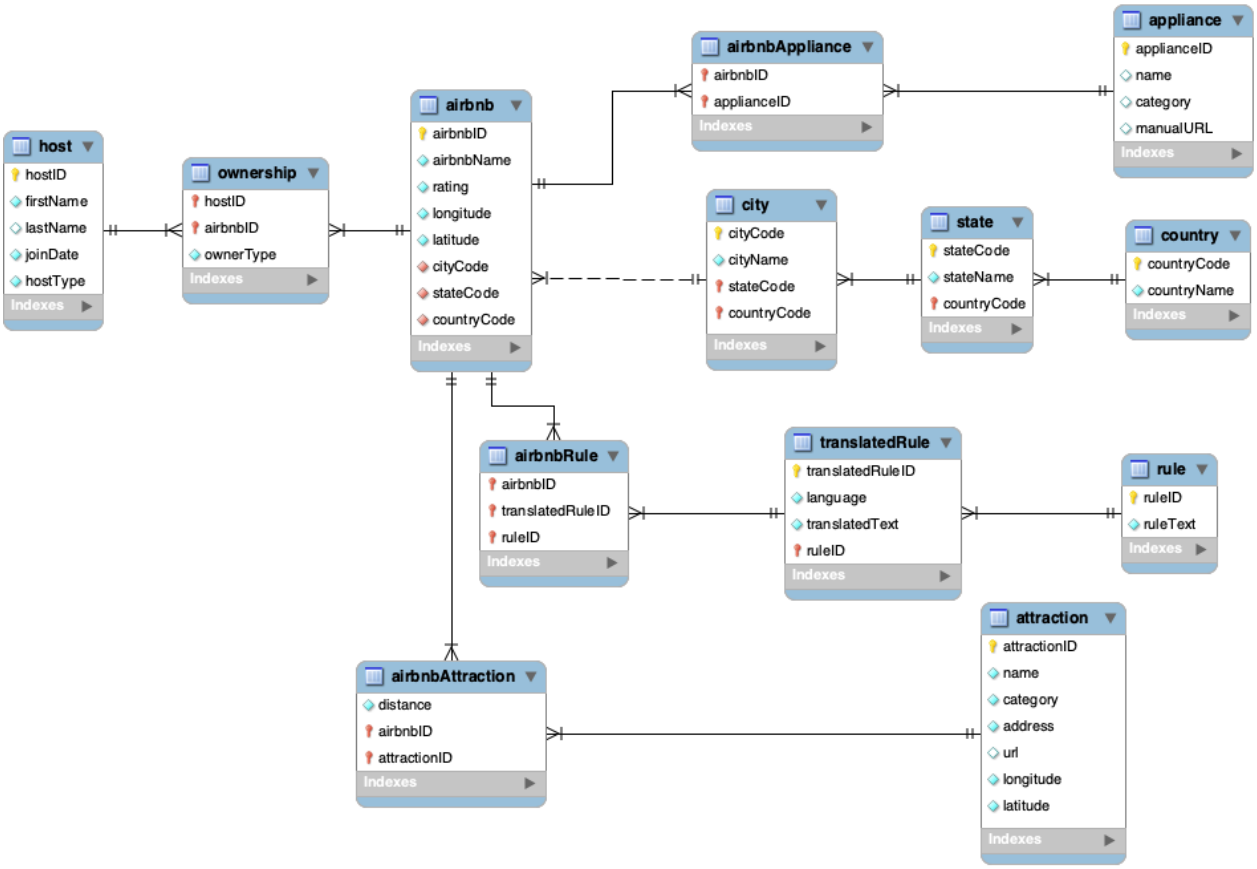
AirBnB - Data Modeling and SQL Queries

Description:

AirBnB hosts provide documentation along with their provided units to users. These documents provide various information such as a unit's appliances, a unit's rules, or even what local attractions are close to the AirBnB units. To organize these documentations for different uses, a database model was designed, and a small amount of data was included for demonstration purpose. SQL queries were performed against the database to extract relevant information. For example, the distance between a specific Airbnb unit and the nearest museum can be found using the database.

Software: MySQL Workbench

Data Model:



Queries:

#1: An Airbnb host wants to check important rules that appear in every airbnb and show all available translated versions for those rules. Language, ruleID, and TranslatedRuleid are also shown, results are ordered by language.

Execute:

```
> SELECT
   language, ruleid, TranslatedRuleid, TranslationText
FROM
   TranslatedRule
WHERE
   ruleid IN (SELECT
         rule.ruleid
      FROM
         rule
      WHERE
         NOT EXISTS( SELECT
               *
            FROM
               airbnb
            WHERE
               NOT EXISTS( SELECT
                     *
                  FROM
                     airbnbRule,
                     TranslatedRule
                  WHERE
                     airbnb.airbnbID = airbnbRule.airbnbID
                        AND TranslatedRule.ruleid = rule.ruleid
                        AND airbnbRule.TranslatedRuleid = TranslatedRule.TranslatedRuleid)))
ORDER BY language;
```

Output:

```
+ ------------ + ---------- + ------------------ + ------------------- +
| language     | ruleid     | TranslatedRuleid   | TranslationText     |
+ ------------ + ---------- + ------------------ + ------------------- +
| english      | 81101      | 61101              | No pets allowed     |
| english      | 81102      | 61102              | No smoking          |
| german       | 81101      | 61109              | Keine Haustiere erlaubt. |
| german       | 81102      | 61110              | Rauchen im Apartment ist verboten. |
| spanish      | 81101      | 61105              | No se admiten animales de compañía |
| spanish      | 81102      | 61106              | No fumar            |
+ ------------ + ---------- + ------------------ + ------------------- +
```

#2: The rules that have the word “pet” or “smoking” are common rules, it may be more user friendly to have these rules in several different languages. So, the numbers of their translated rules are counted and presented together with the ruleID and ruleText.

Execute:

```
> SELECT
   rule.ruleID,
   ruleText,
   COUNT(TranslatedRuleid) AS TranslationCount
FROM
```

```
    TranslatedRule,
    rule
WHERE
    rule.ruleID = TranslatedRule.ruleId
        AND rule.ruleID IN (SELECT
            rule.ruleID
        FROM
            rule
        WHERE
            ruleText REGEXP 'pet|smoking')
GROUP BY rule.ruleID;
```

Output:

```
+ ----------- + ------------- + --------------------- +
| ruleID      | ruleText      | TranslationCount      |
+ ----------- + ------------- + --------------------- +
| 81101       | No pets allowed | 3                   |
| 81102       | No smoking    | 3                     |
+ ----------- + ------------- + --------------------- +
```

#3: An Airbnb user is curious about the average ratings of airbnb listings in different U.S. cities, the ratings are arranged in descending order.

Execute:

```
> SELECT
    airbnb.cityCode, ROUND(AVG(rating), 2) AS meanRating
FROM
    airbnb
        JOIN
    country ON airbnb.countryCode = country.countryCode
WHERE
    countryName = 'United States'
GROUP BY airbnb.stateCode, airbnb.cityCode
ORDER BY meanrating DESC;
```

Output:

```
+ ------------- + ---------------- +
| cityCode      | meanRating       |
+ ------------- + ---------------- +
| ATH           | 4.83             |
| NYC           | 2.75             |
+ ------------- + ---------------- +
```

#4: The host “Daniel” wants to check if there exists any AirBnB that he owns has rates above the average rate 3.0 using EXISTS.

Execute:

```
> SELECT
    firstName, lastName, airbnbName, rating
FROM
    airbnb
        JOIN
    ownership ON airbnb.airbnbID = ownership.airbnbID
        JOIN
    host ON ownership.hostID = host.hostID
```

```
WHERE
   EXISTS( SELECT
         *
      FROM
         host
      WHERE
         host.hostID = ownership.hostID
            AND ownership.airbnbID = airbnb.airbnbID
            AND firstName = 'Daniel'
            AND rating > 3.0);
```

Output:

```
+ -------------- + ------------- + --------------- + ----------- +
| firstName      | lastName      | airbnbName      | rating      |
+ -------------- + ------------- + --------------- + ----------- +
| Daniel         | Doobey        | DelightfulDen   | 3.50        |
+ -------------- + ------------- + --------------- + ----------- +
```

#5: An AirBnB hosts wants to check if any of AirBnBs in Athens, GA has historic attractions nearby, if so, what are the attraction names, Airbnb names and their corresponding distances?

Execute:

```
> SELECT category, name, distance, airbnbName, cityCode
FROM attraction
JOIN airbnbAttraction
ON attraction.attractionID = airbnbAttraction.attractionID
JOIN airbnb
ON airbnbAttraction.airbnbID = airbnb.airbnbID
GROUP BY category, name, distance, airbnbName, cityCode
HAVING category = "Historic"
AND cityCode = (select cityCode from city where cityName = “Athens”
AND stateCode = (select stateCode from state where stateName = “Georgia”
AND countryCode = (select countryCode from country where countryName = “US”))
AND countryCode = (select countryCode from country where countryName = “US”));
```

Output:

```
+ ------------- + --------- + ------------- + ---------------- + ------------- +
| category      | name      | distance      | airbnbName       | cityCode      |
+ ------------- + --------- + ------------- + ---------------- + ------------- +
| Historic      | Camak House | 5             | BatCaveNY        | ATH           |
| Historic      | Camak House | 12            | SuperHome        | ATH           |
+ ------------- + --------- + ------------- + ---------------- + ------------- +
```

#6 A airbnb customer wants to know the distances between his/her interested airbnb listings (BatCaveNY , SuperHome, RunAwayFromYourTroubles) and local Museum, Aqurium or Market related attractions if any, the distances are arranged in ascending order.

Execute:

```
> SELECT
   airbnbName, name, distance
FROM
   attraction
      JOIN
   airbnbAttraction ON attraction.attractionID = airbnbAttraction.attractionID
```

```
    JOIN
  airbnb ON airbnbAttraction.airbnbID = airbnb.airbnbID
WHERE
  airbnbName IN ('BatCaveNY' , 'SuperHome', 'RunAwayFromYourTroubles')
    AND name REGEXP 'Museum|Aqurium|Market'
ORDER BY distance;
```

Output:

```
+ --------------- + --------- + ------------ +
| airbnbName      | name      | distance     |
+ --------------- + --------- + ------------ +
| BatCaveNY       | Museum of Modern Art | 5            |
| SuperHome       | Museum of Modern Art | 13           |
| RunAwayFromYourTroubles | Namdaemun Market | 15           |
+ --------------- + --------- + ------------ +
```

#7 A airbnb customer wants to find the AirBnB with the most attractions within 5 miles.

Execute:

```
> SELECT
  airbnbName, COUNT(airbnb.airbnbID) AS numberOfAttractions
FROM
  airbnb
    JOIN
  airbnbAttraction ON airbnbAttraction.airbnbID = airbnb.airbnbID
    JOIN
  attraction ON attraction.attractionID = airbnbAttraction.attractionID
WHERE
  distance <= 5
GROUP BY airbnb.airbnbID
ORDER BY COUNT(airbnb.airbnbID) DESC LIMIT 1;
```

Output:

```
+ --------------- + ------------------------ +
| airbnbName          | numberOfAttractions           |
+ --------------- + ------------------------ +
| BatCaveNY           | 3                        |
+ --------------- + ------------------------ +
```

#8 A airbnb customer wants to find which AirBnBs have ratings greater than 3.0 and more than 2 appliances provided.

Execute:

```
> SELECT
  airbnbName,
  rating,
  COUNT(appliance.applianceID) AS numberOfAppliances
FROM
  airbnb
  JOIN
  airbnbAppliance ON airbnbAppliance.airbnbID = airbnb.airbnbID
  JOIN
  appliance ON appliance.applianceID = airbnbAppliance.applianceID
GROUP BY airbnb.airbnbID HAVING rating > 3
   AND COUNT(appliance.applianceID) > 2;
```

Output:

airbnbName	rating	numberOfAppliances
CozyCorner	4.50	4
DelightfulDen	3.50	3
BatCaveNY	5.00	4
SuperHome	5.00	3
RunAwayFromYourTroubles	4.50	6
DragonDen	4.30	4
MoonlightVilla	4.00	3

#9 An airbnb customer wants to find AirBnBs with ratings above average of 3.1. He/she also wants to see how much percentages the corresponding ratings are higher than the average.

Execute:

```
> SELECT airbnbName, (((rating-3.1)/3.1)*100) AS higher_Rating_Percent, rating
FROM airbnb
WHERE rating>3.1
ORDER BY higher_Rating_Percent DESC;
```

Output:

airbnbName	higher_Rating_Percent	rating
BatCaveNY	61.290323	5.00
SuperHome	61.290323	5.00
CozyCorner	45.161290	4.50
RunAwayFromYourTroubles	45.161290	4.50
DragonDen	38.709677	4.30
MoonlightVilla	29.032258	4.00
DelightfulDen	12.903226	3.50
CheeseCastle	12.903226	3.50

Condensed SQL Query Feature Matrix

Query / SQL Feature	1	2	3	4	5	6	7	8	9
Multiple Table Join			x		x	x	x	x	
Subquery	x	x		x	x				
Correlated Subquery				x					
Group by		x	x				x	x	
Group by with Having					x			x	
Order by	x		x			x	x		x
Divide	x								
In or Not In	x	x				x			
Built In Function/ Calculated Field		x	x				x	x	x
Regexp		x				x			
Exists or Not Exist	x			x					