Data Cleaning, Transformation and Modelling

Question 1 : What is Data Loading in Power BI and why is it considered the first step of analysis?

Data Loading in Power BI, often initiated via the "Get Data" feature, is the foundational process of connecting the Power BI Desktop to external data sources and importing that information into the application. It acts as the bridge between raw data storage—whether in Excel files, SQL databases, cloud services, or web pages—and the analytical engine of Power BI.

It is considered the **first and most critical step** of analysis for several reasons:

- **Establishing the Foundation:** We cannot analyze what is not there. Data loading brings the raw material Without this step, the subsequent stages of cleaning, modeling, and visualization simply cannot happen.
- **Defining Scope and Source:** This stage determines *what* data is available for analysis. It involves selecting specific tables or datasets, which sets the boundaries of your entire project.
- **Triggering ETL (Extract, Transform, Load):** Loading data immediately leads into the **Power Query Editor**.

This is where the raw data is cleaned and shaped. Since transformation logic relies on the structure of the incoming data, the loading phase dictates how accurate and efficient our data preparation will be.

Question 2 : Explain the difference between “Load” and “Transform Data” in Power BI.

When you connect to a new data source in Power BI, we are presented with a Navigator window containing two primary action buttons. The choice you make here dictates the immediate next step in your workflow.

1. Load

Selecting **Load** tells Power BI to take the data exactly as it currently appears in the source and import it directly into the **Power BI Data Model**.

- The window closes, and Power BI begins processing rows immediately. The data becomes available in the "Data" and "Report" views.
- We have to use this *only* if your data is already perfectly clean, formatted, and requires no changes (e.g., a highly curated SQL view or a verified Excel table).

- If the data has errors, unnecessary columns, or wrong data types, you will clutter your model and potentially slow down performance.

2. Transform Data (Recommended)

Selecting **Transform Data** (formerly "Edit") redirects the data to the **Power Query Editor** before it enters the Data Model.

- The Power BI interface opens a separate window (Power Query). Here, we can clean the data—filtering rows, renaming columns, changing data types, or replacing values—without affecting the original source.
- This is the **best practice** for most scenarios. It allows us to shape the data to fit your analytical needs, ensuring that only clean, relevant data is loaded into your report.
- It reduces the file size and improves report performance by filtering out unnecessary data *before* import.

Feature	**Load**	**Transform Data**
Destination	Direct to Data Model & Report View	Power Query Editor
Action	Imports data "as is"	Allows cleaning & shaping first
Best For	pre-processed data	Raw, messy, or large datasets
Memory Usage	High (imports everything)	Optimized (imports only what you keep)

Question 3 : What is a Fact Table and a Dimension Table? Give examples from the dataset.

In Power BI and data modeling (specifically the Star Schema), tables are categorized into two types based on the data they hold:

1. Fact Table

A **Fact Table** contains the quantitative data or "metrics" of your business. These are the numbers you want to analyze (sum, average, count). It represents **events** or transactions that happened.

- Long and narrow (many rows, fewer columns), contains numbers and IDs (Foreign Keys) linking to dimension tables.

Example from the Dataset: If we were to model our `given dataset`, the **Fact Table** would be the central table containing the daily statistics.

- **Name:** `Fact_Daily_Cases`
- **Columns included:**
  - **Keys (to link data):** `Date` (or DateID), `Location_ID` (a generated ID linking to Country/State).
  - **Metrics (The "Facts"):**

- `Confirmed_Cases`
- `Recovered_Cases`
- `Deaths`
- `Active_Cases`
- `Tests_Conducted`
- `Hospitalized`

These columns contain numbers that you will aggregate (e.g., "Total Deaths in Brazil" or "Average Hospitalized in UK").

2. Dimension Table

A **Dimension Table** contains descriptive attributes or "context" related to the facts. These are the text fields you use to slice, dice, and filter your data. Short and wide (fewer rows, more descriptive columns), contains unique values and a Primary Key.

Example from your Dataset: From your file, we would extract the descriptive columns to create **Dimension Tables**.

- **Proposed Name:** `Dim_Location`
    - **Columns included:** `Country`, `State`.
    - **Usage:** Used to filter the report (e.g., a Slicer for "Country").
- **Proposed Name:** `Dim_Vaccination_Status`

- ○ **Columns included:** `Vaccination_Status`.

Question 4 : Why is Star Schema preferred over Snowflake Schema in Power BI?

In Power BI, the **Star Schema** is universally preferred over the Snowflake Schema primarily due to **performance** and **usability**.

Power BI runs on the **VertiPaq engine**, which is an in-memory, columnar database. It is highly optimized for scanning single tables and compressing data. While Snowflake schemas save storage in traditional databases by normalizing data (splitting tables to avoid repetition), Power BI's compression engine handles data redundancy so efficiently that this storage benefit is negligible.

Instead, the Snowflake schema introduces a major penalty: **Joins**. Every relationship in Power BI has a computational cost. A Star Schema ensures that any dimension is just *one relationship away* from the fact table. A Snowflake schema often requires traversing chains of tables (e.g., Sales > Product > SubCategory > Category) to filter data. This "relationship hopping" significantly slows down query performance and report rendering.

Furthermore, Star Schemas are **simpler for users**. A business user wants to find "Product Category" in the "Product" table, not hunt for it in a separate linked table. This structure also simplifies **DAX formulas**, avoiding the need for complex, performance-killing bi-directional filters often required in Snowflake designs .

Question 5 : Identify and remove duplicate records based on Date, Country, and State.

1. Open Power Query

- We have to use the **Home** tab in Power BI Desktop.
- Click **Transform data**.

2. Select the Columns

- Hold down the **Ctrl** key on our keyboard.
- Click the headers of the three columns you want to check: `Date`, `Country`, and `State`.

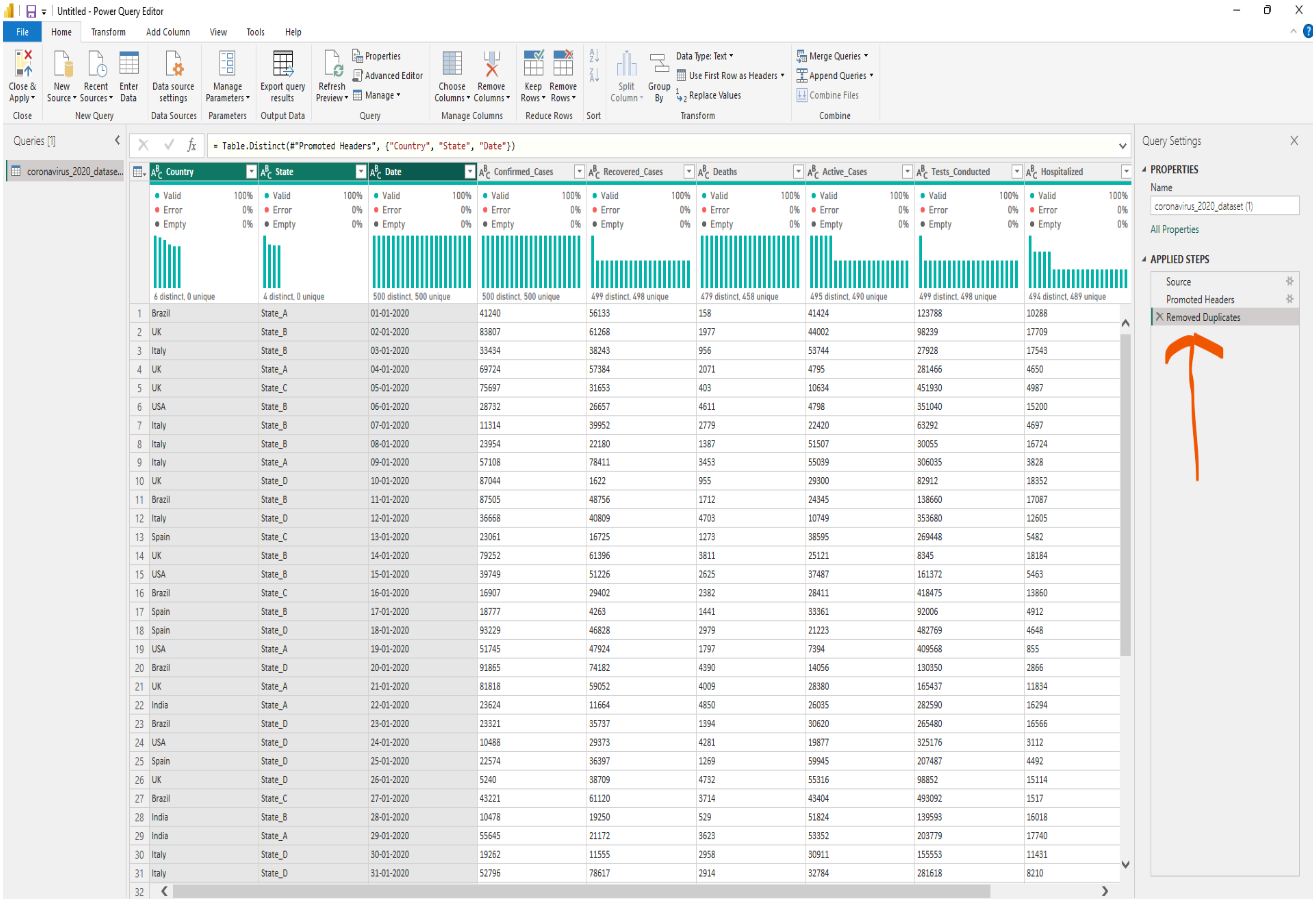
3. Removing Duplicates

- With those three columns still highlighted, **Right-click** on one of the headers clicking the **Ctrl** key .
- Select **Remove Duplicates** from the menu.

4. Check our Work

- Look at the "Applied Steps" panel on the right. We will be able to see a new step called **"Removed Duplicates"**.
- Click **Close & Apply** (top left) to save the changes.

Note : The date format in the raw data was incorrect. Directly changing the datatype in Power BI resulted in an "Error" for the entire column, even when attempting the change locale.



coronavirus_2020_datase...

```
= Table.Distinct(#"Promoted Headers", {"Country", "State", "Date"})
```

	Country	State	Date	Confirmed_Cases	Recovered_Cases	Deaths	Active_Cases	Tests_Conducted	Hospitalized
	● Valid 100%	● Valid 100%	● Valid 100%	● Valid 100%	● Valid 100%	● Valid 100%	● Valid 100%	● Valid 100%	● Valid 100%
	● Error 0%	● Error 0%	● Error 0%	● Error 0%	● Error 0%	● Error 0%	● Error 0%	● Error 0%	● Error 0%
	● Empty 0%	● Empty 0%	● Empty 0%	● Empty 0%	● Empty 0%	● Empty 0%	● Empty 0%	● Empty 0%	● Empty 0%
	6 distinct, 0 unique	4 distinct, 0 unique	500 distinct, 500 unique	500 distinct, 500 unique	499 distinct, 498 unique	479 distinct, 458 unique	495 distinct, 490 unique	499 distinct, 498 unique	494 distinct, 489 unique
1	Brazil	State_A	01-01-2020	41240	56133	158	41424	123788	10288
2	UK	State_B	02-01-2020	83807	61268	1977	44002	98239	17709
3	Italy	State_B	03-01-2020	33434	38243	956	53744	27928	17543
4	UK	State_A	04-01-2020	69724	57384	2071	4795	281466	4650
5	UK	State_C	05-01-2020	75697	31653	403	10634	451930	4987
6	USA	State_B	06-01-2020	28732	26657	4611	4798	351040	15200
7	Italy	State_B	07-01-2020	11314	39952	2779	22420	63292	4697
8	Italy	State_B	08-01-2020	23954	22180	1387	51507	30055	16724
9	Italy	State_A	09-01-2020	57108	78411	3453	55039	306035	3828
10	UK	State_D	10-01-2020	87044	1622	955	29300	82912	18352
11	Brazil	State_B	11-01-2020	87505	48756	1712	24345	138660	17087
12	Italy	State_D	12-01-2020	36668	40809	4703	10749	353680	12605
13	Spain	State_C	13-01-2020	23061	16725	1273	38595	269448	5482
14	UK	State_B	14-01-2020	79252	61396	3811	25121	8345	18184
15	USA	State_B	15-01-2020	39749	51226	2625	37487	161372	5463
16	Brazil	State_C	16-01-2020	16907	29402	2382	28411	418475	13860
17	Spain	State_B	17-01-2020	18777	4263	1441	33361	92006	4912
18	Spain	State_D	18-01-2020	93229	46828	2979	21223	482769	4648
19	USA	State_A	19-01-2020	51745	47924	1797	7394	409568	855
20	Brazil	State_D	20-01-2020	91865	74182	4390	14056	130350	2866
21	UK	State_A	21-01-2020	81818	59052	4009	28380	165437	11834
22	India	State_A	22-01-2020	23624	11664	4850	26035	282590	16294
23	Brazil	State_D	23-01-2020	23321	35737	1394	30620	265480	16566
24	USA	State_D	24-01-2020	10488	29373	4281	19877	325176	3112
25	Spain	State_D	25-01-2020	22574	36397	1269	59945	207487	4492
26	UK	State_D	26-01-2020	5240	38709	4732	55316	98852	15114
27	Brazil	State_C	27-01-2020	43221	61120	3714	43404	493092	1517
28	India	State_B	28-01-2020	10478	19250	529	51824	139593	16018
29	India	State_A	29-01-2020	55645	21172	3623	53352	203779	17740
30	Italy	State_D	30-01-2020	19262	11555	2958	30911	155553	11431
31	Italy	State_D	31-01-2020	52796	78617	2914	32784	281618	8210
32									

PROPERTIES

Name: coronavirus_2020_dataset (1)

APPLIED STEPS

- Source
- Promoted Headers
- Removed Duplicates

Question 6 : Identify and replace null values in Vaccination_Status

Step 1: Open Power Query Editor

1. We have to use the **Home** tab in Power BI Desktop.
2. Click **Transform data**. This opens the Power Query Editor window.

Step 2: Identify the Nulls

1. Locate the **Vaccination_Status** column.
2. We will see some cells that are literally empty.
3. To verify, click the small arrow (filter icon) on the column header. We will see `(blank)` in the list of values.

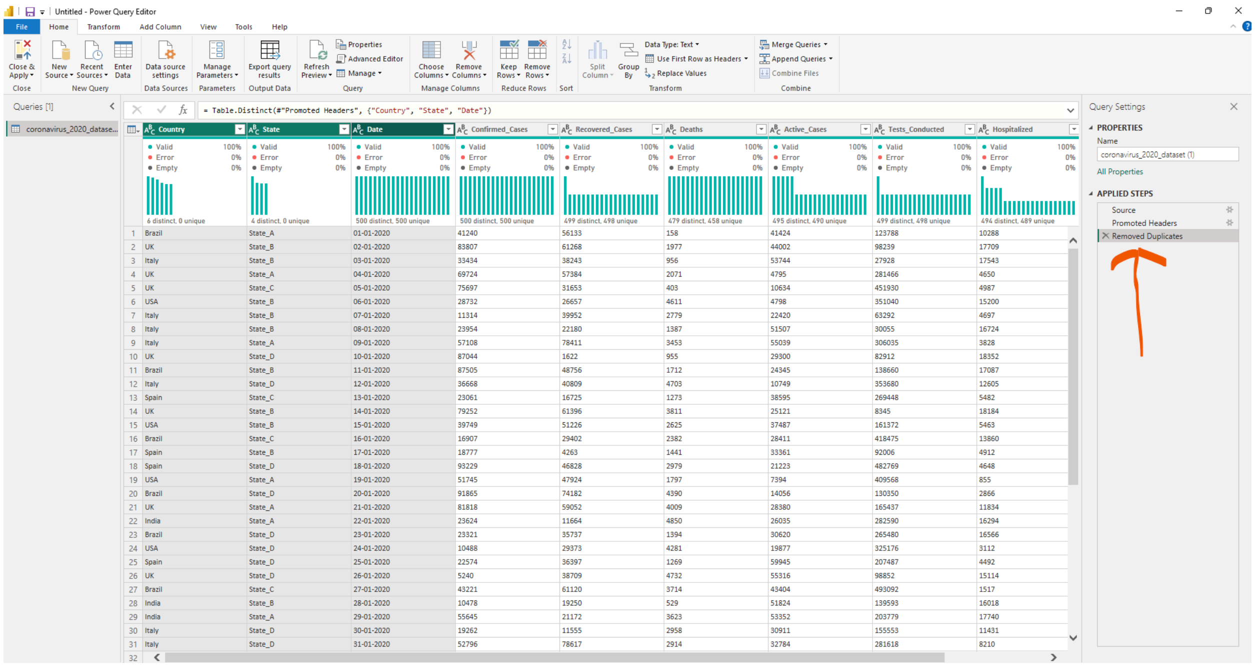
Step 3: Replace Values

1. **Right-click** the header of the`Vaccination_Status` column.
2. Select **Replace Values** from the menu.
3. In the dialog box that appears:
   - **Value To Find:** Type `null` (leave this empty as your cells are just blank).
   - **Replace With:** Type the value you want, such as `Not Mentioned`.

4. Click **OK**.

Step 4: Verify and Save

1. The *blank* values in the column should now instantly change to "`Not Mentioned`".
2. Click **Close & Apply** in the top-left corner to save these changes to your report.



```
= Table.Distinct(#"Promoted Headers", {"Country", "State", "Date"})
```

coronavirus_2020_datase...

Name: coronavirus_2020_dataset (1)

Applied steps:
- Source
- Promoted Headers
- Removed Duplicates

	Country	State	Date	Confirmed_Cases	Recovered_Cases	Deaths	Active_Cases	Tests_Conducted	Hospitalized
	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%
	6 distinct, 0 unique	4 distinct, 0 unique	500 distinct, 500 unique	500 distinct, 500 unique	499 distinct, 498 unique	479 distinct, 458 unique	495 distinct, 490 unique	499 distinct, 498 unique	494 distinct, 489 unique
1	Brazil	State_A	01-01-2020	41240	56133	158	41424	123788	10288
2	UK	State_B	02-01-2020	83807	61268	1977	44002	98239	17709
3	Italy	State_B	03-01-2020	33434	38243	956	53744	27928	17543
4	UK	State_A	04-01-2020	69724	57384	2071	4795	281466	4650
5	UK	State_C	05-01-2020	75697	31653	403	10634	451930	4987
6	USA	State_B	06-01-2020	28732	26657	4611	4798	351040	15200
7	Italy	State_B	07-01-2020	11314	39952	2779	22420	63292	4697
8	Italy	State_B	08-01-2020	23954	22180	1387	51507	30055	16724
9	Italy	State_A	09-01-2020	57108	78411	3453	55039	306035	3828
10	UK	State_D	10-01-2020	87044	1622	955	29300	82912	18352
11	Brazil	State_B	11-01-2020	87505	48756	1712	24345	138660	17087
12	Italy	State_D	12-01-2020	36668	40809	4703	10749	353680	12605
13	Spain	State_C	13-01-2020	23061	16725	1273	38595	269448	5482
14	UK	State_B	14-01-2020	79252	61396	3811	25121	8345	18184
15	USA	State_B	15-01-2020	39749	51226	2625	37487	161372	5463
16	Brazil	State_C	16-01-2020	16907	29402	2382	28411	418475	13860
17	Spain	State_B	17-01-2020	18777	4263	1441	33361	92006	4912
18	Spain	State_D	18-01-2020	93229	46828	2979	21223	482769	4648
19	USA	State_A	19-01-2020	51745	47924	1797	7394	409568	855
20	Brazil	State_D	20-01-2020	91865	74182	4390	14056	130350	2866
21	UK	State_A	21-01-2020	81818	59052	4009	28380	165437	11834
22	India	State_A	22-01-2020	23624	11664	4850	26035	282590	16294
23	Brazil	State_D	23-01-2020	23321	35737	1394	30620	265480	16566
24	USA	State_D	24-01-2020	10488	29373	4281	19877	325176	3112
25	Spain	State_D	25-01-2020	22574	36397	1269	59945	207487	4492
26	UK	State_D	26-01-2020	5240	38709	4732	55316	98852	15114
27	Brazil	State_C	27-01-2020	43221	61120	3714	43404	493092	1517
28	India	State_B	28-01-2020	10478	19250	529	51824	139593	16018
29	India	State_A	29-01-2020	55645	21172	3623	53352	203779	17740
30	Italy	State_D	30-01-2020	19262	11555	2958	30911	155553	11431
31	Italy	State_D	31-01-2020	52796	78617	2914	32784	281618	8210
32									

Question 7 : Create a new column to calculate Recovery Rate.

Method : Using DAX

Go to Data View: Click the Table icon on the left sidebar.

New Column:

- Right-click on your table name in the **Data** panel (right side).
- Select **New column**.

Enter Formula:

```
Recovery Rate =
DIVIDE('coronavirus_2020_dataset (1)'[Recovered_Cases],'coronavirus_2020_dataset (1)'[Confirmed_Cases],0)
```

Question 8 : Create a summarized table showing total confirmed cases by Country.

Click here to check answers (5 to 8)