Part 1: Yelp Dataset Profiling and Understanding

1. Profile the data by finding the total number of records for each of the tables below:

i. Attribute table = 10000

```
               SELECT COUNT(*)
               FROM attribute
```

ii. Business table = 10000

```
               SELECT COUNT(*)
               FROM business
```

iii. Category table = 10000

```
               SELECT COUNT(*)
               FROM category
```

iv. Checkin table = 10000

```
               SELECT COUNT(*)
               FROM checkin
```

v. elite_years table = 10000

```
               SELECT COUNT(*)
               FROM elite_years
```

vi. friend table = 10000

```
               SELECT COUNT(*)
               FROM friend
```

vii. hours table = 10000

```
               SELECT COUNT(*)
               FROM hours
```

viii. photo table = 10000

```
               SELECT COUNT(*)
               FROM photo
```

ix. review table = 10000

```
               SELECT COUNT(*)
               FROM review
```

x. tip table = 10000

```
               SELECT COUNT(*)
               FROM tip
```

xi. user table = 10000

```
               SELECT COUNT(*)
               FROM user
```

2. Find the total distinct records by either the foreign key or primary key for each table. If two foreign keys are listed in the table, please specify which foreign key.

i. Business = 10000

```
               SELECT COUNT(distinct id)
               FROM business
```

ii. Hours = 1562

```
SELECT Count(distinct id)
FROM business
```

iii. Category = 2643

```
SELECT Count(distinct business_id)
FROM category
```

iv. Attribute =1115

```
SELECT Count(distinct business_id)
FROM attribute
```

v. Review = 10000(id), 8090 (business_id), 9581(user_id)

```
SELECT Count(distinct id)
FROM review

SELECT Count(distinct business_id)
FROM review

SELECT Count(distinct user_id)
FROM review
```

vi. Checkin = 493

```
SELECT Count(distinct business_id)
FROM checkin
```

vii. Photo = 10000 (id), (business_id) = 6493

```
SELECT Count(distinct id)
FROM photo

SELECT Count(distinct business_id)
FROM photo
```

viii. Tip = (user_id) 537, (business_id) = 3979

```
SELECT Count(distinct user_id)
FROM tip

SELECT Count(distinct business_id)
FROM tip
```

ix. User = 10000

```
SELECT Count(distinct id)
FROM user
```

x. Friend = (user_id) 11

```
SELECT Count(distinct user_id)
FROM friend
```

xi. Elite_years = (user_id) 11

```
SELECT Count(distinct user_id)
FROM friend
```

Note: Primary Keys are denoted in the ER-Diagram with a yellow key icon.

3. Are there any columns with null values in the Users table? Indicate "yes," or "no."

Answer: No

SQL code used to arrive at answer:

```
        select id, name, review_count, yelping_since, useful, funny, cool,
fans, average_stars,
                          compliment_hot, compliment_more, compliment_profile,
compliment_cute, compliment_list,
                          compliment_note, compliment_plain, compliment_cool,
compliment_funny, compliment_writer, compliment_photos
                from  user
                where   id is null
                             or name is null
                             or review_count is null
                             or yelping_since is null
                             or useful is null
                             or funny is null
                             or cool is null
                             or fans is null
                             or average_stars is null
                             or compliment_hot is null
                             or compliment_more is null
                             or compliment_profile is null
                             or compliment_cute is null
                             or compliment_list is null
                             or compliment_note is null
                             or compliment_plain is null
                             or compliment_cool is null
                             or compliment_funny is null
                             or compliment_writer is null
                             or compliment_photos is null
```

4. For each table and column listed below, display the smallest (minimum), largest (maximum), and average (mean) value for the following fields:

i. Table: Review, Column: Stars

min: 1   max: 5   avg: 3.7082

ii. Table: Business, Column: Stars

min:1   max:5   avg:3.6549

iii. Table: Tip, Column: Likes

min:0   max:2   avg:0.0144

iv. Table: Checkin, Column: Count

```
        min:1          max:53         avg:1.9414
```

v. Table: User, Column: Review_count

```
        min:0          max:2000             avg:24.2995
```

5. List the cities with the most reviews in descending order:

SQL code used to arrive at answer:

```
        select city, sum(review_count)
        from business
        group by city
        order by sum(review_count) desc
```

Copy and Paste the Result Below:

```
+-----------------+-------------------+
        | city            | sum(review_count) |
        +-----------------+-------------------+
        | Las Vegas       |             82854 |
        | Phoenix         |             34503 |
        | Toronto         |             24113 |
        | Scottsdale      |             20614 |
        | Charlotte       |             12523 |
        | Henderson       |             10871 |
        | Tempe           |             10504 |
        | Pittsburgh      |              9798 |
        | MontrÃƒÂ©al       |              9448 |
        | Chandler        |              8112 |
        | Mesa            |              6875 |
        | Gilbert         |              6380 |
        | Cleveland       |              5593 |
        | Madison         |              5265 |
        | Glendale        |              4406 |
        | Mississauga     |              3814 |
        | Edinburgh       |              2792 |
        | Peoria          |              2624 |
        | North Las Vegas |              2438 |
        | Markham         |              2352 |
        | Champaign       |              2029 |
        | Stuttgart       |              1849 |
        | Surprise        |              1520 |
        | Lakewood        |              1465 |
        | Goodyear        |              1155 |
        +-----------------+-------------------+
```

6. Find the distribution of star ratings to the business in the following cities:

i. Avon

SQL code used to arrive at answer:

```
select stars as [Star Rating], count(stars) as [Count]
from business b
where city = 'Avon'
group by stars
```

Copy and Paste the Resulting Table Below (2 columns Ã¢â‚¬â€œ star rating and count):

```
+-------------+-------+
| Star Rating | Count |
+-------------+-------+
|         1.5 |     1 |
|         2.5 |     2 |
|         3.5 |     3 |
|         4.0 |     2 |
|         4.5 |     1 |
|         5.0 |     1 |
+-------------+-------+
```

ii. Beachwood

SQL code used to arrive at answer:

```
select stars as [Star Rating], count(stars) as [Count]
from business b
where city = 'Beachwood'
group by stars
```

Copy and Paste the Resulting Table Below (2 columns Ã¢â‚¬â€œ star rating and count):

```
+-------------+-------+
| Star Rating | Count |
+-------------+-------+
|         2.0 |     1 |
|         2.5 |     1 |
|         3.0 |     2 |
|         3.5 |     2 |
|         4.0 |     1 |
|         4.5 |     2 |
|         5.0 |     5 |
+-------------+-------+
```

7. Find the top 3 users based on their total number of reviews:

SQL code used to arrive at answer:

```
select name, review_count
from user
order by review_count desc
limit 3
```

Copy and Paste the Result Below:

```
+--------+--------------+
| name   | review_count |
+--------+--------------+
| Gerald |         2000 |
| Sara   |         1629 |
| Yuri   |         1339 |
+--------+--------------+
```

8. Does posing more reviews correlate with more fans?
No

Please explain your findings and interpretation of the results:
Gerald with a total of 2000 reviews with 253 fans, averaging 7 fans per review
Sara with a total of 1629 reviews with 50 fans. Therefore we can interpret that posing more reviews does not corelate with more fans because Gerald would have more fans

9. Are there more reviews with the word "love" or with the word "hate" in them?

Answer: There are more reviews with the word "love"

SQL code used to arrive at answer:

```
select (select count(text)
               from review
               where text like "%love%") as  love_text,

           (select count(text)
               from review
               where text like "%hate%") as hate_text
```

```
+-----------+-----------+
| love_text | hate_text |
+-----------+-----------+
|      1780 |       232 |
+-----------+-----------+
```

10. Find the top 10 users with the most fans:

SQL code used to arrive at answer:

```
select name, fans
from user
order by fans desc
limit 10
```

Copy and Paste the Result Below:

```
+-----------+------+
| name      | fans |
+-----------+------+
| Amy       |  503 |
| Mimi      |  497 |
| Harald    |  311 |
| Gerald    |  253 |
| Christine |  173 |
| Lisa      |  159 |
| Cat       |  133 |
| William   |  126 |
| Fran      |  124 |
| Lissa     |  120 |
+-----------+------+
```

Part 2: Inferences and Analysis

1. Pick one city and category of your choice and group the businesses in that city or category by their overall star rating. Compare the businesses with 2-3 stars to the businesses with 4-5 stars and answer the following questions. Include your code.

i. Do the two groups you chose to analyze have a different distribution of hours?

I chose "Las Vegas" as city and "Shopping" as category

Yes, but a slight difference. 2-3 stars has a total of 13 workding days and 4-5 stars has 12

```
SELECT  CASE WHEN stars >= 4 THEN "4-5 stars"
             WHEN stars >= 2 THEN "2-3 stars"
             ELSE "below 2"
             END star_rank,
        city,
        c.category,
        count(distinct business.id) AS company_count,
        count(h.hours) AS working_days
FROM business
JOIN hours h ON business.id = h.business_id
JOIN category c ON business.id = c.business_id
WHERE city = "Las Vegas" AND c.category = "Shopping"
GROUP BY star_rank
```

```
+-----------+-----------+----------+---------------+--------------+
| star_rank | city      | category | company_count | working_days |
+-----------+-----------+----------+---------------+--------------+
| 2-3 stars | Las Vegas | Shopping |             2 |           13 |
| 4-5 stars | Las Vegas | Shopping |             2 |           12 |
+-----------+-----------+----------+---------------+--------------+
```

ii. Do the two groups you chose to analyze have a different number of reviews?

Yes, the total number of reviews froom 4-5 stars is doubled compared to 2-3 stars

SQL code used for analysis:

```
SELECT  CASE WHEN stars >= 4 THEN "4-5 stars"
             WHEN stars >= 2 THEN "2-3 stars"
             ELSE "below 2"
             END star_rank,
        city,
        c.category,
        count(distinct business.id) AS company_count,
        sum(review_count) AS total_review
FROM business
JOIN category c ON business.id = c.business_id
WHERE city = "Las Vegas" AND c.category = "Shopping"
GROUP BY star_rank

+-----------+-----------+----------+---------------+--------------+
| star_rank | city      | category | company_count | total_review |
+-----------+-----------+----------+---------------+--------------+
| 2-3 stars | Las Vegas | Shopping |             2 |           17 |
| 4-5 stars | Las Vegas | Shopping |             2 |           36 |
+-----------+-----------+----------+---------------+--------------+
```

iii. Are you able to infer anything from the location data provided between these two groups? Explain.

Stores with 2-3 stars are located in the same area, where 4-5 stars are apart from each other from the postal codes results

SQL code used for analysis:

```
               SELECT  CASE WHEN stars >= 4 THEN "4-5 stars"
                                       WHEN stars >= 2 THEN "2-3 stars"
                                       ELSE "below 2"
                                       END star_rank,
                              address,
                              neighborhood,
                              city,
                              postal_code
               FROM business
               JOIN category c ON business.id = c.business_id
               WHERE city = "Las Vegas" AND c.category = "Shopping"
               ORDER BY star_rank


+-----------+------------------------------+--------------+-------------+
| star_rank | address                      | neighborhood | postal_code |
+-----------+------------------------------+--------------+-------------+
| 2-3 stars | 3421 E Tropicana Ave, Ste I  | Southeast    | 89121       |
| 2-3 stars | 3808 E Tropicana Ave         | Eastside     | 89121       |
| 4-5 stars | 1000 Scenic Loop Dr          |              | 89161       |
| 4-5 stars | 3555 W Reno Ave, Ste F       |              | 89118       |
+-----------+------------------------------+--------------+-------------+
```

2. Group business based on the ones that are open and the ones that are closed. What differences can you find between the ones that are still open

and the ones that are closed? List at least two differences and the SQL code you used to arrive at your answer.

i. Difference 1: Total review is noticeably higher between open and closed business

ii. Difference 2: average stars given are very close to each other, we can infer which businesses were closed not solely to poor service or quality

```
               SELECT  CASE    WHEN is_open = 1 THEN "STILL OPEN"
                                              WHEN is_open = 0 THEN "CLOSED"
                                              END status,
                               count(distinct id) AS num_company,
                               sum(review_count) AS total_review,
                               round(avg(review_count),2) AS avg_review,
                               round(avg(stars),2) AS avg_stars
               FROM business
               GROUP BY is_open
               ORDER BY status DESC
```

```
               +------------+-------------+--------------+------------+------
-----+
               | status     | num_company | total_review | avg_review |
avg_stars |
               +------------+-------------+--------------+------------+------
-----+
               | STILL OPEN |        8480 |       269300 |      31.76 |
3.68 |
               | CLOSED     |        1520 |        35261 |       23.2 |
3.52 |
               +------------+-------------+--------------+------------+------
-----+
```

3. For this last part of your analysis, you are going to choose the type of analysis you want to conduct on the Yelp dataset and are going to prepare the data for analysis.

Ideas for analysis include: Parsing out keywords and business attributes for sentiment analysis, clustering businesses to find commonalities or anomalies between them, predicting the overall star rating for a business, predicting the number of fans a user will have, and so on. These are just a few examples to get you started, so feel free to be creative and come up with your own problem you want to solve. Provide answers, in-line, to all of the following:

i. Indicate the type of analysis you chose to do:
to find oout what are the most successful businesses in the business category

ii. Write 1-2 brief paragraphs on the type of data you will need for your analysis and why you chose that data:
The required data that is need for this type of analysis is the id, stars, and review count from the business table and categoru froom the category table.

The number of companies within each category and the average stars given by the customers annd the total review given to see if the data is biased and relevant.

By reducing irrelevant data, the categories with 10 companies at least will be analyzed with a average of 3.5+ stars

iii. Output of your finished dataset:

```
+-------------------------+---------------+-----------+---------------+
| category                | num_companies | avg_stars | total_reviews |
+-------------------------+---------------+-----------+---------------+
| Local Services          |            12 |      4.21 |           100 |
| Active Life             |            10 |      4.15 |           131 |
| Health & Medical        |            17 |      4.09 |           203 |
| Home Services           |            16 |       4.0 |            94 |
| Shopping                |            30 |      3.98 |           977 |
| Beauty & Spas           |            13 |      3.88 |           119 |
| American (Traditional)  |            11 |      3.82 |          1128 |
| Food                    |            23 |      3.78 |          1781 |
| Bars                    |            17 |       3.5 |          1322 |
+-------------------------+---------------+-----------+---------------+
```

iv. Provide the SQL code you used to create your final dataset:

```
SELECT  category,
        count(distinct id) AS num_companies,
        round(avg(stars),2) AS avg_stars,
        sum(review_count) total_reviews
FROM business
JOIN category ON business.id = category.business_id
GROUP BY category
HAVING avg_stars >= 3.5 AND num_companies >= 10
ORDER BY avg_stars DESC
```