Database Design

Submitted by Deepthi Suresh

S3991481

Part B

For designing the database, I have initially followed a bottom -up approach by creating an initial schema from the information available and normalizing the initial schema.

Initial Schema – based on each file.

1) **Location.csv**

Locations(location,iso_code, vaccines,last_observation_date,source_name,source_website)

FDs

Iso_code -> location, last_observation_date, source_name,source_website,vaccines

Primary Key

Locations(~~location~~,iso_code, ~~vaccines,last_observation_date,source_name,source_website~~)
PKs - iso_code,

Schema

Locations(location,iso_code, vaccines,last_observation_date,source_name,source_website)

Checking for normalization
1NF - Not in 1NF as it is not meeting the requirements. Vaccine is a multi variate attribute and is not atomic. So, we move vaccine to a table of its own.
Vaccine (Iso_code*,Vaccines, source_name,source_website)

Source name and website is repeated several times for different vaccine, to reduce redundancy, it can be moved to a separate table.

Updated Schema
Locations(location,iso_code,last_observation_date)
Vaccine (Iso_code*,Vaccines)
Source(source_name,source_website)

2NF - satisfied.

3NF – is in 3NF

Updated Schema
Locations(location,iso_code,last_observation_date)
Vaccine (Iso_code*,Vaccines, source_name ,source_website)
Source(source_name,source_website)

2) Vaccination by manufacturer.csv

Manufacturer(location, date, vaccine, total_vaccinations)

FDs

Location, date, vaccine -> total_vaccinations

Primary Key

Manufacturer(location, date, vaccine, ~~total_vaccinations~~)

PK -> Location, date, vaccine

Schema

Manufacturer(location*, date, vaccine*, total_vaccinations)

Is in 1NF as they are all atomic

Is in 2NF as the total_vaccination fully dependent on PK.

Is in 3NF - None of the non primary key attributes are transitively dependent on the primary key.

3) vaccination by age group

Vacc._age(location, date, age_group, people_vaccinated_per_hundred, people_fully_vaccinated_per_hundred, people_with_booster_per_hundred)

FD

location, date, age_group -> people_vaccinated_per_hundred, people_fully_vaccinated_per_hundred, people_with_booster_per_hundred

PKs

PK -> location, date, age_group

Schema

Vacc._age(location, date, age_group, people_vaccinated_per_hundred, people_fully_vaccinated_per_hundred, people_with_booster_per_hundred)

Is in 1 NF as all attributes are atomic.

Is in 2NF as the non primary keys fully dependent on the primary key.

Is in 3NF - None of the non primary key attributes are transitively dependent on the primary key.

4) US_state_vaccination.csv

US(Date, location, total_vaccinations, total_distributed, people_vaccinated, people_fully_vaccinated_per_hundred, total_vaccinations_per_hundred, people_fully_vaccinated, people_vaccinated_per_hundred, distributed_per_hundred, daily_vaccinations_raw, daily_vaccinations, daily_vaccinations_per_million, total_boosters, total_boosters_per_hundred)

Location name changed to “state”. Added iso code to enable future addition of state information for other countries

FD

Iso_code, Date, state -> total_vaccinations, total_distributed, people_vaccinated, people_fully_vaccinated_per_hundred, total_vaccinations_per_hundred, people_fully_vaccinated, people_vaccinated_per_hundred, distributed_per_hundred, daily_vaccinations_raw, daily_vaccinations, daily_vaccinations_per_million, total_boosters, total_boosters_per_hundred

PKs

Iso_code,Date,state

Schema

statewise_distribution (Iso_code, Date, state total_vaccinations, total_distributed, people_vaccinated, people_fully_vaccinated_per_hundred, total_vaccinations_per_hundred, people_fully_vaccinated, people_vaccinated_per_hundred, distributed_per_hundred, daily_vaccinations_raw, daily_vaccinations, daily_vaccinations_per_million, total_boosters, total_boosters_per_hundred)

Is in 1NF
Is in 2 NF since all the non primary keys depend on the PK
Is in 3NF - None of the non primary key attributes are transitively dependent on the primary key

5) Vaccination.csv

Vaccination(location, iso_code, date, total_vaccinations, people_vaccinated, people_fully_vaccinated, total_boosters daily_vaccinations_raw, daily_vaccinations, total_vaccinations_per_hundred, people_vaccinated_per_hundred, people_fully_vaccinated_per_hundred, total_boosters_per_hundred, daily_vaccinations_per_million, daily_people_vaccinated, daily_people_vaccinated_per_hundred)

FD

Iso_code -> location

Iso_code, Date -> total_vaccinations, people_vaccinated, people_fully_vaccinated, total_boosters daily_vaccinations_raw, daily_vaccinations, total_vaccinations_per_hundred, people_vaccinated_per_hundred, people_fully_vaccinated_per_hundred, total_boosters_per_hundred, daily_vaccinations_per_million, daily_people_vaccinated, daily_people_vaccinated_per_hundred

PK

Iso_code,Date

Schema

Vaccination(location,iso_code*, date, total_vaccinations, people_vaccinated, people_fully_vaccinated, total_boosters, daily_vaccinations_raw, daily_vaccinations, total_vaccinations_per_hundred, people_vaccinated_per_hundred, people_fully_vaccinated_per_hundred, total_boosters_per_hundred, daily_vaccinations_per_million, daily_people_vaccinated, daily_people_vaccinated_per_hundred)

1NF - Is in 1NF as they are atomic values
2NF – it is not in 2NF as location is not fully dependent on the primary value. So, we remove location from the original schema and place into a table of its own.

Location (Iso_code,location)

Vaccination(iso_code*, date, total_vaccinations, people_vaccinated, people_fully_vaccinated, total_boosters, daily_vaccinations_raw, daily_vaccinations, total_vaccinations_per_hundred, people_vaccinated_per_hundred, people_fully_vaccinated_per_hundred, total_boosters_per_hundred, daily_vaccinations_per_million, daily_people_vaccinated, daily_people_vaccinated_per_hundred)

Is in 3NF as None of the non primary key attributes are transitively dependent on the primary key.

6) **Australia, England, France, Germany**

CountryName(Location, date, vaccine, source_url, total_vaccinations, people_vaccinated, people_fully_vaccinated, total_boosters)

FD

Iso_code, date, vaccine
-> Vaccine, source_url, total_vaccinations, people_vaccinated, people_fully_vaccinated, total_boosters
Vaccine -> Source_url

Creating a country_vaccineusedbydate table to reduce redundancies. total_vaccinations, people_vaccinated, people_fully_vaccinated, total_boosters are removed as they can be derived from the vaccinations table. Location is replaced by ISO_code. Combining all the countries into one csv file

PK

Iso_code, date, vaccine

1NF - Not in 1NF since vaccine is not atomic and has comma separated values . Vaccine table is already created in the previous steps and can be used for this purpose.

2NF - Not in 2NF as the source_url attribute is not fully dependent on the date. This can be moved into a new table.

Schema

Source(Vaccine*, source_url)
country_vaccineusedbydate (iso_code*, date*, Vaccine*)

Is in 3NF as None of the non primary key attributes are transitively dependent on the primary key.

Normalisation challenges

I have followed a combination of bottoms up and top down approach to come up with a database schema and ER diagram. Initially a schema was drawn from the different csv files, which clearly pointed out the redundancies.

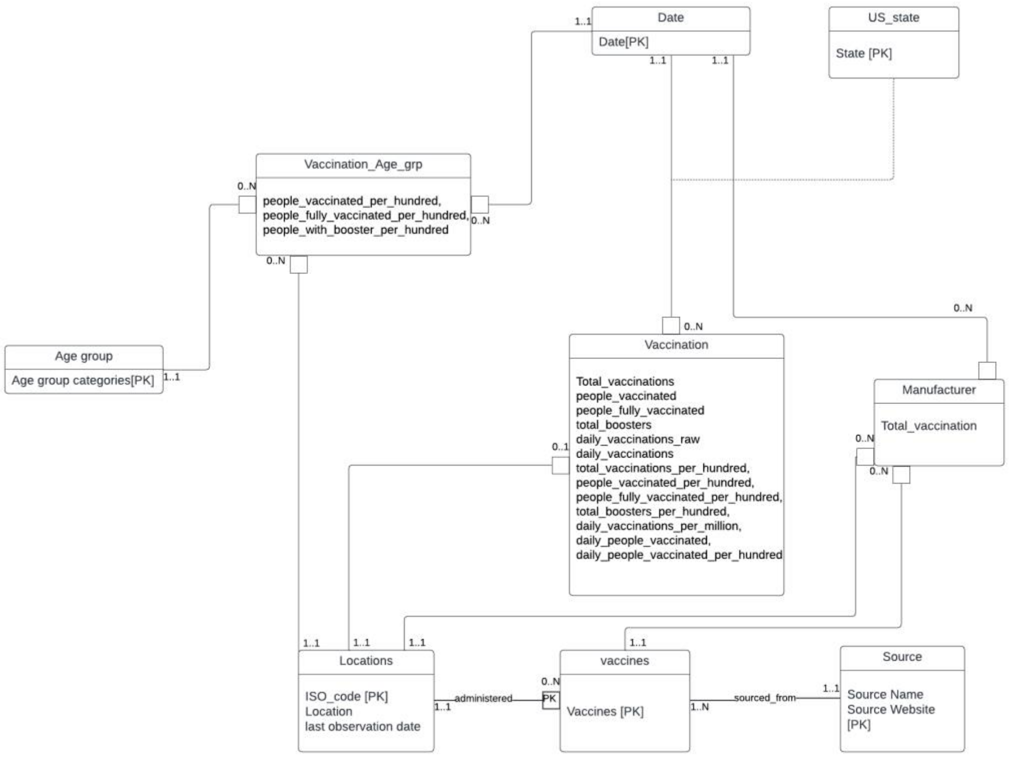
Most of the schemas were not even in the 1NF since variables like vaccines were multi variate and had several vaccine types separated by commas in the same cell. This was corrected by separating them to another table as shown above in the normalisation steps.

Another issue faced was that several variable had repeated observations such as age group, sources etc which leads to redundancies. This was solved by creating separate tables based on the multiplicities. Date was separated into separate tables to avoid any error during future updates.

Some of the variables like location came up in separate normalisations process of the files, which was later combined to create a single schema for location.

Iso_code was used as primary key and replaced location for better readability in some files.

ER Diagram



Final Database Schema

Based on 7 step relational database schema and normalization methods

Locations(iso_code,location, last_observation_date)

Date(Date)

Age_Group (Age_group)

Source (source_name,source_website)

Manufacturer(Iso_code*, date*, vaccine*, total_vaccinations) # replaced location with Iso_code for comparability.

Vacc._age(iso_code*, date*, age_group*, people_vaccinated_per_hundred, people_fully_vaccinated_per_hundred, people_with_booster_per_hundred)

Vaccine (Vaccines)

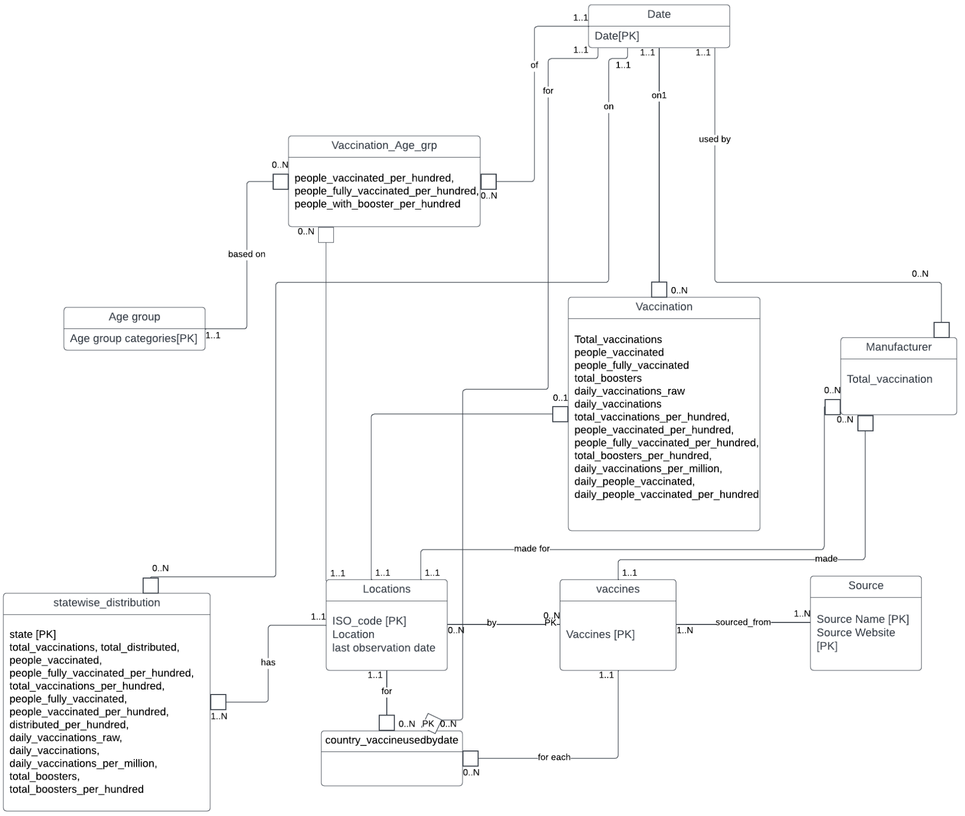
Vaccinebyloc(Iso_code*, vaccine*, source_name*,source_website*)

Vaccination(iso_code*, date, total_vaccinations, people_vaccinated, people_fully_vaccinated, total_boosters daily_vaccinations_raw, daily_vaccinations, total_vaccinations_per_hundred, people_vaccinated_per_hundred, people_fully_vaccinated_per_hundred, total_boosters_per_hundred, daily_vaccinations_per_million, daily_people_vaccinated, daily_people_vaccinated_per_hundred)

country_vaccineusedbydate (iso_code*, date*, Vaccine*)

statewise_distribution (Iso_code*, Date*, state total_vaccinations, total_distributed, people_vaccinated, people_fully_vaccinated_per_hundred, total_vaccinations_per_hundred, people_fully_vaccinated, people_vaccinated_per_hundred, distributed_per_hundred, daily_vaccinations_raw, daily_vaccinations, daily_vaccinations_per_million, total_boosters, total_boosters_per_hundred)

Updated ER Diagram



Assumptions

- Source website and source URL are considered the same.
- Several tables with location as primary key have been replaced with ISO code.

- European union data has been removed from the manufactures csv as confirmed in discussion
- In ER diagram, its assumed that there can be observations with 0 data hence the 0..1/N multiplicity.