# Gaining an understanding of the nature of the business I began by sitting down and speaking with the group that hired me. I'm hoping to find answers to the following questions: 1. What is the nature of the business? 2. What data is currently available? 3. What kinds of data-related questions do they want answered? 4. In a commercial context, how do the many data sources interact with one another?

# From discussions with stakeholders, SMEs, and existing data collection groups, I discovered the following logical linkages between the data; • Shows feature performances.

# • Locations for performances exist.

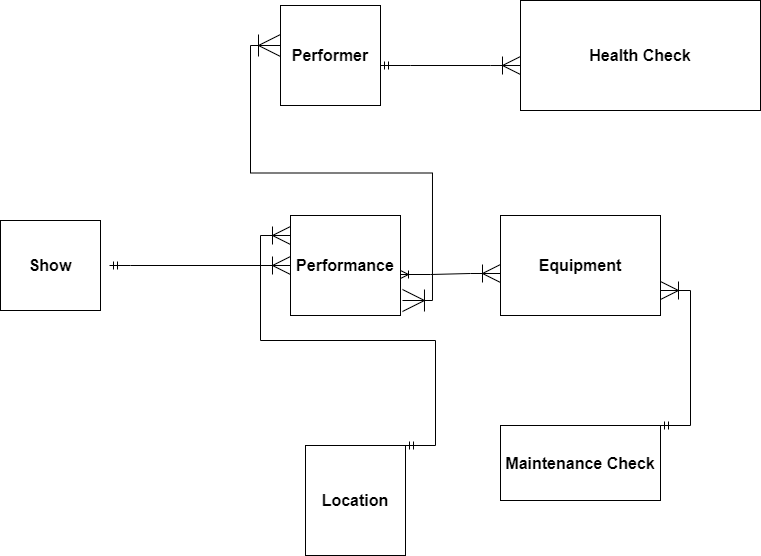
# • Performances have performers.

# • Performances require equipment.

# • Performers undergo medical examinations.

# • Equipment undergoes maintenance checks.

**Designing a logical model based on Entities and Relationships from data collected.**The first step is to start thinking about the data in terms of entities and relationships between them. This will aid considerably with developing a physical data model to persist the data.



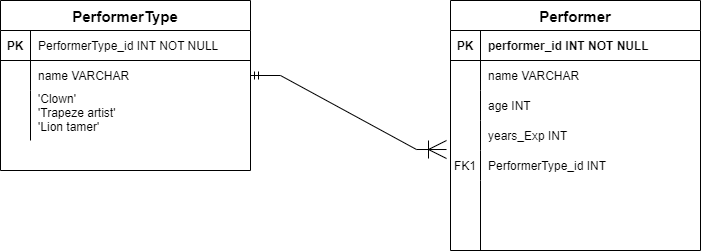
The arrows signal the following relationships between the entities:

1. A show can have multiple performances, but a performance will only correspond to one show. (One to many)
2. A location can be used for many performances, but a performance will only have one location. (one to many)
3. A performance will have many performers, and a given performer can be part of many performances. (many to many)
4. A performer will have many health checks, but a given health check only pertains to one performer. (one to many)
5. A performance will have many pieces of equipment, and a given piece of equipment can be used across many performances. (many to many)
6. A given piece of equipment will have many maintenance checks, but a single maintenance check will only pertain to one piece of equipment(one to many)

# Building a Physical Data Model

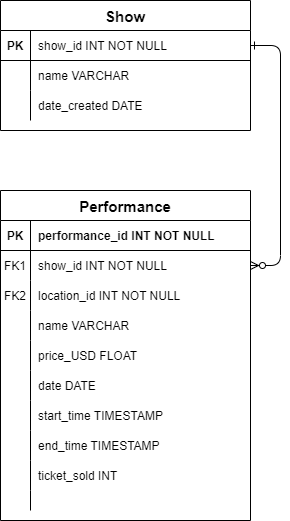
Now that I've established a solid foundation for a logical model of the entities' relationships, it's time to consider how to structure these as tables with attributes and foreign keys between the tables.  
At this point, I realized that the raw data is not in an ideal format for direct transfer to the unified data model. I had to perform certain ETL processes to transform the raw data.  
It's also worth mentioning that I utilized normalization techniques on several of the entities to reduce value redundancy (and so save space) while also ensuring consistency across records.  
Taking for an example few of the ERD diagram with attributes and relationships below:

1. Normalize out reusable labels across rows.



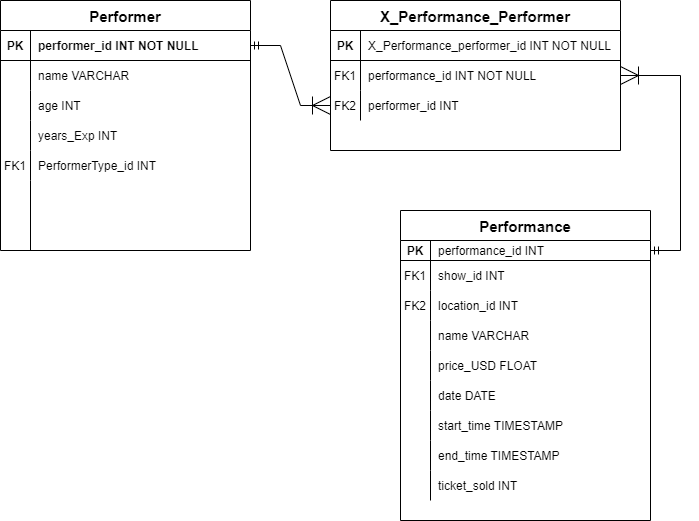
In order to avoid repeating labels over across different performer records, I considered normalizing the column out to a dedicated table to keep a label consistent across all performers that have that type. It also saves space over time avoiding the same label string repeated over and over across performer records. The PerformerType\_id column in the Performer table links a given PerformerType to that performer.

1. **Shows can have many performances**

****

The design above maintains organized properties at the show level while also tracking show instances (performances). Each performance has a lot of structural details that are unique to that performance. Location also has a foreign key location id. The goal is to store repetitive information for locations that are referenced across multiple performances.

1. **Performances can have many performers, and vice versa**



The Join X\_Performance\_Preformer made it possible to model this as a many-to-many relationship. In order to see which performers were part of a given performance instance, joining the X-table with a given performance\_id and then join back to performers table make this possible. Similarly, if stockholders want to see all the performances a given performer has been a part of, this can be achieved by joining the X\_Performance\_Preformer with a given performer\_id and then joining back to the performances table.

This relationship is very similar for Equipment and how it relates to performances.

Putting the components together produces the following ERD diagram for the unified data model with structured attributes:

****

**The data model is in Third normal form and enables the business to start answering questions that span across the different data channels.**

**With this data model in place, my client would be able to answer questions like:**

1. How many days between ‘fine’ and ‘broken’ a piece of equipment goes on the average.
2. How performances are doing in ticket sales relative to the location population.
3. Which performances are accelerating for tickets sold over recent months?