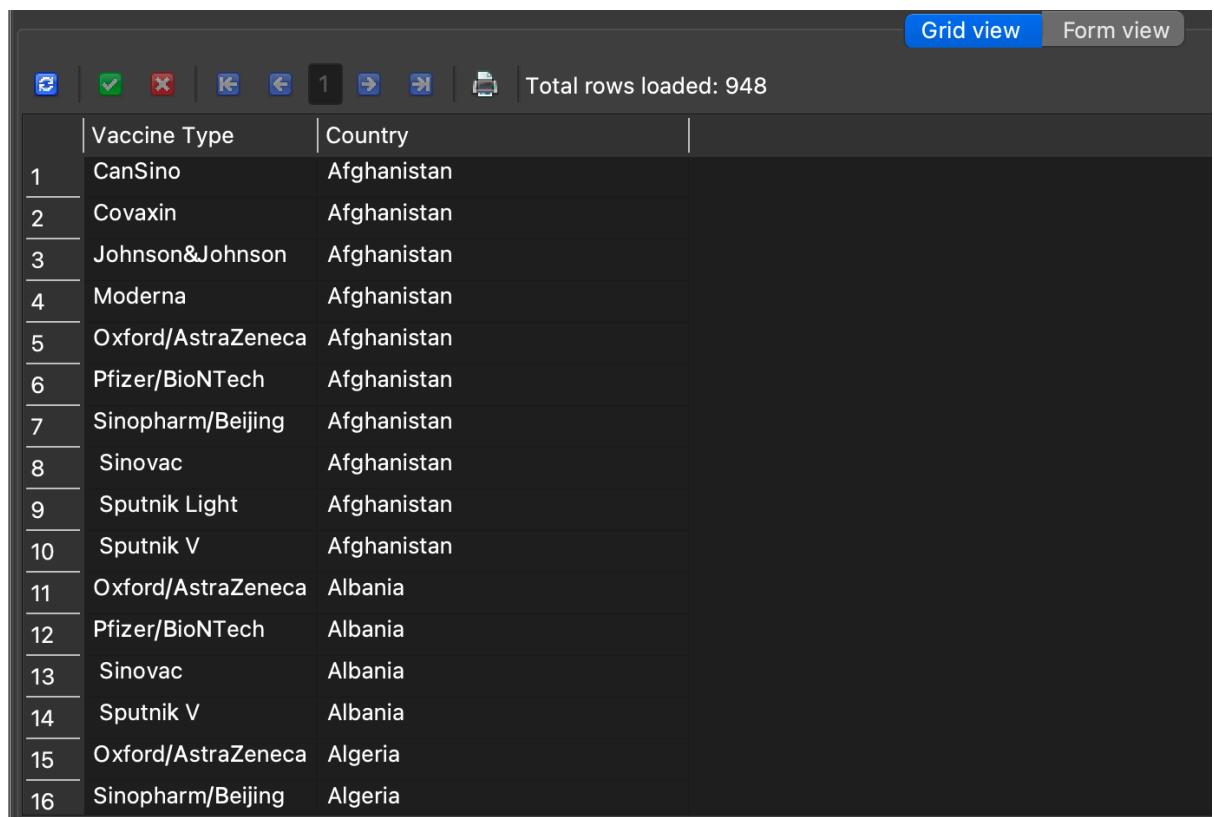


Task D.3

a. SELECT

```
v.vaccines AS 'Vaccine Type',
loc.location AS 'Country'
FROM
    Location_vaccines lv
JOIN
    Locations loc ON lv.iso_code = loc.iso_code
JOIN
    Vaccines v ON lv.vaccine_id = v.vaccine_id
; ;
```

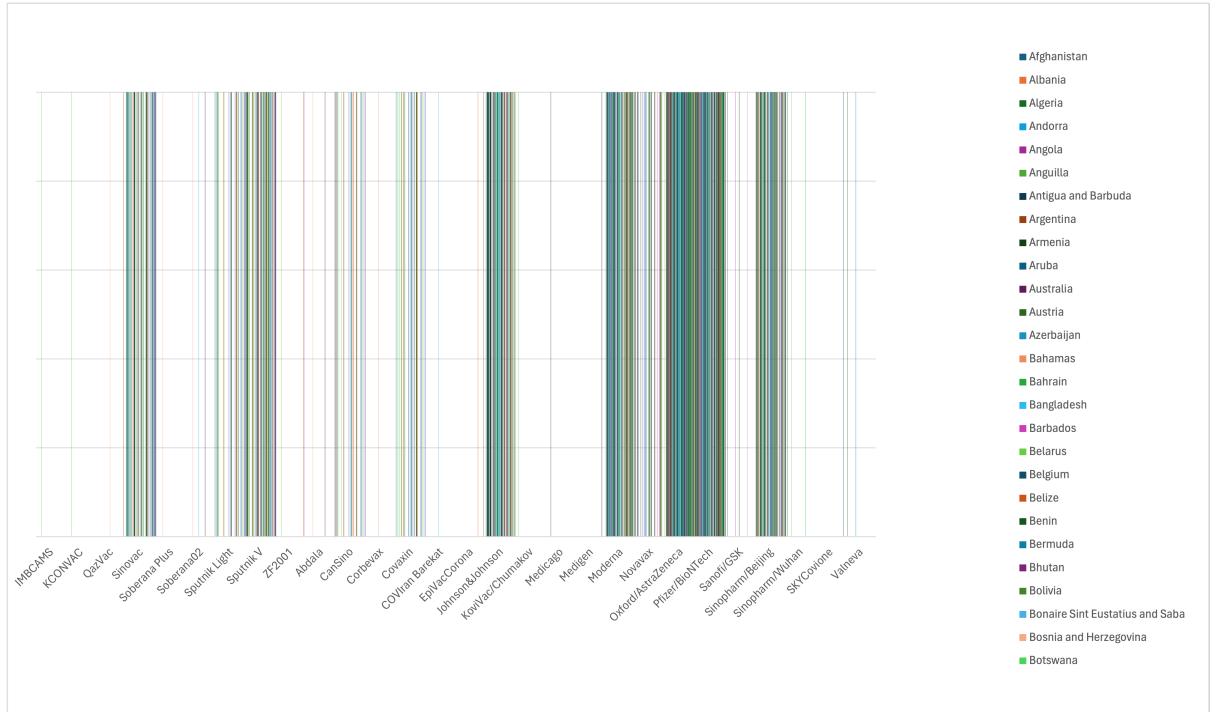
b.



The screenshot shows a database grid view with the following details:

- Header: Grid view (selected), Form view
- Toolbar: Includes icons for search, insert, delete, and navigation.
- Text: Total rows loaded: 948
- Table Headers: Vaccine Type | Country
- Data Rows (16 total):
 - 1 CanSino Afghanistan
 - 2 Covaxin Afghanistan
 - 3 Johnson&Johnson Afghanistan
 - 4 Moderna Afghanistan
 - 5 Oxford/AstraZeneca Afghanistan
 - 6 Pfizer/BioNTech Afghanistan
 - 7 Sinopharm/Beijing Afghanistan
 - 8 Sinovac Afghanistan
 - 9 Sputnik Light Afghanistan
 - 10 Sputnik V Afghanistan
 - 11 Oxford/AstraZeneca Albania
 - 12 Pfizer/BioNTech Albania
 - 13 Sinovac Albania
 - 14 Sputnik V Albania
 - 15 Oxford/AstraZeneca Algeria
 - 16 Sinopharm/Beijing Algeria

C.

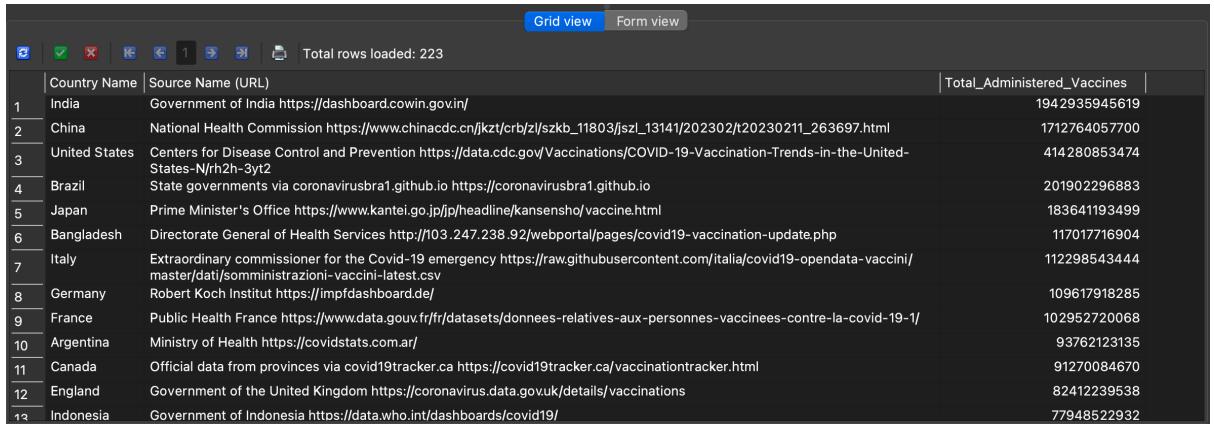


Task D.4

a. SELECT

```
loc.location AS 'Country Name',
ds.source_name || ' ' || ds.source_website AS 'Source Name (URL)',
SUM(vacc.total_vaccinations) AS 'Total_Administered_Vaccines'
FROM
    Vaccinations vacc
JOIN
    -- joining on data source to get source website
    Data_Source ds ON vacc.iso_code = ds.iso_code
JOIN
    -- joining on locations to get country name
    Locations loc ON vacc.iso_code = loc.iso_code
GROUP BY
    loc.iso_code, ds.source_website
ORDER BY
    Total_Administered_Vaccines DESC;
```

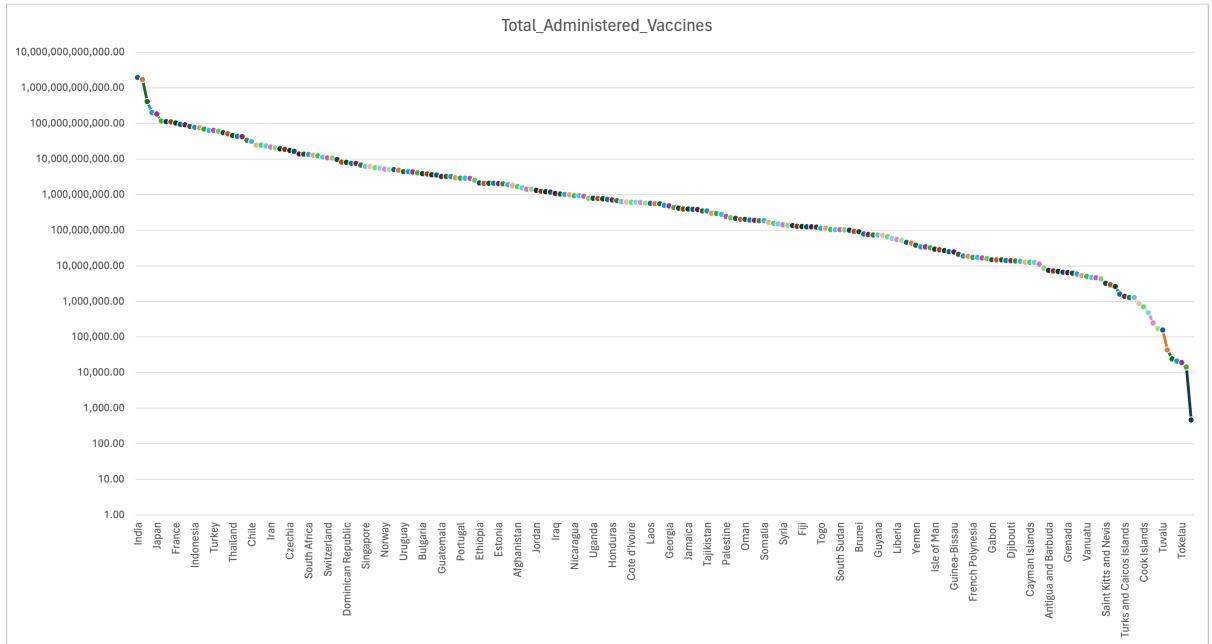
b.



The screenshot shows a database grid view with the following columns: Country Name, Source Name (URL), and Total_Administered_Vaccines. The data is sorted by Total_Administered_Vaccines in descending order. The table has 13 rows, indexed from 1 to 13. Row 1 is India, Row 2 is China, Row 3 is United States, Row 4 is Brazil, Row 5 is Japan, Row 6 is Bangladesh, Row 7 is Italy, Row 8 is Germany, Row 9 is France, Row 10 is Argentina, Row 11 is Canada, Row 12 is England, and Row 13 is Indonesia.

	Country Name	Source Name (URL)	Total_Administered_Vaccines
1	India	Government of India https://dashboard.cowin.gov.in/	1942935945619
2	China	National Health Commission https://www.chinacdc.cn/jkz/crb/z/szkb_11803/jszl_13141/202302/t20230211_263697.html	1712764057700
3	United States	Centers for Disease Control and Prevention https://data.cdc.gov/Vaccinations/COVID-19-Vaccination-Trends-in-the-United-States-N/rh2h-3yt2	414280853474
4	Brazil	State governments via coronavirusbra1.github.io https://coronavirusbra1.github.io	201902296883
5	Japan	Prime Minister's Office https://www.kantei.go.jp/jp/headline/kansensho/vaccine.html	183641193499
6	Bangladesh	Directorate General of Health Services http://103.247.238.92/webportal/pages/covid19-vaccination-update.php	117017716904
7	Italy	Extraordinary commissioner for the Covid-19 emergency https://raw.githubusercontent.com/Italia/covid19-opendata-vaccini/master/dati/somministrazioni-vaccini-latest.csv	112298543444
8	Germany	Robert Koch Institut https://impfdashboard.de/	109617918285
9	France	Public Health France https://www.data.gouv.fr/fr/datasets/donnees-relatives-aux-personnes-vaccinees-contre-la-covid-19-1/	102952720068
10	Argentina	Ministry of Health https://covidstats.com.ar/	93762123135
11	Canada	Official data from provinces via covid19tracker.ca https://covid19tracker.ca/vaccinationtracker.html	91270084670
12	England	Government of the United Kingdom https://coronavirus.data.gov.uk/details/vaccinations	82412239538
13	Indonesia	Government of Indonesia https://data.who.int/dashboards/covid19/	77948522932

C.



Task D.5

a.

```
SELECT
    -- selecting the columns needed for the query results
    Month,
    -- if country name is found return the total_fully_vaccinated people or else
    return 0
    MAX(CASE WHEN Country = 'United States' THEN Total_Fully_Vaccinated ELSE
    0 END) AS 'United States',
    MAX(CASE WHEN Country = 'Wales' THEN Total_Fully_Vaccinated ELSE 0 END)
    AS 'Wales',
    MAX(CASE WHEN Country = 'Canada' THEN Total_Fully_Vaccinated ELSE 0
    END) AS 'Canada',
    MAX(CASE WHEN Country = 'Denmark' THEN Total_Fully_Vaccinated ELSE
    0END) AS 'Denmark'
FROM
(
    SELECT
        strftime('%Y-%m', substr(date, 7, 4) || '-' || substr(date, 4, 2) || '-' ||
        substr(date, 1, 2)) AS Month,
        loc.location AS Country, --selecting to use this in group_by clause
        SUM(cs.people_fully_vaccinated) AS Total_Fully_Vaccinated
    FROM
        Country_statistics cs
    JOIN
        Locations loc ON cs.iso_code = loc.iso_code
    WHERE
        strftime('%Y', substr(date, 7, 4) || '-' || substr(date, 4, 2) || '-' || substr(date, 1,
        2)) IN ('2022', '2023') -- filtering the years
    GROUP BY
        Month, loc.location
)
GROUP BY
    Month
ORDER BY
    Month;
```

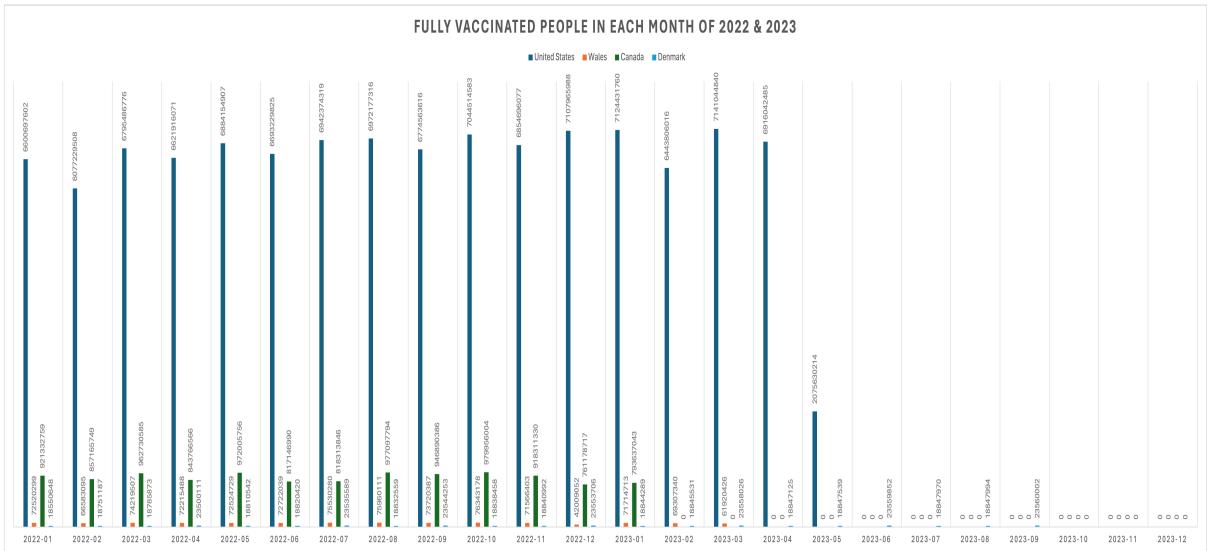
b.

Grid view Form view

Total rows loaded: 24

	Month	United States	Wales	Canada	Denmark
1	2022-01	6600697602	72520299	921332759	18550648
2	2022-02	6077229508	66583095	857165749	18751187
3	2022-03	6795486776	74219507	962730585	18785873
4	2022-04	6621916071	72215488	843766566	23500111
5	2022-05	6884154907	72524729	972005756	18810542
6	2022-06	6693229825	72722039	817146990	18820420
7	2022-07	6942374319	75530280	818313846	23535589
8	2022-08	6972177316	75960111	977097794	18832559
9	2022-09	6774563616	73720387	946890386	23544253
10	2022-10	7044514583	76343178	979956004	18838458
11	2022-11	6854696077	71566403	918311330	18840992
12	2022-12	7107965988	42009052	761178717	23553706
13	2023-01	7124431760	71714713	793637043	18844289

c.



References:

- Usage of substr() - <https://www.sqlitetutorial.net/sqlite-functions/sqlite-substr/>
- Usage of Case expression - <https://www.sqlitetutorial.net/sqlite-case/>
- Usage of Pivot table - <https://support.microsoft.com/en-au/office/create-a-pivottable-to-analyze-worksheet-data-a9a84538-bfe9-40a9-a8e9-f99134456576>
- Usage of Pivot chart - <https://support.microsoft.com/en-au/office/create-a-pivotchart-c1b1e057-6990-4c38-b52b-8255538e7b1c>