ISYS_Databases_Concepts_Assessment4

Name: Mrwan Alhandi. ID: 3969393

Understanding the data

locations.csv

What is the data set about?

- The data set record the vaccine type with the date and source for each country.

Data manipulations

- vaccines attribute is a multi valued attribute. I splited it by column and row.
- All attributes were set to the correct data type.
- No redundent attributes.

us_state_vaccinations.csv

What is the data set about?

- The data set record US data on COVID-19 vaccinations.

Data manipulations

- total_boosters and total_boosters_per_hundred contains empty cells and hence I replaced them with null.
- All attributes were set to the correct data type.
- people_fully_vaccinated + people_vaccinated = total_vaccinations. Therefore, I deleted total_vaccinations attribute since it is redundent.
- The data set description states that any analysis should be done on daily_vaccinations rather than daily_vaccinations_raw. Therefore, I deleted daily_vaccinations_raw because I see it as an extra attribute that's not necessary.
- Since we do not know the population of the country when the data were recorded. Therefore, I can not delete any attribute that depends on the population such as people_fully_vaccinated_per_hundred.

vaccinations.csv

What is the data set about?

- The data set record Country-by-country data on global COVID-19 vaccinations.

Data Manipulations

- total_boosters have empty cells and hence I replaced those with null. Similarly, with total_boosters_per_hundred.
- total_vaccinations and daily_vaccinations_raw deleted.

vaccinations_by_age_group.csv

What is the data set about?

- The data set record the location, the date, and the age of those who vaccinated.

Data Manipulations

- The data set is tidy and inserted with the correct data type.

vaccinations_by_manufacturer.csv

What is this data set about?

- It record the total vaccinations administered of each vaccine type for each country.

Data Manipulations

- The data set is tidy and inserted with the correct data type.

Countries data set

What is the data set about?

- The data set record the number of vaccines administrated with the vaccine details by each country. There are four countries: Australia, Germany, Italy and US.

Data Manipulations

- total_vaccinations deleted from the four data sets.
- vaccine is multi valued attribute. Therefore, I splited the attribute. This is for all countries.

All the unique attributes across all data sets

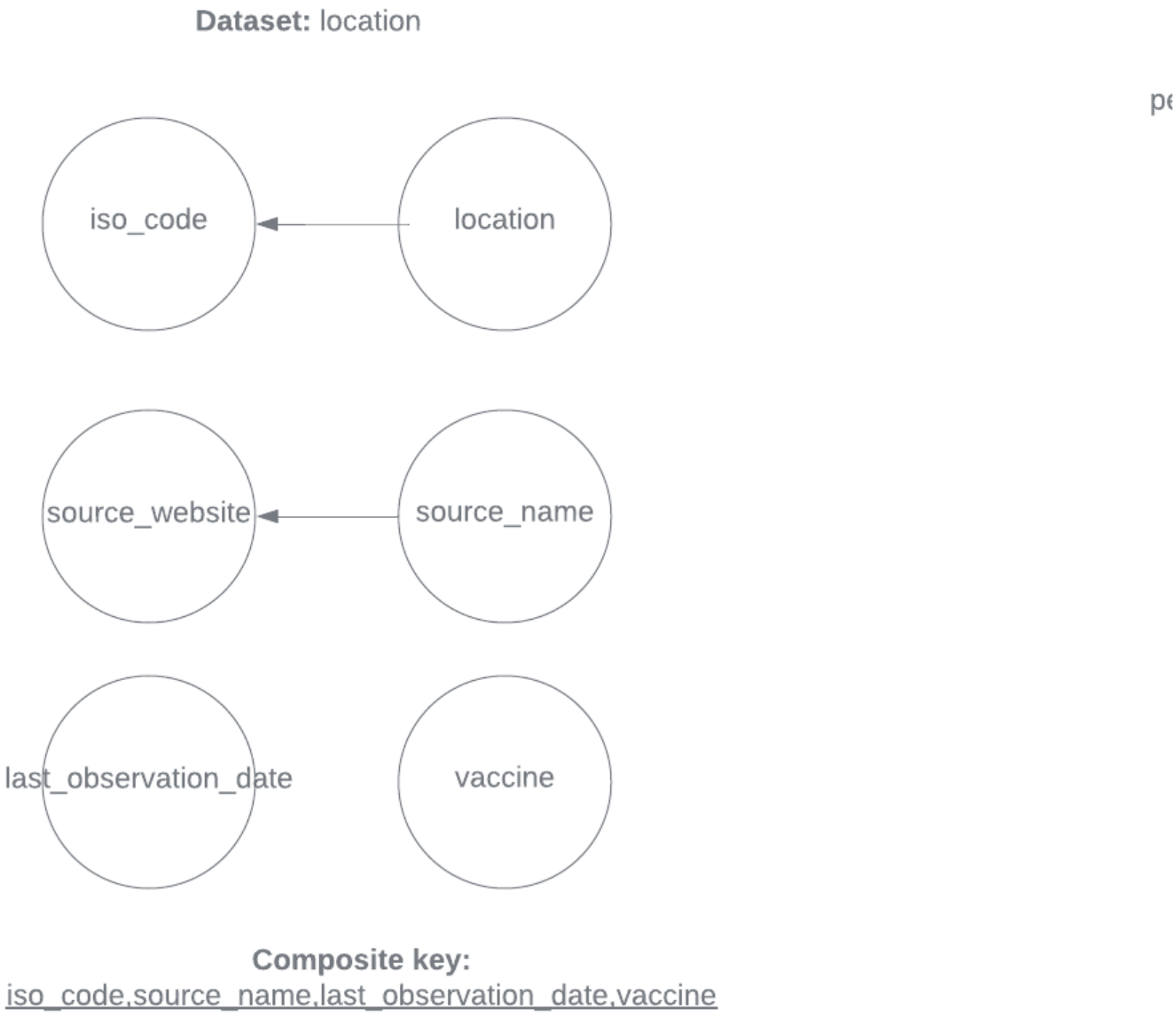
- location: name of the country (or region within a country).
- iso_code: ISO 3166-1 alpha-3 – three-letter country codes.
- vaccine: (it was a multi valued attribute and I seperate it): (was list of vaccines administered in the country up to the current date.) and its just one vaccine.
- last_observations_date: date of the last observation in our data.
- source_name: name of our source for data collection.
- source_website: web location of our source. It can be a standard URL if numbers are consistently reported on a given page; otherwise it will be the source for the last data point.
- date: date of the observation.
- total_distributed: cumulative counts of COVID-19 vaccine doses recorded as shipped in CDC's Vaccine Tracking System.
- people_vaccinated: total number of people who received at least one vaccine dose. If a person receives the first dose of a 2-dose vaccine, this metric goes up by 1. If they receive the second dose, the metric stays the same.
- people_fully_vaccinated_per_hundred: `people_vaccinated` per 100 people in the total population of the state.
- total_vaccinations_per_hundred: `total_vaccinations` per 100 people in the total population of the state.
- people_fully_vaccinated: total number of people who received all doses prescribed by the initial vaccination protocol. If a person receives the first dose of a 2-dose vaccine, this metric stays the same. If they receive the second dose, the metric goes up by 1.
- people_vaccinated_per_hundred: `people_fully_vaccinated` per 100 people in the total population of the state.
- total_distriburted_per_hundred: cumulative counts of COVID-19 vaccine doses recorded as shipped in CDC's Vaccine Tracking System per 100 people in the total population of the state.
- daily_vaccinations: new doses administered per day (7-day smoothed). For countries that don't report data on a daily basis, we assume that doses changed equally on a daily basis over any periods in which no data was reported. This produces a complete series of daily figures, which is then averaged over a rolling 7-day window.
- daily_vaccinations_per_million: `daily_vaccinations` per 1,000,000 people in the total population of the state.
- share_doses_used: share of vaccination doses administered among those recorded as shipped in CDC's Vaccine Tracking System.
- total_boosters: total number of COVID-19 vaccination booster doses administered (doses administered beyond the number prescribed by the initial vaccination protocol)
- total_boosters_per_hundred: `total_boosters` per 100 people in the total population.
- age_group: age group that are vaccinated.

Designing the Database

Functional dependecies, Primary/Composite key, entites, and attributes for each dataset

- The first step is to identify all the functional dependecies from each data set.
- Then, from these functional dependecies, we can identify the primary/composite key for each data set.
- Also, from the functional dependecies we can identify any new entities and attributes. Some of them are repeating entities from other datasets with new attributes.
- Since I must link a relationship between all datasets entites, making relationships before finding all the enities currently is not ideal. An overview of all relationships between entities will emerge once I have done with each dataset.

Dataset: location



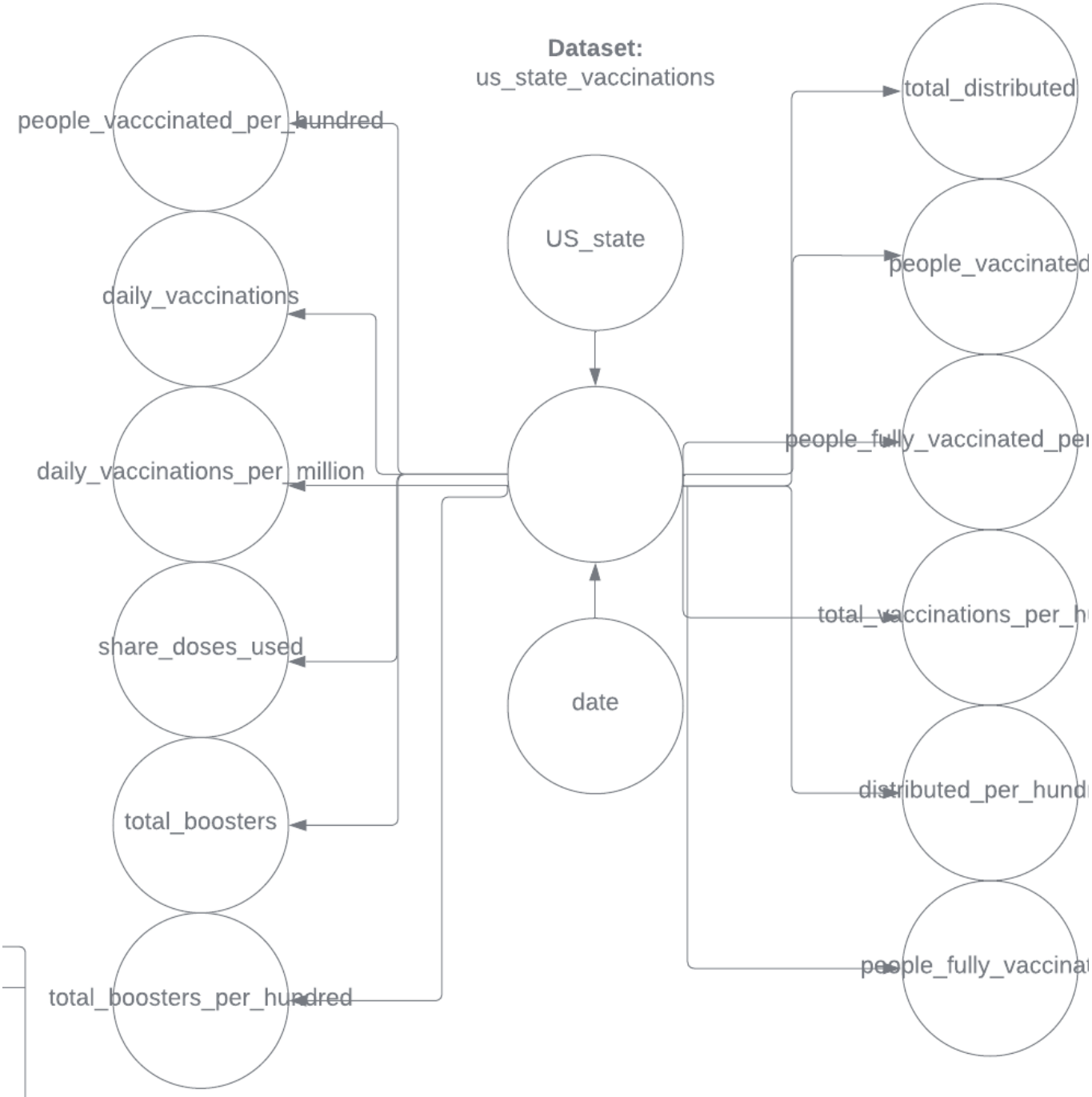
VaccineSource

source_name
source_website

Country

iso_code
location

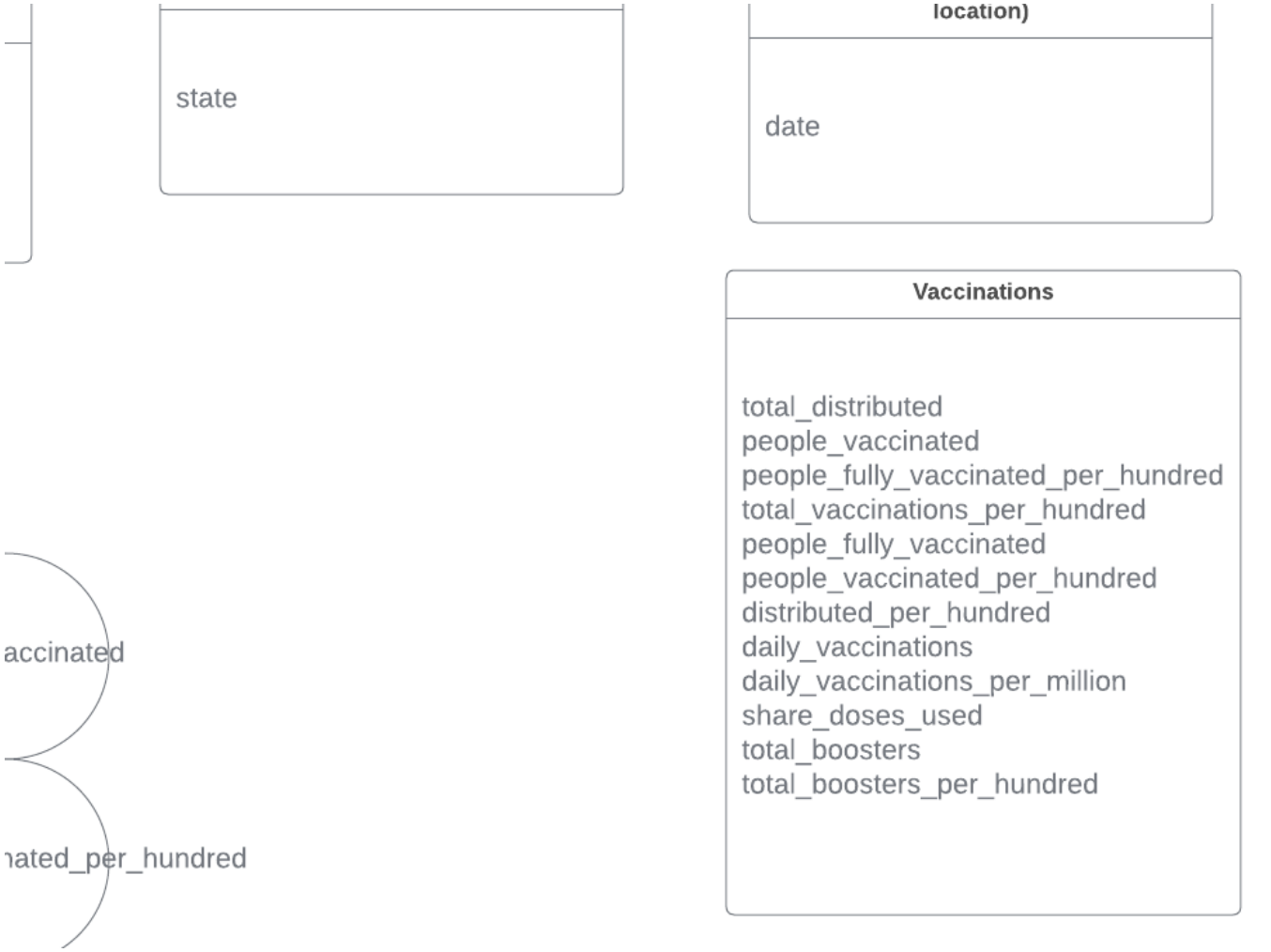
Dataset: us_state_vaccinations



Composite key:
date,US_state

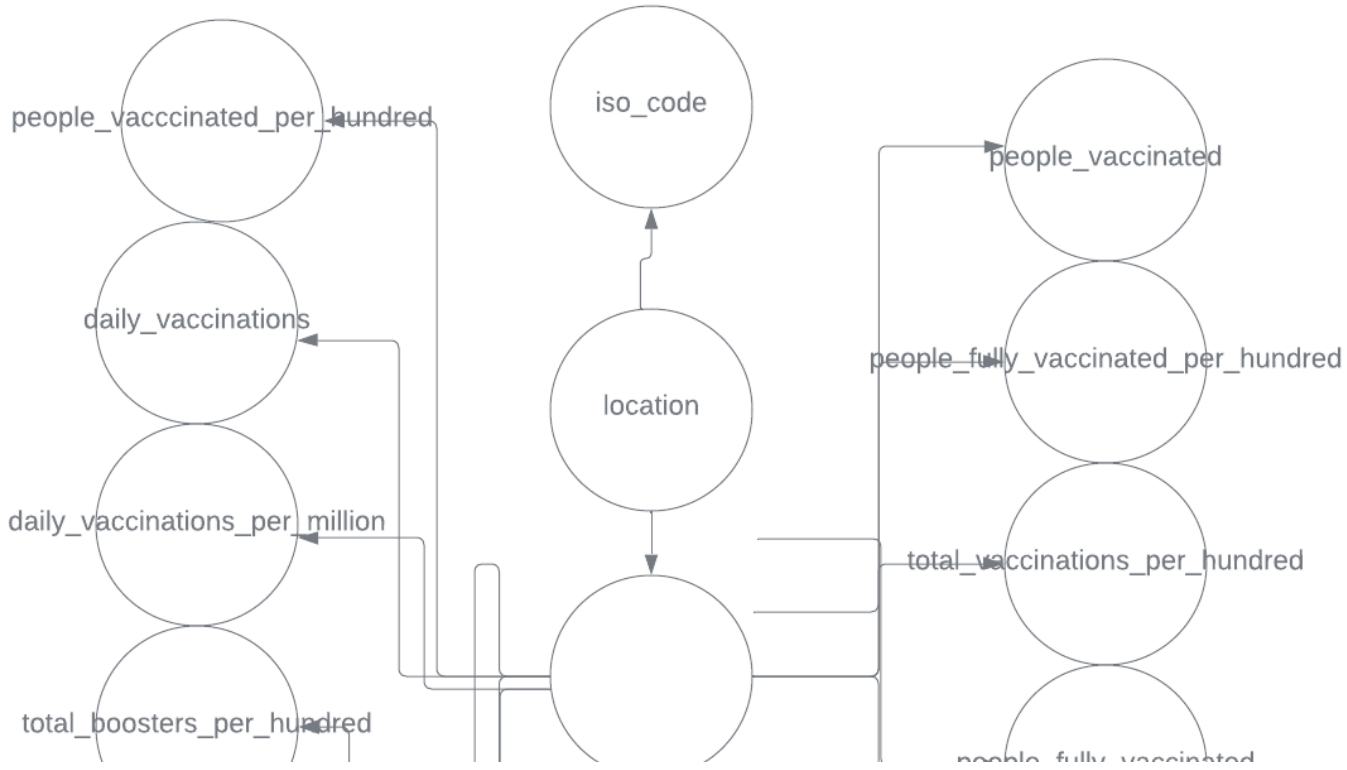
US_state

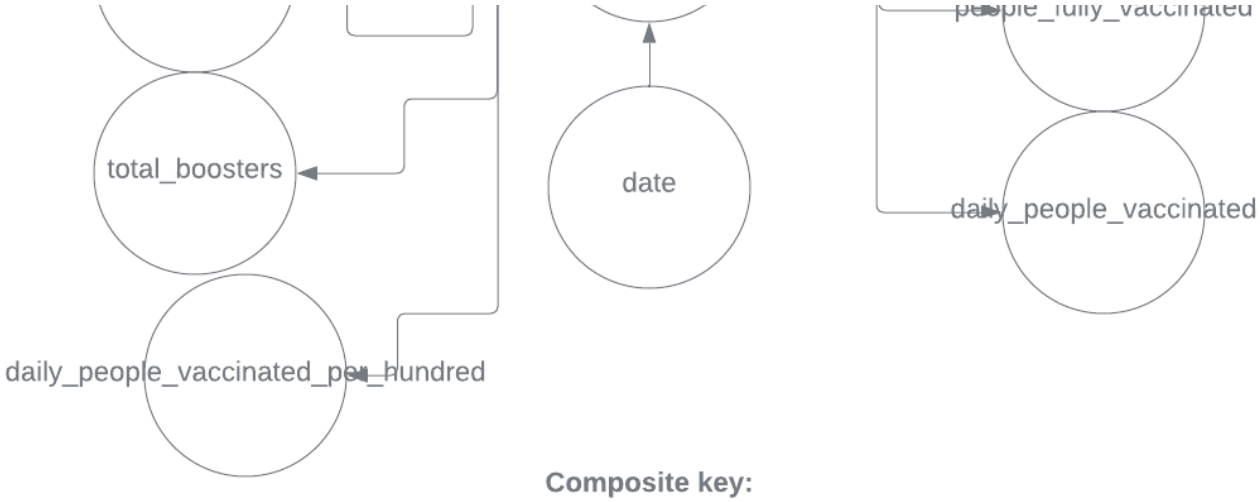
Date (repeated entity from dataset:



Dataset: vaccinations

Dataset: vaccinations





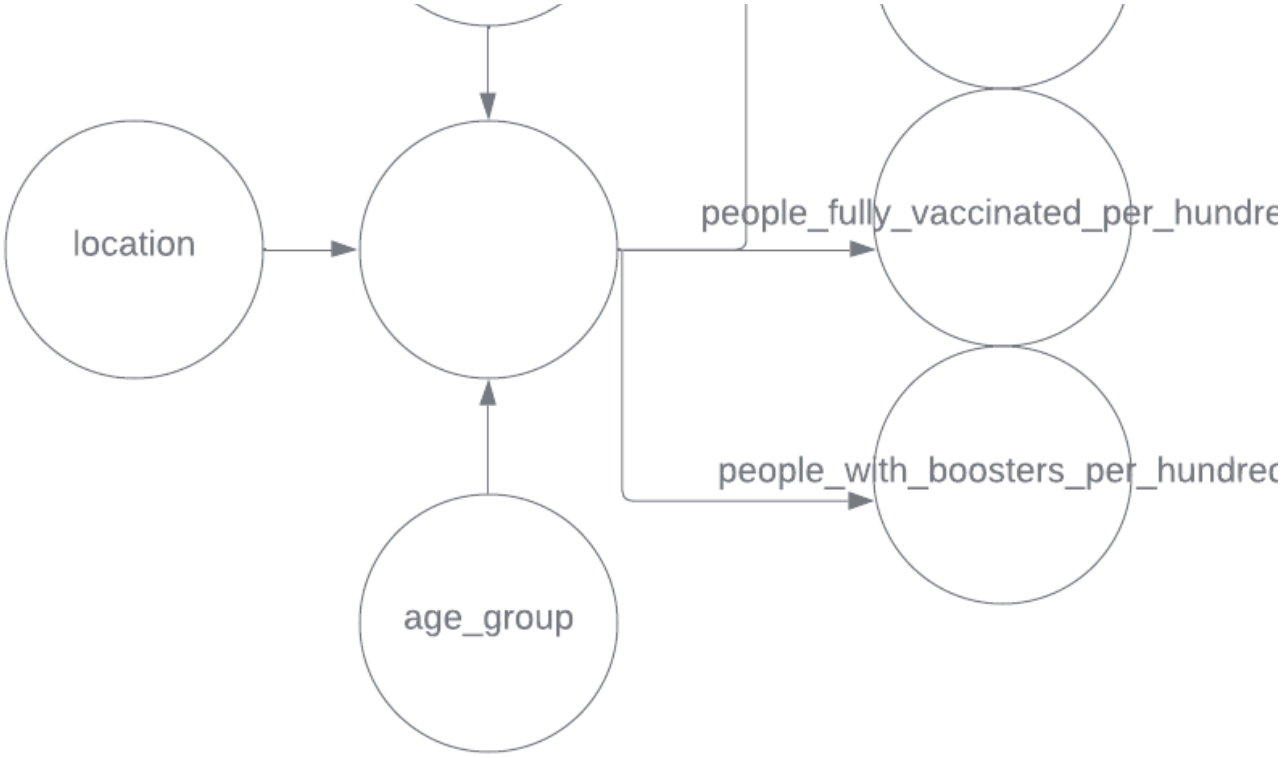
Composite key:
date,location

Country (repeated from dataset: location)
iso_code location

Vaccinations (repeated from dataset us_state_vaccinations but with some missing attributes and additional attributes
total_distributed **(missing)** people_vaccinated people_fully_vaccinated_per_hundred total_vaccinations_per_hundred people_fully_vaccinated people_vaccinated_per_hundred distributed_per_hundred **(missing)** daily_vaccinations daily_vaccinations_per_million share_doses_used **(missing)** total_boosters total_boosters_per_hundred daily_people_vaccinated **(additional attribute) which means its missing from data set: us_state_vaccinations.** daily_people_vaccinated_per_hundred **(additional)**

Dataset: vaccinations_by_age_group

Dataset:
vaccinations_by_age_group



Composite key:
location,date,age_group

Country (repeated from dataset: location)
iso_code location

Date (repeated from dataset: location)
date

Vaccinations (repeated from dataset vaccinations but with some missing attributes and one additional attribute)
total_distributed **(missing)** people_vaccinated **(missing)** people_fully_vaccinated_per_hundred total_vaccinations_per_hundred **(missing)** people_fully_vaccinated **(missing)** people_vaccinated_per_hundred distributed_per_hundred **(missing)** daily_vaccinations **(missing)** daily_vaccinations_per_million **(missing)** share_doses_used **(missing)** total_boosters **(missing)** total_boosters_per_hundred **(missing)** daily_people_vaccinated **(missing)** daily_people_vaccinated_per_hundred **(missing)**

(missing)
people_with_boosters_per_hundred
(additional)

Dataset: vaccinations_by_manufacturer

dataset:
vaccinations_by_manufacturer

location

date

vaccine

Composite key:
location,date,vaccine

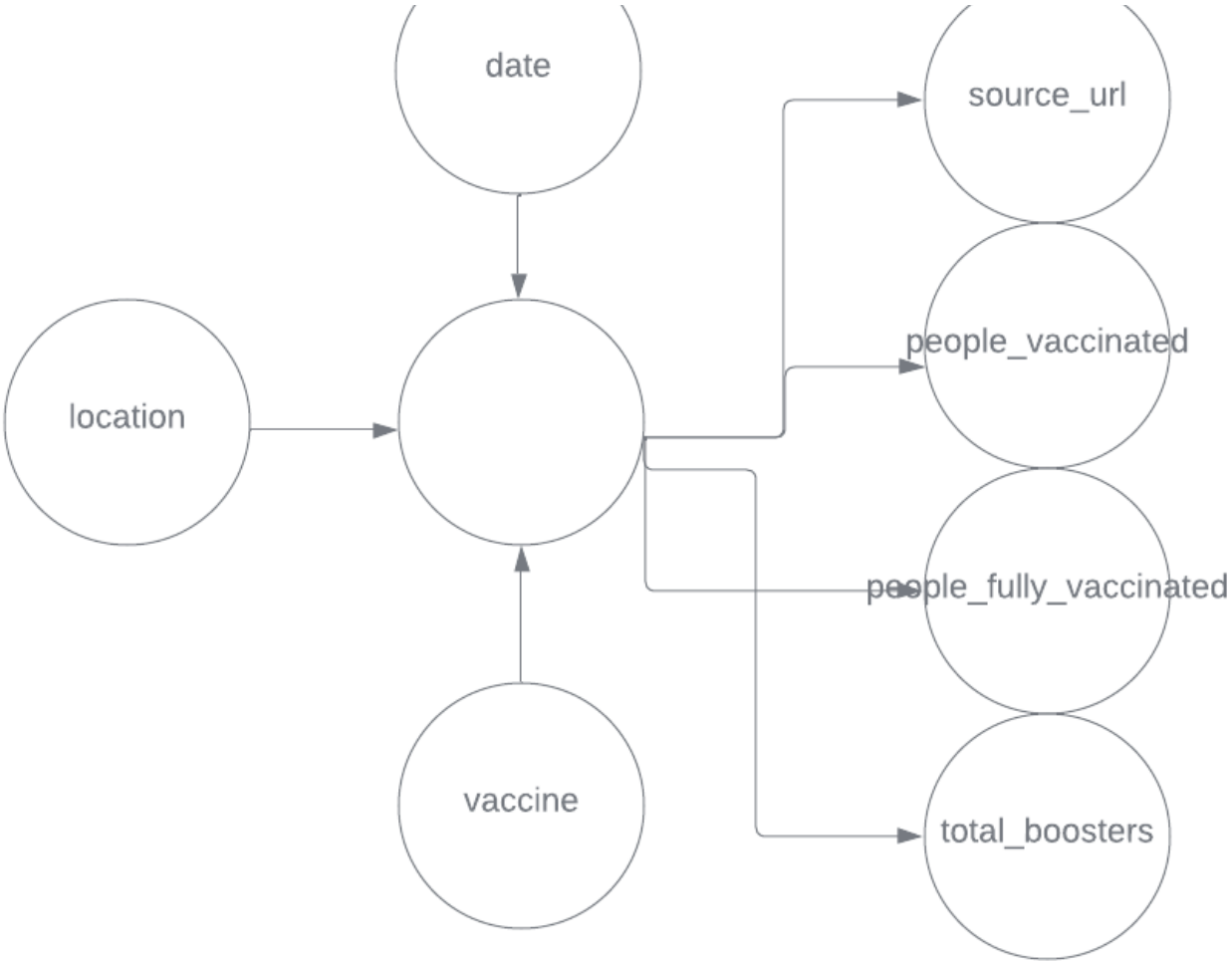
Country (repeated from dataset: location)
iso_code **(missing)** location

Date (repeated from dataset: location)
date

Vaccine (repeated from dataset: location)
vaccine

Dataset: Australia,Germany,Italy,US

dataset:
Australia,Germany,Italy,US



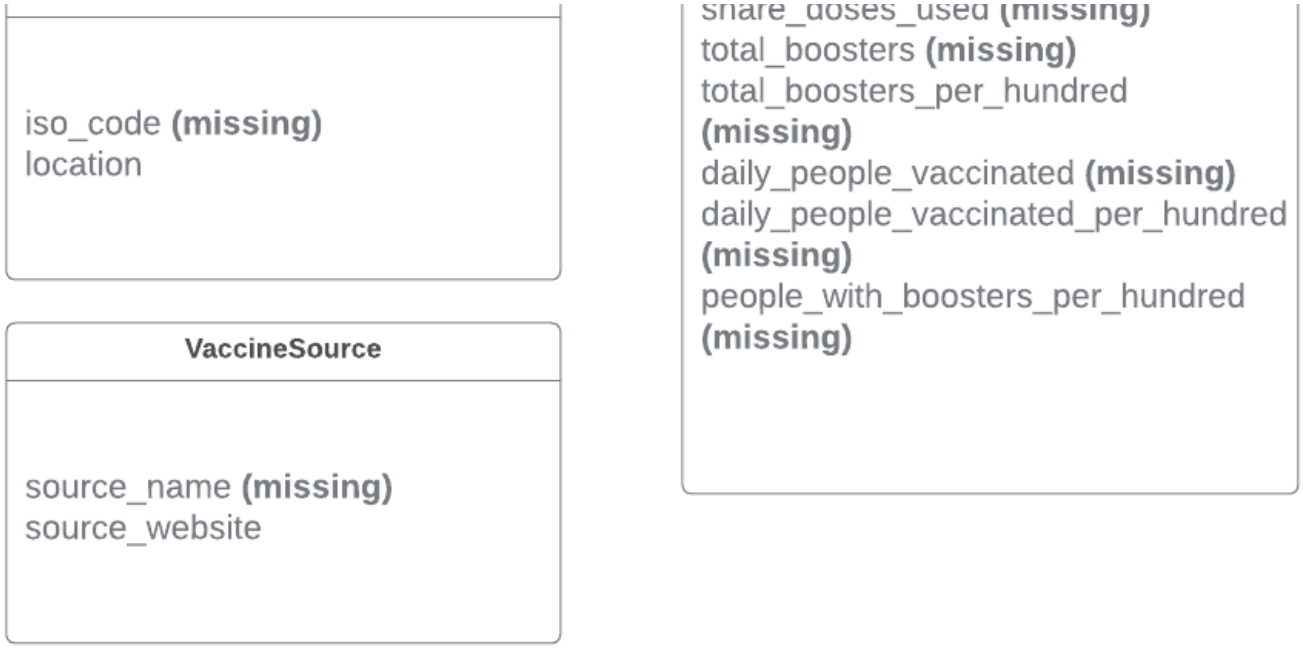
Composite key:
date,location,vaccine

Date (repeated from dataset: location)
date

Vaccine (repeated from dataset: location)
vaccine

Country (repeated from dataset: location)

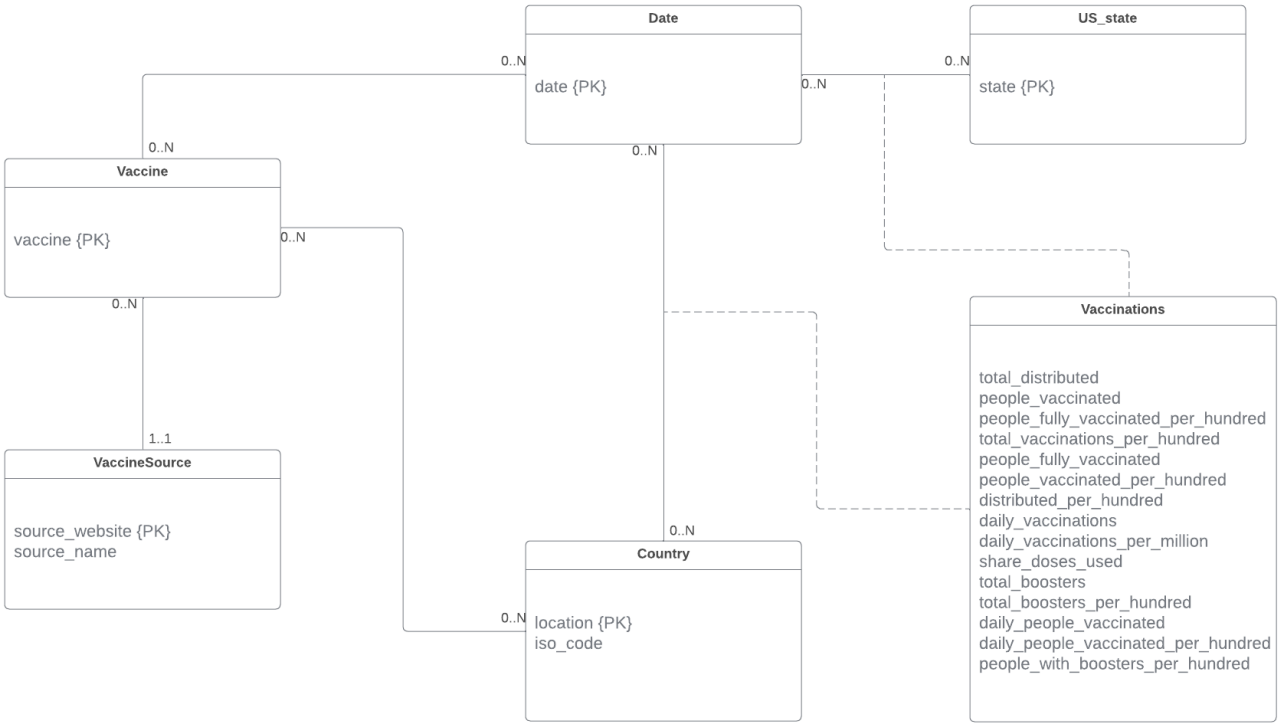
Vaccinations (repeated from dataset vaccinataions_by_age_group but with some missing attributes)
total_distributed **(missing)** people_vaccinated **(missing)** people_fully_vaccinated_per_hundred **(missing)** total_vaccinations_per_hundred **(missing)** people_fully_vaccinated **(missing)** people_vaccinated_per_hundred **(missing)** distributed_per_hundred **(missing)** daily_vaccinations **(missing)** daily_vaccinations_per_million **(missing)** share_doses_used **(missing)**



Unique entites across all datasets

Relationships between entites

- Now that I have all the unique entities and attributes from all datasets, the next step is to identify the relationships between the entites.



9-Steps approach

Step 1: Mapping strong entites

- Strong entites are those that can stand alone.
- For each strong entity, create a relation that include all the simple attributes of that entity.

Vaccine(vaccine)
Date(date)
Country(location,iso_code)
US_state(state)

Step 2: Mapping weak entites

- Weak entites are those that can not stand alone.
- For each weak entity, create a relation that include all the simple attributes of that entity and copy the primary key of the owner and make it a foreign and primary key inside the weak entity.

VaccineSource(vaccine*,source_name,source_website)

Step 3: One-to-many binary relationships

- For each one-to-many binary relationship, the entity on the "one side" of the relationship is designated as the parent entity and the entity on the "many side" is designated as the child entity.
- Copy the primary key of the parent "one side" entity to the child "many side" entity.

One-to-many

Vaccine(vaccine,source_website*)

Step 4: One-to-one binary relationships

- There are no one-to-one binary relationships in the database ER diagram.

Step 5: One-to-one recursive relationships

- There are no one-to-one recursive relationships in the database ER diagram.

Step 6: Superclass/subclass relationships

- There are no such relationships in the database ER diagram.

Step 7: Many-to-many relationships

- For each many-to-many relationships, create a relation to represent the relationships and include any attributes that are part of the relationship. We post a copy of the primary key attribute(s) of the entities that participate in the relationship into the new relation, to act as foreign keys. One or both of these foreign keys will also form the primary key of the new relation, possibly in combination with one or more of the attributes of the relationship.

USVaccinations(date*,state*,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,daily_vaccinations_per_million,
share_doses_used,total_boosters,total_boosters_per_hundred,daily_people_vaccinated,daily_people_vaccinated_per_hundred
people_with_boosters_per_hundred

CountryVaccinations(date*,location*,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,daily_vaccinations_per_million,
share_doses_used,total_boosters,total_boosters_per_hundred,daily_people_vaccinated,daily_people_vaccinated_per_hundred
people_with_boosters_per_hundred

VaccineDate(vaccine*,date*)

VaccineCountry(vaccine*,location*)

-

Step 8: Complex relationships

- There are no such relationships in the database ER diagram.

Step 9: Multi-valued attributes

- There are no multi-valued attributes.

Final database schema (before normalization)

Country(location,iso_code)
US_state(state)
Date(date)

USVaccinations(date*,state*,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,daily_vaccinations_per_million,
share_doses_used,total_boosters,total_boosters_per_hundred,daily_people_vaccinated,daily_people_vaccinated_per_hundred
people_with_boosters_per_hundred

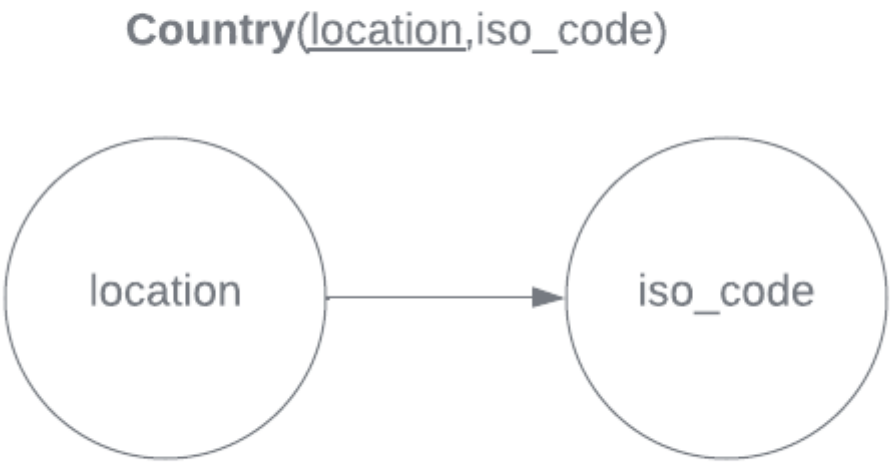
CountryVaccinations(date*,location*,total_distributed,people_vaccinated,
people_fully_vaccinated_per_hundred,total_vaccinations_per_hundred,people_fully_vaccinated,
people_vaccinated_per_hundred,distributed_per_hundred,daily_vaccinations,daily_vaccinations_per_million,
share_doses_used,total_boosters,total_boosters_per_hundred,daily_people_vaccinated,daily_people_vaccinated_per_hundred
people_with_boosters_per_hundred

Vaccine(vaccine,source_website*)
VaccineSource(vaccine*,source_name,source_website)
VaccineDate(vaccine*,date*)
VaccineCountry(vaccine*,location*)

Final database schema (after normalization)

- Everything must be dependent on the primary key so that the relation is in 1NF
- Everything must be dependent on the primary key so that the relation is in 1NF and also everything must be dependent on the whole key (composite key) to be in 2NF.
- Already in 2NF and there's no transitive dependencies meaning: an attribute that's not a key that depends on an attribute that's also not a key.

Relation Country



- Relation already in 3NF.

Relation US_state

- Relation have only one attribute so it is already in 3NF.

Relation Date

- Relation have only one attribute so it is already in 3NF.

Relation USVaccinations

- Previous functional dependecies is given in the dataset: us_state_vaccinations.
- Everything depends on a primary key (1NF pass).
- Everything depends on a composite key (2NF pass).
- There are no transitive dependecies (3NF pass).

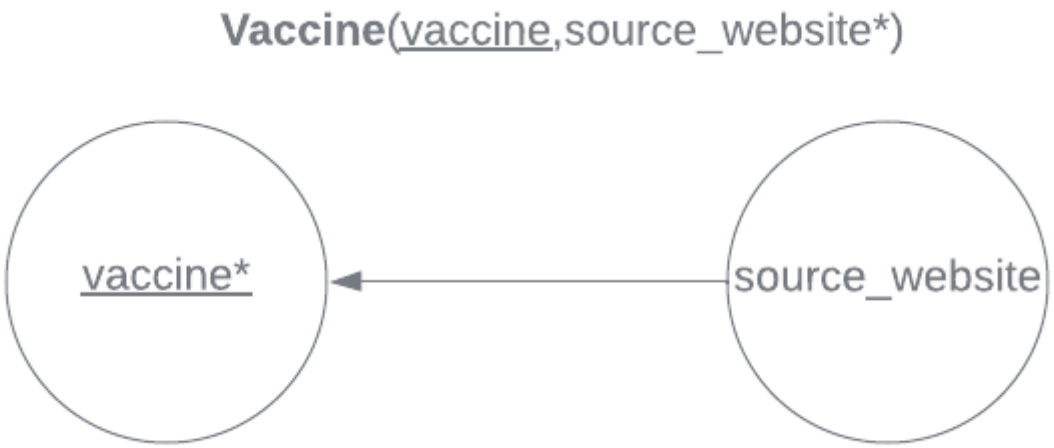
Relation CountryVaccinations

- Previous functional dependecies is given in the dataset: location.
- Everything depends on a primary key (1NF pass).
- Everything depends on a composite key (2NF pass).
- There are no transitive dependecies (3NF pass).

Relation Vaccine

- There were previous errors in finding the functional dependecies for the dataset location. Hence, this relation have the primary key wrong.

- The functional depencies should be source_website -> vaccine.



The new relation becomes:

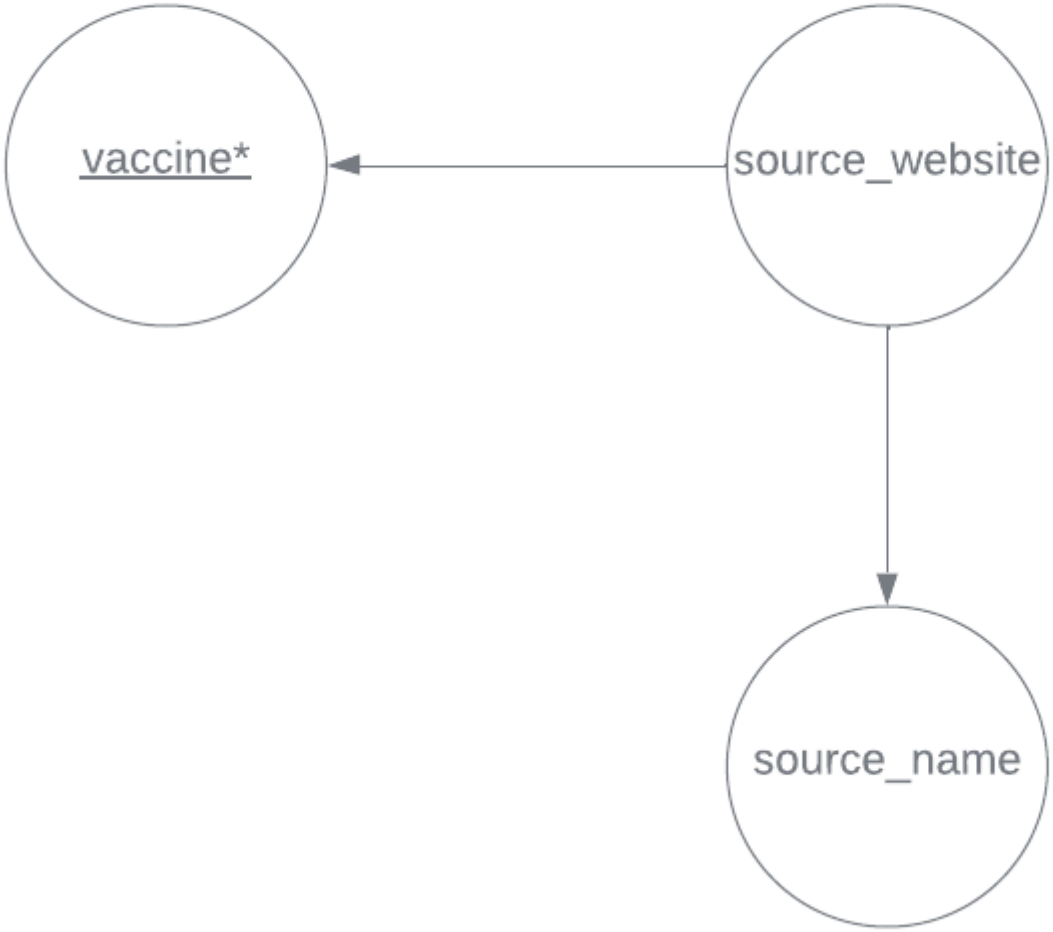
Vaccine(source_website,vaccine)

- The relation now is in 3NF.

Relation VaccineSource

- Due to the functional dependecy error in relation Vaccine, this relation also have the same error.

VaccineSource(vaccine*,source_name,source_website)



The new relation becoms:

VaccineSource(source_website,vaccine,source_name)

- There's also no need for relation Vaccine because it contain data redundency.

Relation VaccineDate

- Relation is in 3NF.

VaccineDate(vaccine*,date*)

Relation VaccineCountry

- Same as the relation VaccineDate. Therefore, the relation is in 3NF.

The final Database schema

Country(location,iso_code)
US_state(state)

Date(date)

USVaccinations(date*,state*,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,daily_vaccinations_per_million,
share_doses_used,total_boosters,total_boosters_per_hundred,daily_people_vaccinated,daily_people_vaccinated_per_hundred
people_with_boosters_per_hundred

CountryVaccinations(date*,location*,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,daily_vaccinations_per_million,
share_doses_used,total_boosters,total_boosters_per_hundred,daily_people_vaccinated,daily_people_vaccinated_per_hundred
people_with_boosters_per_hundred

VaccineSource(source_website,vaccine,source_name)

VaccineDate(vaccine*,date*)

VaccineCountry(vaccine*,location*)